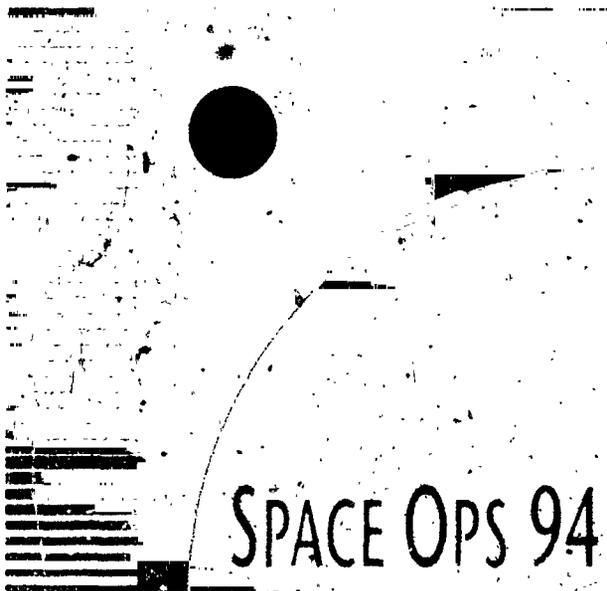


NASA Conference Publication 3281

Third International Symposium on Space Mission Operations and Ground Data Systems Part 1



*Proceedings of a conference held at
Greenbelt Marriott Hotel
Greenbelt, Maryland, USA
November 15-18, 1994*

(NASA-CP-3281-Pt-1) THIRD
INTERNATIONAL SYMPOSIUM ON SPACE
MISSION OPERATIONS AND GROUND DATA
SYSTEMS, PART 1 (NASA, Goddard
Space Flight Center) 666 p

N95-17179
--THRU--
N95-17257
Unclas

H1/17 0031736



NASA Conference Publication 3281

Third International Symposium on Space Mission Operations and Ground Data Systems Part 1

Edited by
James L. Rash
NASA Goddard Space Flight Center
Greenbelt, Maryland

Proceedings of a conference held at
Greenbelt Marriott Hotel
Greenbelt, Maryland, USA
November 15-18, 1994



National Aeronautics
and Space Administration

Goddard Space Flight Center
Greenbelt, Maryland 20771

1994

This publication is available from the NASA Center for Aerospace Information,
800 Elkridge Landing Road, Linthicum Heights, MD 21090-2934, (301) 621-0390.

EXECUTIVE SUMMARY

The Third International Symposium on Space Mission Operations and Ground Data Systems (SpaceOps 94) is being held November 14-18, 1994, in Greenbelt Maryland, USA, and is hosted by the NASA Goddard Space Flight Center. More than 400 people from nine countries are attending. This symposium follows the Second International Symposium that was hosted by the Jet Propulsion Laboratory in Pasadena, California, during November 1992. The First International Symposium on Ground Data Systems for Spacecraft Control, conducted in June 1990, was sponsored by the European Space Agency and the European Space Operations Centre.

The theme of this Third International Symposium is "Opportunities in ground data systems for high efficiency operations of space missions". Accordingly, the Symposium features more than 150 oral presentations in five technical tracks:

- Mission Management
- Operations
- Data Management
- Systems Engineering
- Systems Development

These five tracks are subdivided into over 50 sessions, each containing three presentations. The presentations focus on improvements in the efficiency, effectiveness, productivity, and quality of data acquisition, ground systems, and mission operations. New technology, techniques, methods, and human systems are discussed. Accomplishments are also reported in the application of information systems to improve data retrieval, reporting, and archiving; the management of human factors; the use of telepresence and teleoperations; and the design and implementation of logistics support for mission operations.

FOREWORD

We welcome you to SpaceOps 94! The Goddard Space Flight Center is pleased to host and sponsor our biennial symposium this year. We intend to maintain the same high standards set by our predecessors--the Jet Propulsion Laboratory in 1992, and the European Space Agency with the European Space Operations Centre in 1990.

Like other participating organizations, we benefit from the shared knowledge and combined experiences that are topics of discussion at the SpaceOps 94 symposium. Best of all, we benefit from seeing each other face-to-face and having the opportunity to discuss in person technical issues of mutual, often compelling interest.

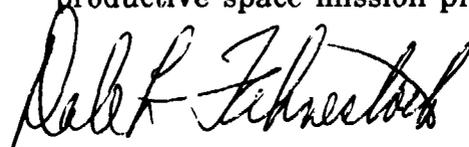
The large number of papers submitted to the SpaceOps 94 committee for acceptance and the projected attendance of over 400 of our colleagues should mean we are in for another splendid symposium this year. We believe these numbers mean that biennial meetings of our international space mission operations community are needed and are viewed as productive.

During the four days of our Symposium, more than 400 people from nine countries will hear more than 150 papers presented, as well as keynote, plenary, and panel talks by individuals from throughout the world. The papers in this proceedings document describe a wide range of ideas and experiences in our field that are developed from the perspectives of international space programs and their supporting industries.

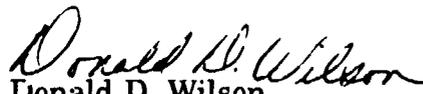
Our review of the papers indicates that future space mission operations will be strongly influenced by the following kinds of challenges and objectives:

- Empowering operators to perform at higher intellectual levels by the increased use of artificial intelligence
- Standardizing protocols, formats, databases, and operations to enable simultaneous and economical support of multiple missions
- Dealing with the science data avalanche
- Converting yesterday's and today's mission experiences into the "corporate knowledge" databases of tomorrow
- Sharing national resources in cooperative space ventures.

We wish you a rewarding week. We also wish for, and look forward to, greater interaction between our people and our countries--not just at our symposia, but in our everyday working world as we learn to achieve increasingly successful and productive space mission programs.



Dale L. Fahnestock
General Chair

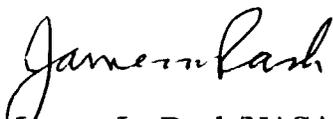


Donald D. Wilson
Executive Committee Chair

PREFACE

I would like to acknowledge the fine support of Laura Capella, Todd Del Priore, and April Johnson in the preparation of the manuscript for this document, which included entering data and creating FileMaker Pro scripts on the Macintosh computer to produce the the table of contents and author index.

If you have Internet access, I invite you to navigate to the NASA "Hot Topics" page using URL address http://hypatia.gsfc.nasa.gov/NASA_homepage.html. Possibly, using this path, you already may have accessed the World Wide Web information pages on SpaceOps 94, and we solicit your comments on what you find there. It is reasonable to assume that the call for papers and other information on the next SpaceOps (in 1996) will be similarly accessible a few months in advance. Please inform potentially interested colleagues regarding this information resource.



James L. Rash/NASA/GSFC
Editor
Publications Committee Chair

1994 SYMPOSIUM EXECUTIVE COMMITTEE

Dale L. Fahnestock
General Chair

Donald D. Wilson
*Executive Committee
Chair*

Peter M. Hughes
Technical Program

J. Michael Moore
Paper Selection

James L. Rash
Publications

Jessica N. Katz
Logistics and Registration

Roberta A. Valonis
*Publicity and Technical
Tours*

Michael R. Bracken
*Exhibitions and
Industry Liaison*

Lori M. Keehan
*Demonstrations and
Industry Liaison*

Dr. Sajjad Durrani
*Technical Program
Support*

Richard V. Nguyen
*Technical Program
Support*

Roger E. Wye
*Technical Program
Support*

SPACEOPS EXECUTIVE COMMITTEE

Charles F. Fuechsel
Co-Chair
NASA/HQ

Herwig Laue
Co-Chair
ESA/ESOC

Valery I. Altunin
NASA/JPL

Jean Latour
CNES

Konstantin G. Sukhanov
Lavochkin Association

Kurt Heftman
Consultant

T. D. Linick
NASA/JPL

Giulio Varsi
NASA/HQ

Joachim Kehr
DI.R/GSOC

Kimihiko Nakayama
NASDA

Donald D. Wilson
NASA/GSFC

PREVIOUS PAGE BLANK NOT FILMED

TABLE OF CONTENTS

Data Management

1. Communications		Page 1
DM.1.a	Telecommunications End-to-End Systems Monitoring on TOPEX/Poseidon: Tools and Techniques <i>Bruno J. Calanche</i>	3-9
DM.1.b	Communication Network for Decentralized Remote Tele-Science During the Spacelab Mission IML-2 <i>Uwe Christ, Klaus-Jürgen Schulz, Marco Incollingo</i>	11-20
DM.1.c	Earth Observing System (EOS) Communication (Ecom) Modeling, Analysis, and Testbed (EMAT) Activity <i>Vishal Desai</i>	21-28
DM.1.d *	S-Band and Ku-Band Return Service Interference Between TDRSS Users <i>Linda Harrell</i>	29-40
DM.1.e *	Attempt of Automated Space Network Operations at ETS-VI Experimental Data Relay System <i>Kiyoomi Ishihara, Masayuki Sugawara</i>	41-47
DM.1.f *	Table-Driven Configuration and Formatting of Telemetry Data in the Deep Space Network <i>Evan Manning</i>	49-56
DM.1.g	End-To-End Communication Test on Variable Length Packet Structures Utilizing AOS Testbed <i>W. H. Miller, V. Sank, W. Fong, J. Miko, M. Powers, J. Folk, B. Conaway, K. Michael, P.-S. Yeh</i>	57-64
DM.1.h	Demand Access Communications for TDRSS Users <i>David J. Zillig, Aaron Weinberg, Robert McOmber</i>	65-74
DM.1.i	The GRO Remote Terminal System <i>David J. Zillig, Joe Valvano</i>	75-82

Data Management

2. Data Handling		Page 83
DM.2.a	Overview on METEOSAT Geometrical Image Data Processing <i>Frank J. Diekmann</i>	85-93
DM.2.b *	UARS CDHF Conversion to DEC AXP Platform <i>Stuart Frye</i>	95
DM.2.c	Use of a Multimission System for Cost Effective Support of Planetary Science Data Processing <i>William B. Green</i>	97-103

* Presented in Poster Session

DM.2.d	A Second Generation 50 Mbps VLSI Level Zero Processing System Prototype <i>Jonathan C. Harris, Jeff Shi, Nick Speciale, Toby Bennett</i>	105-115
DM.2.e	A Corporate Memory for the GSFC Mission Operations Division <i>Beryl Hosack</i>	117
DM.2.f *	Fast Computational Scheme of Image Compression for 32-bit Microprocessors <i>Leonid Kasperovich</i>	119-123
DM.2.g	The Development and Operation of the International Solar-Terrestrial Physics Central Data Handling Facility <i>Kenneth Lehtonen</i>	125-132
DM.2.h	Economical Ground Data Delivery <i>Richard W. Markley, Russell H. Byrne, Daniel E. Bromberg</i>	133-138
DM.2.i	NASA Johnson Space Center Life Sciences Data System <i>Hasan Rahman, Jeffery Cardenas</i>	139-147
DM.2.j	A Processing Centre for the CNES CE-GPS Experimentation <i>Norbert Suard, Jean-Claude Durand</i>	149-154
DM.2.k	Single Stage Rocket Technology's Real Time Data System <i>Steven D. Voglewede</i>	155-168
DM.2.l	International Data Transfer for Space Very Long Baseline Interferometry <i>Alexandria B. Wiercigroch</i>	169-175
DM.2.m	ARACHNID: A Prototype Object-Oriented Database Tool for Distributed Systems <i>Herbert Younger, John O'Reilly, Bjorn Frogner</i>	177-183

Data Management

3. Telemetry Processing

Page 185

DM.3.a	A Low-Cost Transportable Ground Station for Capture and Processing of Direct Broadcast EOS Satellite Data <i>Don Davis, Toby Bennett, Nicholas M. Short, Jr.</i>	187-195
DM.3.b	Applications of Massively Parallel Computers in Telemetry Processing <i>Tarek A. El-Ghazawi, Jim Pritchard, Gordon Knoble</i>	197-204
DM.3.c	Test Telemetry and Command System (TTACS) <i>Alvin J. Fogel</i>	205-212
DM.3.d	Ground Equipment for the Support of Packet Telemetry and Telecommand <i>Wolfgang Hell</i>	213-224

* Presented in Poster Session

DM.3.e	Process and Methodology of Developing Cassini G&C Telemetry Dictionary <i>Edwin P. Kan</i>	225-232
DM.3.f	Multimission Telemetry Visualization (MT ^V) System: A Mission Applications Project From JPL's Multimedia Communications Laboratory <i>Ernest Koeberlein, III, Shaw Exum Pender</i>	233-240
DM.3.g	VLSI Technology for Smaller, Cheaper, Faster Return Link Systems <i>Kathy Nanzetta, Parminder Ghuman, Toby Bennett, Jeff Solomon, Jason Dowling, John Welling</i>	241-246
DM.3.h	Telemetry Distribution and Processing for the Second German Spacelab Mission D-2 <i>E. Rabenau, W. Kruse</i>	249-256
DM.3.i	Lessons Learned Supporting Onboard Solid-State Recorders <i>Jeff Shi, Tony Mao, Tim Clotworthy, Gerald Grebowsky</i>	257-264
DM.3.j	Experience With the EURECA Packet Telemetry and Packet Telecommand System <i>Erik Mose Sørensen, Paolo Ferri</i>	265-272
DM.3.k	A Modular, Software Reprogrammable, Telemetry Preprocessor for Open Systems Backplane Architectures <i>Steve Talabac</i>	273

Mission Management

1. Mission Planning		Page 275
MM.1.a	JPL Operations Cost Model for Flight Projects <i>John Carraway</i>	277
MM.1.b	Galileo Mission Planning for Low Gain Antenna Based Operations <i>R. Gershman, K. L. Buxbaum, J. M. Ludwinski, B. G. Paczkowski</i>	279-286
MM.1.c	A Distributed Planning Concept for Space Station Payload Operations <i>Jeff Hagopian, Theresa Maxwell, Tracey Reed</i>	287-294
MM.1.d	Mission Planning for Space-Based Satellite Surveillance Experiments With the MSX <i>R. Sridharan, T. Fishman, E. Robinson, H. Viggh, A. Wiseman</i>	295-303
MM.1.e	Engineering <i>Ulysses</i> Extended Mission <i>Shaun Standley</i>	305-309
MM.1.f	Accuracy Analysis of TDRSS Demand Forecasts <i>Daniel C. Stern, Allen J. Levine, Karl J. Pitt</i>	311-318

* Presented in Poster Session

MM.1.g	Joint Operations Planning for Space Surveillance Missions on the MSX Satellite <i>Grant Stokes, Andrew Good</i>	319-326
MM.1.h	Multi-Mission Operations From the Headquarters Perspective <i>Guenter Strobel</i>	327
MM.1.i	International Mission Planning for Space Very Long Baseline Interferometry <i>James S. Ulvestad</i>	329-336

Mission Management

2. Operations Planning		Page 337
MM.2.a	Geostationary Satellite Positioning by DLR/GSOC Operations and Management Methods <i>Peter Brittinger</i>	339-346
MM.2.b	Magellan Project: Evolving Enhanced Operations Efficiency to Maximize Science Value <i>Allan R. Chevront, James C. Neuman, J. Franklin McKinney</i>	347-354
MM.2.c	Grand Mission: Versus Small Ops Team: Can We Have Both? <i>Raúl García-Pérez</i>	355-360
MM.2.d	The Role of Mission Operations in Spacecraft Integration and Test <i>Raymond J. Harvey</i>	361-369
MM.2.e	Payload Operations Management of a Planned European SL-Mission Employing Establishments of ESA and National Agencies <i>Rolf Joensson, Karl L. Mueller</i>	371-376
MM.2.f	Evaluating Space Network (SN) Scheduling Operations Concepts Through Statistical Analysis <i>Carl Kwadrat, Nadine Happell</i>	377-
MM.2.g	SPOT4 Management Centre <i>Yves Labrune, X. Labbe, A. Roussel, P. Vielcanet</i>	389-396
MM.2.h	Costs Optimization for Operations Concepts of Small Satellite Missions <i>Jean-Michel Oberto</i>	397
MM.2.i	Nickel Cadmium Battery Operations and Performance <i>Gopalakrishna Rao, Jill Prettyman-Lukoschek, Richard Calvin, Thomas Berry, Robert Bote, Mark Toft</i>	399-408
MM.2.j	Mission Operations Management <i>David A. Rocco</i>	409-416

* Presented in Poster Session

MM.2.k	GRTS Operations Monitor/Control System <i>Richard A. Rohrer</i>	417-423
MM.2.l	International Space Station Alpha User Payload Operations Concept <i>Ronald A. Schlagheck, Elaine F. Duncan, William B. Crysel, James W. Rider</i>	425-433
MM.2.m	Shared Mission Operations Concept <i>Gary L. Spradlin, Richard P. Rudd, Susan H. Linick</i>	435-442
MM.2.n	Mars Pathfinder Mission Operations Concepts <i>Francis M. Sturms, Jr., William C. Dias, Albert Y. Nakata, Wallace S. Tai</i>	443-450

Mission Management

3. Planning Tools

Page 451

MM.3.a	Towards a New Generation of Mission Planning Systems: Flexibility & Performance <i>A. Gasquet, Y. Parrod, A. De Saint Vincent</i>	453-460
MM.3.b	Generic Mission Planning and Scheduling: The AXA ² Solution <i>O. T. Guffin, M. Newhouse</i>	461-466
MM.3.c *	A PC Based Tool for Mission Plan Production <i>Jean-Pierre Joli, C. Yven</i>	467-471
MM.3.d	Towards a Class Library for Mission Planning <i>Olivier Pujol, Simon T. Smith, Paul Starkey, Thilo Wolff</i>	473-480
MM.3.e	Autonomous Mission Planning and Scheduling - Innovative, Integrated, Responsive <i>Charisse Sary, Simon Liu, Larry Hull, Randy Davis</i>	481-488
MM.3.f *	A Mission Planning Concept and Mission Planning System for Future Manned Space Missions <i>Martin Wickler</i>	489-497

Operations

1. Automation

Page 499

OP.1.a	Operations Automation <i>Charles Thomas Boreham</i>	501-505
OP.1.b	X-Analyst, A Generic Tool for Automatic Flight Data Analysis and Spacecraft Performance Assessment <i>Jean-Marc Brenot</i>	507
OP.1.c	Production and Quality Assurance Automation in the Goddard Space Flight Center Flight Dynamics Facility <i>K. B. Chapman, C. M. Cox, C. W. Thomas, O. O. Cuevas, R. M. Beckman</i>	509-516

* Presented in Poster Session

OP.1.d	Graphic Server: A Real Time System for Displaying and Monitoring Telemetry Data of Several Satellites <i>Stéphane Douard</i>	517-523
OP.1.e *	Advanced Technologies in the ASI MLRO Towards a New Generation Laser Ranging System <i>Thomas Varghese, Giuseppe Bianco</i>	525-530
OP.1.f *	The Sequence of Events Generator: A Powerful Tool for Mission Operations <i>Hubertus Wobbe, Armin Braun</i>	531-537

Operations

2. Command and Control		Page 539
OP.2.a	Designing an Autonomous Environment for Mission Critical Operation of the EUVE Satellite <i>Annadiana Abedini, Roger F. Malina</i>	541-546
OP.2.b	Increases in Efficiency and Enhancements to the Mars Observer Non-stored Commanding Process <i>Robert N. Brooks Jr., J. Leigh Torgerson</i>	547-558
OP.2.c	SCL: An Off-The-Shelf System for Spacecraft Control <i>Brian Buckley, James Van Gaasbeck</i>	559-568
OP.2.d	Mission Operations Data Analysis Tools for Mars Observer Guidance and Control <i>Edwin P. Kan</i>	569-576
OP.2.e	Generic Trending and Analysis System <i>Lori Keehan, Jay Reese</i>	577-582
OP.2.f *	A Graphic System for Telemetry Monitoring and Procedure Performing at the TELECOM S.C.C. <i>Jean Philippe Loubeyre</i>	583-590
OP.2.g	Small Scale Sequence Automation Pays Big Dividends <i>Bill Nelson</i>	591-597
OP.2.h	Safety Aspects of Spacecraft Commanding <i>N. Peccia</i>	599-605
OP.2.i	The Development and Validation of Command Schedules for SeaWiFS <i>Robert H. Woodward, Watson W. Gregg, Frederick S. Patt</i>	607-614

* Presented in Poster Session

Operations

3. Control Centers		Page 615
OP.3.a	Towards Cheaper Control Centers <i>Lionel Baize</i>	617-624
OP.3.b *	The SILEX Experiment System Operations <i>B. Deinelenne</i>	625-631
OP.3.c	EURECA Mission Control Experience and Messages for the Future <i>H. Hübner, P. Ferri, W. Wimmer</i>	633-640
OP.3.d	SCOS II: ESA's New Generation of Mission Control System <i>M. Jones, N. C. Head, K. Keyte, P. Howard, S. Lynenskjold</i>	641-647
OP.3.e	CCS-MIP: Low Cost Customizable Control Centre <i>Christian Labezin, Pierre Vielcanet</i>	649-654
OP.3.f	The IUE Science Operations Ground System <i>Ronald E. Pitts, Richard Arquilla</i>	655-662
OP.3.g	Efficient Mission Control for the 48-Satellite Globalstar Constellation <i>Dan Smith</i>	663-670
OP.3.h	Mini All-Purpose Satellite Control Center (MASCC) <i>Gérard Zaouche</i>	671-678

Operations

4. Expert Systems		Page 679
OP.4.a	Delivering Spacecraft Control Centers With Embedded Knowledge-Based Systems: The Methodology Issue <i>S. Ayache, M. Haziza, D. Cayrac</i>	681-688
OP.4.b	SCOSII OL: A Dedicated Language for Mission Operations <i>Andrea Baldi, Dennis Elgaard, Steen Lynenskjold, Mauro Pecchioli</i>	689-696
OP.4.c	DIAMS Revisited: Taming the Variety of Knowledge in Fault Diagnosis Expert Systems <i>M. Haziza, S. Ayache, J.-M. Brenot, D. Cayrac, D.-P. Vo</i>	697-706
OP.4.d	Using Graphics and Expert System Technologies to Support Satellite Monitoring at the NASA Goudard Space Flight Center <i>Peter M. Hughes Gregory W. Shirah, Edward C. Luczak</i>	707-712
OP.4.e	Development and Use of an Operational Procedure Information System (OPIS) for Future Space Missions <i>N. Illmer, L. Mies, A. Schön, A. Jain</i>	713-717

* Presented in Poster Session

OP.4.f	SCOSII: ESA's New Generation of Mission Control Systems: The User's Perspective <i>P. Kaufeler, M. Pecchioli, I. Shurmer</i>	719-725
OP.4.g *	An Object Model for Multi-Mission Command Management System <i>Jon Kuntz</i>	727
OP.4.h	A Software Architecture for Automating Operations Processes <i>Kevin J. Miller</i>	729-734
OP.4.i	Spacecraft Command and Control Using Expert Systems <i>Scott Norcross, William H. Grieser</i>	735-740
OP.4.j	SEQ_GEN: A Comprehensive Multimission Sequencing System <i>Jose Salcedo, Thomas Starbird</i>	741-748
OP.4.k *	The PACOR II Expert System: A Case-Based Reasoning Approach to Troubleshooting <i>Charisse Sary</i>	749-756
OP.4.l	An Intelligent Automated Command and Control System for Spacecraft Mission Operations <i>A. William Stoffel</i>	757-763

Operations

5. Orbit Determination		Page 765
OP.5.a	DSN Co-Observing Operations to Support Space VLBI Missions <i>Valery I. Altunin, Thomas B. Kuiper, Pamela R. Wolken</i>	767-772
OP.5.b	Implementation of a Low-Cost, Commercial Orbit Determination System <i>Jim Corrigan</i>	773-783
OP.5.c *	Development of a Prototype Real-Time Automated Filter for Operational Deep Space Navigation <i>W. C. Masters, V. M. Pollmeier</i>	785-789
OP.5.d *	Magnetometer-Only Attitude and Rate Determination for a Gyro-less Spacecraft <i>G. A. Natanson, M.S. Challa, J. Deutschmann, D.F. Baker</i>	791-798
OP.5.e *	TDRS Orbit Determination by Radio Interferometry <i>Michael S. Pavloff</i>	799-806

* Presented in Poster Session

Operations

6. Small Explorers		Page 807
OP.6.a	Cost Efficient Operations for Discovery Class Missions <i>G. E. Cameron, J. A. Londshof, G. W. Whitworth</i>	809-816
OP.6.b	Ground Station Support for Small Scientific Satellites <i>R. Holdaway, E. Dunford, P. H. McPherson</i>	817-824
OP.6.c	Design of Ground Segments for Small Satellites <i>Guy Macé</i>	825-835
OP.6.d	The SAX Italian Scientific Satellite. The On-Board Implemented Automation as a Support to the Ground Control Capability <i>Andrea Martelli</i>	837-846
OP.6.e	Small Satellite Space Operations <i>Keith Reiss</i>	847-853

Systems Development

1. Architectural Approaches		Page 855
SD.1.a	Embedded Parallel Processing Based Ground Control Systems for Small Satellite Telemetry <i>Michael L. Forman, Tushar K. Hazra, Gregory M. Troendly, William G. Nickum</i>	857-864
SD.1.b	Open Solutions to Distributed Control in Ground Tracking Stations <i>Wm. Randy Heuser</i>	865-877
SD.1.c	An Agent-Oriented Approach to Automated Mission Operations <i>Walt Truszkowski, Jidé Odubiyi</i>	879-887
SD.1.d	Advanced Ground Station Architecture <i>David Zillig, Ted Benjamin</i>	889-896

Systems Development

2. Development Tools		Page 897
SD.2.a	Automating Testbed Documentation and Database Access Using World Wide Web (WWW) Tools <i>Charles Ames, Brent Auernheimer, Young H. Lee</i>	899-904
SD.2.b *	Towards an Integral Computer Environment Supporting System Operations Analysis and Conceptual Design <i>E. Barro, A. Del Bufalo, F. Rossi</i>	905-913
SD.2.c	SEQ_POINTER: Next Generation, Planetary Spacecraft Remote Sensing Science Observation Design Tool <i>Jeffrey S. Boyer</i>	915-922

* Presented in Poster Session

SD.2.d *	Knowledge-Based Critiquing of Graphical User Interfaces With CHIMES <i>Jianping Jiang, Elizabeth D. Murphy, Leslie E. Carter, Walter F. Truszkowski</i>	923-928
SD.2.e	SEQ_REVIEW: A Tool for Reviewing and Checking Spacecraft Sequences <i>Pierre F. Maldague, Mekki El-Boushi, Thomas J. Starbird, Steven J. Zawacki</i>	929-936
SD.2.f	Simplifying Operations With an Uplink/Downlink Integration Toolkit <i>Susan Murphy, Kevin Miller, Ana Maria Guerrero, Chester Joe, John Louie, Christine Aguilera</i>	937-943
SD.2.g	ELISA, A Demonstrator Environment for Information Systems Architecture Design <i>Chantal Panem</i>	945-952
SD.2.h	Software Interface Verifier <i>Tomas J. Soderstrom, Laura A. Krall, Sharon A. Hope, Brian S. Zupke</i>	953-960

Systems Development

3. Methods		Page 961
SD.3.a	OOD/OOP Experience in the Science Operations Center Part of the Ground System for X-Ray Timing Explorer Mission <i>Abdur Rahim Choudhary</i>	963-970
SD.3.b	Mission Operations Development: A Structured Approach <i>Michael Fatig</i>	971
SD.3.c *	The Cooperative Satellite Learning Project: Space Missions Supporting Education <i>Michael Fatig</i>	973
SD.3.d	A Proven Approach for More Effective Software Development and Maintenance <i>Rose Pajerski, Dana Hall, Craig Sinclair</i>	975-983
SD.3.e	XMM Instrument On-Board Software Maintenance Concept <i>N. Peccia, F. Giannini</i>	985-992
SD.3.f	Integration of a Satellite Ground Support System Based on Analysis of the Satellite Ground Support Domain <i>R. D. Pendley, E. J. Scheidker, D. S. Levitt, C. R. Myers, R. D. Werking</i>	993-1000
SD.3.g *	Software Process Assessment (SPA) <i>Linda H. Rosenberg, Sylvia B. Sheppard, Scott A. Butler</i>	1001-1008

* Presented in Poster Session

SD.3.h *	Taking Advantage of Ground Data Systems Attributes to Achieve Quality Results in Testing Software <i>Clayton B. Sigman, John T. Koslosky, Barbara H. Hageman</i>	1009-1014
SD.3.i	SCOS II - An Object Oriented Software Development Approach <i>Martin Symonds, Steen Lyneskjold, Christian Müller</i>	1015-1022

Systems Development

4. Modeling		Page 1023
SD.4.a	Evaluating Modeling Tools for the EDOS <i>Gordon Knoble, Frederick McCaleb, Tanweer Aslam, Paul Nester</i>	1025-1030
SD.4.b	Solar and Heliospheric Observatory (SOHO) Experimenters' Operations Facility (EOF) <i>Eliane Larduinat, William Potter</i>	1031-1038
SD.4.c	Galileo Spacecraft Modeling for Orbital Operations <i>Bruce A. McLaughlin, Erik N. Nilsen</i>	1039-1044
SD.4.d	The Advanced Orbiting Systems Testbed Program: Results to Date <i>John F. Otranto, Penny A. Newsome</i>	1045-1054
SD.4.e	NCCDS Performance Model <i>Eric Richmond, Antonio Vallone</i>	1055-1062
SD.4.f	Evaluation of NASA's End-to-End Data Systems Using DSDS+ <i>Christopher Rouff, William Davenport, Philip Message</i>	1063-1069
SD.4.g	Analysis of Space Network Loading <i>Mark Simons, Gus Larrison</i>	1071-1077
SD.4.h	Modeling ESA's TT&C Systems <i>Enrico Vassallo</i>	1079-1090

Systems Development

5. Simulation		Page 1091
SD.5.a	A General Mission Independent Simulator (GMIS) and Simulator Control Program (SCP) <i>Paul L. Baker, J. Michael Moore, John Rosenberger</i>	1093-1100
SD.5.b	A Reusable Real-Time Object Oriented Spacecraft Simulator <i>Eric Beser</i>	1101
SD.5.c *	Test/Score/Report: Simulation Techniques for Automating the Test Process <i>Barbara H. Hageman, Clayton B. Sigman, John T. Koslosky</i>	1103-1109

* Presented in Poster Session

SD.5.d	Spacecraft Data Simulator for the Test of Level Zero Processing Systems <i>Jeff Shi, Julie Gordon, Chandru Mirchandani, Diem Nguyen</i>	1111-1120
--------	--------------------------------------------------------------------------------------------------------------------------------------------	-----------

Systems Engineering

1. Re-engineering		Page 1121
SE.1.a	Re-engineering the Multimission Command System at the Jet Propulsion Laboratory <i>Scott Alexander, Jeff Biesiadecki, Nagin Cox, Susan Murphy, Tim Reeve</i>	1123-1131
SE.1.b	Re-Engineering Nascom's Network Management Architecture <i>Brian C. Drake, David Messent</i>	1133-1141
SE.1.c	Reengineering NASA's Space Communications to Remain Viable in a Constrained Fiscal Environment <i>Rhoda Shaller Hornstein, Donald J. Hei, Jr., Angelita C. Kelly, Patricia C. Lightfoot, Holland T. Bell, Izeller E. Cureton-Snead, William J. Hurd, Charles H. Scales</i>	1143-1150
SE.1.d *	A System Study for Satellite Operation and Control in Next Generation <i>K. Nakayama, T. Shigeta, T. Gotanda, K. Yamamoto, Y. Yokokawa</i>	1151-1157

Systems Engineering

2. Reusable Systems		Page 1159
SE.2.a	Transportable Payload Operations Control Center Reusable Software: Building Blocks for Quality Ground Data Systems <i>Ron Mahmot, John T. Koslosky, Edward Beach, Barbara Schwarz</i>	1161-1169
SE.2.b	Customizing the JPL Multimission Ground Data System: Lessons Learned <i>Susan C. Murphy, John J. Louie, Ana Maria Guerrero, Daniel Hurley, Dana Flora-Adams</i>	1171-1175
SE.2.c	Configurable Technology Development for Reusable Control and Monitor Ground Systems <i>David R. Uhrlaub</i>	1177-1184

* Presented in Poster Session

Systems Engineering

3. Standards		Page 1185
SE.3.a	A New Communication Protocol Family for a Distributed Spacecraft Control System <i>Andrea Baldi, Marco Pace</i>	1187-1195
SE.3.b	Standardizing the Information Architecture for Spacecraft Operations <i>C. R. Easton</i>	1197-1204
SE.3.c	A Standard Satellite Control Reference Model <i>Constance Golden</i>	1205-1212
SE.3.d	Standard Protocol Stack for Mission Control <i>Adrian J. Hooke</i>	1213-1220
SE.3.e	The Space Communications Protocol Standards Program <i>Alan Jeffries, Adrian J. Hooke</i>	1221-1228
SE.3.f	The ESA Standard for Telemetry & Telecommand Packet Utilisation P.U.S. <i>J.-F. Kaufeler</i>	1229-1234
SE.3.g	Packet Utilisation Definitions for the ESA XMM Mission <i>H. R. Nye</i>	1235-1242
SE.3.h	Use of Data Description Languages in the Interchange of Data <i>M. Pignède, B. Real-Planells, S. R. Smith</i>	1243-1251
SE.3.i	Cross Support Overview and Operations Concept for Future Space Missions <i>William Stallings, Jean-Francois Kaufeler</i>	1253-1260
SE.3.j	The CCSDS Return All Frames Space Link Extension Service <i>Hans Uhrig, John Pietras, Michael Stoloff</i>	1261-1269
SE.3.k	Proposal for Implementation of CCSDS Standards for Use With Spacecraft Engineering/Housekeeping Data <i>Dave Welch</i>	1271-1276

Systems Engineering

4. Systems Architectures		Page 1277
SE.4.a	Ground Segment Strategies and Technologies in Support of Cost Reductions <i>K. Debatin</i>	1279
SE.4.b	Mission Operations Centers (MOCs): Integrating Key Spacecraft Ground Data System Components <i>Randy Harbaugh, Donna Szakal</i>	1281-1288

* Presented in Poster Session

SE.4.c	ATOS: Integration of Advanced Technology Software Within Distributed Spacecraft Mission Operations Systems <i>M. Jones, J. Wheadon, W. O'Mullane, D. Whitgift, K. Poulter, M. Niézette, R. Timmermans, Iván Rodríguez, R. Romero</i>	1289-1296
SE.4.d	The NASA Mission Operations and Control Architecture Program <i>Paul J. Ondrus, Richard D. Carper, Alan J. Jeffries</i>	1297-1303
SE.4.e	Renaissance Architecture for Ground Data Systems <i>Dorothy C. Perkins, Lawrence B. Zeigenfuss</i>	1305-1316
SE.4.f	Architecture of a Distributed Multimission Operations System <i>Takahiro Yamada</i>	1317-1324

Systems Engineering

5. Systems Engineering Tools		Page 1325
SE.5.a	Re-engineering the Mission Life Cycle With ABC & IDEF <i>Daniel Mandl, Michael Rackley, Jay Karlin</i>	1327-1334
SE.5.b	MO&DSD Online Information Server and Global Information Repository Access <i>Diem Nguyen, Kam Ghaffarian, Keith Hogie, William Mackey</i>	1335-1342
SE.5.c	Orbital Mechanics Processing in a Distributed Computing Environment <i>Randolph Nicklas</i>	1343
SE.5.d	The Requirements Generation System: A Tool for Managing Mission Requirements <i>Sylvia B. Sheppard</i>	1345-1351
SE.5.e	An Opportunity Analysis System for Space Surveillance Experiments With the MSX <i>Ramaswamy Sridharan, Gary Duff, Tony Hayes, Andy Wiseman</i>	1353-1360
SE.5.f	Matrix Evaluation of Science Objectives <i>Randii R. Wessen</i>	1361-1368

Systems Engineering

6. Systems Operations		Page 1369
SE.6.a	A New Systems Engineering Approach to Streamlined Science and Mission Operations for the Far Ultraviolet Spectroscopic Explorer (FUSE) <i>Madeline J. Butler, George Sonneborn, Dorothy C. Perkins</i>	1371-1376
SE.6.b *	Risk Reduction Methodologies and Technologies for the Earth Observing System (EOS) Operations Center (EOC) <i>Richard K. Hudson, Nelson V. Pingitore</i>	1377-1381

* Presented in Poster Session

SE.6.c	EDOS Operations Concept and Development Approach <i>Gordon Knoble, C. Garman, G. Alcott, C. Ramchandani, J. Silvers</i>	1383-1390
SE.6.d	Concurrent Engineering: Spacecraft and Mission Operations System Design <i>J. A. Landshof, R. J. Harvey, M. H. Marshall</i>	1391-1397

Author Index Page No. A-1

** Presented in Poster Session*

Data Management

1. Communications		Page 1
DM.1.a	Telecommunications End-to-End Systems Monitoring on TOPEX/Poseidon: Tools and Techniques <i>Bruno J. Calanche</i>	3-9
DM.1.b	Communication Network for Decentralized Remote Tele-Science During the Spacelab Mission IML-2 <i>Uwe Christ, Klaus-Jürgen Schulz, Marco Incollingo</i>	11-20
DM.1.c	Earth Observing System (EOS) Communication (Ecom) Modeling, Analysis, and Testbed (EMAT) Activity <i>Vishal Desai</i>	21-28
DM.1.d *	S-Band and Ku-Band Return Service Interference Between TDRSS Users <i>Linda Harrell</i>	29-40
DM.1.e *	Attempt of Automated Space Network Operations at ETS-VI Experimental Data Relay System <i>Kiyoomi Ishihara, Masayuki Sugawara</i>	41-47
DM.1.f *	Table-Driven Configuration and Formatting of Telemetry Data in the Deep Space Network <i>Evan Manning</i>	49-56
DM.1.g	End-To-End Communication Test on Variable Length Packet Structures Utilizing AOS Testbed <i>W. H. Miller, V. Sank, W. Fong, J. Miko, M. Powers, J. Folk, B. Conaway, K. Michael, P.-S. Yeh</i>	57-64
DM.1.h	Demand Access Communications for TDRSS Users <i>David J. Zillig, Aaron Weinberg, Robert McOmber</i>	65-74
DM.1.i	The GRO Remote Terminal System <i>David J. Zillig, Joe Valvano</i>	75-82

* Presented in Poster Session

TELECOMMUNICATIONS END-TO-END SYSTEMS MONITORING ON TOPEX/POSEIDON: TOOLS AND TECHNIQUES

Bruno J. Calanche
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California, USA

ABSTRACT

The TOPEX/Poseidon Project Satellite Performance Analysis Team's (SPAT) roles and responsibilities have grown to include functions that are typically performed by other teams on JPL Flight Projects. In particular, SPAT Telecommunication's role has expanded beyond the nominal function of monitoring, assessing, characterizing, and trending the spacecraft (S/C) RF/Telecom subsystem to one of End-to-End Information Systems (EEIS) monitoring. This has been accomplished by taking advantage of the spacecraft and ground data system structures and protocols.

By processing both the received spacecraft telemetry minor frame ground generated CRC flags and NASCOM block poly error flags, bit error rates (BER) for each link segment can be determined. This provides the capability to characterize the separate link segments, determine science data recovery, and perform fault/anomaly detection and isolation. By monitoring and managing the links, TOPEX has successfully recovered ~99.9% of the science data with an integrity (BER) of better than 1×10^{-8} .

This paper presents the algorithms used to process the above flags and the techniques used for EEIS monitoring.

Key words: Telecom OPS, Link Monitoring and Management, End-to-End Information Systems (EEIS) Monitoring, Data Recovery and Integrity.

1. INTRODUCTION

TOPEX/Poseidon was launched on Monday, August 10, 1992 as a joint US/French venture. Its mission is to map the ocean surface using radar altimetry. The measurement is so accurate that even at an altitude of 1336 km

(830 mi), it can detect height changes in the ocean surface as small as 5.0 cm. Two primary precision orbit determination (POD) systems are used - laser and doppler shift (French Doris System). Both are limited in coverage because they are land based. A third method for POD utilizes an experimental global positioning system (GPS) receiver and offers the potential for higher accuracy.

In order to take full advantage of the science information obtainable and to provide it with a high integrity for science data processing, a program was implemented by SPAT Telecom Operations to perform End-to-End Information Systems (EEIS) monitoring. The program takes full advantage of the data structures and protocols used during data transfer - i.e. TOPEX to White Sands (WSGT) via TDRS and WSGT to JPL Project Operations Control Center (POCC) via NASCOM; and, provides Telecom Ops with an end-to-end view of data recovery.

2. EEIS MONITORING CONCEPT

The end-to-end telemetry and science recovery monitoring is inherent in/and an offshoot of the telemetry data structure used on TOPEX and the NASCOM block packaging and design. On TOPEX, the science and telemetry data is packaged in minor frames (MMS data structure - this predates CCSDS standards for telemetry packaging). All engineering and science telemetry minor frame data is packaged with a 16 bit CRC control word (See Figure 2.1) and convolutionally encoded ($R=1/2$, $K=7$). The C.E. data modulates (QPSK) a carrier and is transmitted to White Sands (WSGT) via TDRSS. On the Q-Channel (recorder playback mode) the C.E. data is also interleaved prior to modulation. At WSGT, demodulation and Viterbi decoding utilizing soft detection occurs. The Q-Channel C.E. data is also de-interleaved prior to decoding. The recovered minor frame data plus frame error

control word is packaged in NASCOM blocks (~4.5 minor frames per block) with a 22 bit poly error check and is transmitted to JPL. At JPL's TOPEX/Poseidon Project Operations Control Center (POCC), the NASCOM Front End Processor (NFEP) receives the NASCOM block, deblocks the minor frames and generates the minor frame and NASCOM block CRC flags. All minor frames from an error detected NASCOM block are tagged with an NASCOM block error flag. The TOPEX data transmission/recovery scheme is depicted in Figure 2.2.

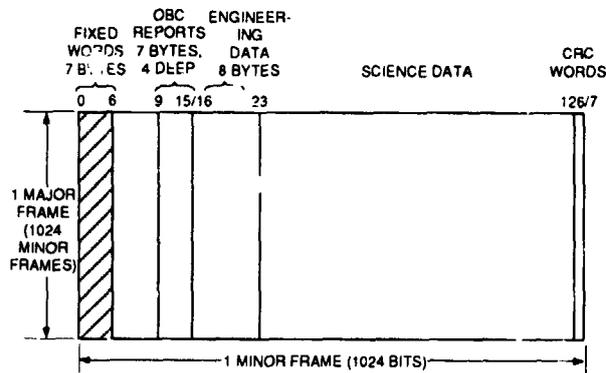


Figure 2.1. TOPEX Telemetry Stream-Major Frame Structure

By processing the received spacecraft minor frame ground generated CRC flags and the minor frame NASCOM block poly error flags (See Figure 2.2), the following can be determined:

- 1) Actual TOPEX-to-WSGT via TDRS return link BER performance – used for link calibration, margin determination, an assessment/impact of non link budget (non RF) losses
- 2) NASCOM link BER performance – used for link assessment, fault/anomaly detection and isolation, and link management

; and, performed

- 3) End-to-End Systems (EEIS) monitoring
- 4) Tape recorder characterization
- 5) Science recovery assessment

3.0 MINOR FRAME ERROR FLAG PROCESSING: BER DETERMINATION

End-to-End Telemetry and Science Data Recovery Monitoring (and therefore EEIS monitoring) involves the calculation of a metric for assessing data recovery performance. Two metrics are available. One is the percentage of

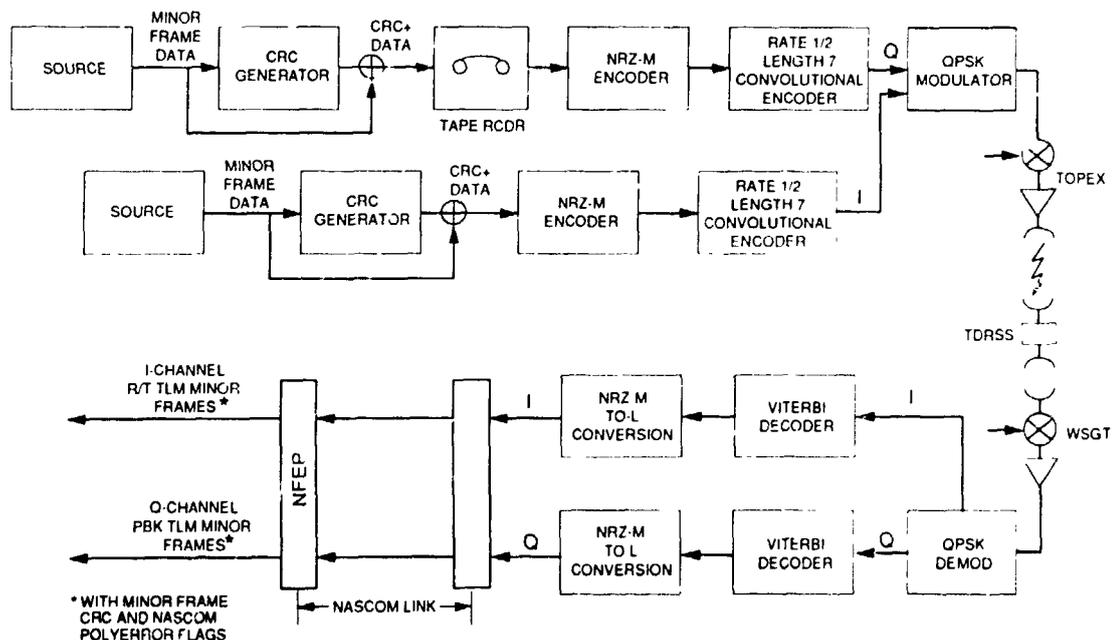


Figure 2.2. TOPEX Data Transmission/Recovery Scheme¹

¹RECOVERY SCHEME SHOWS TAPE RECORDER PLAYBACK MODE (THE Q-CHANNEL S/C INTERLEAVING AND GROUND DE-INTERLEAVING PROCESSES ARE NOT SHOWN)

data recovered and the other is the integrity of the data recovered – i.e., the BER. The calculation of the BER is based on the CRC flags.¹ The return link BER is calculated from the received minor frame files as:

$$(3.1) \text{ R/L BER} = \frac{\text{\# of frame CRC errors}}{\text{\# of minor frames (Bits/Minor frame)}}$$

where the bits/minor frame = 1024

The NASCOM BER is also calculated from the minor frames files as:

$$(3.2) \text{ NASCOM BER} = \frac{\text{\# of blocks with CRC error}}{\text{\# of blocks (bits/block)}}$$

where bits/block = 4800; and,

$$\text{\# of blocks} = \frac{\text{\# of minor frames}}{\text{(minor frames/block)}}$$

The # of blocks with CRC errors (estimate) is given as:

$$(3.3) \text{ \# of blocks with CRC error (estimate)} = \frac{\text{\# of minor frames with block CRC flagset}}{\text{(minor frames/block)}}$$

and, the minor frames/block = 4624/1024 (~4.5).

A minor frame file includes all the minor frames received during a scheduled TDRS event. TOPEX schedules TDRS events based on pass type. The pass types used on the TOPEX Project are given in Table 3.0. Three types of minor frame files are processed – the 16 kbps R/T files, the MRO files, and the PBK (512 kbps) frame files. The MRO (Memory ReadOut) data from the On Board Computer (OBC) is not packaged with the S/C generated 16 bit CRC control word.

For the determination of # of minor frames, # of frame CRC errors, and # of minor frames with block CRC flagset used in equations 3.1, 3.2 and 3.3, an integration time of 2 minutes is used. The results of equation 3.1 are labeled as RLPBKBER – for the Q-Channel PBK_SA data and RLR16BER – for the 16 kbps I or

Q-Channel R/T Telemetry (TLM) data. The results of equation 3.2 are labeled NSPBKBER – for the PBK_SA Q-Channel data returned to JPL via NASCOM links and NSR16 BER – for the 16 kbps R/T Telemetry return to JPL via the NASCOM links.

Table 3.0 TOPEX Return Link Events (Pass Types)

Pass Type	Channel/Data Rate	OPS Mode
LOAD_MRO_SA	I-Channel/16 kbps Telemetry Q-Channel/32 kbps OBC Data*	R/T Eng. TLM Memory Readout
LOAD_MkO_MA	IBID	IBID
MONITOR_MA	Q-Channel/16 kbps Telemetry I-Channel/16 kbps Telemetry	R/T Eng. TLM Not Returned to JPL
PBK_SA	Q-Channel/512 kbps I-Channel/16 kbps Telemetry	Recorded Science and Eng. Data R/T Eng. TLM

*Not packaged with the S/C generated 16 bit CRC control word

4.0 APPLICATION: END-TO-END LINK MONITORING

Typical LOAD_MRO_SA I-Channel 16 kbps performance is given in Figure 4.1. The LOAD_MRO_SA pass was scheduled via TDRS_East on day 1994-187 from 20:35:00 to 21:15:00. The NASCOM BER shows no NASCOM hits and the return link 16 kbps I-Channel shows BER performance of 2.64×10^{-8} . Errors occur at the start of the pass. This is typical. Occasionally, the RF link shows error free performance; and, the performance is not TDRS dependent – i.e. East or West.

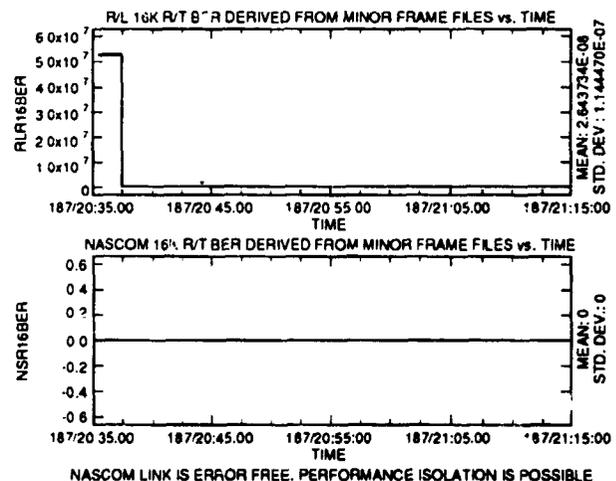


Figure 4.1. Typical LOAD_MRO_SA Performance (Day 1994-187)

A MONITOR_MA pass, Q-Channel 16 kbps return is given in Figure 4.2. This pass occurred on day 1993-191 from 00:17:00 to 00:57:00. This pass shows random RF bit hits (not NASCOM induced). The BER performance was $\sim 5.2 \times 10^{-8}$.

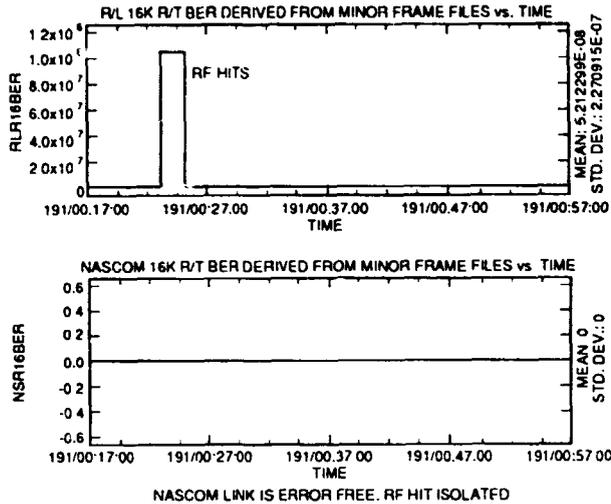


Figure 4.2. MONITOR_MA Pass (1993-191) Showing RF Bit Errors

A PBK_SA pass showing NASCOM bit errors coupling on to the received minor frame data is given in Figure 4.3. This pass occurred on day 1994-166 and was scheduled from 18:09:00 to 18:34:00 via TDRS_East. The pass was scheduled for playback of tape recorder B.

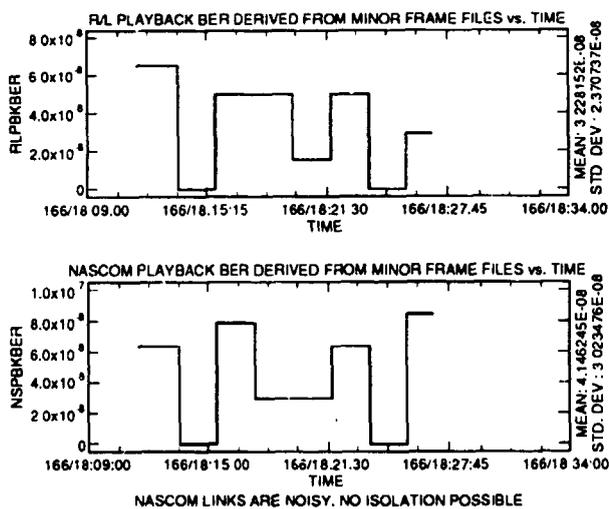


Figure 4.3. NASCOM Link Errors Coupling Onto the Received Playback Minor Frames (Tape Recorder B Playback)

Tape recorder playback characteristics and science recovery performance is given in sections 4.1 and 4.4, respectively. Playback rate is at 512 kbps and takes ~ 15 minutes (8 hours worth of recorded data).

The technique clearly isolates errors to the NASCOM links – fault/anomaly detection and isolation.

4.1 APPLICATION: TAPE RECORDER PERFORMANCE CHARACTERIZATION

TOPEX/Poseidon operates with three tape recorders. These are designated as A, B, and C. Each records science and engineering data for \sim eight hours a day. Three PBK_SA passes are scheduled daily for TOPFX to WSGT via TDRS playback, and subsequent transmission to the JPL POCC via the NASCOM links. Tape recording begins at the beginning of tape (BOT) and playback is reversed, bringing the tape back to BOT for subsequent recording.

Because of the tape recording and playback operations; and, due the fact that the S/C generated minor frame data (including the 16 bit CRC word) are recorded, the tape recorders can be characterized and tape media errors can be isolated. Figure 4.4 shows typical tape recorder (T/R) – C performance. This T/R-C playback occurred on day 1994-185 during a PBK_SA pass scheduled from 02:13:00 to 02:38:00. Errors and/or data gaps occur in the tape "soft spot" on $\sim 98\%$ of the playbacks. On occasion ($\sim 2\%$) T/R-C displays error free

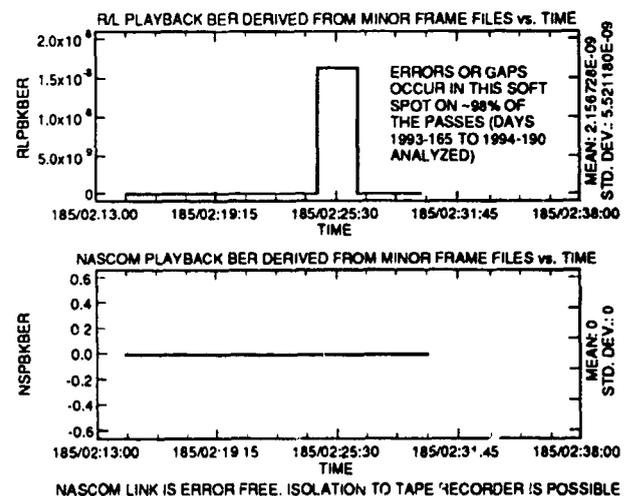


Figure 4.4. Tape Recorder C BER Signature

performance. On ~39% of the passes, T/R-C shows errors in this region (T/R-C soft spot). On the other ~59% of the passes, minor frame data gaps occur (4 to 8 minor frames).

Figure 4.4 is known as the T/R-C BER signature. Tape recorder-A's BER signature is given in Figure 4.5. Tape recorder-A has one soft spot and errors occur in that region on ~3% of the passes. Tape recorder-B has four soft spots - designated as 1, 2, 3, and 4. Errors occur as follows in those regions:

- 1 - 61% of the passes
- 2 - 2.9% of the passes
- 3 - 59% of the passes
- 4 - 56% of the passes

For regions 1, 3, and 4 data gaps occur ~39%, 41%, and 44% of the passes, respectively. The data gaps range from 4 to 8 minor frames. Tape Recorder B has multiple BER signatures, one is presented in Figure 4.6.

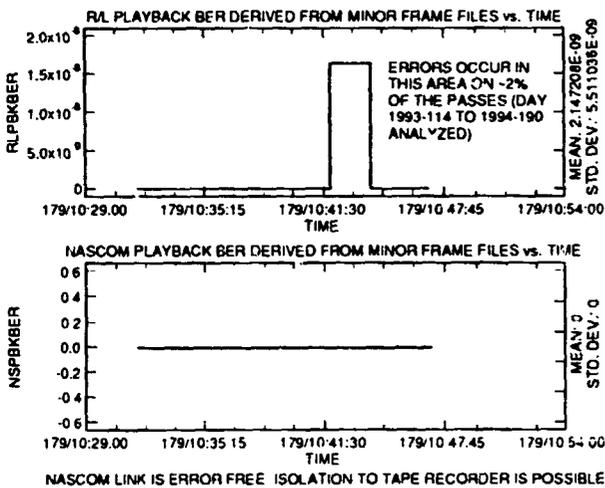


Figure 4.5. Tape Recorder A BER Signature

The overall tape recorder performance based on the PBK_SA return link (R/L) BER is given in Table 4.1. The data was taken from 1993-081 to 1993-339.

Table 4.1 Tape Recorder Performance Based on PBK_SA R/L BER

Tape Rcdr	Days 081 to 113	Days 114 to 164	Days 165 to 248	Days 249 to 339
A	1.2 x 10 ⁻¹⁰	1.9 x 10 ⁻¹⁰	1.55 x 10 ⁻¹⁰	2.5 x 10 ⁻¹⁰
C	6.45 x 10 ⁻¹⁰	1.16 x 10 ⁻⁹	1.24 x 10 ⁻⁹	1.0 x 10 ⁻⁹
B	7.16 x 10 ⁻⁹	6.27 x 10 ⁻⁹	6.32 x 10 ⁻⁹	5.83 x 10 ⁻⁹

Tape recorder performance has remained stable (shown no degradatio:).

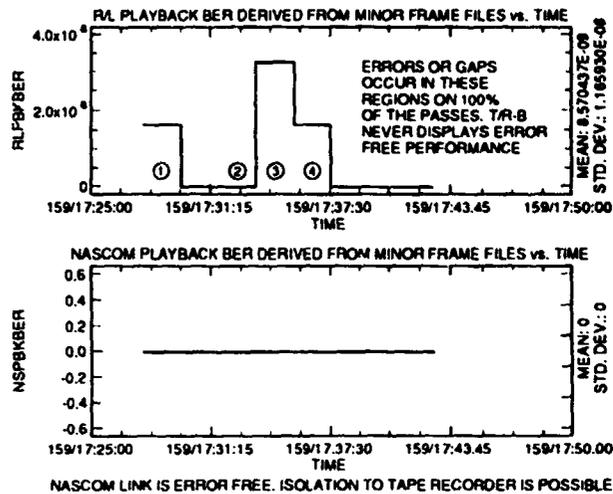


Figure 4.6. One of Many Tape Recorder B Signatures (All Soft Spots Are Shown)

4.2 APPLICATION : NASCOM LINK PERFORMANCE DETERMINATION

Using the NASCOM playback BER data (NSPBKBER), NASCOM link performance during the PBK_SA passes was analyzed early during TOPEX's project (1993-093 to -339). The results are summarized in Table 4.2. The NASCOM configuration being used is designated.

Table 4.2 NASCOM Return Link BER Performance (PBK_SA 512 kbps Playback)

Days	NASCOM Configuration	BER	Total Passes
093 to 189	768 kbps	9.14 x 10 ⁻¹⁰	286
190 to 227	536 kbps	3.19 x 10 ⁻⁸	112
228 to 248	544 kbps	5.71 x 10 ⁻¹⁰	63
249 to 339	544 kbps	2.5 x 10 ⁻⁹	266

Using the NASCOM return link BER data (NSR16BER) for the 16 kbps real time telemetry, the NASCOM link performance was determined for days 1993-179 to 234. The results are given in Table 4.3.

Table 4.3 NASCOM Return Link Performance (16 kbps Real Time Telemetry)

Passtype	Channel	BER	Total Passes
PBK_SA	I	3.47 x 10 ⁻⁹	158
LOAD_MRO_SA	I	1.69 x 10 ⁻⁹	62
LOAD_MRO_MA	I	2.93 x 10 ⁻⁹	69
MONITOR_MA	Q	4.4 x 10 ⁻⁹	624

The NASCOM links are monitored by TOPEX and NASCOM is informed of link performance. Based on this monitoring, TOPEX requested a link configuration change from 536 kbps to 544 kbps for the tape recorder playbacks.

4.3 APPLICATION: TOPEX TO WSGT RETURN LINK PERFORMANCE DETERMINATION

By accounting for NASCOM link performance and tape recorder performance, the actual TOPEX to WSGT via TDRS link performance was determined. Passes where NASCOM link performance could not be decoupled from the TOPEX to WSGT RF link were omitted from the analysis – but included in Section 4.2 analysis. The observed TOPEX to WSGT via TDRS performance is given in Table 4.4.

Table 4.4 TOPEX to WSGT via TDRS Return Link Performance

Passtype/Channel	BER	Margin		Passes
		Actual ¹	Expected ²	
LOAD_MRO_SA (I-Channel)	4.72×10^{-8}	1.6 dB	13.4 dB	61
LOAD_MRO_MA (I-Channel)	1.06×10^{-8}	2.0 dB	9.3 dB	65
MONITOR_MA (Q-Channel)	2.95×10^{-10}	2.7 dB	9.3 dB	572
PBK_SA				
• I-Channel	8.09×10^{-8}	1.5 dB	13.4 dB	153
• Q-Channel	4.53×10^{-11}	3.1 dB	5.3 dB	536

¹Based on Eb/No = 4.5 dB for Viterbi decoding (soft detection)

²Based on link budgets

The difference in the actual and expected performance can be explained by non-RF losses – i.e. phase noise, AM/PM, synchronization losses, etc. at WSGT. Based on the formalism presented in Reference 2, WSGT appears to have a performance and error floor for I-Channel performance (PN code "on") at $\sim 1 \times 10^{-8}$, a performance and error floor for Q-Channel performance (PN code "on") at $\sim 1 \times 10^{-10}$; and, a performance and error floor for Q-Channel performance (PN code "off") at $\sim 1 \times 10^{-11}$. Based on the above, during the LOAD_MRO_SA/MA and PBK_SA I-Channel 16 kbps telemetry returns, WSGT displays the performance of a marginal system (Ref. 2 formalism). During the MONITOR_MA Q-Channel return passes, WSGT displays

performance of a better than marginal system. During the PBK_SA Q-Channel 512 kbps playback, WSGT displays performance closer to a "well-equalized practical system" (Ref. 2 formalism).

4.4 APPLICATION: PLAYBACK (SCIENCE RECOVERY) PERFORMANCE

The TOPEX overall playback performance (accounting for tape recorder performance, TOPEX to WSGT via TDRS link performance and NASCOM performance) is given in Table 4.5. This is based on days 1993-081 to -339 data.

Table 4.5 TOPEX Overall Playback (Science Recovery) Performance

Link Element	BER (Average)	Comments
Tape recorders	3.0×10^{-9}	Limited by T/R-B Performance
PBK_SA Link Performance (TOPEX to WSGT via TDRSS)	4.54×10^{-11}	Limited by WSGT Equipment
NASCOM Link	6.24×10^{-9}	Limited by TDMA Configuration
Overall Performance Playback	$\sim 9 \times 10^{-9}$	Limited by NASCOM Performance

4.5 APPLICATION: STGT VERSUS WSGT PERFORMANCE

The second TDRS Ground Terminal (STGT) will replace the White Sands Ground Terminal (WSGT) by the end of this year – 1994. TOPEX has performed limited tests using the STGT and TDRS_S(E). These tests included PBK_SA passes – scheduled for days 1994-045, -059, -087, and -095 and modified MON_MA (I-Channel data returned) tests – scheduled for days 1994-043, -057, -080, -084, -089, -095, -097 and -108. STGT BER results are given in Table 4.6 and are compared to WSGT performance.

Table 4.6 STGT vs WSGT Performance

Passtype/Channel	BER Performance	
	STGT	WSGT
PBK_SA/Q	3.33×10^{-8}	4.53×10^{-11}
PBK_SA/I	1.06×10^{-7}	8.09×10^{-8}
MONITOR_MA/I	9.19×10^{-8}	1.06×10^{-8} *

* Based on LOAD_MRO_MA performance

Based on the limited STGT tests, STGT displays the performance of a marginal system (Ref. 2 formalism) with a performance and error floor at $\sim 1 \times 10^{-8}$. Playback performance (PBK_SA Q-Channel) for STGT appears to be substantially worse than WSGT's performance. The original STGT results are documented in Reference 3.

5.0 CONCLUSION

Jacques Cousteau stated⁴ that "remote sensing, backed by telemetry, is undoubtedly one of the fastest developing, as well as one of the most promising technologies of our times...". By taking advantage of spacecraft telemetry and ground data system structures and protocols, TOPEX/Poseidon has successfully implemented an end-to-end information systems (EEIS) monitoring program that allows the project scientist to take full advantage of the radar altimetry and POD data obtained.

ACKNOWLEDGEMENT

The research described in this paper was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

FOOTNOTES/REFERENCES

¹ BER calculation equating frame error rate to BER was presented at the "TOPEX/Poseidon Telecom Subsystem Review - Part II", July 1992, B. Calanche. Based on the expected link margins, one CRC error is equated to one bit error. This assumption falls apart for the tape recorder channel. Even then, a pseudo BER can be calculated based on equating 1 minor frame loss to a single CRC error, since the science people exclude minor frames with a CRC flag set.

² "Telecommunications Measurements, Analysis and Instrumentation", Dr. Kamilo Feher, Prentice-Hall, Englewood Cliffs, New Jersey, 1987.

³ JPL Interoffice Memorandum (IOM) 3395-94-09, "Second TDRSS Ground Terminal (STGT) Tests: Telecom Performance Through Day 1994-108", April 19, 1994, B. Calanche. (Internal Memo)

⁴ Excerpt from Captain Jacques Cousteau's speech, "A Time to Choose", to the Eurospace Society, Monaco, October 1975.

Communication Network for Decentralized Remote Tele-Science during the Spacelab Mission IML-2

Dr. Uwe Christ, Dr. Klaus-Jürgen Schulz, Marco Incollingo
European Space Agency (ESA), European Space Operation Center (ESOC)
Robert-Bosch-Str. 5, 64293 Darmstadt
Tel.: +49-6151-90-0, Fax: +49-6151-90-3046

ABSTRACT

The ESA communication network for decentralized remote tele-science during the Spacelab mission IML-2, called Interconnection Ground Subnetwork (IGS), provided data, voice conferencing, video distribution/conferencing and high rate data services to 5 remote user centers in Europe. The combination of services allowed the experimenters to interact with their experiments as they would normally do from the Payload Operations Control Center (POCC) at MSFC. In addition, to enhance their science results, they were able to make use of reference facilities and computing resources in their home laboratory, which typically are not available in the POCC. Characteristics of the IML-2 communications implementation were the adaptation to the different user needs based on modular service capabilities of IGS and the cost optimization for the connectivity. This was achieved by using a combination of traditional leased lines, satellite based VSAT connectivity and N-ISDN according to the simulation and mission schedule for each remote site. The central management system of IGS allows to minimize the staffing and the involvement of communications personnel at the remote sites. The successful operation of IGS for IML-2 as a precursor network for the Columbus Orbital Facility (COF) has proven the concept for communications to support the operation of the COF decentralized scenario.

1. INTRODUCTION

For the Columbus Orbital Facility (COF) as part of the International Space Station a distributed European ground segment has been defined. A dedicated network, the Interconnection Ground Subnetwork (IGS) of ESA/ESOC, is in development to provide the necessary communication services to connect the scientists in their remote centers with their experiments in space. In the preparation phases a number of space missions are supported in Columbus-like pre-cursor scenarios to provide proof of concept for the anticipated tele-science/tele-operations infrastructure.

In April 1993 ATLAS-2, a Spacelab mission, was launched for which IGS provided the communications support to perform remote operations of the two European payloads from the Principal Investigators (PI) site in Brussels. These services included data exchange, voice and video conferencing. For the first time an experimenter, who remained in his home base, exercised full control over his experiment aboard Spacelab. After this successful demonstration

in July 1994 five European remote user centers participated in a remote experiment operations scenario in the international Spacelab mission IML-2 in which they were able to monitor and adjust their experiments by commands directly from their home bases. This scenario was again based on IGS. The approved remote operations sites in Europe were: CADMOS in Toulouse (France), DUC in Amsterdam (the Netherlands), MARS in Naples (Italy), MUSC in Cologne (Germany), SROC in Brussels (Belgium).

IGS communications uses the Marshall Space Flight Center (MSFC) in Huntsville, Alabama as the relay center to the shuttle and provides the scientists live access to activities aboard the Spacelab in form of video, voice and data transmissions. From MSFC the data are sent to ESA/ESOC in Darmstadt by undersea cable, where the IGS central node and the network management is located. From here the data was routed to the European sites of IML-2.

2. COMMUNICATIONS SCENARIO FOR IML-2

The communications scenario for IML-2 is depicted in figure 1. Communications from the Spacelab to MSFC was carried by the NASA Tracking and Data Relay Satellite System (TDRSS) via the White Sands based ground terminal. At the Huntsville Operations Support Center (HOSC), one of the MSFC facilities, ESA has established an IGS Relay which represents the network relay for Spacelab communications and mission operations to Europe.

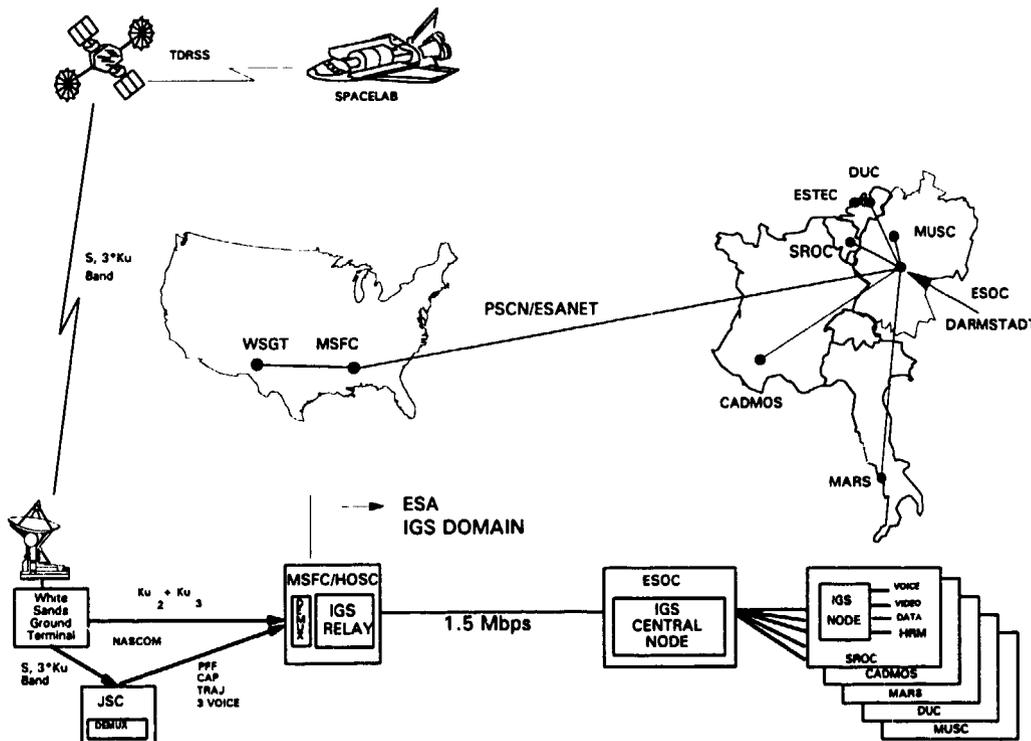


Figure 1 IGS communications scenario for IML-2

For cost reasons in this precursor demonstration the available resources of the general purpose networks of NASA (PSCN) and ESA (ESANET) were chosen as carrier providers for the trans-Atlantic link. At ESOC the complementary IGS central node terminated this 'trunk' and provided the connectivity of the integrated services system for voice, video and data services to the remote sites in Europe. At all remote sites IGS nodes were installed rendering the end-to-end network management capabilities which are required for the reliable operation of networks of this complexity.

For IML-2 the implementation of the **voice** system had to be based on the extension of the NASA Huntsville Voice Data System (HVoDS) with its proprietary formats and signaling. The remote sites can access up to 32 voice loops at the same time.

The analog **video** signals (NTSC at NASA and PAL at the European sites) are digitized and compressed to 384 kbps. Besides simultaneous distribution of on-board video to multiple sites a digital video multipoint control unit at ESOC provides the capability to support any video conferencing configuration between the remote sites, ESOC and NASA.

The IGS frame relay network provides the connectivity for **data** exchange between the workstations which are connected to the LANs at the different sites. This includes the HOSC LAN from where data bases and planning data can be accessed. Since the latter also is interconnected via several other networks including TDRSS with the on-board LAN of Spacelab, the European remote operators were able to directly communicate with their experiment in space, i.e. to send commands and to receive 'low rate science data'.

For the distribution of experiment **high rate data** which were multiplexed aboard Spacelab, IGS provided a special communication service as detailed in paragraph 3.2.

Since the IGS **network management system** at ESOC provides the capability to remotely monitor and control all remote IGS nodes, no communications expertise is required at the remote sites and maintenance interventions could be reduced to a minimum. These are conducted on request and under remote guidance from the communications operations team (IGS Control) at ESOC. The complement team of IGS Control on NASA side is HOSC Comm Control.

Since a low cost approach had to be taken for the remote operations support of the IML-2 mission no backup systems or redundant communication links were foreseen. A reduced prime investigator team for each center was present at MSFC to take over experiment operations in case of a communications failure.

2.1 Implementation Phases

Four major tests preceded the mission: CPS-1, CPS-2, JIS-1 and JIS-2. The last test defined the configuration freeze for the mission. Since only limited capabilities for the remote operations support during some of these tests were required the connectivity cost was optimized according to the actual needs. Three major technologies were used to achieve this:

1. Leased lines with initially lower data rates which were later increased to the bandwidth as required for the mission,
2. Satellite based connectivity with mobile ground stations (VSAT) that provided on-demand establishment of links with fixed data rate,
3. On-demand N-ISDN connectivity using inverse multiplexing techniques in order to combine multiple B-channels to higher aggregates as a substitute for leased lines.

The commonality of all of these types of connectivity is that they interface to the switching system via framed E1 or T1 interfaces, which is a software configurable interface to allow data rates between 64 kbps and 1936 kbps in increments of 64 kbps.

For cost saving reasons the initial bandwidth requirements were reduced or in other cases on-demand connectivities were requested which in later phases were replaced by leased lines. Figure 2 provides an overview of the implementation phases.

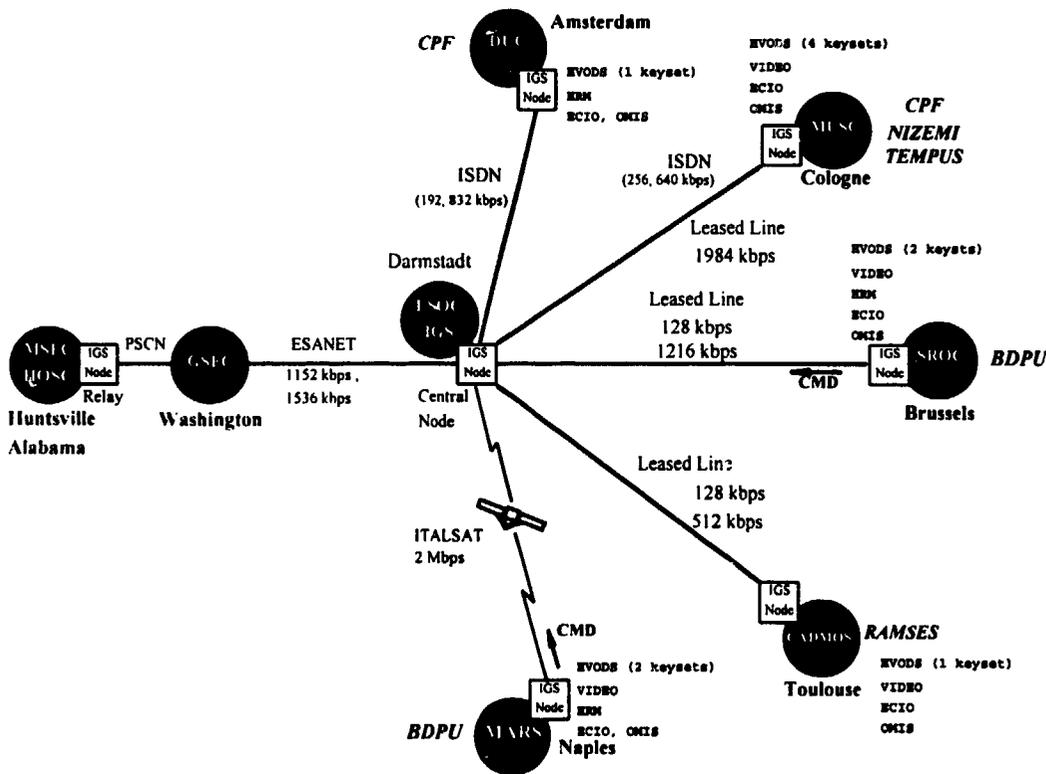


Figure 2 Communication links and implementation phases

3. COMMUNICATION SERVICES

The communication services provided by IGS for IML-2 were derived from the remote operations requirements of each site which are listed below:

1. Low rate housekeeping telemetry reception (ECIO)
2. Low rate experiment telemetry reception (ECIO)
3. High rate experiment telemetry reception of different data rates (HRM)
4. Fixed telecommand sending
5. Variable telecommand sending
6. Access to operations and management information system (OMIS)
7. Voice conferencing
8. Video reception of science video from the experiments aboard Spacelab
9. Video reception of NASA Select to follow the launch and other activities
10. Video conferencing for the science operations planning group (SOPG) meetings

These remote operation requirements resulted in the following IGS services:

1. Data services over IP and DECnet encapsulated in frame relay for 1, 2, 4, 5, 6
2. High Rate Mux Service over switched circuits for requirement 3
3. Voice conferencing service over switched circuits for requirement 7
4. Video distribution and conferencing service for requirements 8, 9, 10
5. A remote management access service over X.25.

These IGS services were supported by an integrated switching system that allows to combine circuit and packet traffic, i.e. frame relay and X.25.

The following subchapters describe the IGS services highlighting the management domains.

3.1 Data Services

The IP and DECnet data services are realized by a multi-protocol router network over a frame relay service provided by the integrated switching system. The management domains are depicted in figure 3. The routers are managed by IGS Control. The end systems connected to the various LANs are managed by their users.

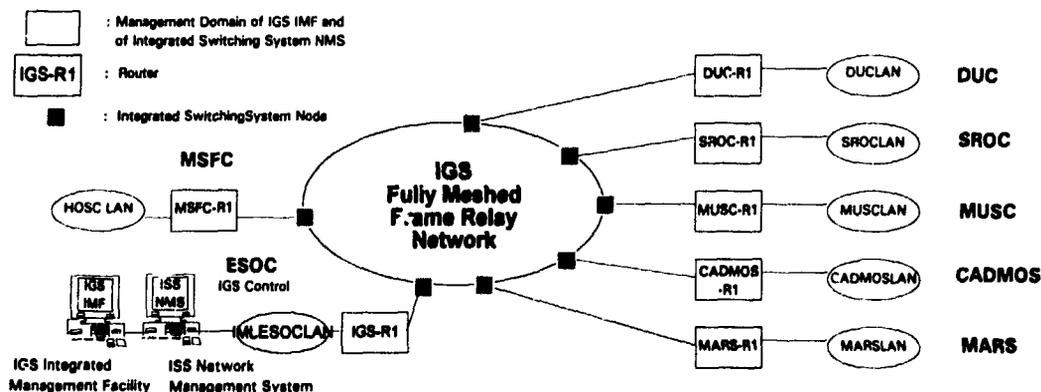


Figure 3 Data Services

3.2 High Rate Mux Service

The high rate mux service is realized by special rate adapters that convert the "odd" data rates of 307.2 and 400 kbps to multiples of 64 kbps that can then be routed as circuits through the integrated switching system.

Figure 4 depicts the management domains. IGS Control provides the switching of the circuit through the switches to the final destination. The rate adapters are managed by HOSC Comm Control.

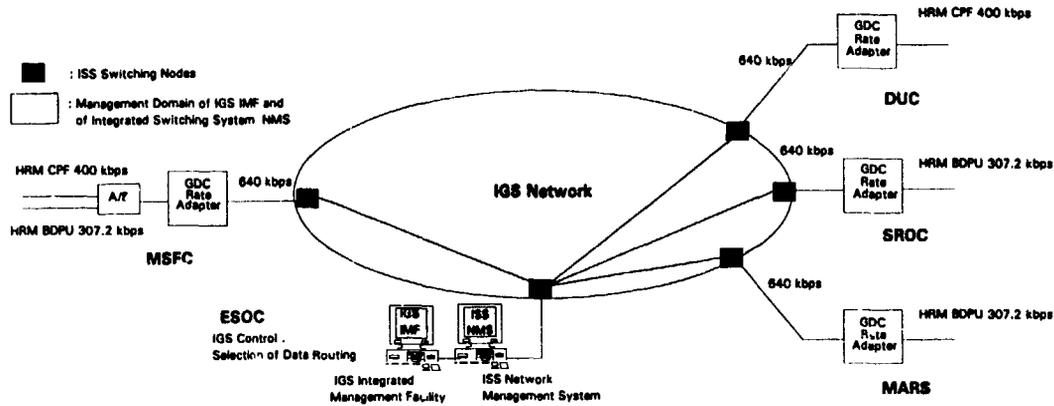


Figure 4 High Rate Mux Service

3.3 Voice Conferencing Service

The voice conferencing service is realized by special dual trunk adapters (DTA) and dual phone adapters (DPA) of the NASA HVoDS system. DTA and DPA pairs are connected by circuits through the integrated switching system.

Figure 5 shows the management domains. The HVoDS DTAs, DPAs and attached keysets are managed by HOSC Comm Control. IGS Control ensures the routing of the circuits through the switches.

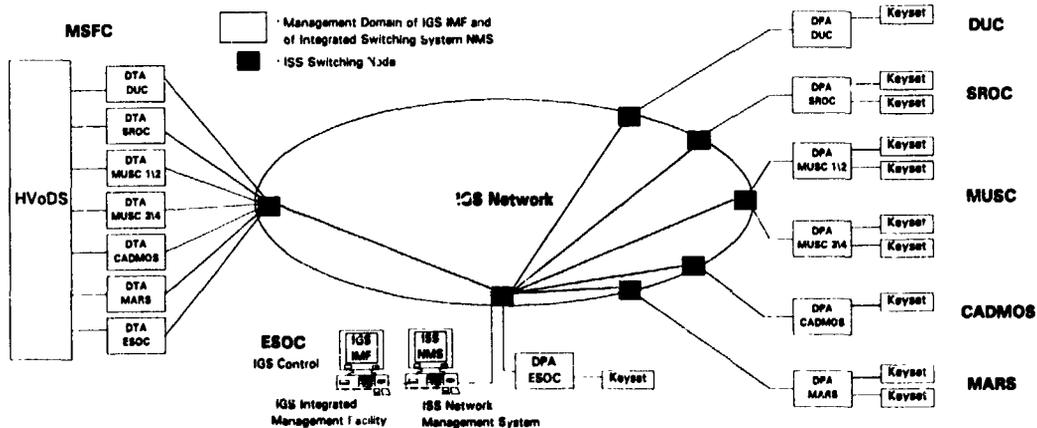


Figure 5 Voice Conferencing Service

3.4 Video Distribution and Conferencing Service

The video distribution and conferencing service is based on video codecs performing H.261 coding and a digital video multipoint control unit. The digitized video is routed as circuits through the main switching system. Figure 6 shows the management domains. The switching of the video streams is conducted from IGS control. The video input from NASA's Huntsville Video Data System (HViDS) is managed by HOSC Comm Control.

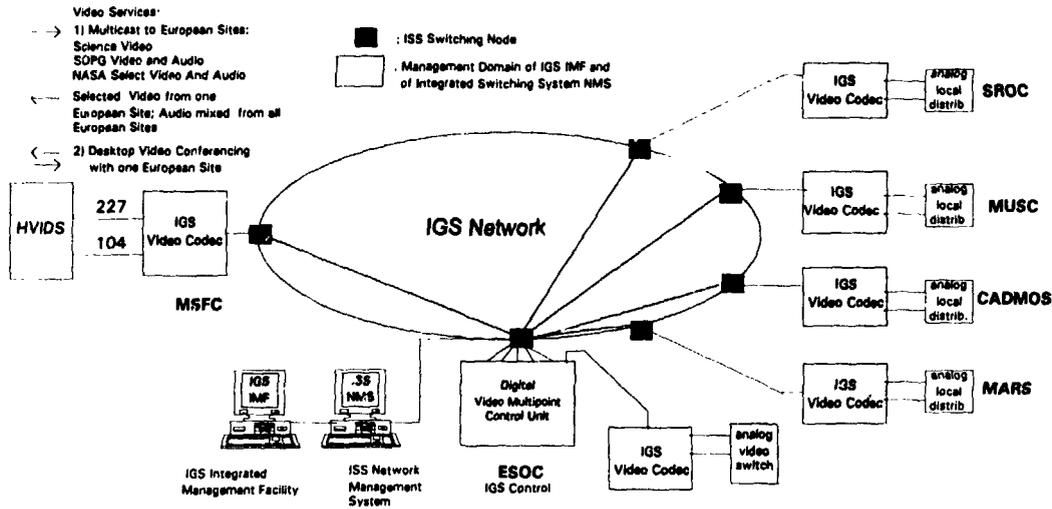


Figure 6 Video Distribution and Conferencing Service

3.5 Remote Management Access Service

The remote management access service is realized over a X.25 service provided by the integrated switching system network connecting packet assemblers/disassemblers (PADs). The management domains are depicted in figure 7. The PADs are managed as part of the main switching system.

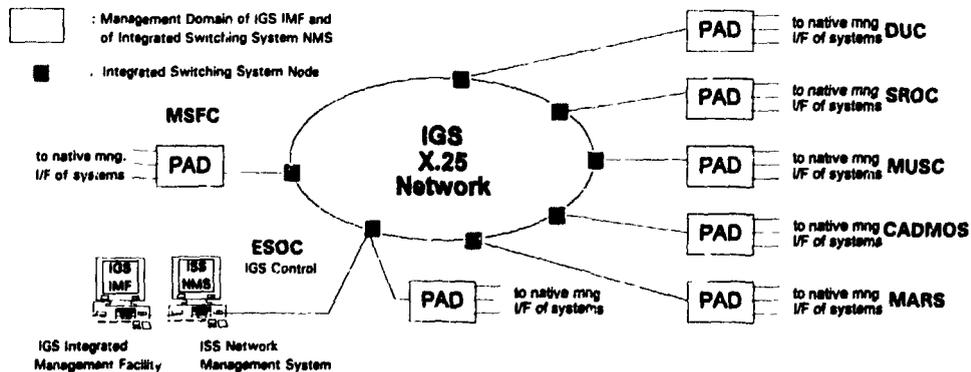


Figure 7 Remote Management Access Service

4. NETWORK MANAGEMENT

The integration of the communication services has been accomplished in two ways: first, at level of transfer, in the sense that the different packet and circuit services are multiplexed together over the link resources and routed through the same integrated switching nodes; second, at the level of network management, because all communications services and the systems which provide them are managed by the same centralized management platforms.

Two management platforms are used: the integrated switching system Network Management System (NMS), and the IGS Integrated Management Facility (IMF). The integrated switching system NMS is the proprietary management system of the core switching nodes on which the network is based. The IGS IMF is a management platform based on an expert-system, which was customized for IML-2 to integrate the management of the heterogeneous subsystems used in the IGS in a single network management system.

The integrated switching system NMS covers all areas of network management for the switching nodes and bases itself on a distributed S/W architecture, thus allowing basic processing of the management information already in the nodes, limiting the traffic on the network trunks to 3-4 kbps of packet bandwidth. Information is displayed on a window-based MMI and colors are used for status indication.

The management domain of the IGS IMF includes the video codecs, the digital video multipoint control unit, and the ISDN inverse multiplexers. These systems do not offer a standard management protocol and are accessed remotely through their native control interfaces.

Knowledge bases in the IMF have been defined for those systems, based on experience gained and with emphasis on the specific use for IML-2. The IMF collects events from all managed systems, including the switching nodes, and evaluates them in a reasoning process, showing correlation between network problems, thus representing a tool to ease problem solutions. Sequences of multiple commands and timed actions are realized as single operations to simplify operator activities. Routers are also managed by the IMF through standard SNMP protocol.

The key aspect of the management architecture that has been described is the centralization of the IGS operations, both for routine and trouble shooting operations.

During routine operations considerable in-service monitoring capabilities allow to exercise constant performance evaluation of the link resources and the services provided, in order to timely react in case degradations are observed.

Trouble shooting or out-of-services network management operations are also conducted in a centralized manner. Complete reconfiguration of faulty systems can be conducted from ESOC, provided that a communication link e.g. a dial-up back-up link with the remote site exists. Loop-backs can be initiated from ESOC in various points of the network for fault isolation. Transmission of test frames and subsequent checking can be also performed from any node

location, eliminating the need and the complications of installing test equipment at the remote sites to trace down difficult problems.

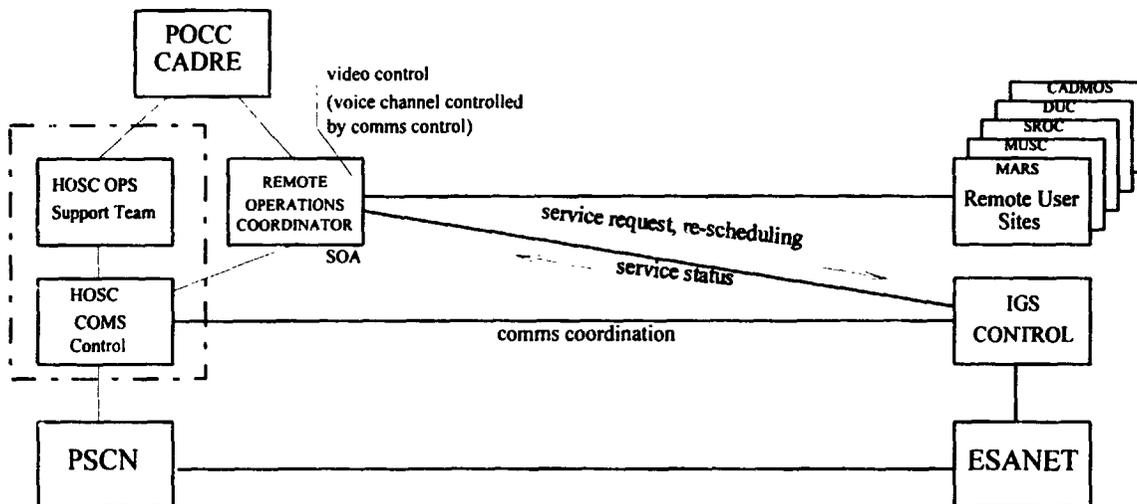
Support at the remote sites is required in principle only for hardware replacement.

5. OPERATIONAL ASPECTS

The timeline defining the activation of experiment aboard of Spacelab and the duration was scheduled well in advance of the IML-2 mission. From this overall timeline the remote operations timeline was derived which in principle represents the scheduled communication service requirements for the operation of IGS. The planned operational activities were:

- switching different high rate science data to either DUC, SROC or MARS,
- configuring the ISDN link to DUC into low- or high-data rate mode,
- distribution of NASA select video to the remote user sites,
- configuration of video conferences on request.

Unplanned on-board experiment or resource failures require real-time changes of the timeline for communications operation.



SOA Science Operations Area
 REMOPS Remote Operations Coordination

Figure 8 Nominal communications operation

Figure 8 describes the nominal communications operation scenario for IML-2. The IGS operations team (IGS Control) monitored and configured the network by means of the integrated network management system as described above. IGS Control was in permanent contact with the remote operations coordinator who resided during the mission in the science operations area at MSFC via the voice conferencing service. The remote operations coordinator issued requests to IGS Control to perform service changes and received reports on the service status. Some service changes required the support of the HOSC Comm Control, e.g. provision of Spacelab high rate data flow to the IGS Relay at MSFC. For this and similar reasons IGS Control remained in permanent contact with its MSFC complement.

IGS Control permanently monitors the performance of all IGS resources and services and informs the remote operations coordinator on any identified or potential problem. Interfaces to the trunk providing networks PSCN and ESANET are activated only in case carrier problems are identified. Trouble shooting and failure close-out require a close cooperation between the above identified operational positions.

CONCLUSIONS

The successful operation of IGS for IML-2 has demonstrated that decentralized remote telescience can be performed in a reliable and cost effective manner. The modular service capability of the IGS network allows easy adaptation to different user needs. The minimization of the connectivity cost, which is the major cost driver for remote telescience, was achieved by phased implementations and employing the optimum connectivity techniques. The approach of central management of IGS has proven to be a big advantage allowing to minimize staffing and the required communications expertise in the remote sites.

IGS today includes state-of-the art technologies. The strategy which is consequently followed will ensure the direct migration capabilities into the future connectivity techniques e.g. B-ISDN whenever these services demonstrate to be more cost effective.

Subsequent missions which will reuse the proven capabilities of IGS are the Spacelab ATLAS-3 mission October 1994 and other follow-on missions. Also for EUROMIR 95 a scenario based on the available resources of IGS is investigated.

Acknowledgments

The authors would like to thank Dr. W. Frank and Dr. C. Reinhold from Columbus Operation, R. Joennson from Columbus Utilisation, as well as J. C. Degavre from the ESA/ESTEC technical directorate for their constant support as well as our NASA colleagues from HOSC engineering and operation and the IGS team for their high motivation.

EARTH OBSERVING SYSTEM (EOS) COMMUNICATION (Ecom) MODELING, ANALYSIS, AND TESTBED (EMAT) ACTIVITY

**Vishal Desai
Ecom Project Engineer
NASA Communications Division
NASA/Goddard Space Flight Center
Greenbelt, MD 20771**

ABSTRACT

This paper describes the Earth Observing System (EOS) Communication (Ecom) Modeling, Analysis, and Testbed (EMAT) activity performed by Code 540 in support of the Ecom project. Ecom is the ground-to-ground data transport system for operational EOS traffic. The National Aeronautic and Space Administration (NASA) Communications (Nascom) Division, Code 540, is responsible for implementing Ecom.

Ecom interfaces with various systems to transport EOS forward link commands, return link telemetry, and science payload data. To understand the complexities surrounding the design and implementation of Ecom, it is necessary that sufficient testbedding, modeling, and analysis be conducted prior to the design phase. These activities, when grouped, are referred to as the EMAT activity. This paper describes work accomplished to date in each of the three major EMAT activities: modeling, analysis, and testbedding.

1.0 OVERVIEW

NASA has begun the implementation of the EOS as part of the Mission to Planet Earth (MTPE) initiative. The MTPE initiative is NASA's main contribution to the interagency Global Change Research Program. EOS supports this initiative by providing a capability to observe and study Earth from multiple, low-earth orbiting spacecraft. In addition, EOS also provides the capability to collect, process, and distribute the observed data to the world wide scientific community.

EOS consists of three main components: EOS Space Measurement System (EOSSMS), EOS ground system, and the EOS Scientific Research Program. The EOSSMS is composed of a series of spacecraft that provide remote sensing capabilities to collect scientific data relating to the Earth's atmosphere, oceans, and land surface. This scientific data is then processed and distributed among the various members of the user community by the EOS ground system. The EOS ground system also provides the data acquisition, command generation, and delivery capabilities. The EOS ground system includes the EOS Data and Information System (EOSDIS) and GSFC-managed NASA institutional support elements, as well as elements from several other non-NASA sources, including U.S. Government agencies, user support facilities, and international partners.

EOSDIS, a key element of the EOS ground system, is a geographically distributed data information system that supports the operation and management of in-orbit EOS spacecraft. EOSDIS consists of EOS Data and Operations System (EDOS), Ecom, EOSDIS Core System (ECS), Science Computing Facilities (SCFs), and NASA Science Internet (NSI). Figure 1, shown on the next page, presents the overall EOS data systems architecture from Ecom's perspective.

The primary objective of Ecom is to provide connectivity between EDOS sites, and to support data transfers between EDOS sites and Distributed Active Archive Centers (DAACs). Ecom also interfaces with the ECS segments and the Mission Operations and Data Systems Directorate (MO&DSD) institutional systems to support Space Network (SN) scheduling and spacecraft orbit and altitude determination. In addition, Ecom interfaces with the Ground Network (GN), Deep

Space Network (DSN), and Wallops Orbital Tracking Station (WOTS) to support EOS contingency requirements.

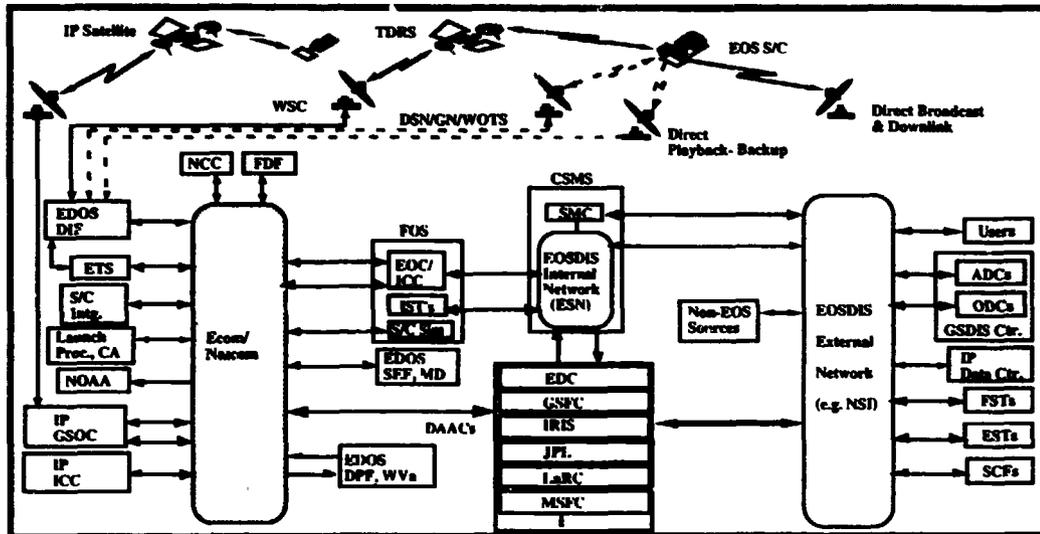


Figure 1 - EOS Data Flow

To understand the complex nature of the Ecom system and its interfaces, it was necessary to conduct sufficient investigation early in the project life cycle to resolve key implementation and technology issues. To assist in this investigation process, Nascom developed an in-house laboratory environment to perform the necessary modeling, analysis, and testbedding. This environment is referred to as the EMAT activity.

2.0 HISTORICAL PERSPECTIVE

In support of the Government Open Systems Interconnection (OSI) Profile (GOSIP) mandate, a Code 540 Research, Technology, and Operations Plan (RTOP) project, known as the Nascom OSI protocol (NOSIP) testbed, was formed in 1989. The purpose of the NOSIP testbed was to test the functionality and performance of the first four layers of the OSI protocol stack. Additionally, the testbed was to evaluate commercial off-the-shelf (COTS) equipment (e.g., hubs, routers) that implemented these protocols. From 1989 to 1993, exhaustive NOSIP tests were conducted in various local and wide area network configurations using different vendors' communications equipment. Results documenting the capabilities and limitations of these protocols and associated equipment were published in the semi-annual RTOP presentations and reports.

In 1992, a high level EDOS/Ecom requirements were formulated and provided to Nascom. Subsequent documents outlining Ecom requirements also became available in the following months. Accordingly, the need to address Ecom related issues became more pressing. After carefully weighing the available resources, Nascom decided to withdraw from the RTOP program in 1993 to support the Ecom project. However, due to the RTOP's direct applicability to several Ecom issues, the work, equipment, and staff were absorbed by the newly created EMAT activity.

The EMAT activities formally began at the beginning of fiscal year (FY) 1993, and will continue until Ecom's Critical Design Review (CDR), scheduled for May 1995. After CDR, EMAT work

will be incorporated under the Ecom testing activity, which will emphasize Ecom system testing and integration. The modeling, analysis, and testbedding work performed after the Ecom testing and integration phases will be performed via the Ecom Sustaining Engineering Facility (SEF).

3.0 OBJECTIVE

The objective of EMAT is to perform the necessary modeling, analysis, and testbedding to support the Ecom project design, test, and integration phases. To meet this objective, EMAT will:

- perform testing to verify if COTS equipment can meet Ecom requirements
- recommend candidate hardware and software elements for Ecom
- identify requirements beyond COTS' capabilities
- perform availability, error, and trade-off analyses
- use modeling tools to obtain the candidate wide area network (WAN) topologies
- conduct periodic market surveys and assessments.

4.0 ORGANIZATION AND RESPONSIBILITIES

The EMAT activities are performed by Nascom personnel and their System Engineering, Analysis, and Support (SEAS) contractor. EMAT personnel also perform joint testing with AT&T to help define Federal Telecommunications System (FTS) 2000 services that Ecom/Nascom can use in the future. Pertinent results obtained from this joint testing activity are incorporated with the EMAT results. The Ecom Project Manager, in conjunction with the Ecom Project System Engineer, determine the EMAT work priorities and scheduling.

5.0 REPORTING MECHANISM

The EMAT team periodically presents its results in the *EMAT Reports*. These reports have been issued prior to each major Ecom project review. To date, three separate *EMAT Reports* have been delivered to the Ecom project.

The *EMAT Reports* primarily consist of four major sections. The first three sections detail the work accomplished in the three major areas, namely, modeling, analysis, and testbedding. The last section of the *EMAT Reports* presents results, conclusions, and recommendations organized by issues (e.g., delay, throughput, error sensitivity). This section allows the authors to integrate all related results and arrive at an overall conclusion for each major issue.

6.0 MODELING

The purpose of the EMAT modeling effort is to address and resolve those areas of Ecom that are not feasible for testbedding. There are two such areas that have been identified by the EMAT team. They include WAN topology modeling and reliability, maintainability, and availability (RMA) modeling.

The WAN topology modeling effort is accomplished in two phases. In the first phase, Ecom and EDOS periodically update a jointly developed EDOS and Ecom traffic model (EETM). This model contains assumptions and equations that allow both projects to interpret and process information stated in the *EDOS and Ecom Requirements* document. The output of this model is a database containing traffic volumes and types between individual source-destination pairs. The second phase uses this information to develop candidate Ecom WAN topologies.

A COTS modeling tool is used to develop the WAN topologies. This tool uses the EETM results, along with Ecom design rules and assumptions as an input. The resultant output contains candidate Ecom topologies optimized by performance and cost. The optimization is primarily accomplished by the tool's algorithms and tariff database. Minimal manual intervention is required to complete the optimization process. The resultant, candidate topologies, along with the design rules and assumptions, are presented in the *Ecom System Design Specification*.

The EMAT WAN modeling effort has provided important insight into designing WANs using common carrier circuits. Additionally, it has provided EMAT with the necessary platform to model "what if" scenarios in support of the Ecom and EOS costing exercises. The output of these "what if" exercises have already resulted in modifying, and in some instances eliminating, the cost sensitive EOS requirements.

The second type of modeling performed in this phase is RMA modeling. An in-house developed modeling tool called Automated Reliability, Availability, and Maintainability (ARAM) is used to produce these models. These models allow EMAT personnel to analyze the RMA characteristics of various network configurations. The ARAM modeling tool requires as input, network configurations with accompanying RMA numbers for each element within that configuration. Operations concepts regarding manpower availability and staffing are also required as an input into the ARAM tool. The output of this modeling activity provides, at a minimum, the mean time to restore service (MTTRS) and availability numbers for the modeled network configuration.

Every candidate Ecom design undergoes RMA modeling to ensure that the MTTRS and availability requirements are met. The modeling duration is sufficient to simulate failures throughout the life of the Ecom system. The Ecom design identified in the *Ecom System Design Specification* has successfully undergone RMA modeling verification.

7.0 ANALYSIS

There are two kinds of analysis activities supported in the EMAT environment: mathematical analysis and tradeoff analysis. The mathematical analysis addresses the verification of theoretical performance via computational means. The tradeoff analysis focuses on conducting market surveys to perform technology versus cost studies and performance versus cost studies. No special tools are required to perform either of the above activities. However, to facilitate the research process associated with each, EMAT has established a reference library containing vendor brochures, equipment catalogues, textbooks, reference manuals, and conference papers.

The mathematical analysis conducted so far within the EMAT activity has focused on obtaining the communications protocol overhead and determining the Ecom error performance. The communications protocol overhead analysis was performed to determine the additional bandwidth required to transmit the individual user data streams. The obtained communications protocol overhead numbers were used to develop the equations appearing in the EETM.

The Ecom error performance analysis was conducted to characterize Ecom's real-time and science services. This analysis was performed on several end-to-end configurations. Past test results, current common carrier performance numbers, and theoretical calculations were used to obtain the pertinent results. These results, expressed as error free seconds and packet loss ratios, are included in the *Ecom Design Specification*.

In addition to conducting mathematical analysis, EMAT personnel periodically perform market surveys to keep abreast of new COTS products and technologies. These surveys are primarily accomplished by mapping vendor literature and presentations to Ecom requirements. The latest survey performed in EMAT focused on Asynchronous Transfer Mode (ATM) switches. This

survey was performed to initiate the procurement process to acquire three ATM switches. Over the past year, EMAT has also conducted market surveys to evaluate Network Management System (NMS) packages, T3 multiplexers, routers, and local area network (LAN) analyzers. These surveys usually culminate in informal product recommendation lists. Products appearing on these lists are then evaluated in the EMAT laboratory.

8.0 TESTBED

The purpose of the testbed portion of the EMAT activity is to create and maintain an operational laboratory, procure and/or borrow communications equipment, develop test scripts and scenarios, and evaluate the functional and performance characteristics of COTS communication equipment in candidate Ecom configurations. This purpose is accomplished in two phases.

The first phase addresses the development of an operational laboratory, as well as completion of low (0-10Mbps) and high (10-100 Mbps) speed tests using Transmission Control Protocol (TCP)/Internet Protocol (IP) and Transport Protocol (TP) 4/Connectionless Network Protocol (CLNP) protocols over Fiber Distributed Data Interface (FDDI) and Ethernet LANs. The necessary test equipment, workstations, and other required tools are procured in this phase.

The second part focuses on ATM and Synchronous Optical Network (SONET) testing. The objective in this phase is to evaluate COTS ATM switches and interfaces in multiple LAN-WAN configurations. The design, configuration, and integration of the Simple Network Management Protocol (SNMP) based COTS network management (NM) package is also addressed in this phase.

8.1 PHASE I TESTING

In the first phase of testing, the TCP/IP, TP4/CLNP, Ethernet, and FDDI protocols are used to characterize and evaluate the functionality and performance of COTS hubs, routers, and multiplexers. The COTS equipment is configured and tested in selected LAN-WAN-LAN configurations. Key functional and performance parameters collected and analyzed from these tests are described below:

- i) No Loss Point: Determines the maximum single stream data rate that the hub, router, or multiplexer can sustain without losing data.
- ii) Loss: Describes the pattern and amount of data loss at rates above the No Loss Point.
- iii) Service Restoral: Finds the time and effect of restoring connectivity due to either a hub, router, multiplexer, or link failure.
- iv) Delay: Provides delay incurred by a packet due to either transport delay or equipment latency.
- v) Filters: Characterizes the effect on No Loss Point due to packet filtering.

To obtain the above parameters, exhaustive testing is performed using different packet sizes, test durations, and vendor equipment. As a standard operating procedure, short term (1- 5 min.) tests are performed initially to prove the test functionality and to obtain a preliminary No Loss Point. Long duration (1-36 hr.) tests are then performed to obtain accurate results. To simulate a "real world" Ecom environment, the SNMP polling feature is enabled, the router's packet filtering option is selected, and the WAN simulators are programmed to simulate circuit delay and errors.

A typical Phase I test configuration is shown below in Figure 2. In this configuration, the LAN analyzer provides generation and capture capability of IP or OSI packets. The FDDI concentrators provide the connectivity between the analyzer and the two FDDI LANs. The routers reside on these LANs and are connected to each other via WAN simulators. Depending on the individual test scenario, variations to this configuration are made to include additional LAN analyzers, multiplexers, end systems, and Ethernet hubs.

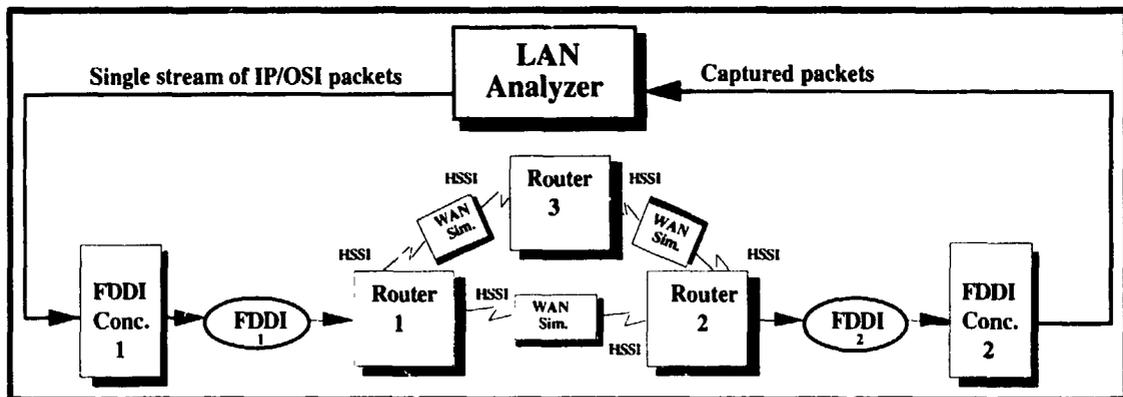


Figure 2 - Typical Phase I Test Configuration

The test results obtained to date in this phase have verified the capability of COTS equipment to process standard protocols at high data rates (1-25 Mbps) in Ecom specific configurations. The Ecom delay and service restoral requirements have also been validated. Additional testing still needs to be performed as newer equipment becomes available in the marketplace. Due to vendor agreements, the actual results are classified as sensitive material and, accordingly, are not included in this paper. Interested readers should contact the Ecom Project Manager to request these results.

8.2 PHASE II TESTING

In Phase II, the objective is to evaluate ATM switches to determine if they can be used for Ecom. Additionally, this phase addresses the integration and development of an SNMP NMS with a COTS Structured Query Language (SQL) database and trouble ticketing system. To meet these objectives, COTS ATM switches and NMS packages are purchased/borrowed, tested, and evaluated in the EMAT laboratory.

8.2.1 ATM TESTING

The ATM switch evaluation began in April 1994 with the acceptance of three, COTS ATM switches. After resolving initial switch power-on configuration issues, actual testing work began. Specific issues investigated included switch failover, circuit failover, bandwidth management, cell loss, switch management, and router/switch integration.

The switch failover tests addressed the redundancy features of the switches and the capability to automatically transfer to a back-up switch. The circuit failover tests explored the re-establishment of Permanent Virtual Circuits (PVCs) in the event of a failed common carrier circuit. The bandwidth administration management tests focused on call admission control, bandwidth allocation, traffic shaping, traffic policing, congestion control, and virtual circuit prioritizing mechanisms. The cell loss tests characterized cell loss occurring in the switches under various traffic loads. The switch management tests determined the vendor-specific management capabilities, as well as the ability to manage the switches via a third-party, multi-vendor SNMP based NMS. The router/switch integration effort interfaced the routers and workstations with

ATM switches to support end-to-end testing. The preliminary results obtained from these tests are documented in the ATM test reports.

The generic ATM testing configuration used to resolve ATM issues is shown in Figure 3. In this configuration, three ATM switches are connected to each other on the WAN side via DS-3 channel simulators. On the user side, these switches are connected to workstations, routers, and ATM analyzers. This configuration is modified depending on the specific issue that is being investigated.

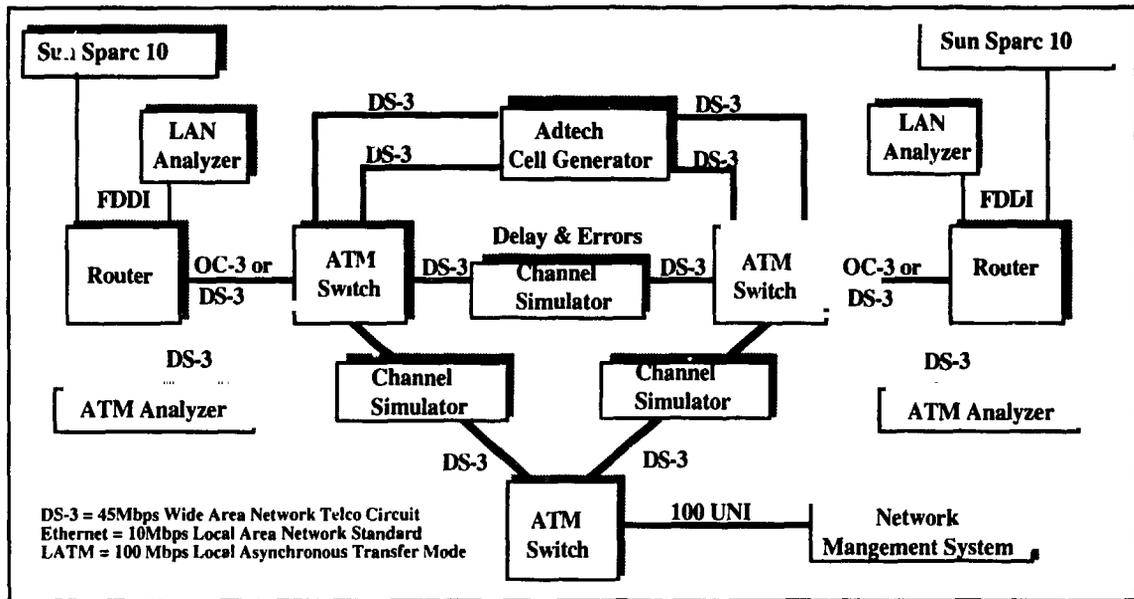


Figure 3 - ATM Test Configuration

The testing and evaluation of the EMAT procured switches is nearly complete. Work has already begun to evaluate other vendors' switches. As ATM technology and associated products mature and become readily available in the marketplace, they will be brought in and evaluated in the EMAT laboratory to ensure compliance with Ecom requirements. The test results obtained from these evaluations will be documented in future ATM test reports.

8.2.2 NMS TESTING

EMAT is currently prototyping a state-of-the-art, integrated NMS. The objective of this effort is to obtain an integrated NMS that extracts network health/status information, processes this information for alarm conditions, stores and reports this information, tracks fault conditions, and sends selected data to the EDOS NMS.

To accomplish this objective, COTS products are evaluated and development work is initiated. The COTS products evaluated include network management applications (NMApPs), SQL databases, and trouble ticket systems (TTS). The NMApPs packages configure and monitor the network. In addition, they collect the network's health/status data. The SQL database programs allow design and development of databases that facilitate data storage and retrieval mechanisms. The TTS provides an automated trouble ticket generation, processing, and tracking capability.

The NMApPs, SQL database, and TTS are integrated in EMAT to obtain a complete NMS. This integration requires some in-house development work. This work focuses primarily on developing the interface between the NMApPs software and the SQL database, and between the NMApPs

software and the EDOS management system. Minimal development work may also be required to integrate the ATM management system with the NMAApps. Major portions of this integration work is either under way or completed. The results and evaluations obtained to date are presented in the *EMAT Reports*.

9.0 CONCLUSION

Over past few years, EMAT has provided an excellent environment in which to conduct modeling, analysis, and testbedding activities. The results of the EMAT activity have helped verify that Ecom requirements can be met via COTS equipment, allowed Ecom to identify unrealistic requirements, and enabled Ecom to characterize the performance associated with the design. This effort has not only helped Ecom and Nascom, but also other projects within the MC&DSD directorate. As communications technology evolves and newer and better products become available in the marketplace, EMAT will continue to provide the government with the capability to test and evaluate products, and thereby minimize risk, prior to the design and implementation of communication networks.

10.0 ACKNOWLEDGMENTS

The information appearing in this paper represents a body of work performed over the past several years by Ecom project personnel. Civil servants, in conjunction with SEAS contractors, have worked diligently to create and operate the EMAT environment. The author appreciates the assistance provided by these personnel. In addition, the author would like to specifically thank Steven Smith, Bradford Torain, John Steedman, and Felicia Grice of the Ecom project for providing their reviews and comments on this paper.

S-Band and Ku-Band Return Service Interference Between TDRSS Users ²¹ _{P-12}

Stanford Telecom
 7501 Forbes Boulevard
 Seabrook, Maryland 20706
 Author: Linda Harrell
 Phone: (301) 464-8900

Abstract

The Tracking Data Relay Satellite System (TDRSS) return service performance can be degraded by interference from another user when two or more spacecraft communicate with the same Tracking Data Relay Satellite (TDRS) at the same time. This paper describes the S-band and Ku-band return service self-interference environment expected in the 1996 - 2010 timeframe and shows the self-interference expected for selected TDRSS users based on Communications Link Analysis and Simulation System (CLASS) Automated Conflict Resolution System (ACRS) and Interference Monitor (IM) tools. The results show:

- a. which user links are susceptible to interference from other users,
- b. the interference statistics,
- c. whether or not interference can be avoided with appropriate interference mitigation techniques such as scheduling, cross-polarization, or Pseudorandom Noise (PN) spreading.

The analysis results enable Space Network (SN) managers to determine the impacts of self-interference on the TDRSS service availability. They also enable project offices to determine whether they should (a) select return service communications parameters, such as polarization and PN spreading, to minimize the probability of being impacted by self-interference, (b) try to schedule TDRSS support around other user spacecraft communications schedules, or (c) accept communication outages due to self-interference.

1.0 Analysis Approach

This analysis uses the CLASS ACRS [1, 2] and IM [1] software packages to assess the return link performance for selected TDRSS users in the presence of self-interference. The selected TDRSS users are Space Transportation System (STS), Bilateral Transponder System (BRTS), Earth Observing System (EOS), Extreme Ultraviolet Explorer (EUVE), Gamma Ray Observatory (GRO), Hubble Space Telescope (HST), Space Station Freedom (SSF), and Ocean Topography Experiment (TOPEX).

ACRS is an interference prediction tool designed to analyze communications problems arising from two or more spacecraft transmitting return links to the same TDRS simultaneously. ACRS is used in this analysis to calculate interference threshold angles. The interference threshold angle is the angle at the TDRS antenna formed by the vectors from the TDRS to each of the users as shown in Figure 1. It is defined as the minimum angle that provides sufficient antenna discrimination to ensure that the desired link achieves a 10^{-5} Bit Error Rate (BER) (10^{-4} for some STS links) in the presence of interference from another user. ACRS also provides BER performance curves as a function of interference levels which are useful in assessing why interference occurs between users.

IM is a software tool that calculates interference statistics between TDRSS users for a given interference threshold angle. The statistics include the percentage of time that interference occurs on average and in a worst-case week. IM predicts the orbital trajectory of two users over a 25-year period and calculates the probability of interference between these users for a given interference threshold angle. It assumes that both users communicate continuously and interference occurs whenever the angle formed by the vectors from the

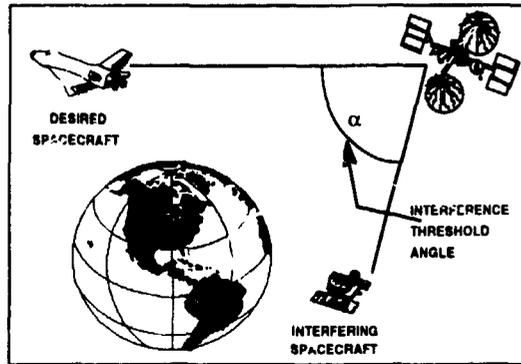


Figure 1. Interference Threshold Angle

TDRS antenna to each of the users is less than the given interference threshold angle. (The statistics do not consider passage through the Zone of Exclusion (ZOE).)

Figure 2 shows a block diagram of the interference analysis approach.

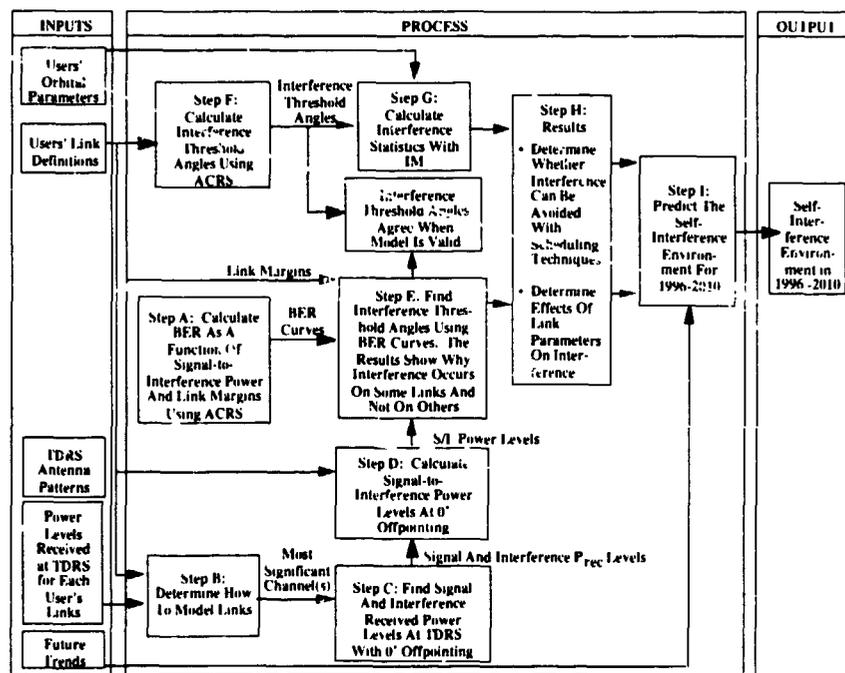


Figure 2. Analysis Approach

2.0 S-band Interference Analysis

There are 48 unique S-band links considered in this analysis. ACRS software can calculate the interference threshold angle for all possible combinations of desired and interfering links. However, one of the objectives of this analysis is to explain why some link combinations are not susceptible to interference (i.e. the interference threshold angle is zero) and other links require large offpointing angles (i.e. a large interference threshold angle). This is done with the use of BER curves that are plotted as a function of the signal-to-interference power ratio. It is desirable to show the link BER performance for all possible interfering and desired link combinations with a minimum number of curves. [3] has found that each link considered in this

analysis can be modeled as a Binary Phase Shift Key (BPSK) signal, without loss of accuracy. This reduces the number of BER curves needed in the analysis.

2.1 TDRS Antenna Discrimination

This analysis uses the antenna patterns defined in [4] and [5], which are shown in Figures 3 and 4.

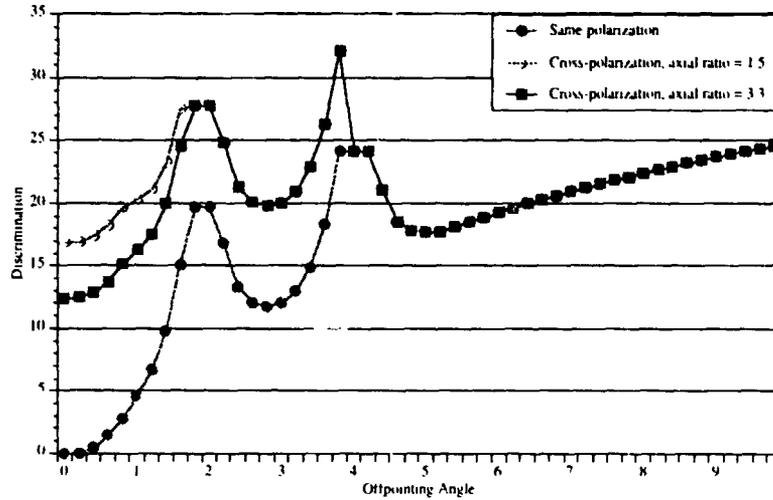


Figure 3. SSA Antenna Discrimination

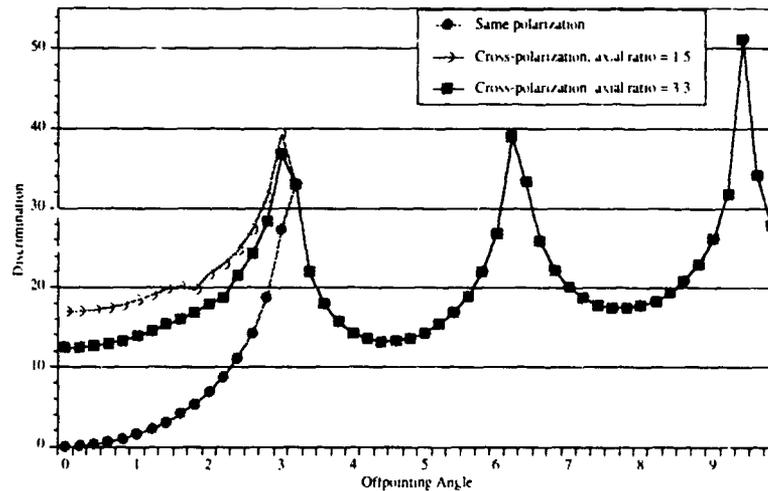


Figure 4. SMA Antenna Discrimination

2.2 BER Performance in the Presence of Self-Interference

The following indicators determine the BER of the desired link in the presence of an interfering link when both users operate at the same frequency:

- Received power level of the interfering signal relative to the desired signal.
- Symbol rate of the interfering link as compared with the desired link.
- Link margin of the desired link.

2.2.1 Performance of Nonspread Links

Figures 5 and 6 show the BER performance of nonspread links for different combinations of desired signal symbol rates relative to interfering signal symbol rates and different signal margins. These figures also have arrows pointing to the BER at 0° offpointing for several interfering and desired user link combinations. (The links are defined in [3].) The interference threshold angle is also shown for each combination.

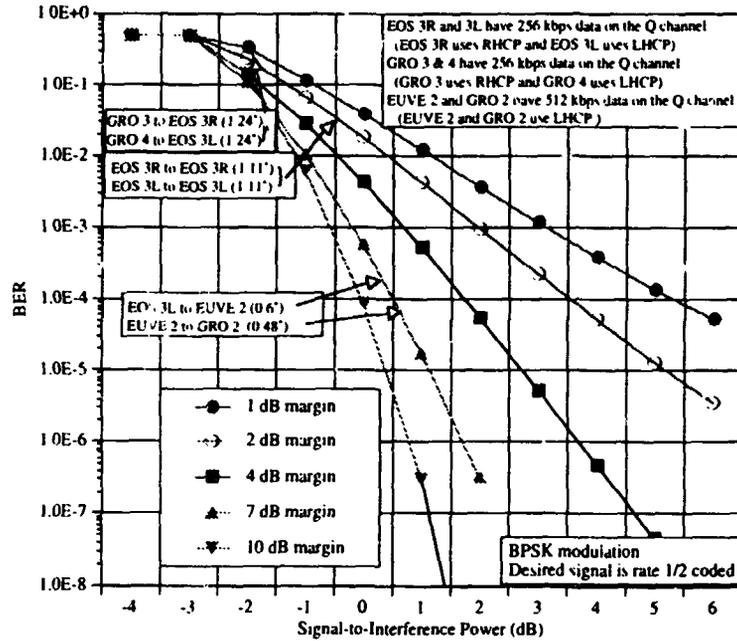


Figure 5. Performance of Nonspread Links Due to Interference When the Desired Symbol Rate is \geq the Interfering Link Symbol Rate

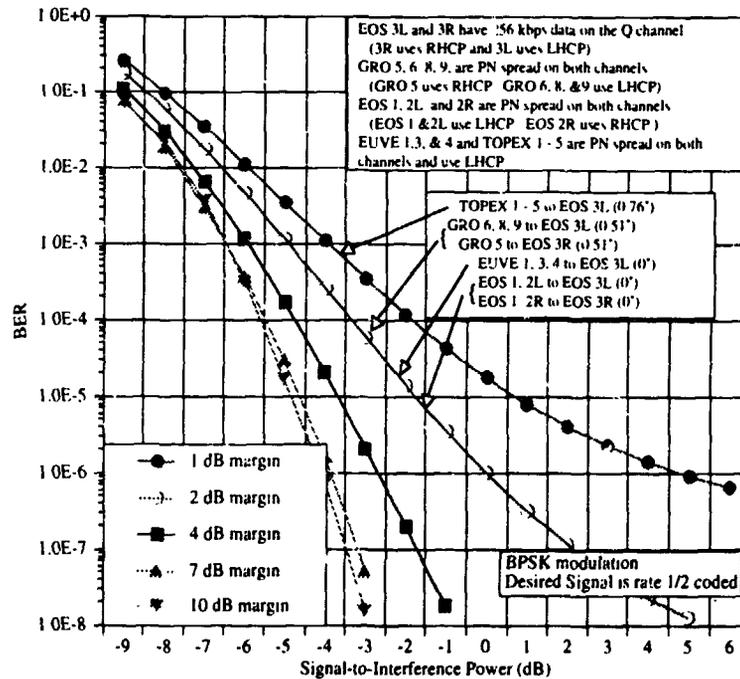


Figure 6. Performance of 256 kbps Nonspread Links With Interference from a PN Spread Link

Figures 5 and 6 show that there are three ways to improve the BER performance when the desired user's link is nonspread:

- Decrease the symbol rate of the desired signal relative to the interfering signal. Reducing the symbol rate of the desired signal reduces the desired signal bandwidth, which means that more of the interferer's power is filtered in the receiver. A comparison of Figures 5 (desired signal rate to interfering symbol rate ≥ 1) and 6 (desired signal rate to interfering symbol rate $\approx 1/6$) shows the effect that this filtering has on the BER performance.
- Increase the desired user's signal margin. This improves performance because increasing the user's signal margin reduces the sensitivity of the signal to noise.
- Increase the Signal-to-Interference power ratio.

2.2.2 Performance of PN Spread Links

The performance of PN spread links depends on the desired user's data rate and signal margin. It does not depend on the symbol rate of the interferer since the interfering signal has a symbol rate equal to 3 Mcps after the PN despreader. Figure 7 shows the BER performance of PN spread links for various data rates and signal margins on the desired link. This figure also has arrows pointing to the BER at 0° offpointing for several interfering and desired user link combinations. The interference threshold angle calculated by ACRS is also shown in parenthesis for each combination.

Figure 7 shows that there are three ways to improve the BER performance when the desired user's link is PN spread:

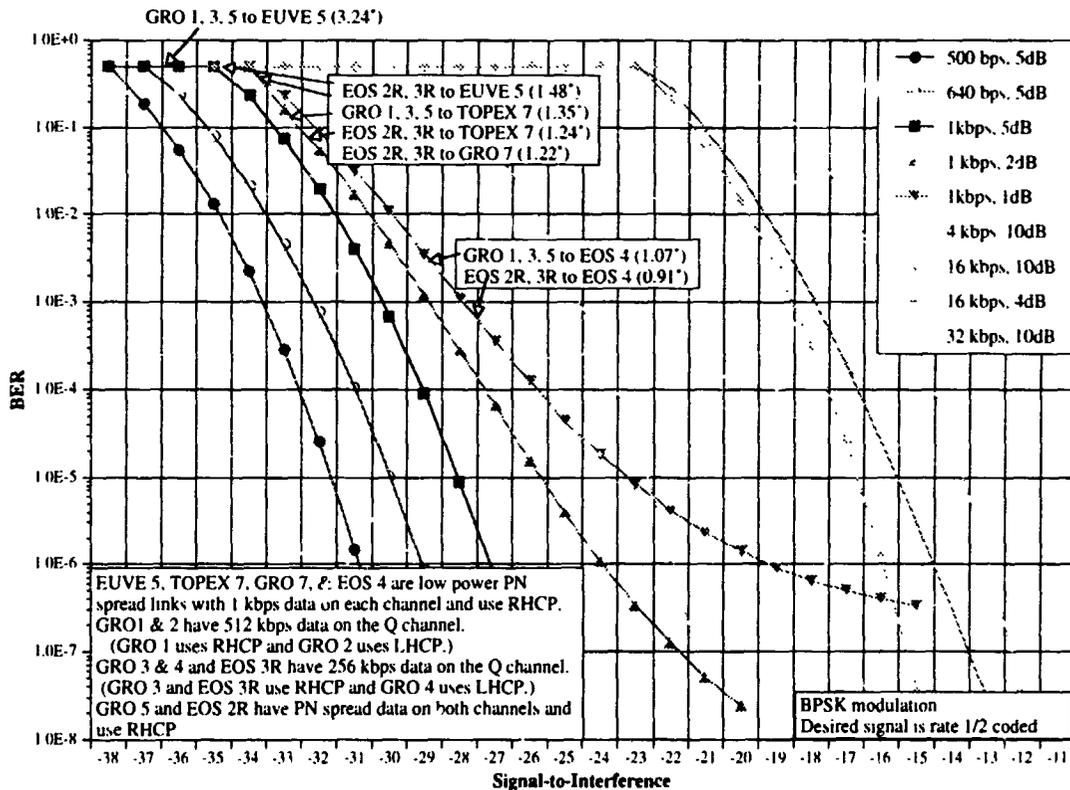


Figure 7. Performance of PN Spread Links in the Presence of Interference

- Decrease the data rate of the desired signal. At the receiver, despreading the desired signal spreads the interfering signal. Therefore, the interfering signal's symbol rate in the receiver is the chip rate of the desired PN spread signal, 3 Mcps, and the interfering signal's received bandwidth is 6 MHz. Reducing the data rate of the desired signal reduces the desired signal bandwidth, which means that more of the interferer's power is filtered in the receiver.
- Increase the desired user's signal margin.
- Increase the Signal-to-Interference power ratio.

2.2.3 Performance of STS Links

Figure 8 shows the BER performance of the 192 kbps STS link for various data rates on the interfering link. This figure also has arrows pointing to the BER at 0° offpointing for several interfering and desired user link combinations. The interference threshold angle calculated by ACRS is also shown in parenthesis for each combination.

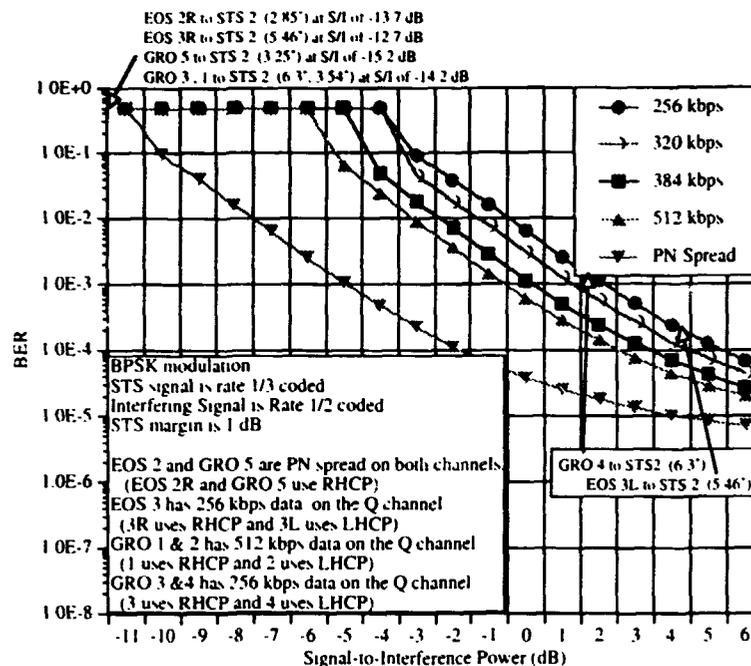


Figure 8. Performance of the STS 192 kbps Link in the Presence of Interference

2.3 Links that are Susceptible to Interference

ACRS results show that:

- Interference only affects the 2287.5 MHz S-band Single Access (SSA) links. The S-band Multiple Access (SMA) links are not susceptible to interference due to PN spreading.
- STS links are very susceptible to interference from other users' links via the High Gain Antenna (HGA) even if the other user is cross-polarized. This is because the STS signal is nonspread, STS power levels received at the TDRS are very low relative to other USAT HGA links, and STS signals have very low signal margins.
- The majority of self-interference occurs when the desired signal is nonspread and both the desired user and the interferer transmit on the HGA with the same polarization.

- d. Low power omni links with Right Hand Circular Polarization (RHCP) are susceptible to interference from GRO and EOS if these two users transmit on the HGA antenna with RHCP.
- e. EUVE's low power omni link is susceptible to interference from HST if HST transmits on 2287.5 MHz with the HGA antenna, even though EUVE link 5 and HST links are cross-polarized. This is because EUVE omni link has insufficient margin (-1 dB) and HST has the highest transmit power of all the users considered.

2.4 Self-Interference Statistics

IM simulations show that most of the interference between any two users at a time occurs less than 1.2% of the time on average or 7% of the time in a worst case week. There are only five link combinations which experience interference more often than this. Interference to STS from GRO, another STS, TOPEX and EOS can occur up to 80%, 20%, 17%, and 10% (respectively) of the time in a worst-case week and up to 14.9%, 1%, 10.8%, and 7.5% (respectively) on average. Interference to EUVE from GRO can occur up to 15% in a worst-case week and 6.1% on average.

2.5 Interference Mitigation Techniques

2.5.1 PN Spreading

A comparison of Figure 7 with Figures 5 and 6 shows that the signal-to-interference ratio required to achieve a 10^{-5} BER is much lower for PN spread signals than for nonspread signals. Therefore, PN spreading is a very effective mitigation technique. In fact, none of the PN spread signals transmitted via the HGA are susceptible to interference. However, the PN spread signals transmitted via the omni antenna are susceptible to interference from other users unless the interfering signal is cross-polarized.

2.5.2 Cross-Polarization Discrimination

We define the interference attenuation needed as the difference between the signal-to-interference ratio needed to achieve a 10^{-5} BER (10^{-4} for STS) and the signal-to-interference ratio with 0° offpointing. Consider the case where low levels (<11.8 dB for SSA return signals and < 12.4 dB for SMA return signals) of interference attenuation is needed. Figures 2 and 3 show that this attenuation is achieved at all offpointing angles if the signals are cross-polarized. Therefore cross-polarization is an effective interference mitigation technique when low levels of attenuation are needed. Figures 2 and 3 also show that high levels (greater than 17.8 dB for SSA return signals and 13.1 dB for SMA return signals) of interference attenuation can only be achieved at large offpointing angles where the antenna discrimination of cross-polarized signals is the same as for signals that use the same polarization. Therefore, the cross-polarization discrimination is not helpful in mitigating the interference when large interference attenuation levels are needed.

2.5.3 Scheduling

Each user transmits several links and only some of these links receive interference from and cause interference to other TDRSS users. Interference between users can be minimized if users avoid transmitting on the links that interfere with each other at the same time (whenever the angle at the TDRS formed by the vectors pointing to each user is less than the interference threshold angle). For example, GRO and EOS can both transmit with RHCP and Left Hand Circular Polarization (LHCP) polarization. Interference between these users can be avoided if GRO and EOS use opposite polarization when the angle at the TDRS is less than the interference threshold angle.

A single TDRS can support five SMA users and 2 SSA users. SMA links do not receive interference due to PN spreading. The problem is that each SSA user can receive interference from the remaining six users. It could be difficult to avoid interference by selecting links and scheduling support times for all 7 users all the time.

2.6 Self-Interference Environment for 1996 - 2010

All the self-interference events occurring on the 2287.5 MHz SSA links considered in this analysis fall into one of the following three categories:

- a. STS links. These links are very susceptible to interference from other user's HGA links even if the interferer is cross-polarized and/or uses PN spreading.
- b. Nonspread links. The majority of self-interference occurs when the desired signal is nonspread (Q channel of mode DG1-3) and both the desired user and the interferer transmit on the HGA and with the same polarization.
- c. Low power omni links. These links are susceptible to interference from other user's that transmit on the HGA antenna.

Since STS is very susceptible to interference from other user's HGA links (regardless of whether the other user uses cross-polarization or PN spreading), it is likely that any new user supported by TDRSS at 2287.5 MHz will interfere with STS.

Nonspread signals are required for tape recorder dumps and many users require them. It is likely that these nonspread links operating at 2287.5 MHz will receive interference from other user's links and will cause interference to other user's links.

PN spread low power omni links are only used for backup and contingencies. Due to the fact that they are used infrequently or in emergency situations, it is expected that the interference to these links can be avoided by scheduling.

Therefore, there are two areas of concern for S-band self-interference in 1996 - 2010 as the number of TDRSS users increases. First is the likelihood of interference to STS. Second is the possibility of interference to all the nonspread links from any of the other user's HGA links.

3.0 Ku-Band Interference Analysis

The analysis approach at Ku-band is the same as at S-band with the only exception being with regards to how the links were modeled. The S-band analysis considered 48 links. In order to show the link performance in the presence of self-interference for all possible interfering and desired link combinations with a minimum number of BER curves, the S-band analysis modeled links with BPSK signals. This was not necessary for the Ku-band analysis since the Ku-band analysis only considers four Ku-band links: two STS links, one EOS link, and one SSF link.

3.1 TDRS Antenna Discrimination

This analysis uses the antenna pattern in the CLASS database which was obtained from [4] and is shown in Figure 9.

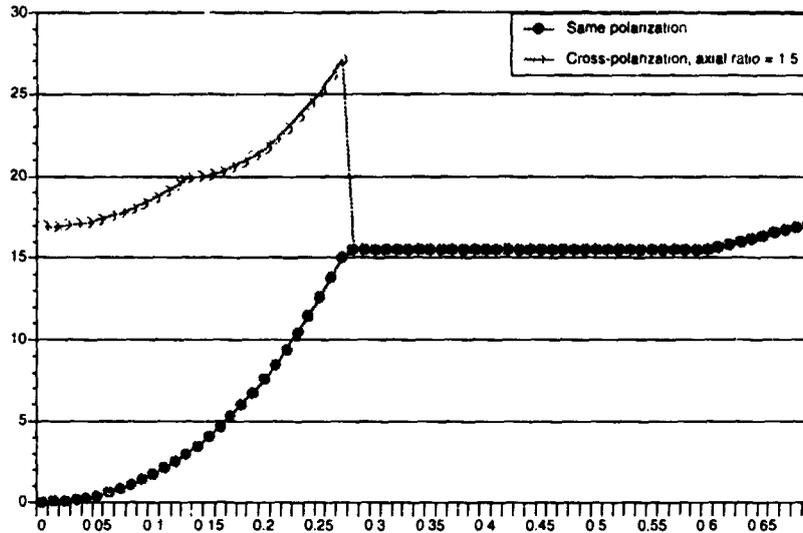


Figure 9. KSA Antenna Discrimination

3.2 STS, SSF, and EOS Link Polarizations

STS Ku-band links are RHCP, the SSF Ku-band link is LHCP, and the EOS link is either RHCP or LHCP. (EOS links 5R and 5L represent the EOS links with RHCP and LHCP, respectively, in the BER curves shown in Section 3.3.)

3.3 BER Performance in the Presence of Self-Interference

3.3.1 Performance of STS Links

Figure 10 shows the performance of the STS channels in the presence of interference from EOS. This figure also has arrows pointing to the BER at 0° offpointing showing the interference from the EOS Ku-band link with RHCP. Figure 10 shows that STS channels 2 and 3 experience interference in the presence of this EOS link. None of the STS channels is susceptible to interference from an EOS signal with LHCP since the cross-polarization discrimination ensures that the signal-to-interference power is greater than -1 dB, which is sufficient to achieve a 10^{-5} BER (10^{-4} for the STS channel 1).

3.3.2 Performance of EOS Links

The EOS Ku-band link has the highest symbol rate of all the Ku-band links, except for Shuttle Channel 3 with 50 kbps data. Figure 11 shows the performance of the EOS link for the two cases: first, when the EOS symbol rate is greater than or equal to the interfering signal's symbol rate; and second, when the interfering signal is the STS channel 3 with 50 kbps data. This figure also has arrows pointing to the BER at 0° offpointing showing the interference from the other users. It shows that EOS links can receive interference signal from STS, SSF and another EOS even if the other user is cross-polarized. This is because the EOS signal has the highest symbol rate and a very low signal margin, and because the EOS power level received at the TDRS is low relative to the other USAT power levels. The cross-polarization discrimination is insufficient to mitigate interference from STS and SSF because the antenna discrimination required to achieve a 10^{-5} BER is only available at large offpointing angles, where there is no cross-polarization discrimination. (See Figure 9).

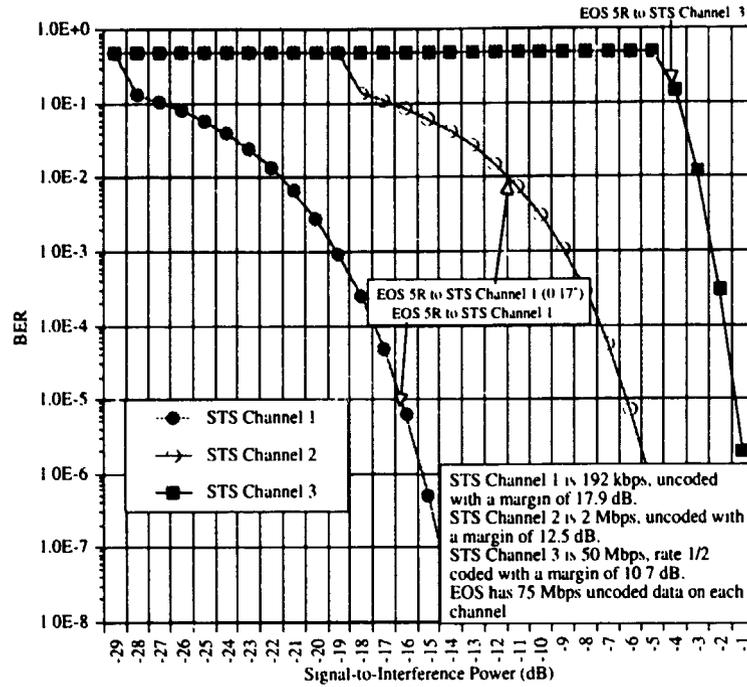


Figure 10. Performance of STS Ku-band Links in the Presence of Interference from EOS

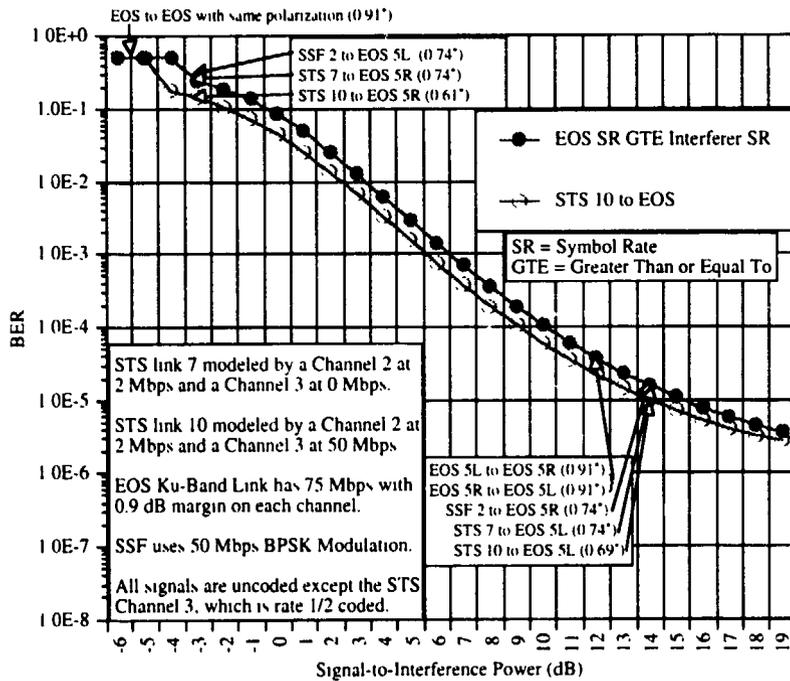


Figure 11. Performance of EOS Ku-band Links in the Presence of Interference from STS and SSF

3.3.3 Performance of SSF Links

The SSF Ku-band link has a higher symbol rate than the other Ku-band links, except for Shuttle channel 3 with 50 kbps data and the EOS link. Figure 12 shows the performance of the SSF link for the three cases:

first, when the SSF symbol rate is greater than the interfering signal's symbol rate; second, when the interfering link includes STS channel 3 with 50 kbps data; and third, when the interfering link is the EOS link. This figure shows that only the EOS link with LHCP can interfere with SSF communications. This is because both users transmit similar symbol rates, the SSF power level received at the TDRS is low relative to the EOS power level and the SSF signal has a very low signal margin. Both STS and the EOS link with RHCP cannot interfere with SSF communications because these links are cross-polarized providing a signal-to-interference power that is greater than 13 dB and a BER less than 10^{-5} .

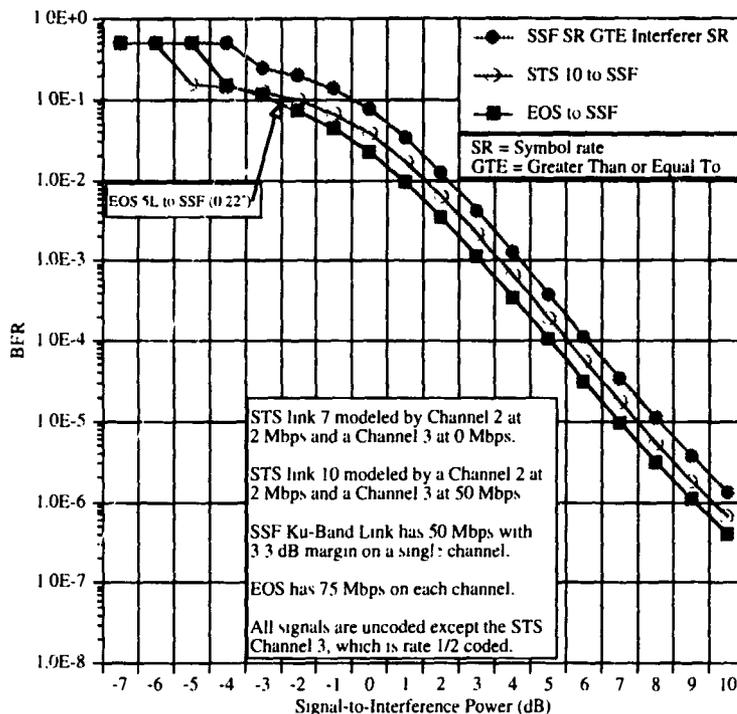


Figure 12. Performance of SSF Ku-band Links in the Presence of Interference from STS and EOS

3.4 Results

Section 3.3 and [3] shows that:

- Cross-polarization discrimination is sufficient to ensure that STS and SSF do not interfere with each other.
- EOS links can receive interference from either STS link, the EOS link, or the SSF link even if the other link is cross-polarized. The cross-polarization discrimination is ineffective in mitigating interference to EOS.
- Both STS links can interfere with each other.
- The EOS link with RHCP can interfere with both STS links, but the EOS link with LHCP cannot.
- The EOS link with LHCP can interfere with the SSF link, but the EOS link with RHCP cannot.

3.5 Self-Interference Statistics

IM simulations provide the interference statistics between STS, SSF, and EOS. The results show that interference occurs less than 0.8% of the time on average but can occur up to 3% of the time in a worst-case week. These averages are significantly lower than the averages obtained for S-band interference primarily because the Ku-band antenna pattern has a much narrower beamwidth than the S-band antenna pattern.

3.6 Interference Mitigation Techniques

The only interference mitigation techniques that are available in the current system is cross-polarization and scheduling. Cross-polarization is very effective, but it is not able to mitigate interference from any of the users to an EOS spacecraft since EOS has the highest symbol rate and lowest power of the three users.

Fortunately, each TDRS can only support two users at a time and the percentage of time that interference occurs is small due to the small beamwidths at Ku-Band. Therefore, scheduling should be sufficient to avoid interference with one exception. The most serious concern is when two or more STS spacecraft are in orbit. Each STS spacecraft may require 100% coverage when not in ZOE, but there is no way to avoid interference between two STS spacecraft since both of the STS links interfere with each other. It should also be noted that since STS and SSF require 100% coverage when not in ZOE and both of these users have a higher priority than EOS, EOS communications must be scheduled to avoid interference with STS, SSF, and other EOS spacecraft. This, however, should be achievable as EOS only requires 20 minutes of service every orbit, and it can use LHCP to avoid interfering with STS and RHCP polarization to avoid interfering with SSF.

[6] provides an example showing how interference to STS can be avoided with scheduling.

3.7 Self-Interference Environment for 1996 - 2010

It is anticipated that more and more users will be using the Ku-Band service. However, it should be possible to avoid interference by scheduling support times since each TDRS only supports two users at a time.

4.0 Conclusions

The S-band analysis showed that the percentage of time that interference occurs between any two users is less than 1.2% on average and 7% in a worst-case week most of the time. However, there are some cases where interference occurs up to 15% of the time on average and 80% in a worst-case week.

Interference at S-band can be avoided with appropriate scheduling techniques. However, this can become more difficult with many users. Furthermore, self-interference events at S-band can be expected to increase in the future as data rates increase. There are two areas of concern for self-interference in 1996 - 2010. First is the likelihood of interference to STS. Second is the possibility of interference to all the nonspread links from any of the other user's HGA links.

The Ku-band analysis showed that the percentage of time that interference occurs between any two users is less than 0.8% on average and 3% in a worst-case week. Since each TDRS only supports two Ku-band users at a time, it should be possible to avoid interference with appropriate scheduling techniques.

5.0 Acknowledgements

The author would like to thank Robert Godfrey and Nancy Smith of GSFC, Code 531.1, for their guidance and support of this analysis.

6.0 References

1. T. Kaplan, J. Freedman, D. Wampler, A. Musliner, C. Puseau, CLASS Interference Analysis System, National Telesystems Conference, May 19th and 20th, 1992, Washington D.C.
2. T. Kaplan, A. Musliner, D. Wampler, The Automated Conflict Resolution System, Networks Technology Conference, Greenbelt, Maryland, November 9, 1993.
3. L. Harrell, T. Kaplan, SN Requirements Analysis, STel 004R1493, February 16, 1994.
4. "Default SA Antenna Gain Pattern Envelopes", Memo to Nancy Smith from Jim Rash GSFC/Code 531.1, Mar 8, 1990.
5. "Space Station Freedom MA/SMA Jamming Analysis", Memo to Denver Herr from Tim Rand, STel, May 10, 1991.
6. Y. Wong, J. Rash, A Method for Interference Mitigation in Space Communications Scheduling, Space Ops 92 Proceedings, November 16-20, 1992, Pasadena, CA

Attempt of automated space network operations at ETS-VI experimental data relay system

Kiyoomi ISHIHARA Masayuki SUGAWARA

National Space Development Agency of Japan (NASDA)
Tsukuba, Japan

ABSTRACT

National Space Development Agency of Japan (NASDA) is to perform experimental operations to acquire necessary technology for the future inter-satellite communications configured with data relay satellite. This paper intends to overview functions of the experimental ground system which NASDA has developed for the Engineering Test Satellite VI (ETS-VI) Data Relay and Tracking Experiment, and to introduce Space Network System Operations Procedure (SNSOP) method with an example of Ka-band Single Access (KSA) acquisition sequence. To reduce operational load, SNSOP is developed with the concept of automated control and monitor of both ground terminal and data relay satellite. To perform acquisition and tracking operations fluently, the information exchange with user spacecraft controllers is automated by SNSOP functions

1. INTRODUCTION

NASDA has launched several types of satellites since 1975. The tracking and data acquisition operations for these satellites have been conducted by using NASDA's ground stations network, four S-band ground tracking stations, 3 stations in Japan and one in Sweden, but with no space network. NASDA has yet no experience of performing the satellite mission operations with a space network.

ETS-VI Space Network system described in this paper has been developed by NASDA under the Space Operations and Data System (SODS) program, as an experimental system to obtain the necessary technology for the inter-satellite communications. As a follow-on program, NASDA will launch COMETS in 1997 and enhance NASDA's space network

technology of ETS-VI for the future operational system.

2. ETS-VI SN SYSTEM

ETS-VI Space Network (SN) system consists of space segment and ground segment. Figure-1 illustrates the ETS-VI SN system.

2.1 Space Segment

ETS-VI will be launched by H-II launch vehicle in 1994. It will be located at 153.8 degrees East longitude. ETS-VI mounts two experiment equipments for the experiment: S-band Inter-satellite Communications Equipment (SIC) and Ka-band Single Access Equipment (KSA).

SIC can provide one forward service and two return services simultaneously to support user spacecraft on low earth orbit. The 19 elements phased array antenna attached to the body of spacecraft provides return link service for up to 2 users simultaneously. Only 16 of the 19 elements are used to provide forward link service to one user. The SIC operates on a fixed frequency (2106.4 MHz forward/2287.5 MHz return)

KSA provides high-data rate support by using new frequency band: 23 GHz forward/ 26 GHz return. The 23/26 GHz KSA will be the first inter-satellite communications equipment on orbit. For experiments, only 80 cm parabolic antenna provides one forward and one return link services at a time.

2.2 Ground Segment

ETS-VI SN ground segment is located in Tsukuba Space Center. It consists of 6 major functions.

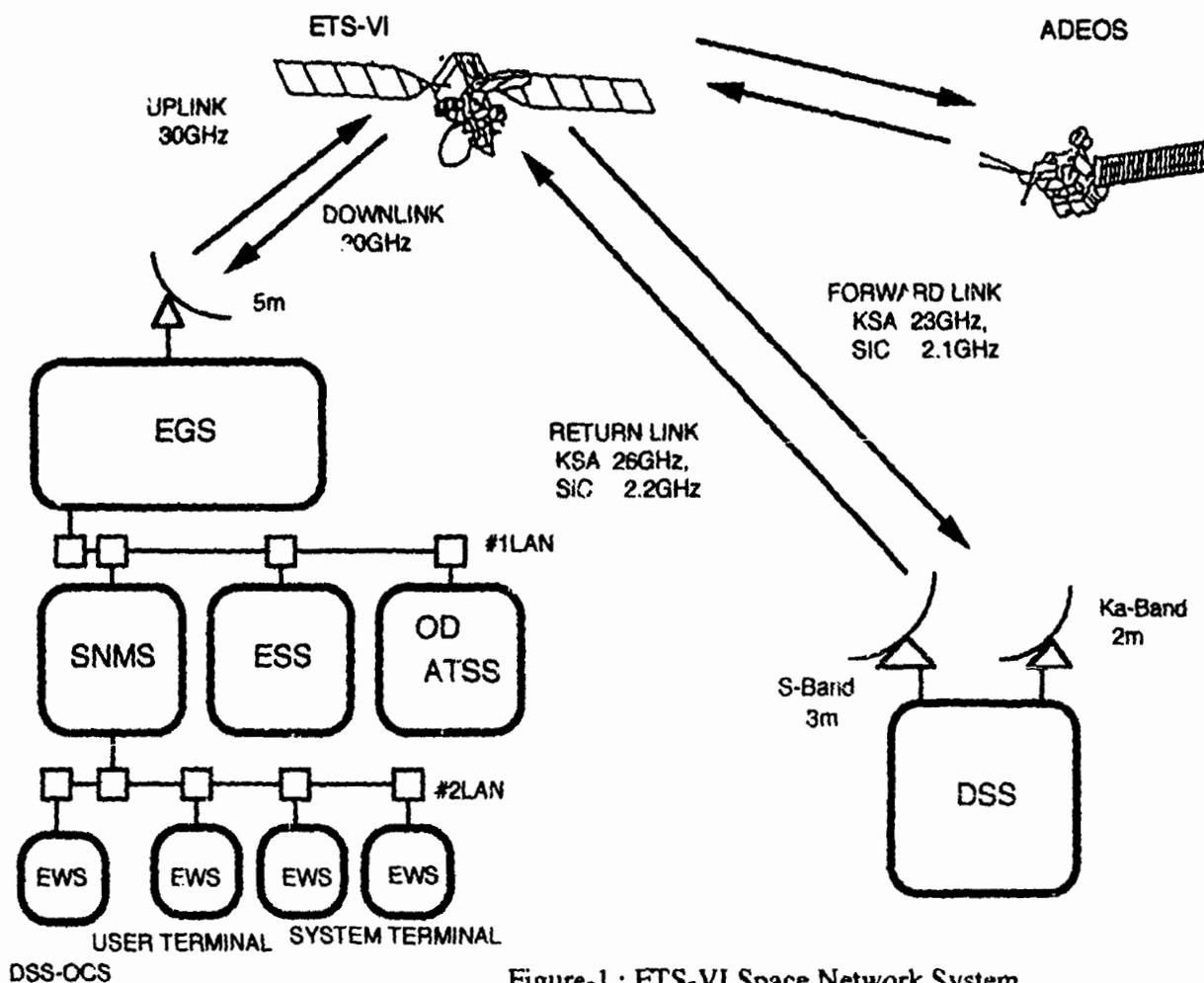


Figure-1 : ETS-VI Space Network System

(1) Experimental Ground Station (EGS)

The EGS provides the telemetry and command functions for ETS-VI and also provides the telecommunication functions necessary for transmitting and receiving user data through the ETS-VI. The EGS interfaces with the ETS-VI through Ka-band (30 GHz uplink/20 GHz downlink) Space-to-Ground Feeder Link. These uplinks and downlinks are done through a 5-meter antenna.

(2) Dummy Satellite Station (DSS)

The DSS provides the telecommunication equipments to emulate the user spacecraft. The DSS has functions of receiving SIC/KSA forward links and transmitting SIC/KSA return links to ETS-VI. The SIC forward/return services are sent/received through a 3-meter antenna, KSA services use a 2-meter antenna. The DSS is normally located in the Tsukuba Space Center, Japan, and is transportable for experimental purposes.

(3) Space Network Management Subsystem

The SNMS is a core of the ETS-VI SN group segment. ETS-VI SN system control is provided by the SNMS, which consists of a configuration of computer setup to control and monitor the ETS-VI and the EGS. The SNMS also provides a interface with the User Operations Control Center (OCC), through a user terminal for the scheduling and real time operations.

(4) Orbit Determination/Acquisition and Tracking Support Subsystem (OD/ATSS)

The OD provides orbit computation for earth orbiting spacecraft and tracking performance assessments for the SN.

The ATSS provides all system calculations necessary for the initial acquisition and tracking operations. One computer (EWS) is shared between OD and ATSS.

(5) Experiment Support Subsystem (ESS)
 The ESS provides the following functions.
 -Accumulation of the experiment data
 -Analysis of the experiment data
 -Fault Isolation analysis (KSA system)

(6) DSS-OCS
 The Dummy Satellite Station-Operations Control Subsystem(DSS-OCS) provides DSS control. Its primary functions is to monitor the DSS conditions and generate the commands to the DSS.

3.ACQUISITION SEQUENCE

During the initial acquisition, many complicated processes occur in combination with the user OCC, the ETS-VI, and the user spacecraft in order to establish the inter-satellite communication links between the two satellites. Understanding of these processes is required. Obtaining these operations techniques is one of the most important objectives of the experiments.

The acquisition and tracking operations for space network is much complex in comparison to the usual direct communications with the ground station network. The last tendency of mass data and high rate transmission, for example, requires data relay satellite to have antenna bigger and frequency higher, and eventually make the operations more complicated. In fact, although the KSA antenna of ETS-VI is only 80-cm in diameter, the KSA has a very narrow radiation beam because the frequency band is so high, 23/26 GHz band. Eventually the KSA is required to point its antenna precisely toward the user spacecraft during acquisition.

For the ETS-VI SN experiments, NASDA plans to use the Advanced Earth Observation Satellite (ADEOS), as a user spacecraft, which will be launched in 1996 from Tanegashima Space Center, NASDA. Table-1 shows the KSA typical acquisition sequence between the two satellites.

The acquisition sequence shown in table-1 is nominal procedure. However in case ETS-VI fails, for example, to acquire the return beacon signal from ADEOS at step E4, ETS-VI will start the antenna movement for signal search.

Table-1: Typical Acquisition Sequence

ETS-VI	ADEOS
Step-E1 Update orbit elements stored at onboard computer	Step-A1 Update orbit elements
Step-E2 Antenna point toward ADEOS	Step-A2 Antenna point toward ETS-VI
Step-E3 Transmit Forward beacon signal	Step-A3 Acq. Forward Beacon Start Antenna auto tracking mode
Step-E4 Acq. RTN signal Start ANT auto track	Step-A4 Transmit return signal
Completion of initial acquisition	
Start Forward and Return Services	

4. AUTOMATED SN OPERATIONS

4.1 Automated SN operations

Establishing an automated SN operations system is one of the key-elements that NASDA intends to develop by the era of our operational Data Relay Satellite System. NASDA thus made an attempt to develop a new automatic operations method, SNSOP (Space Network System Operations Procedure), as a experimental method at the ETS-VI Experimental SN ground system. This method concentrated on the following points:

- To control both ETS-VI and EGS automatically in accordance with the pre-determined acquisition sequence.

- To monitor the network conditions effectively.
- To move into recovery procedures automatically as much as possible if any of the checks fail
- To create/modify the operations sequences easily.
- To exchange the operational information with the user OCC timely.

2. Hardware

All control and monitor of the EGS and ETS-VI is performed by SNSOP installed in the SNMS, although manual intervention is still possible. The SNMS consists of the host computer and two EWSs (one for SIC, the other for KSA). The SNSOP process can be divided into two processes; (1) a client process in the EWS, (2) a server process in the host computer. Figure-2 shows their functional assignments.

4.3 SNSOP

The SNSOP is a flow chart list which consists of the combination of process boxes and flow lines. The process box can define a statement, an arithmetic expression, a logical expression, a control command for SNSOP and so forth. The flow line is a line, which can define one-way direction, for connecting between two process boxes. The SNSOP can be composed of a main-SOP, a sub-SOP and a mini-SOP structurally. The SNMS provides a MMI for creating and modifying the SNSOP easily. Process boxes used in the ETS-VI SN system are shown in figure-3 and as follows:

- Start Box : States the start of main-SOP
- End Box : States the end of main-SOP
- Entry Box : States the start of sub/mini-SOP
- Return Box : States the end of sub/mini-SOP
- Process Box: Defines the free text (ex.arithmetic expression,etc.)
- Exec1 Box : Defines the execute command for ETS-VI
- Exec2 Box : Defines the execute command for EGS
- Exec3 Box : Defines the execute command

- for other ground elements
- Check Box : Defines the TLM# to compare with data base value.
- Branch Box : Defines the branch condition.
- Loop Box : Defines the loop process based on a branch condition.
- Store Box : Defines the stored CMDs for ETS-VI
- StoreExe Box: Defines the execution of stored commands
- Sub SOP Box: Defines the sub/mini-SOP#
- SN/OpsMSG Box: Defines SN or OpsMsgID

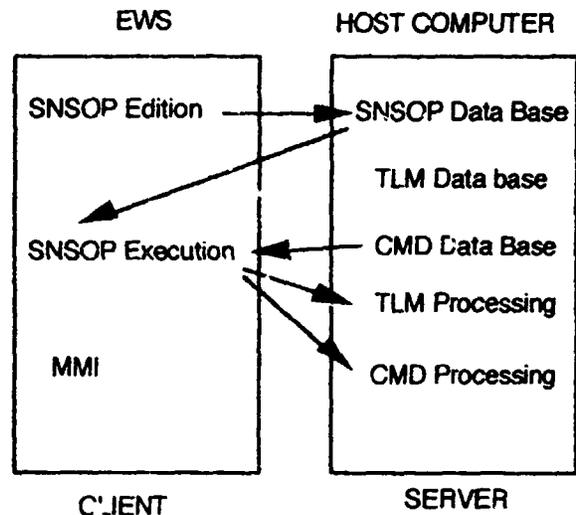


Figure-2 : SNMS functional assignment

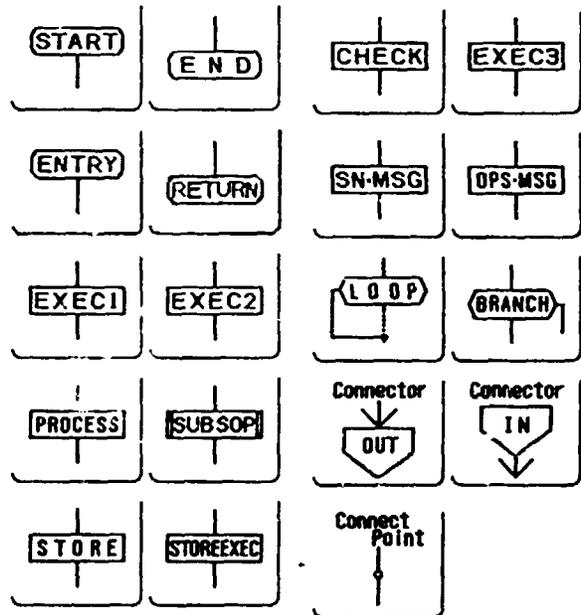


Figure-3 : Process Box

4.4 Operation Description Data (ODD)

The ODD is a unique language developed by NASDA to be used in SNSOP. The system operator can easily describe a free text, e.g., an arithmetic expression, a logical expression, branch condition, etc., in a process box by using the ODD. It can basically be divided into following groups:

(1) Arithmetic Assignment

- Constant
- Variable
- Array
- Arithmetic operator (+, -, x, /)
- Functions
- Assignment

(2) Logical Assignment

The assignment method is the same as arithmetic assignment.

- Logical operator
and, or, not, xor

(3) Judgement condition

- >, <, >=, <=, =, !=

(4) SNSOP control commands

The control commands used at ETS-VI SN system are as follows;

- Test** : Compare the received TLM with canned value in data-base.
- Wait** : Wait before moving to the next step for an appointed period of time.
- Wake** : Break the current job and restart at the appointed time.
- Input** : Input the value to the appointed variable name.
- Display** : Display the value in the appointed variable name.
- Print** : Print the value in the appointed variable name.
- Interrupt** : Prohibit the interruption, or release it.
- GetTLM** : Set the appointed TLM value to the appointed variable name.
- StoreCancel** : Cancel the stored commands
- Save** : Store the value in the appointed variable name to the appointed file
- Load** : Get the value into the appointed variable name.
- Beep** : Sound buzzer for the appointed time
- Exec1** : Execute the commands for ETS-VI
- Exec2** : Execute the commands for EGS
- Exec3** : Execute the commands for other ground element.
- Store** : Store the ETS-VI commands

- StoreExe** : Execute the stored commands
- OpsMsgGet** : Get the operation message

4.5 Execution of SNSOP

The SNSOP is executed one after another automatically in accordance with the flow chart list, and the manual operation is also possible. The automatic execution will be stopped temporarily when the system operator selects "Pause Mode" or sets a break point on the expected process box of flow chart list, or when the system operator is asked for his judgement by SNSOP. Following are the three modes during manual operation:

- Step Exec. mode** : Execute one step only.
- Step Skip mode** : Skip one step only.
- Box Skip mode** : Skip one process box.

The transition from manual to automatic mode is performed by choosing "Restart Mode" and "Abort mode" is for the SNSOP forced end.

4.6 Control Requests from user-OCC

The changes to the operating conditions or configurations of the ETS-VI SN ground segment can be initiated by the "Operations-Control" message from user-OCC to the SNMS. These messages are categorized into two groups: Class A and Class B.

The class A Ops-control message is executed automatically after completing the valid check at SNMS. The class B is performed with the SN operator's decision.

The operating SNSOP will be interrupted after the completion of validity check when the Ops-control message is received at SNMS, and another SNSOP for an Ops-control message will be called and started automatically.

4.7 Monitor requests from user-OCC

The real-time monitoring of SN ground system at user-OCC is achieved via "SN message" and "SN message ID". The SN message is a telemetry data of ETS-VI SN system, and is sent to user-OCC every 2 seconds during support period. The SN message ID is used when the SN system wants to notify the key event of the current executing sequence, e.g.,

Completion of EGS setup, Bringing forward beacon signal up, etc., to the user-OCC. The message ID is sent to user-OCC by SNSOP.

The user-OCC can get information of which step the SN system is executing at a certain time via the SN message and SN message ID.

5. SNSOP DATA BASE

A total number of 61 main-SNSOPs are stored in the data base of SNMS. Many sub and mini-SOPs are linked with the main-SOP. The main-SOP can be classified as follows:

(1) System Operations SOP

There are 11 SOPs for system operations, e.g., EGS initial setup, Pilot links on/off, TT&C links on/off, etc.

(2) KSA Service SOP

4 SOPs are prepared for the KSA service operations based on the combination of types of initial acquisition sequence mode (Sequence#1, Sequence#2A, Sequence#2B) and KSA antenna pointing modes (CPU mode, Direct mode).

(3) SIC Service SOP

4 SOPs are prepared for the SIC forward/return service operations based on the SIC antenna pointing modes (Orbit elements mode, Program mode, Real time mode, Direct mode).

(4) Recovery SOP

The recovery SOP is called when any of the checks in the main-SOP fails. 11 SOPs are stored in the data base.

(5) Ops-Control SOP

30 SOPs are stored in the data base. The Ops-control SOP is used when it is requested from the user-OCC.

Figure-4 shows the KSA service operations display for CPU/sequence#2A mode. The left window shows the main-SOP and the right is the sub-SOP of the second process box from the top of main-SOP.

6. SN Real-Time OPERATIONS

The real-time operations period is the time frame in which the user-OCC and the SN system perform necessary activities to support the command, telemetry, and tracking operations of a user spacecraft. These operations can be categorized as those occurring prior to the scheduled support start time, those occurring during the real support period, and those occurring post-support.

The following are the typical operations of each phase with an example of the KSA service operation (CPU mode).

6.1 Pre-support Phase

(1) User-OCC

a. Sending the KSA service requests to the SNMS by at least 30 minutes prior to scheduled start of service, through the user terminal.

(2) ATSS

a. Receiving the state vectors of both ETS-VI and user spacecraft if updated.
b. Processing of them by 30 minutes prior to

the scheduled start of service.

c. Calculating the parameters necessary for the acquisition and tracking operations, and sending them to the SNMS when required from the SNMS.

(3) SNMS

a. Choosing the next support service by at least 30 minutes prior to the scheduled start time.

b. Sending the calculation request of acquisition parameters to the ATSS and receiving it from the ATSS within a certain time.

c. Setting the parameters calculated at step b into the system variables of KSA service SOP.

d. Configuring the ETS-VI by SNSOP.

(4) EGS

a. Configuring the EGS by SNSOP.

6.2 Support Phase

(1) SNMS

a. Controlling KSA antenna operations by

SNSOP.

- b. Initiating/terminating the services by SNSOP.
- c. Generating and transmitting the SN message and SN message ID to the user terminal located at the user-OCC.
- d. Operating service.
- e. Monitoring the SN performance.
- f. Verifying and processing the Ops-control from the user-OCC through user terminal if received.
- g. Initiating the recovery operations based on the recovery SOP if any of the checks fails.

(2) EGS

- a. Acquisition of service channel.
- b. Operating service.
- c. Reconfiguring/reacquisition service.
- d. Receiving and logging the tracking data if required.

(3) User-OCC

- a. Monitoring the SN performance via SN message and message ID.
- b. Sending the Ops-control if needed.

6.3 Post-support Phase

(1) SNMS

- a. Dumping the operations history
- b. Conducting post-support debriefing with the user-OCC.

(2) OD/ATSS

- a. Receiving and logging the tracking data from the EGS in accordance with the direction from the SNMS.

7. CONCLUSION

NASDA is now in its final process of the ETS-VI launch scheduled on August '94. The experiment will begin from November '94 after completing the initial mission checkouts on orbit. We would like to verify the availability of the SNSOP method through the ETS-VI data relay experiment period and reflect it to our future data relay satellite system.

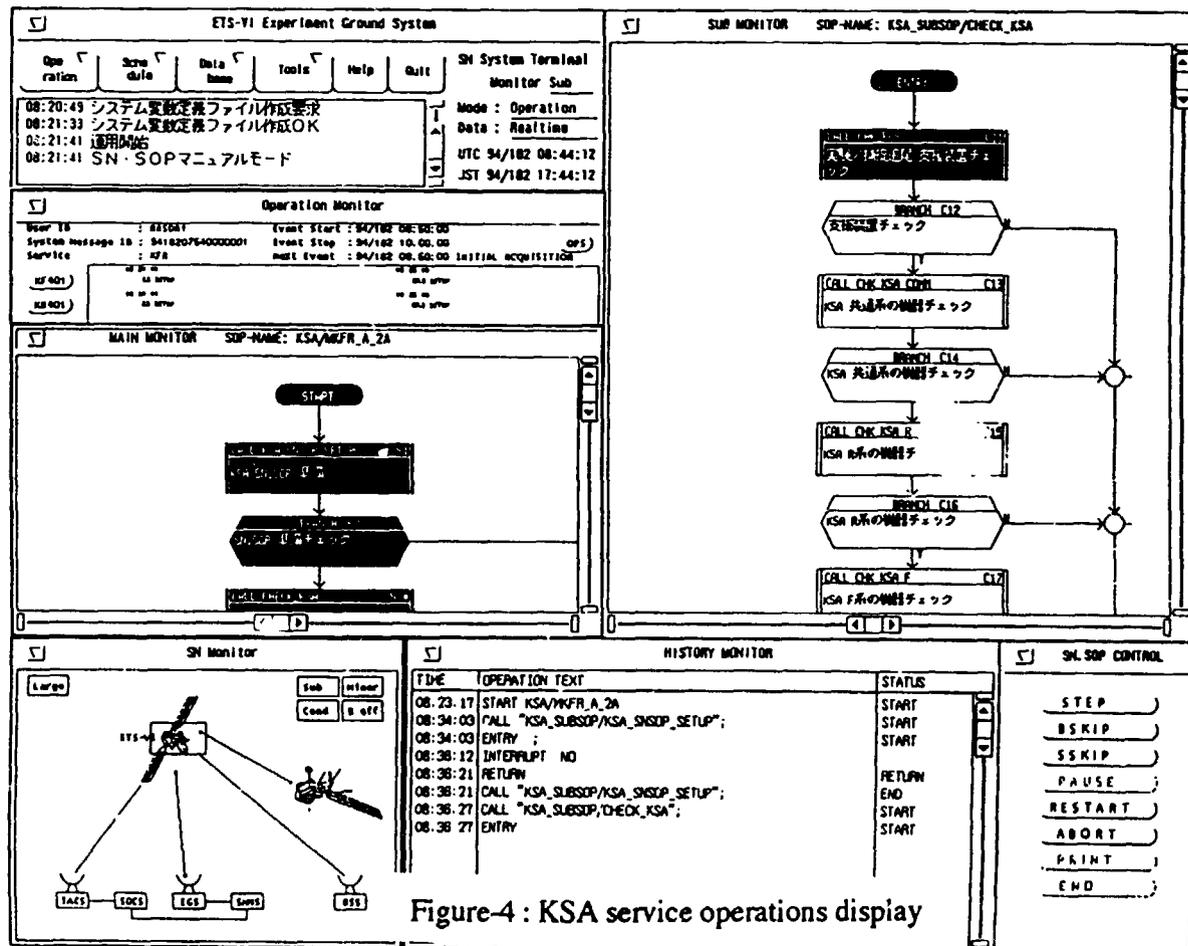


Figure-4 : KSA service operations display

Table-Driven Configuration and Formatting of Telemetry Data in the Deep Space Network

Evan Manning

Telos Information Systems
320 N Halstead, Suite 260
Pasadena, CA 91107
manning@alumni.caltech.edu

Abstract

With a restructured software architecture for telemetry system control and data processing, the NASA/Deep Space Network (DSN) has substantially improved its ability to accommodate a wide variety of spacecraft in an era of "better, faster, cheaper."

In the new architecture, the permanent software implements all capabilities needed by any system user, and text tables specify how these capabilities are to be used for each spacecraft. Most changes can now be made rapidly, outside of the traditional software development cycle. The system can be updated to support a new spacecraft through table changes rather than software changes, reducing the implementation-test-and-delivery cycle for such a change from three months to three weeks. The mechanical separation of the text table files from the program software, with tables only loaded into memory when that mission is being supported, dramatically reduces the level of regression testing required.

The format of each table is a different compromise between ease of human interpretation, efficiency of computer interpretation, and flexibility.

1. Deep Space Network Telemetry 1990

In 1990 NASA's Deep Space Network (DSN) supported fewer than a dozen spacecraft, all them using minor variants on a single NASA output format. Each new spacecraft was a major event, frequently accompanied by upgrades of DSN hardware and software.

In addition, support for frequent minor processing changes was creating a bottleneck because each change required a formal software build and delivery, and regression testing. One example of such a change is an increase in data rate and frame size as an encounter approaches.

2. Changing Environment

In recent years both the number and variety of missions supported by DSN has grown explosively. Today, the DSN supports over seventy deep-space and near-Earth spacecraft operated by NASA/JPL, other NASA centers (e.g., GSFC), other US government agencies (e.g., NOAA), and other nations' space agencies (e.g., CNES, ISAS). And these spacecraft are beginning to be produced with shorter lead times.

It would be impossible to support all these spacecraft with the old system.

3. Tables Save the Day

Spacecraft specifics had to be removed from the main build-test-delivery cycle.

Text tables presented an opportunity to isolate mission-specifics from the telemetry processing software, and thus from much of the delivery cycle's cost in time and money. With a software architecture where tables are clearly read in by the computer only when the appropriate mission is commanded, table changes need no software build and little regression testing.

Three tables are sufficient to encapsulate all mission specific behaviors of telemetry processing: spacecraft initialization tables, rules tables, and format tables.

3.1 Spacecraft Initialization Tables

The spacecraft initialization table (SIT) configures devices based on mission-specific telemetry parameters (subcarrier frequency, coding type, frame length, Reed-Solomon interleave depth, etc.). Their format is almost exactly the same as that used for interactive Operator Directives (ODs) except for the addition of comments. Implementation of these tables was integrated with implementation of a warmstart file capability. Both send commands to the existing front end for interpretation, and so are easy to implement despite their power.

Example 1 is a SIT table.

3.1.1 Tradeoffs in SIT Table Design

SIT table design is natural, giving ease of operator use through familiarity and ease of implementation through use of existing interpretation facilities. The only loss in

going to a table driven approach is a loss in speed because each command must be interpreted each spacecraft tracking pass.

3.2 Rules Tables

Realtime changes of certain key configuration parameters sometimes require changes to other related configuration parameters. For example, a change of bit rate may imply changes to frame length, coding type, and data output format.

Rules tables reconfigure devices and data output formats when key parameters change. The current implementation can accept only data rate as the key parameter because that is the only key parameter needed by any existing mission for changes to anything but format.

3.2.1 Rules Tables Format

The format of rules tables also uses the existing operator directive format as much as possible. The only enhancement is that these tables have two columns of ODs: key parameters in the first column, corresponding ODs to be processed in the second column. Although the first column is identical in format to the telemetry bit rate (TBR) OD, its meaning in context is different. When the operator enters "TBR 32000" the TGC reacts by configuring hardware to expect incoming data to change data rate to 32,000 bits per second (including any commands specified in the rules table). "TBR 32000" in the first column of a rules table directs the TGC to execute the corresponding ODs whenever a TBR OD is received with a new bit rate closer to 32,000 than to any other bit rate in the rules table.

Before the latest DSN upgrade there was no

Example 1. Spacecraft Initialization Table (partial)

```
#####
# SPACECRAFT:  DSPSE
#
# Characteristics:  GSFC data type
#                  Single channel
#                  MCD coded
#                  NRZ-M to Bi0-L conversion
#####
#
PGM    DSPSE
OFT1   DSPSE
#
# XBBM is internal form of the BBM OD
#
XBBM   UU
#
TBR1   128000  C
FSU1   S
FOM1   A
FLG1   P=2048  S=0
FBT1   IL=1   OL=1
FLT1   IL=2   OL=2
FSW1   32    EB90146F
APC1   D
FES1   E
MCV1   CCSDS
MLT1   3     3048
MCA1   MNR 256 2048
DIFD1  NM
RSU1   D
#
# XPRn replaces IGP and USP ODs
#
XPR1   D D D D D D
#
PTO1   0 000000
MNO1   ACC=0.0  AGN=0.0  DOP=0.0  SCF=0.0  SNR=0.0  TBR=0.0
BBS    1, 9, 2, 10
LBW1   M 3
SER1   D
SLT1   -2.5
SNR1   0.0
SCF1   1700000.0
# Input symbol is NRZ-M
PCM1   NL
DSA1   E
```

Example 2. Rules Table

```
#
# SPACECRAFT:  DSPSE1
# DATE CREATED: 09/15/93 - First edition
#
TBR 125      TSO DSID=ED
TBR 250      TSO DSID=EE
TBR 500      TSO DSID=EF
TBR 1000     TSO DSID=F0
TBR 2000     TSO DSID=F1
TBR 6000     TSO DSID=F2
TBR 61000    TSO DSID=F3
TBR 128000   TSO DSID=F4
```

corresponding capability. Operators had to enter all ODs manually every time bit rate changed.

Example 2 is a rules table.

3.2.2 Tradeoffs in Rules Tables Design

Like SIT tables, rules tables use the same interface as ODs, making them easy from both a human and a machine perspective.

Conceptually, rules tables are almost part of SIT tables. Both are implemented from the same documents with invariant fields going to a SIT table while bit-rate dependent fields go to a rules table. Care is needed on the part of the table implementor to make sure all fields go to one or the other and to put only the needed fields in rules, as these will override operator-entered commands whenever bit-rate changes.

3.3 Formatter Tables

Format tables specify the packaging of data from an internal representation to the format required by the mission. (Formats currently supported include 1200- and 4800-bit asynchronous and frame-synchronous blocks as well as variable-length Standard Formatted Data Units.) In addition to data, formatted output generally incorporates a variety of information about processing: bit rate, sequence number, signal strength, Earth Receive Time, etc.

3.3.1 Formatting in the Bad Old Days

Before the recent DSN upgrade formatting was implemented directly in computer language, with a separate executable overlay for each supported mission. So every

change to a formatter required an entire build and delivery, and the computer language implementation left open the possibility that typos could create apparently unrelated errors.

3.3.2 Format Tables

Format tables are much more complex than SIT or Rules tables. In essence they are almost a mini-language, but this language is focused only on the ability to format telemetry data.

The first part of each table (after a conventional comment header including name and change history) is a preamble that specifies whether the rest of the table will use 8-, 16-, or 32-bit words and whether word and bit counting will number from zero or one. This makes it easy to translate any document to a format table.

The rest of the table is divided into event blocks. Each event block specifies actions to be taken at a specific event:

- when the format table is first loaded
- when new information on upstream processing is received
- when new data is received
- when a new output block is started
- when an output block is completed

Within each event block is a series of table entries. These entries are executed in the order they are encountered. Generally it is preferable to order entries within each section by increasing address of destination within output data, but sometimes dependencies among fields make it necessary to vary this order.

The first column of each entry specifies the source, the second column specifies the destination, and the optional third column

specifies operations to be performed on the data.

Source and destination fields can be constants (source only) (numeric, restricted ASCII, or symbolic), fields from within the `formatted data block (/[<start-word>].<start-bit>:[<word-length>].<bit-length>/)`, or named data store fields. Data stores correspond to formatter structures. There are structures associated with each input and output data block, with status information for display, with the formatting process, and with upstream processing information. A few fields are available for internal use when more than one operation is needed at a time.

Operations can be simple replacement, bitwise-or with current contents, floating point conversions, addition of a constant, table lookup, or if-style flow control.

Example 3 is a format table.

3.3.3 Formatter Implementation

For reasons of speed, operational software uses binary forms of format tables. The translation from text to binary format is normally performed when the delivery media is prepared.

Tables can also be modified and "binarized" (compiled) in the field if necessary using a standard text editor.

Inside the binarizer, the 'C' preprocessor is used in a way called Plastic List Manipulation to allow use of C structure field names to access those fields from within tables. The binarizer translates field names to structure offsets and lengths. Version checkwords make sure that the appropriate version of the binarizer was used

on each file, so problems are not created when structures change.

3.3.4 Formatter Tradeoffs

Formatter design was essentially unconstrained by prior art, leaving many decisions to the implementer. The two major considerations were ease of implementation and ease of use. Ease of use seems best served by making the format of the tables as similar as possible to the format of the documents that specify them. In cases where document format could not work, ease of use is best served by similarity to familiar computer languages.

But with limited implementation effort, many decisions were made to favor easy one-pass interpretation. These include separating out event blocks instead of allowing pure ordering by address and placing the optional operation field last.

It is worth noting, however, that the binary file implementation leaves open the possibility that a "friendlier" binarizer could be written, producing the same binary format and therefore not impacting the formatter software at all.

It is also worth noting that user-friendliness is relative. Even the week that might be required for a new user to implement a new spacecraft is a great improvement over the previous "hard-coded" method.

4. Difficulties of Working with Tables

Adding table capability to any program increases its complexity and therefore its upfront costs.

Example 3. Format Table (continued)

```
sap.scn                /4.1:8/                ; Spacecraft Number
;
;
on(COND_IN)           ; Tag meaning following operations are to take place when
; each input block is received.
;
;
on(COND_BIT_1)        ; Tag meaning following operations are to take place when
; a new output block has been started.
;
;
on(COND_OUT)          ; Tag meaning following operations are to take place when
; all data to be sent has been placed in the outgoing block.
;
; Hard-coded calculations will determine the out.out_dest used here
; from DEST specified above and TSO commands and the_VCID.
out.out_dest          /3.1:8/                ; Destination code
out.out_dsid          /4.9:8/                ; Data Stream ID
sap.bsn_0              /5.9:8/                ; Message Block Count
; First three bits of word 6 are flags set to zero. Ignored because all
; fields are zeroed.
out.fill_next         /6.4:13/              ; Number of telemetry bits in
; data field.
; First two bits of word 7 are flags set to zero. Ignored because all
; fields are zeroed.
; following fields are timestamp of first bit in data field.
out.doy               /7.3:9/                ; Day of year
out.msec_day          /7.12:1.11/           ; milliseconds of UTC
out.usec_msec         /9.7:10/              ; microseconds of millisec
; Telemetry data (and filler) words 10-296
sap.bsn_0              /297.1:16/           ; Block Serial Number
; Zero-filled CGF Error fields words 298-300
```

4.1 Documentation

When a new interface is invented for tables, as for format tables, it must be documented. The format table documentation, including examples, index, and descriptions for each data store field, runs to over 200 pages. The work of documenting an interface must be included in the cost of adding table capability. The size and complexity of the documentation seems to increase as tradeoffs are made to simplify the implementation of tables at the cost of more work in making tables.

4.2 Speed

The extra level of indirection introduced by tables carries with it a significant run-time computational expense. I estimate it at 10x for formatting header fields but no added expense in the move of actual data, averaging out to a 2x to 3x cost penalty. Current software and hardware can pay this penalty and still process data at the highest rate currently needed: 2.2 Mbits/sec. If higher rates are needed, special hardware may be needed.

5. Conclusion

DSN has now been operationally using format tables for nearly three years and SIT and rules tables for 1 year for all telemetry adaptation and in that time has successfully added over a dozen new missions without any required software changes. The only modification to the software has been to accommodate NIMBUS frame stripping, where most of the data is discarded because most of the instruments are no longer functioning. Even this could have been done with format tables, but would have been too slow.

In addition, turnaround time for implementation of mission-specific changes from inception to installation has been reduced by a factor of four.

6. References

Deep Space Network Telemetry Group Controller Software Operator's Manual. TDA/DSN No. UG-DOT-5464-OP Rev. A. NASA Ident No. 23835. May 2, 1994.

Deep Space Network Telemetry Channel Assembly Software User's Guide. TDA/DSN No. UG-DOT-5464-OP Rev. B. NASA Ident No. 23835. July 16, 1993

7. Acknowledgements

The work described in this paper was accomplished by Telos Corporation under contract to the Jet Propulsion Laboratory, California Institute of Technology and sponsored by the National Aeronautics and Space Administration.

I would especially like to thank Michael Stoloff of JPL for his foresight and encouragement.

Within Telos, this effort benefitted from the support of Tom Soderstrom and the DSN telemetry operations expertise of Ron Holden.

END-TO-END COMMUNICATION TEST ON VARIABLE LENGTH PACKET STRUCTURES UTILIZING AOS TESTBED

W. H. Miller*, V. Sank⁺, W. Fong*, J. Miko*, M. Powers*, J.
Folk**, B. Conway⁺, K. Michael*, P.-S. Yeh*

* Goddard Space Flight Center

+ Fredrick Herold and Associates

** Advanced Technology and Research

Abstract

This paper describes a communication test, which successfully demonstrated the transfer of losslessly compressed images in an end-to-end system. These compressed images were first formatted into variable length Consultative Committee for Space Data Systems (CCSDS) packets in the Advanced Orbiting System Testbed (AOST). The CCSDS data Structures were transferred from the AOST to the Radio Frequency Simulations Operations Center (RFSOC), via a fiber optic link, where data was then transmitted through the Tracking and Data Relay Satellite System (TDRSS). The received data acquired at the White Sands Complex (WSC) was transferred back to the AOST where the data was captured and decompressed back to the original images. This paper describes the compression algorithm, the AOST configuration, key flight components, data formats, and the communication link characteristics and test results.

1.0 INTRODUCTION

With the advent of sophisticated scientific satellites, space data communication systems are becoming more complicated in order to handle advanced instruments which generate variable data rates and formats. The desire to provide international cross-support across different platforms in order to better utilize the science data globally has prompted the international CCSDS to issue a

recommended standard on space data system architecture specified in the Advanced Orbiting System (AOS) Blue Book [1]. This architecture provides flexibility to transport space data between platforms, ground stations and commercial data networks. To demonstrate the capability of this architecture, Goddard Space Flight Center (GSFC) has been developing a testbed for the AOS. The key components of the AOST is implemented in hardware in order to provide insight regarding achievable speed and limitations for actual flight hardware. The block diagram in Figure 1 shows these key components including an instrument simulator followed by a packet generator, a high-speed multiplexer, additional instrument simulators, and a virtual channel transfer frame generator.

The testbed is capable of implementing the packet data architecture specified in the standards book and re-illustrated in Figure 2. A salient feature of the data architecture is the ability to transport variable-length CCSDS packet, as opposed to the conventional fixed-length packet structures. This structure allows packet data from different instruments to be multiplexed in a much more flexible way in the data system. For data originating from one single instrument, the variable-length packet is also a natural structure for holding the variable-length bit string resulting from losslessly compressing fixed-length instrument data, such as from a scan line of image data.

In parallel with AOST effort, GSFC is also engaged in the development of data compression technology. Data compression provides a viable means to alleviate the demands on onboard storage, communications bandwidth, station contact time and ground archive requirements. There are two types of data compression: a lossless technique, which guarantees full reconstruction of the data; and a lossy technique, which generally gives higher data compaction ratio but incurs distortion in the reconstructed data. Lossless compression generally results in variable length compressed data due to statistical nature of the original data. To satisfy the many science disciplines, lossless data compression has become the priority for development. After extensive research, the Rice algorithm [2,3] was chosen and developed into hardware. In 1991, a hardware engineering model was built in an application specific integrated circuit (ASIC) for proof of concept. This particular chip set was named the Universal Source Encoder/Universal Source Decoder (USE/USD) (Venbrux, 92)[4]. Later, it was redesigned with several additional capabilities and implemented in Very Large Scale Integration (VLSI) circuits using gate arrays suitable for space missions. The flight circuit is referred to as Universal Source Encoder for Space (USES). The fabricated USES chip is capable of processing data up to 20 Msamples/second and will take data of quantization from 4-bit to 15-bit [5]. In the following sections, we will provide a brief description of the data compression algorithm, the overall communication system, the AOST and physical link characteristics.

2.0 THE LOSSY LOSS DATA COMPRESSION ALGORITHM

The architecture of the Rice algorithm is shown in Fig. 3. It consists of a preprocessor to decorrelate data samples and subsequently map them into symbols suitable for the entropy coding module. This entity is a collection of options operating in parallel over a large entropy range. The option yielding the least number of coding bits will be selected. This selection is performed over a block

of J , typically 16, samples to achieve adaptability to scene statistics. An identification field of a fixed number of bits, determined by the input sample quantization levels, is used to signal the selected option for the block. The performance of this algorithm has been shown to be the same as that of a collection of Huffman codes on typical imagery [6] and has been tested on various instrument data [7].

3.0 SYSTEM DESCRIPTION

3.1 End to End System Description

The end-to-end system is depicted in Figure 4. The AOST is linked via an optical fiber to the RF SOC, which transmits the packetized data to the White Sands Complex (WSC) via a TDRS on a Ku band carrier. Data was recorded at the WSC and later transmitted to the AOST via NASA communication (NASCOM).

3.2 Source Equipment

3.2.1 Data Source

The source data can be either simulated instrument data or a video frame of data acquired from a CCD camera. In both cases, the data is first loaded into a frame buffer before each scan line is passed to the compression hardware which incorporates the USE chip. Each compressed scan line is then passed to the packetizer for further processing.

3.2.2 Packetizer and Multiplexer System Description

The packetizer takes data from the instrument, encapsulates it into CCSDS packets [8], and sends them over a fiber optic transmitter-receiver interface (FOXI) at a burst transfer rate of 80 Mbps. A separate packet is formed for each video scan line with the segmentation flag in the packet header used to treat an entire video frame as a large data block. The segment flag is set to "beginning of segment" for the first scan line of a video frame; it is set to "continuation segment" for intermediate scan lines; and it is set to "end of segment" for the final video scan line of a video frame. Frame syn-

chronization is derived through control signals in the FOXI interface.

The multiplexer operates in two modes: 1) path service mode where the multiplexer passes CCSDS packets through to the Wideband Transfer Frame Formatter (WTFF) without processing and 2) virtual channel access (VCA) service mode where the multiplexer produces multiplexing protocol data units (MPDU) and transmits them to the WTFF. Access to the output channel is granted based on availability and a round robin polling sequence. This polling occurs once every 400 ns, which is rapid enough that it results in a statistical multiplexing function. In general, the higher the packet rate for a channel, the more the number of requests and grants are given to that channel, causing access to be data rate driven. Details of the hardware are provided in [9].

3.2.3 Wideband Transfer Frame Formatter (WTFF)

The WTFF system [10] is designed to serve as a gateway providing transfer frame generation using a subset of the AOS services, as defined in Reference 1, for up to seven user virtual channels (VC) plus an idle channel. Data messages arriving from any one of the user VCs are buffered and then inserted into CCSDS standard format data transfer frames. These frames are padded with frames from the idle channel as necessary to maintain a preset data rate and are output on a single serial line. CCSDS Grade-2 service is provided by including a Reed-Solomon (RS) (255,223) error correcting code in each of the eight virtual channel circuits to form coded virtual channel data units (CVCDUs) formed from VCDUs or MPDUs, Fig. 2).

Virtual Channel: A VC unit receives user data and formats it into virtual channel frames (i.e., CVCDUs) at rates up to 100 Mbps. The frames are composed of five interleaved RS code words containing 255 bytes each. Each CVCDU is thus 1275 bytes (10,200 bits) long, including the RS encoding check symbols. When CVCDUs are appended with a frame synchronization pattern (32 bits), a channel access data unit (CADU) is created, which

can be transmitted over I or Q output data streams. Each VC is configured by the system controller upon initialization or during system re-configuration and has a unique ID (VCID) set by hardware. Data can be received as a fixed length data unit (MPDU in VCA service) or as CCSDS packets (Path Service).

PN Code Transition Generator: To ensure bit transition, the pseudo-noise (PN) transition generator is utilized. When it is, each byte of the CVCDU is XORed with a stored PN pattern before being sent through the multiplexer to the I or Q data outputs. The frame synchronization pattern is generated separately and is neither RS coded or changed by the PN generator.

3.3 Data Capture Equipment

All packetized data received at the WSC on the I channel was transferred to a workstation and processed predominantly with software tools. The Q channel signal was sent to a communications bit error rate (BER) test set for real time monitoring. The capture and analysis equipment is composed of a 32 Mbyte solid state memory connected to a Sun workstation via an ethernet; frame detection software; a hardware RS decoder; a software RS encoder; virtual channel and packet detection software; a software data decompressor; and a software image display package for the workstation.

3.3.1 Frame Detector

The original data as organized into frames by the WTFF is a series of bytes forming a data frame structure. Once transmitted from the WTFF, the bits in each byte are serialized and knowledge of the byte boundary is lost. The frame detection software searches the received file on a bit by bit basis to find the frame synchronization marker. To ensure that it is the frame marker and not a sequence of data bits that happened to be the same as the frame marker, the file position pointer was moved one frame length from the initial marker and data was examined for a second marker. In all of the data collected, a bit slip (or bit addition) was

never observed, making the need for a more elaborate frame synchronization strategy unnecessary.

3.3.2 Reed Solomon Decoding

After byte alignment and frame detection, RS decoding was performed. Each frame output from the RS decoder has a 16-byte status block appended to it. During the data flow with the compressed variable length image packets, the error rates were never severe enough to cause uncorrectable frames. During the portion of the test where the purpose was to evaluate the Ku band physical link using the CCSDS structure, frames that were found to be uncorrectable were not deleted. They were examined for burst error statistics along with the correctable frames. When uncorrectable frames occurred, the data portion of the frame was corrected by computer using knowledge of the data structure, and the parity portion was generated by re-encoding the data. CCSDS idle frames were also retained and RS decoded. The received raw data was compared with RS corrected data to determine burst error statistics.

3.3.3 Channel and Packet Detection

The file created after RS decoding was then processed by a software program called CHANNEL which examined the CCSDS frame header and produced a separate output file for each virtual channel identification (VCID) number that was found. Compressed video packets were assigned a particular application identification (APID) on a particular virtual channel. The PACKET program was run on the VCID file of interest and produced separate output files for each APID found in that VC. The APID file of interest contained variable length packets of compressed video data.

3.3.4 Decompressor

The packet file associated with the image frame data was further broken down into a separate file containing one compressed image frame. These 512 variable-length lines were decoded to generate one fixed-length image frame. In software, this

decoding routine performed the decompression algorithm and simulated the operation performed by the Universal Source Decoder (USD) chip, realizing the lossless data decompression algorithm. The routine used the reference sample at the start of each compressed 512 sample line as well as the header information at the start of each compressed 16 sample block to convert the data back to its uncompressed format. The resulting 512x512x8 bit binary image file was then accessed by a commercial display software package.

4.0 TEST PARAMETERS AND LINK ANALYSIS

4.1 Test Parameters

Testing of the AOST using variable length packet structures was performed by using losslessly compressed images. The first image transmitted was a prestored Landsat image. After the Landsat image was received, real time images from the video camera and packetized data from the simulators were routed through the source equipment and output on the WTFF I channel at 80 Mbps while the WTFF Q channel was not used. PN data transmitted on the RF Q channel was used to monitor the link BER. The WTFF I channel was configured to run at a continuous output rate of 80 Mbps (including idle channel fill frames) by using an 80 MHz crystal on its high speed output board. The WTFF configuration was as follows:

Virtual Channel #	Virtual Channel Mode	Data Source	Data Rate
1	Path Service	Simulator	16.0 Mbps
2	VCA Service	CCD camera	~28 Mbps
3	Path Service	Simulator	16.0 Mbps
63	Idle channel	WTFF	as needed

4.2 Link ANALYSIS

In addition to evaluating compressed variable length packets, this test allowed an examination of the channel characteristics of the K-band Single Access (KSA) return link through TDRSS. The primary objective was to evaluate a BCH code proposed for a LANDSAT-7 300 Mbps return link.

It was important to understand the error characteristics of the channel because the proposed binary BCH code (1023,993,3) can only correct 3 bit errors in a block of 1023 bits. This section will concentrate on the performance of the link used in this experiment in terms of BER vs. Eb/No. Link analysis was performed by transmitting a PN data pattern of NRZ-M data at 300 Mbps, QPSK modulated (150 Mbps on I, 150 Mbps on Q) through the KSA return channel. The data was recorded in one minute samples, and six pairs of C/No and BER measurements were taken at WSC with $2^{23}-1$ and 2^7-1 PN coded data. The measurements are summarized in Table 1. Using the measured carrier to noise density, the required effective isotropic radiated power (EIRP) values were also calculated.

TABLE 1. C/No, BER and RF SOC EIRP Data Points

Measured C/No (dB)	Measured Bit Error Rate	EIRP Required for Measured C/No
95.1	0.8e-3	51.2
95.3	3.5e-4	51.4
96.5	9.0e-5	52.6
98.9	2.0e-5	55.0
99.3	3.0e-6	55.4
99.9	2.0e-7	56.0

A plot of the six data points are shown in Figure 5 along with the ideal BER vs. Eb/No curve. The separation from the ideal curve varied with each measurement, about an average of 3.6 dB for five of the data points. This value was taken as the implementation loss. It is important to note that for the required EIRP calculation, several loss parameters were assumed based on estimates and the weather conditions of that day (which was cloudy with light rain) at the transmit site.

5.0 RESULTS

Since the WTFF used the CCSDS recommended (255,223) RS code, it was expected that as long as the BER was about 1×10^{-4} or better, the RS decoding would correct all of the errors. It was therefore expected that the decompression process would be error free and the reproduced image would be

identical to the digital version of the original. This was found to be true.

5.1 Image Quality

No streaks or drop outs occurred in the images. Since the compression technique was applied independently to each scan line, a decompression error would be expected to corrupt an entire line. Loss of a user data CCSDS frame would impact several image lines, but no corruptions were observed.

5.2 Channel Performance

One of the concerns was to evaluate the burst errors on the TDRS Ku band link. A test cannot examine all the parameters that vary over a satellite's lifetime, but it can at least provide a snapshot of some of those parameters. In order to simulate an end-to-end system, the data was recorded at the WSC and retransmitted to GSFC before being finally decoded.

During the link portion of the test, differential coding (NRZ-M) was used to avoid data inversion. The disadvantage of NRZ-M is that it causes each error to appear as two errors which may or may not be consecutive. In the data observed, no errors greater than 2 bits in length (errors were always consecutive) were observed when the proposed Landsat 7 power level was used. At lower power settings, where the BER was greater, longer burst lengths were seen (Table 2). For this analysis, a burst was arbitrarily defined as a group of incorrect and correct bits where there were no more than 11 consecutive correct bits. A burst always starts and ends with an error.

TABLE 2. BER vs. Error Burst Characteristics

BER	Max Length of Burst	Max Errors Per Burst
2.2e-5	2	2
1.3e-4	14	4
8.5e-4	22	6

The CCSDS frame sequence count was continuous with no gaps. Bit slips (or additions) were never observed in this or other AOST tests with the TDRSS.

6.0 CONCLUSION

The CCSDS recommendations for AOS data architecture have been put to a physical test with compressed data being multiplexed with several separate instrument channels. Losslessly data compressed images were received and decompressed without any distortion. The achieved compression ratio is about 1.8 for the Landsat image. This type of compressed data is very sensitive to channel errors which, if they occur, cause long streaks in the recovered images as results of the decompression operation. Therefore one can say that the data generation and recovery system worked as expected. No problems were introduced by the variable length packets resulting from lossless compression. The Ku band TDRSS link contained errors consistent with a purely thermal (random) environment for data transmitted from the GSFC. This analysis is based on statistics gathered during a short period, therefore no statement can be made about the burst environment that would be observed when the TDRSS antenna is pointed toward other areas of the Earth.

References

1. Advanced Orbiting Systems: Networks and Data Links: Architectural Specification, Consultative Committee for Space Data Systems, CCSDS 701.0-B-1 Blue Book, October 1989. Available from CCSDS Secretariat, Communications and Data systems Division (Code-TS), National Aeronautics and Space Administration, Washington, DC 20546, USA.
2. Rice, Robert F., "Some practical universal noiseless coding techniques," JPL Publication 79-22, 1979. Available from author, JPL, 4800 Oak Grove Drive, Pasadena, CA. 91109, USA.
3. Rice, Robert F., Pen-Shu Yeh and Warner H. Miller, "Algorithms for High-Speed Universal Noiseless Coding," Proceedings of the AIAA Computing in Aerospace 9 Conference, San Diego, CA, Oct. 19-21, 1993.
4. Venbrux, Jack, Pen-Shu Yeh and Muye N. Liu, "A VLSI Chip Set for High-Speed Lossless Data Compression," IEEE Trans. on Circuits and Systems for Video Technology, Vol. 2, No. 4, Dec. 1992.
5. MRC, "Universal Source Encoder for Space - USES," Preliminary Product Specification, Version 2.0, Microelectronics Research Center, University of New Mexico, 1993.
6. Yeh, Pen-Shu, Robert F. Rice and Warner H. Miller, "On the Optimality of a Universal Noiseless Coder," Proceedings of the AIAA Computing in Aerospace 9 Conference, San Diego, CA, Oct. 19-21, 1993.
7. Yeh, Pen-Shu and Miller, Warner, "Application Guide for Universal Source Encoding," NASA technical paper No. 3441, 1993.
8. Packet Telemetry Blue Book CCSDS 102.0-B-2 January 1987.
9. Fong, Wai, "AOST Technical Description: SEMP hardware description," Goddard Space Flight Center, 1994.
10. Betancourt, Jose and Folk, John, "AOST Technical Description: WTFF hardware description," Goddard Space Flight Center, 1994.

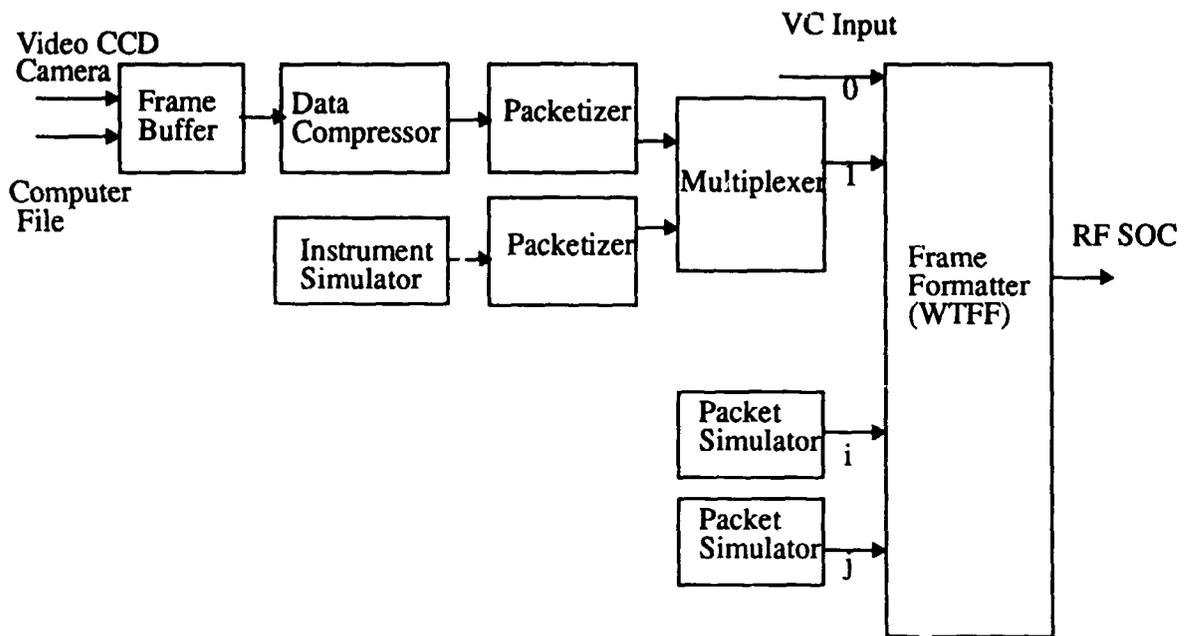


Figure 1. AO3 Test Bed Key Components

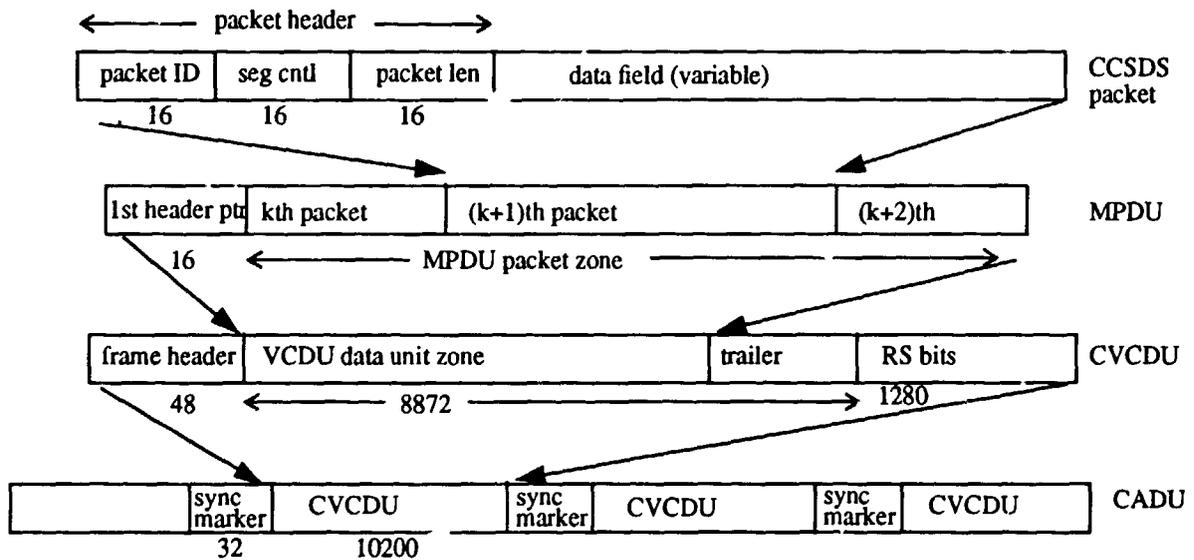


Figure 2. CCSDS Data Unit Structure

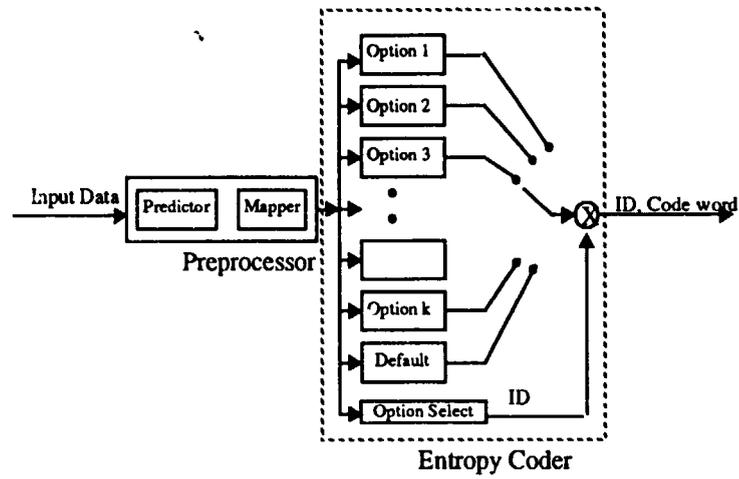


Figure 3. Rice Compression Algorithm

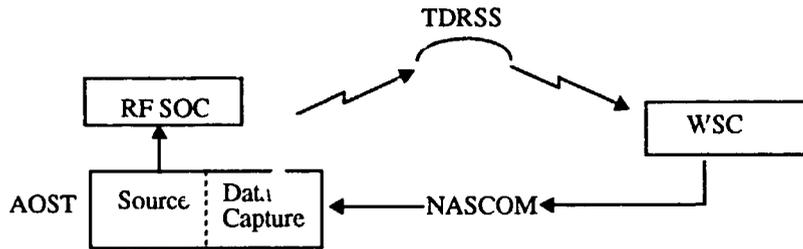


Figure 4. End to End Test Block Diagram

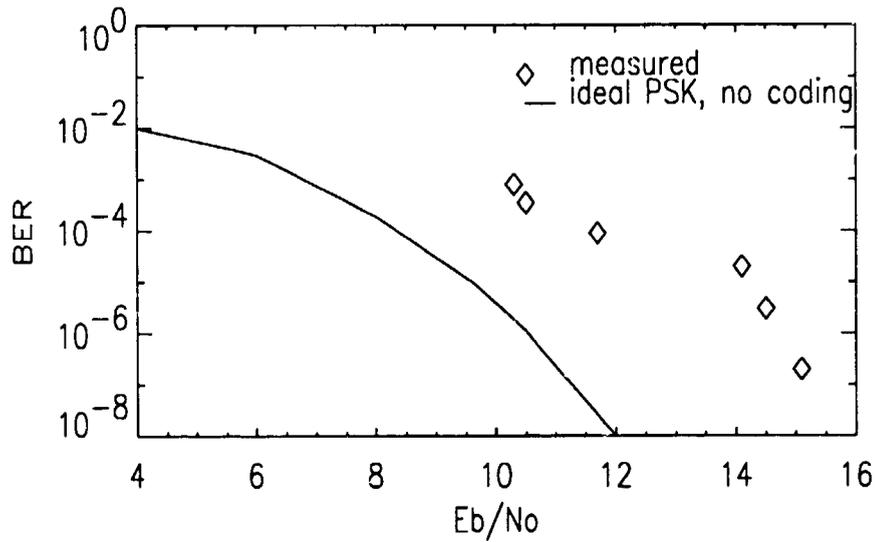


Figure 5. End-to-End Test BER vs. E_b/N_0

DEMAND ACCESS COMMUNICATIONS FOR TDRSS USERS

Mr. David Zillig
NASA GSFC, Code 531
Goddard Space Flight Center
Greenbelt, MD 20771

Dr. Aaron Weinberg
Mr. Robert McOmber
Stanford Telecommunications, Inc.
1761 Business Center Drive
Reston, VA 22090

p-10

ABSTRACT

The Tracking and Data Relay Satellite System (TDRSS) has long been used to provide reliable low- and high-data rate relay services between user spacecraft in Earth orbit and the ground. To date, these TDRSS services have been implemented via prior scheduling based upon estimates of user needs and mission event timelines. While this approach may be necessary for large users that require greater amounts of TDRSS resources, TDRSS can potentially offer the planned community of smaller science missions (e.g., the small explorer missions), and other emerging users, the unique opportunity for services on demand. In particular, innovative application of the existing TDRSS Multiple Access (MA) subsystem, with its phased array antenna, could be used to implement true demand access services without modification to either the TDRSS satellites or the user transponder, thereby introducing operational and performance benefits to both the user community and the Space Network.

In this paper, candidate implementations of demand access service via the TDRSS MA subsystem are examined in detail. Both forward and return link services are addressed and a combination of qualitative and quantitative assessments are provided. The paper also identifies further areas for investigation in this ongoing activity that is being conducted by GSFC/Code 531 under the NASA Code O Advanced Systems Program.

1.0 INTRODUCTION

For over a decade, the Tracking and Data Relay Satellite System (TDRSS) has been providing reliable, low- and high-data rate, two-way relay services between low-earth orbit user spacecraft and the ground. To date, these TDRSS services -- both single access (SA) and multiple access (MA) -- have been provided to users via structured scheduling. The scheduling is completed days in advance of the actual service event, and based upon estimates of user needs and mission event timelines. This approach has historically been, and may continue to be, necessary for certain classes of users and operational scenarios (e.g., real-time relay of time-critical, wideband science

data). On the other hand, newly emerging users and operational scenarios may be capable of taking advantage of certain TDRSS services on demand. Such users may include emerging small-satellites and certain non-space users (e.g., aircraft).

Toward this end, Code 531 at NASA/GSFC, under the sponsorship of the Code O Advanced Systems Program, has been identifying and assessing a variety of Demand Access (DA) concepts that reflect the following Statement of Need:

Dramatically enhance user accessibility to TDRSS, by accommodating service requests on demand. The new DA services should:

- Support the broadest possible range of users, with particular emphasis on emerging small-sats and other unique users (e.g., NASA aircraft).
- Emphasize low-data-rate TT&C services.
- Ensure low-cost Space-Network (SN)/user operations.

Within the framework of SN operations, the above DA service needs are addressed here by focusing on the innovative utilization of the TDRSS Multiple Access (MA) Forward and Return services. The rationale for MA utilization -- in contrast to Single Access (SA) -- is due to the unique nature of the electronically steerable MA antenna, its amenability to very rapid configuration, and its much higher availability than SA (especially on the MA Return link). Further insights into MA service utilization for DA are provided via subsequent discussions in the body of the paper.

The purpose of this paper is to provide representative, interim results of ongoing DA study activities that are being conducted by GSFC Code 531. The organization of this paper is as follows. Section 2 provides an overview of DA operations, including an architectural definition and a description of candidate DA service applications. Sections 3 and 4 follow with respective descriptions and unique features of candidate Forward and Return link DA operations concepts; qualitative and quantitative performance results are also presented. Section 5 concludes with a Summary and Observations.

2.0 DEMAND ACCESS OVERVIEW

A first logical question to ask is: **What is meant by "ideal" Demand Access?** Within the present SN framework, this question is addressed in Figure 1. As seen, the key ingredients of interest, for both the Forward and Return links, may be summarized as follows:

- No NCC scheduling.
- Essentially immediate SN reaction to a user service request.
- No contention with other users (i.e., 100% service satisfaction).

Given that "ideal" DA is not achievable via the SN, the second question that arises is: **How close to "ideal" DA can be achieved via innovative utilization of MA service capabilities?** As will be shown in Sections 2 and 3, the answer to this question is: **Remarkably close to "ideal" DA is achievable via: suitable user operations concepts, modest ground augmentations, and appropriate applications of the highly capable MA resources!** To be emphasized is the fact that the proposed DA capabilities are achievable with the existing on-orbit TDRSS spacecraft and existing user spacecraft transponders.

Prior to proceeding with the detailed discussions it is useful to gain some insight into potential applications of DA. A listing of candidate DA services and relevant observations is given in Table 1. As seen, the DA services can benefit both SN operations (e.g., BRTS) and user communications and tracking, by introducing simplicity, flexible and efficient use of resources, robustness, enhanced performance, and the accommodation of new/unique applications.

To complement Table 1, Figure 2 illustrates a few candidate DA scenarios. As seen, the applications are diverse, and the Forward and Return portions may be applied either separately or jointly. It should also be noted that a conscious effort is being made here to include the potential for a direct, real-time INTERNET interface between the user spacecraft and the Principal Investigator.

It is apparent that a variety of system-level, cost, performance, and technology considerations must be addressed in assessing candidate DA concepts.

Representative considerations and evaluation criteria being applied as part of the GSFC Code 531 study are as follows:

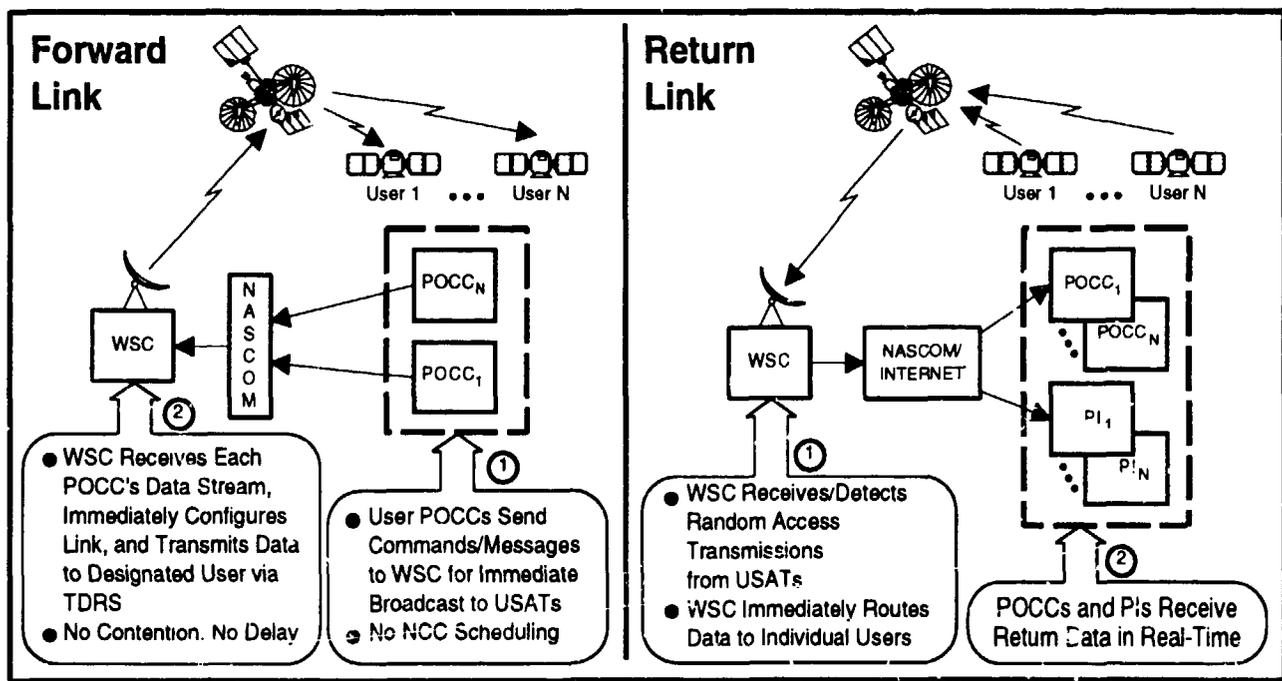


Figure 1: What is "Ideal" Demand Access?

Table 1: Candidate DA Services

DA Service	Observations
TDRS tracking; BRTS no longer scheduled for nominal TDRS tracking	<ul style="list-style-type: none"> • SN benefit • Potentially improved tracking performance
User one-way return tracking	<ul style="list-style-type: none"> • Via single TDRS with sufficiently stable oscillator • Differenced doppler for portion of user orbit, cancels oscillator drift • Potential to eliminate coherent turnaround and reduce transponder power consumption
Low-rate command load	<ul style="list-style-type: none"> • Multiple times per day • As desired by user
Low-rate telemetry	<ul style="list-style-type: none"> • As desired by POCC and/or PI • Echoes or ACKs of DA CMDs • Immediate access to ground problems (via transmission initiated by spacecraft) • Immediate access to data on unexpected Targets of Opportunity • Potential E-mail interface to PI via WSC/INTERNET interface
User or SN testing without loading SN scheduled services	<ul style="list-style-type: none"> • FWD and RTN user tests • SN SMA FWD/RTN tests via BRTS, or via cooperative spacecraft, or via non-ops spacecraft (e.g., COBE)
<ul style="list-style-type: none"> • Apply inner TDRSs for DA and outer TDRSs for scheduled service • Option: DA augmentation via GRTS 	<ul style="list-style-type: none"> • Would provide near-global, 24 hour DA service • Would simplify DA/WSC operational interface (may enhance automation potential) • May simplify supporting HW/SW upgrades required at WSC • Low complexity DA operations may be ideal application of GRTS
Provide FWD messages to user community whenever MA FWD is not otherwise being used	<ul style="list-style-type: none"> • Maximizes utilization of MA FWD resource • Can be used to periodically provide entire user community with useful housekeeping data; e.g.: <ul style="list-style-type: none"> - Time of day - SN schedule information that is unclassified - Clock/oscillator corrections; periodic synchronization to WSC CTFS - TDRS and USAT state vector updates (effectively eliminates need for on-board nav)
FWD/RTN DA can also accommodate unique ground users (e.g., polar)	Take advantage of increased inclination of aging TDRSs (e.g., F1)

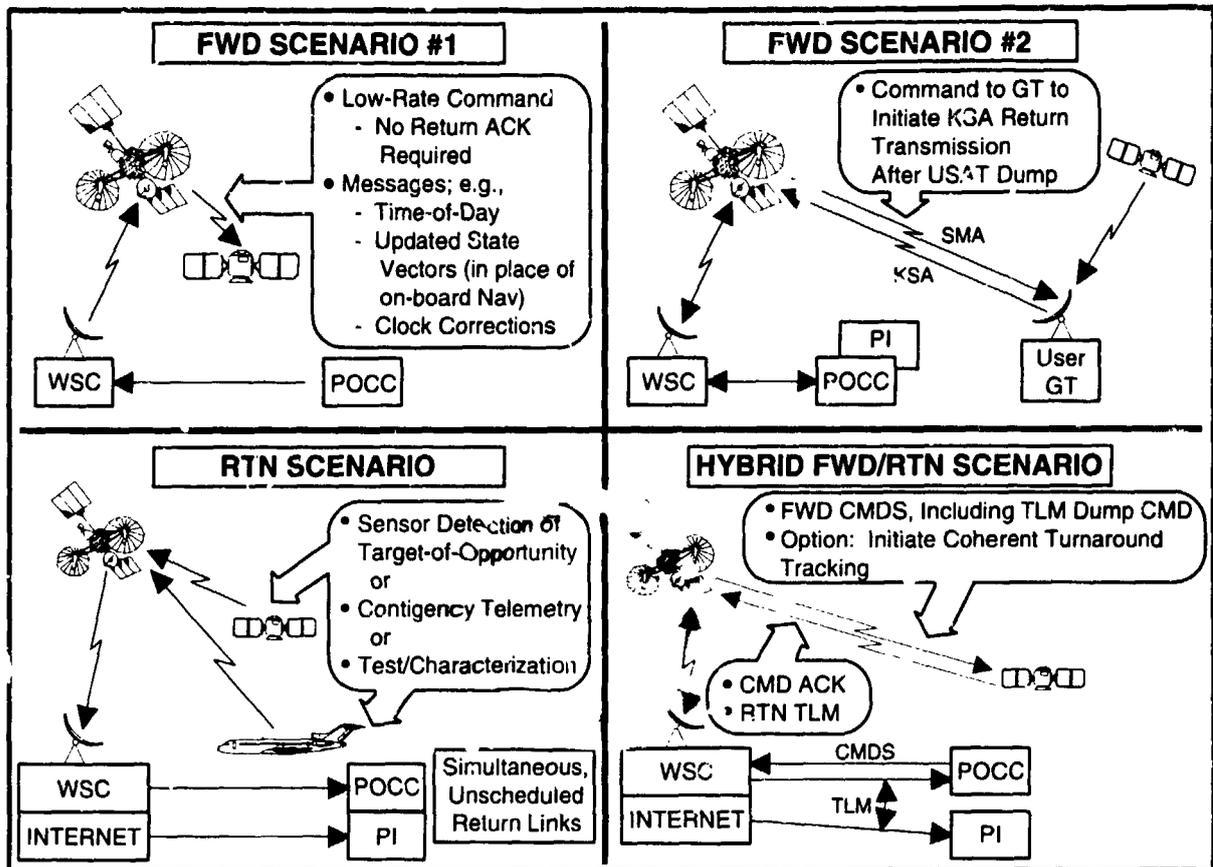
- FWD/RTN link availability (user satisfaction; waiting time).
- FWD/RTN link data throughput.
- SN impacts -- implementation and cost (e.g., White Sands Complex (WSC); new elements and interfaces; application of 1 vs 2 vs 3 TDRS constellation nodes for DA).
- User impacts -- implementation and cost (e.g., POCC; transponder).
- Operational risk and robustness (e.g., prime/backup; transition, robust accommodation of an expanding user population that desires DA service).
- End-to-end cost per bit (overall NASA perspective).

3.0 MA FORWARD DEMAND ACCESS (MAFDA)

Preliminaries

The TDRS MA capability relies on the 30 element phased array antenna on the TDRS spacecraft, with each element providing a 2.6° beamwidth (i.e., greater than earth diameter from geostationary orbit). The MA capability provides both FWD and RTN link services. The FWD capability, and its application to DA, is addressed in this section.

The MA FWD capability involves application of 8 - 12 of the 30 elements, which are phased via ground control, to point to and service a single user at a time.



07/15/94 MIS0976/KN1067

Figure 2: Candidate Demand Access Scenarios

As such, the MA FWD link is generally considered "a scarce resource" and must be applied wisely to maximize its applicability to DA. The TDRS MA FWD EIRP is 34 dBW, which accommodates .5 - 1 kbps, via a user near-omni antenna, if the FWD data is convolutionally encoded.

Because of the "one user at a time" feature, an appropriate quantitative performance assessment of MAFDA requires explicit utilization of a user mission model. For the purpose of this paper, the mission model employed reflects the seven baseline SN users for the year 2000, augmented with 10 hypothetical users that reflect a diversity of current and planned small-sat characteristics (Table 2). The rationale for this augmentation is to "stress" the proposed MAFDA system, and determine how robustly the system performs under high loading conditions, and ultimately what the system capacity is.

Description and Assessment of Candidate MAFDA Concept

As part of the Code 531 study activity, several candidate MAFDA concepts have been addressed to date. The present paper addresses a concept that appears to be a leading contender. The high level architecture, and associated operations concept ingredients, are illustrated in Figure 3. In this figure, the circled numbers represent the order time sequence of events. Of particular importance is the ability of each POC to send messages, as desired, to its respective user spacecraft via the TDRSS ground terminal (WSC). Note that the NCC is not in the service request path, but does receive associated status information, as it must. Also to be noted is that the POC transmissions are relayed to WSC via a preprocessor that:

- Queues messages on a first-come-first-serve basis (i.e., the proposed DA scheme precludes priorities).

Table 2: Representative Small-Sat Orbital Characteristics

#	Basis of Orbit	Eccentricity	Inclination (deg)	Period (minutes)	Altitude (km)	Argument of Perigee (deg)	Right Ascension (deg)	Spacecraft Altitude Rotation Axes
1	TIMED Ellipt.	0.3089	95	152	--	0	0	Pitch Axis
2	TIMED Circular	0	49	92	400	0	40	Pitch Axis
3	SAMPEX	0.0089	82	97	612	0	80	Pitch Axis, Yaw Axis
4	TIMED Circular	0	49	92	400	0	100	Pitch Axis
5	SAMPEX	0.0089	82	97	612	0	225	Pitch Axis, Yaw Axis
6	TIMED Elliptical	0.3089	95	152	--	270	180	Pitch Axis
7	Low Inclination	0	28.5	92	400	0	100	Pitch Axis
8	Low Inclination	0	28.5	97	600	0	200	Pitch Axis
9	Critical Inclination	0.1	63.4	120	--	270	0	Pitch Axis
10	Critical Inclination	0.1	63.4	120	--	90	180	Pitch Axis

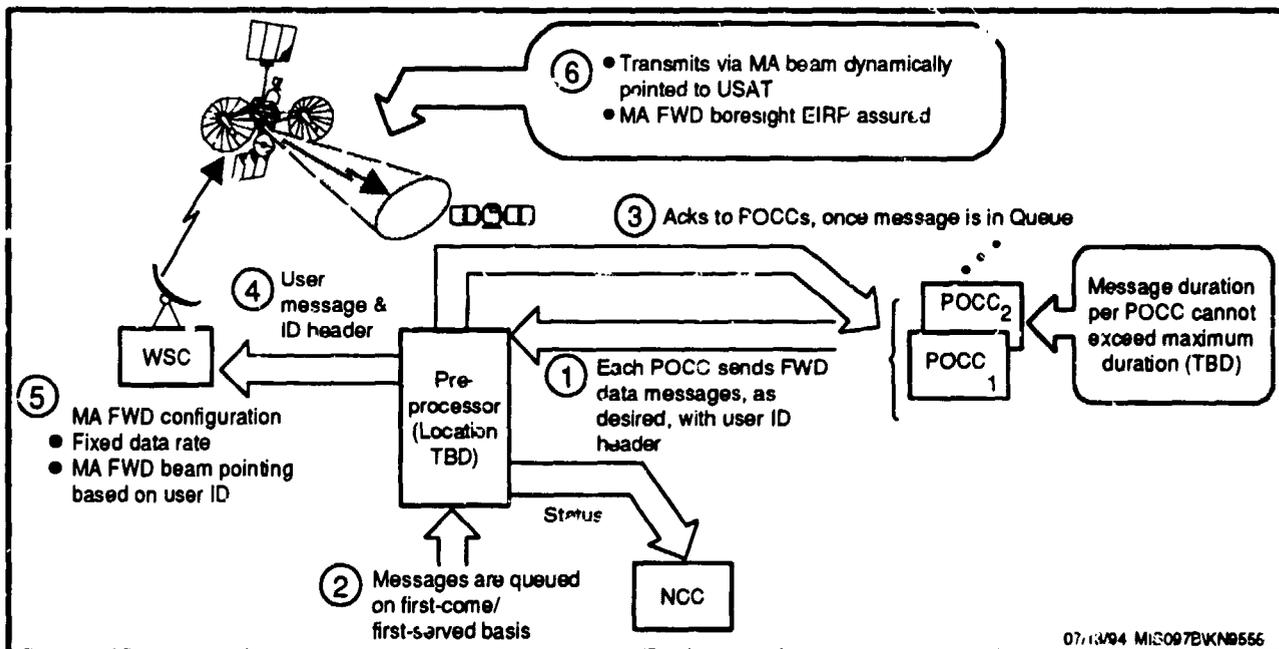


Figure 3: Candidate MAFDA Approach

- Sends acknowledgement to the POCC, once the message is successfully queued.
- Identifies the POCC's ID and sends the appropriate information to WSC for rapid link configuration.

Within this framework, WSC can configure and establish the MA FWD link within 30 seconds. As such, almost immediate service accommodation is provided, as long as the queue is empty when a POCC transmits a message. Queue contents vs time is addressed in more detail shortly.

More insight into the ground interface is presented in Figure 4. The middle block shown represents the preprocessor of interest, which is anticipated to be an automated/low-rate/low-complexity processor, that perhaps can be embedded in a small workstation. Its physical location, yet to be established, is currently not envisioned to be a critical system factor. Several additional points of interest are as follows:

- For simplicity, the present concept assumes that the MAFDA data rate is the same for all users.
- The message duration per DA service is assumed to be fixed (e.g., 2 minutes).
- WSC, as currently implemented, contains all necessary user database information to permit rapid extraction of key user link parameters (e.g., state vectors and oscillator frequency).

Figure 4 also illustrates a representative structure for a user message. The duration of each such message,

and the transmission duty cycle per POCC, are key system design parameters. Discussion follows.

Given that a single MA FWD link exists per TDRS, it is clear that the user satisfaction, via the proposed DA service concept, will be high only if the message duration and transmission duty cycle per POCC are reasonably sized. To gain quantitative insight into these matters, as well as insight into how many TDRSS constellation nodes should be allocated to MAFDA, a comprehensive and flexible simulation capability has been developed. The simulation propagates all user orbits of interest, permits variation of message duration and duty cycle, and randomly inserts POCC messages into the queue. The simulation can also assess DA operations via one, two or three TDRSS constellation nodes.

Figure 5 provides one illustrative set of simulation results, wherein the 10 small-sats of Table 2 are treated as DA users and are accommodated via the single TDRS node at 85°E; BRTS is also included in the simulation as a priority user, given the requirement for TDRS orbit determination. All other TDRSS users -- i.e., the nominal year 2000 users (such as HST) -- are accommodated as regularly scheduled users via the other two TDRS constellation nodes. The following key observations result:

- Messages of up to 2.5 minute duration per orbit can be accommodated with little or no queue waiting time; < 1% probability that waiting time exceeds 2.5 minutes.

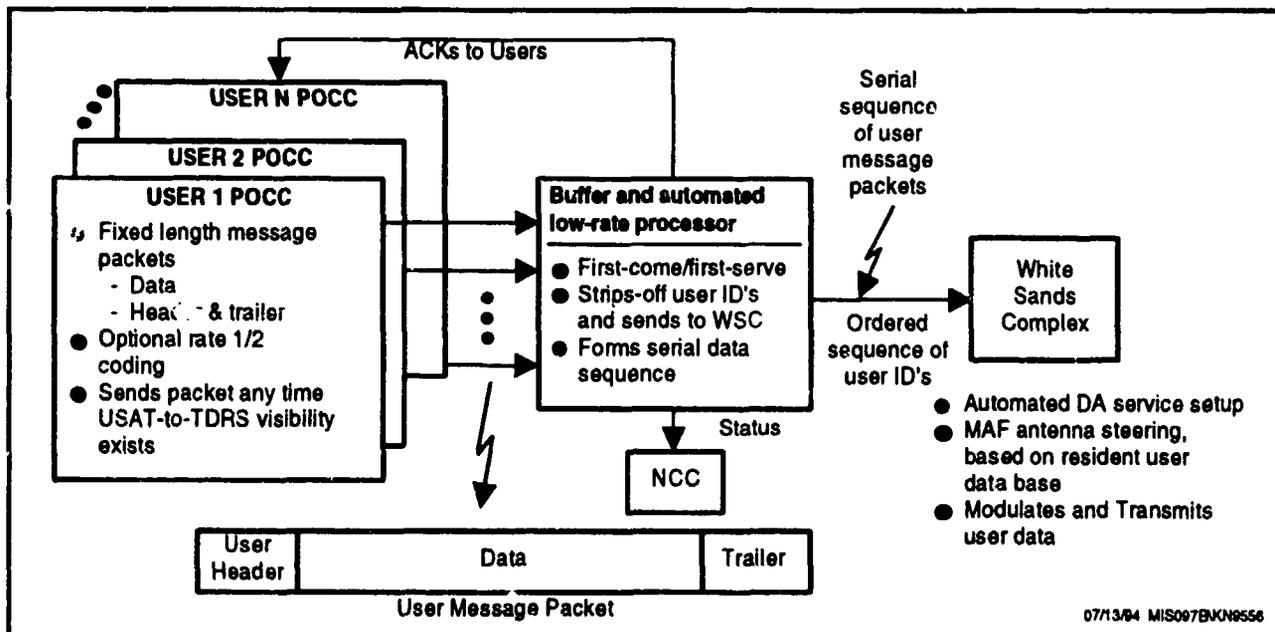


Figure 4: Candidate User Ground Interface for MAFDA

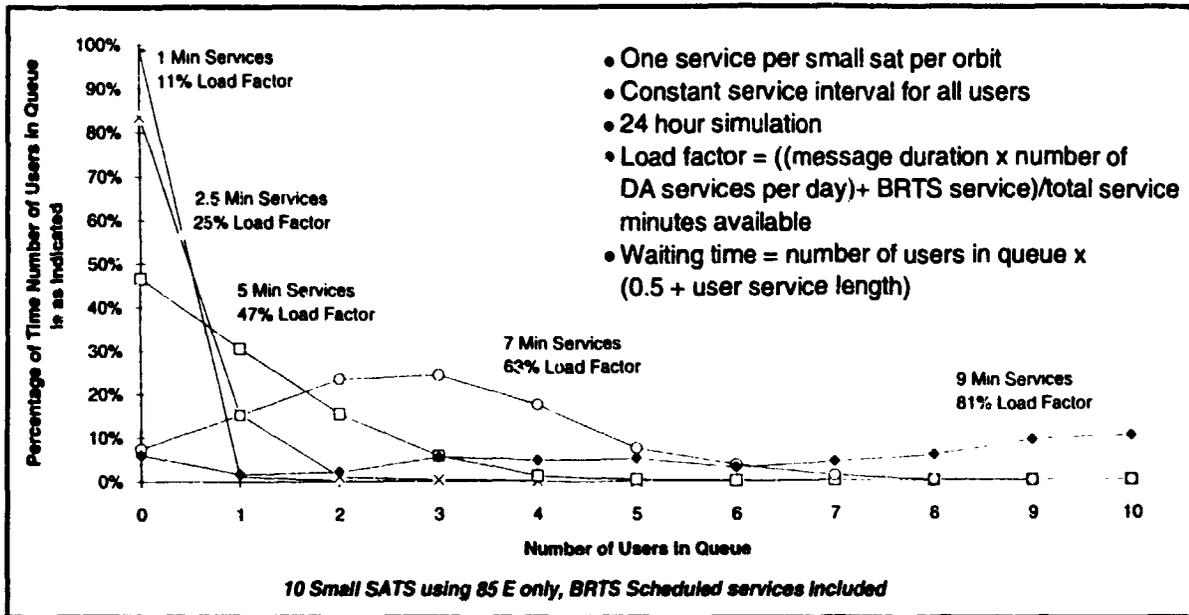


Figure 5: MAFDA Simulation Results

- Waiting time increases to a maximum of 15 minutes for 5 minute message duration per orbit.
- The queue grows unacceptably for message durations > 5 minutes, and instability occurs for message durations exceeding 8 minutes.

Additional simulation results were generated for DA services via 2 and 3 nodes, including results that combine scheduled and DA services per constellation node. General conclusions, to date, include:

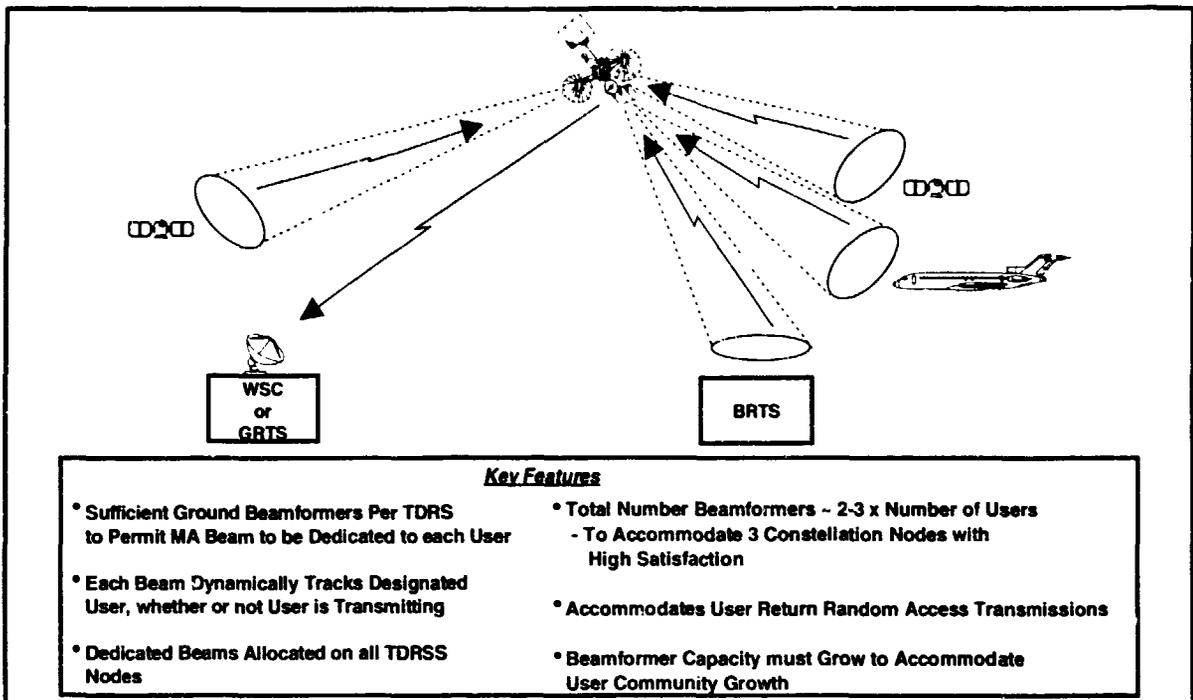
- High DA service satisfaction and extremely short waiting times are achievable, even for a significant user population, subject to a reasonably designed DA operations concept; e.g.,
 - One POCC message per unit orbit.
 - Message duration on the order of 2 - 5 minutes, with larger durations acceptable if more than one TDRS node allocated to DA.
- Load factor determines DA performance, regardless of number of nodes allocated to DA.
 - Negligible waiting time for $\leq 25\%$ load factor.
 - Maximum weighting time increases to ~ 3 x message length for load factor $\sim 50\%$.
 - Instability occurs for load factor $> 75\%$.
- Dedication of node(s) to DA leads to slightly higher satisfaction with reduced operational flexibility, than integration of scheduled and DA users on a node.

- One service per small sat per orbit
- Constant service interval for all users
- 24 hour simulation
- Load factor = $((\text{message duration} \times \text{number of DA services per day}) + \text{BRTS service}) / \text{total service minutes available}$
- Waiting time = $\text{number of users in queue} \times (0.5 + \text{user service length})$

4.0 MA RETURN DEMAND ACCESS (MARDA)

For the MA RTN link, all 30 elements of the phased array are employed, and a unique ground-based beamforming capability is applied to enable support of many users simultaneously; the formed-beam G/T is on the order of 0 dB/K, which supports at least 1 - 2 kbps data rate for a user EIRP of -7 dBW. The current baseline, for operations with the new Second TDRSS Ground Terminal (STGT), is simultaneous support of 10 users. To be noted, however, is that this can be greatly expanded via utilization of additional ground-beamformers. As such, the MA RTN capability is not a scarce resource, and considerable operational flexibility is achievable.

As part of the Code 531 study activity, two primary candidate MARDA concepts have been examined to date. The first of these concepts is illustrated in Figure 6a. To accommodate random access return link user transmissions, each user is continuously covered by a dedicated, dynamically steered TDRSS MA RTN antenna beam. A key requirement in this approach is that enough beamformers and demodulators are available at WSC. Since WSC equipment chains are dedicated to each TDRSS node, the possibility of an uneven distribution of users among the TDRSS nodes means that the total number of needed beamformers/demodulators exceeds the number of MARDA users. Simulations to date have indicated that 10 to 11 beamformer/demodulator combinations per each of three TDRSS nodes are



7/15/94 MIS097BKN5210

Figure 6a: Candidate MARDA Approach 1 - Dedicated MA RTN Beams

required to assure continuous coverage of all MARDA users for the stressed mission model previously described (7 nominal users plus 10 new small-sats). This quantity of MA beamformers/demodulators approximately reflects the baseline STGT/WSGT capacity, but MA augmentation would be required for the site supporting the 85°E node. Clearly, however, additional MA augmentation would be required for larger user populations and/or users that are more "stationary" in nature than spacecraft. However, with limited user data rates and the rapidly advancing state-of-the-art in communications technology, the cost of such beamformer/demodulator combinations may be kept within acceptable bounds.

By keeping an MA RTN antenna beam centered on each user, the users are assured the full MA RTN G/T during MARDA operations. The corresponding operational complexity is the need for dynamic MA antenna steering and dynamic receiver configuration at WSC to account for both user dynamics and handovers between the TDRSS nodes. Associated operational assessments are in progress.

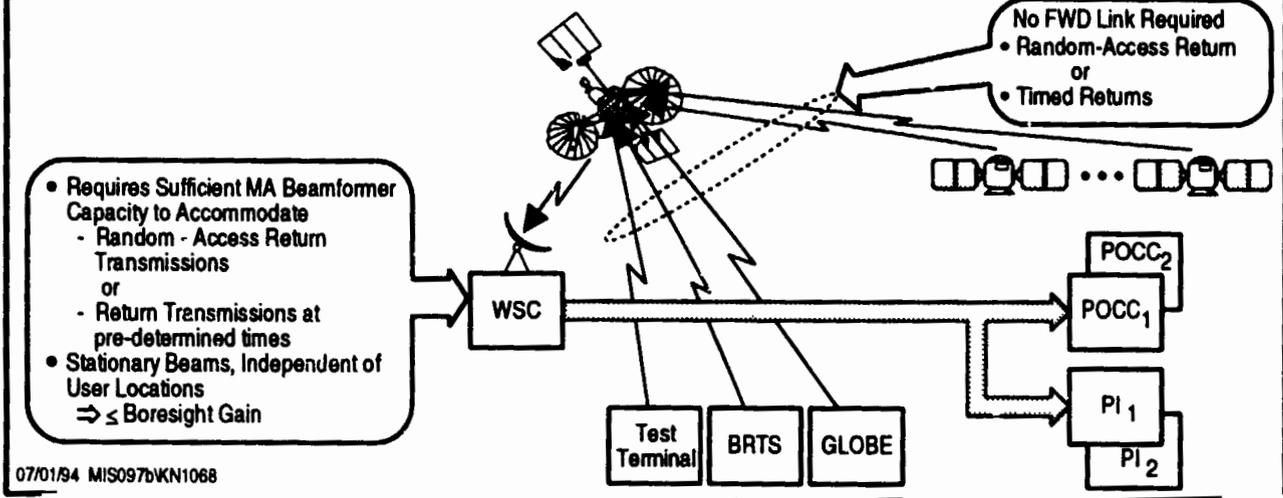
The second approach to MARDA implementation is illustrated in Figure 6b. The approach uses a set of stationary MA RTN beams at WSC to cover the field-of-view of each TDRSS node. A set of low-rate demodulators is provided for each beam, with each

demodulator matched to a user-unique PN code. Each such demodulator is always available to acquire and demodulate a user's transmissions as it passes from beam to beam. As in the first approach, full random access transmissions by TDRSS users are supported. However, unlike the earlier architecture, no prior knowledge of user position or dynamic MA antenna steering are required. But note that a user does not achieve the full TDRSS MA RTN boresite antenna G/T if it is near the edge of one of the fixed beams.

Analyses to date have indicated that a pattern of 19 beams per each of three TDRSS nodes can be used to provide near full-time coverage of TDRSS users as illustrated in Figure 7. The beamwidth used in this Exhibit is 4.34° -- achieved using defocusing of the TDRSS MA RTN array which has a normal beamwidth of ~3.2°. The number of beams per TDRS is approximately doubled if the 3.2° beamwidth is desired. Array defocusing is at the cost of some loss in TDRSS MA RTN G/T performance; the cost and performance trades between the number of needed MA RTN beams and the potential for TDRSS array defocusing, is continuing to be addressed.

Figure 8 illustrates a candidate implementation of this second MARDA approach -- showing the bank of low-rate demodulators associated with each of the

CANDIDATE MARDA APPROACH 2



07/01/94 MIS097bKN1068

Figure 6b: One-Way MA Return

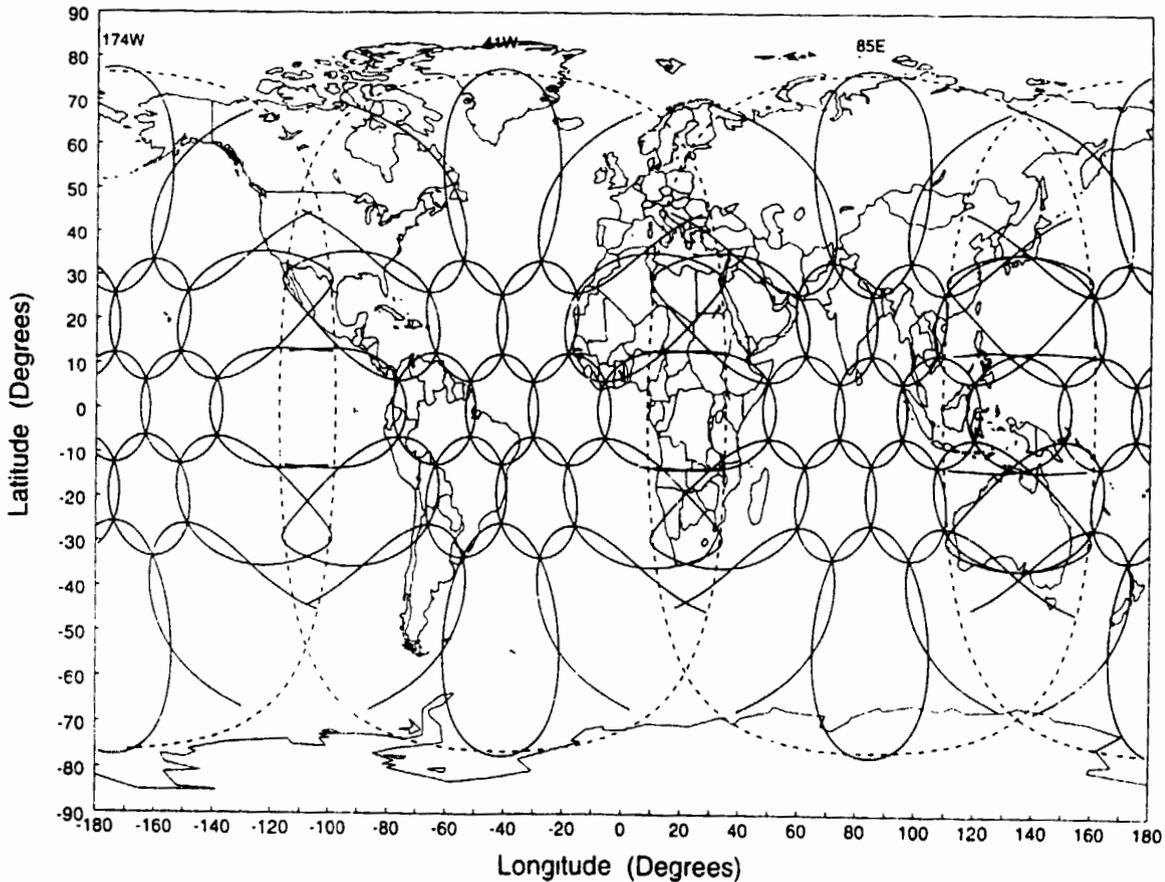
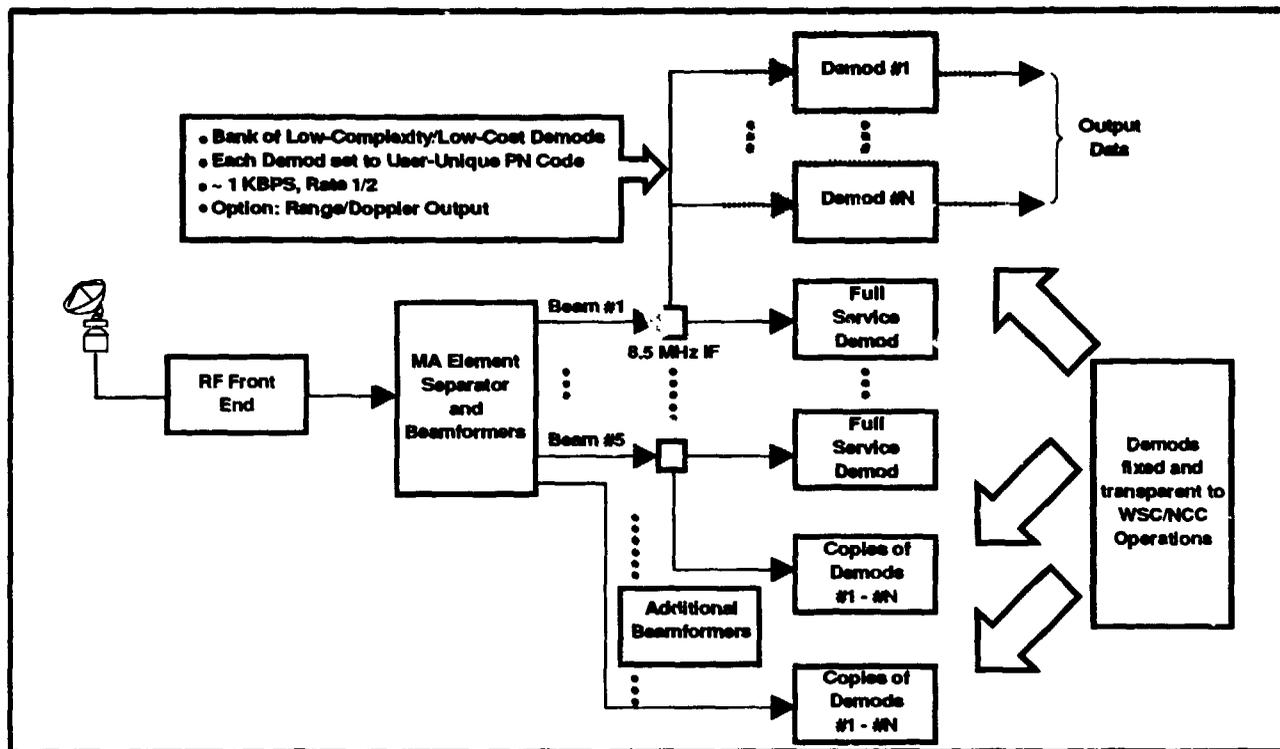


Figure 7: TDRS MA Return Beams on the Surface of the Earth; Earth Coverage Pattern
 TDRS Elevation Mask: 5 Degrees, Inclination: 0 Degrees, Beamwidth: 4.34 Degrees



7/05/94 MIS007BKN5212

Figure 8: Candidate WSC/GRTS MARDA Augmentation

fixed MA RTN beams. Because the number of beamformers is fixed and independent of the number of TDRSS users, the architecture has the potential to provide service to a very large number of TDRSS MA users with only the addition of demodulators needed to add new users. As long as user data rates are kept low (i.e., on the order of a few kbps), self-interference among the CDMA users can be kept negligible. Based on advancing technology, beamformer and demodulator size and cost can be kept low, and this represents an active area for examination.

While this second MARDA approach is oriented towards support to a larger DA user community, it also offers the opportunity to provide DA TDRSS services to new user types not previously considered. For example, the use of fixed MA RTN beams which cover Earth means that an appropriate user on the surface of the Earth could obtain TDRSS return service on demand regardless of location. Such service could greatly benefit geographically dispersed sets of low data rate users. One such example is the GLOBE Program -- a U.S. Government initiative to establish an international partnership for environmental monitoring by students on a worldwide basis. In its initial phases, the GLOBE Program will use a limited set of fixed TDRSS MA RTN beams to demonstrate transfer of science data between remotely

located students and science processing centers. Such a set of users is entirely consistent with this implementation of the MARDA architecture.

5.0 SUMMARY AND OBSERVATIONS

In Section 2.0, the concept of "ideal" user demand access service was defined as service initiation whenever desired, with no NCC scheduling, and little or no contention for service with other users. As described above, innovative application of the TDRSS MA forward and return service capability appears well suited to providing near ideal demand access services to low-rate TDRSS users. The approaches for implementing forward and return DA service have the key advantage of not requiring changes in the user transponder implementation or in the existing constellation of TDRSS satellites.

On-going GSFC Code 531 activity is oriented towards detailed examination of the relative merits of each of the available service options described above. In particular, the operational and implementation impacts associated with each approach are currently being addressed. It is expected that the current effort will lead to definition of a candidate demand access capability that provides both enhanced service to the TDRSS user community while at the same time simplifying Space Network operations.

The GRO Remote Terminal System

David J. Zillig (NASA/GSFC/Code 531.2)

Joe Valvano (AlliedSignal Technical Services Corporation)

ABSTRACT

In March 1992, NASA HQ challenged GSFC/Code 531 to propose a fast, low-cost approach to close the Tracking Data Relay Satellite System (TDRSS) Zone-of-Exclusion (ZOE) over the Indian Ocean in order to provide global communications coverage for the Compton Gamma Ray Observatory (GRO) spacecraft. GRO had lost its tape recording capability which limited its valuable science data return to real-time contacts with the TDRS-E and TDRS-W synchronous data relay satellites, yielding only approximately 62% of the possible data obtainable. To achieve global coverage, a TDRS spacecraft would have to be moved over the Indian Ocean out of line-of-sight control of White Sands Ground Terminal (WSGT). To minimize operations life cycle costs, Headquarters also set a goal for remote control, from the WSGT, of the overseas ground station which was required for direct communications with TDRS-1.

On August 27, 1992, Code 531 was given the go-ahead to implement the proposed GRO Relay Terminal System (GRTS). This paper describes the Remote Ground Relay Terminal (RGRT) which went operational at the Canberra Deep Space Communications Complex (CDSCC) in Canberra, Australia in December 1993 and is currently augmenting the TDRSS constellation in returning between 80-100% of GRO science data under the control of a single operator at WSGT.

INTRODUCTION

The GRO Remote Terminal System (GRTS) was implemented in a fast-paced, low-cost effort to close the gap in TDRSS coverage over the Indian Ocean to increase the science data return from the GRO spacecraft. To cover the ZOE, which is caused by earth-blockage of the 2-TDRS constellation, the oldest TDRS spacecraft, TDRS-1, was drifted to 85 degrees east longitude. At that location it could provide the additional coverage for GRO but, as a result, was out of line-of-sight communications with WSGT. To provide control and monitoring of the TDRS and to return the data relayed from GRO to the US, two new nodes (connected by Intelsat links) were required to be

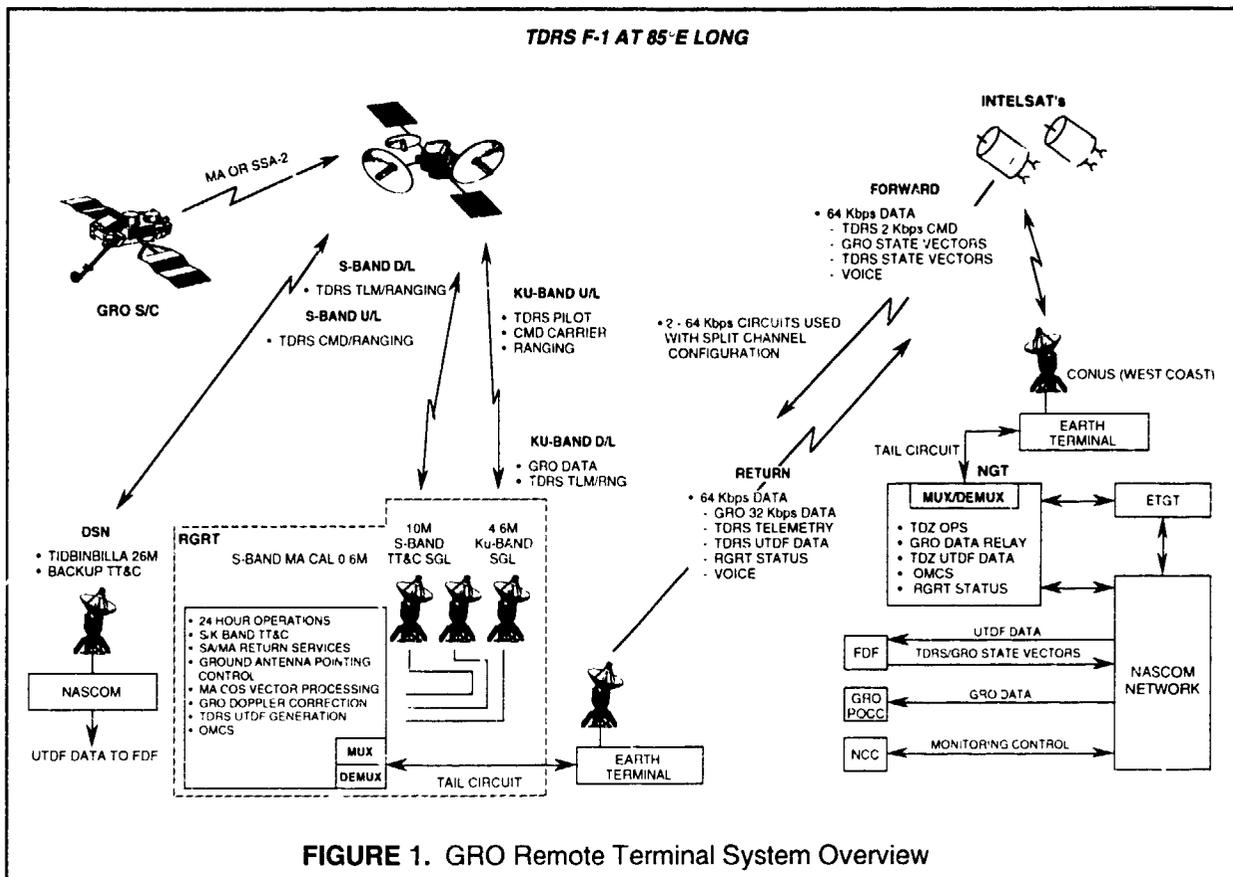
added to the Space Network, one at WSGT in New Mexico and the other at the new overseas relay site in Australia. Figure 1 provides an overview of the GRO Remote Terminal System.

The node at WSGT, named the Extended TDRSS Ground Terminal (ETGT) retains all the command and telemetry processing and unique software for TDRS-1, the spacecraft controller personnel and the interfaces to GSFC for GRO data, the NCC and FDF. As with the other TDRS spacecraft, complete control remains at WSGT/ETGT. In this case, however, control is exercised via redundant NASCOM 64 kb/s Intelsat links to the RGRT at CDSCC in Australia.

The node at CDSCC, RGRT, emulates (at a much reduced scale) the ground terminal equipment at WSGT (the antennas, receivers, transmitters and computer controls). The commands are received by commercial carrier from ETGT and then transmitted from RGRT up to TDRS-1 and the status telemetry is downlinked from TDRS-1 to RTGT and then relayed back to ETGT via Intelsat. Range and Doppler measurements for TDRS orbit determination are made at the RGRT and then communicated through WSGT to the Flight Dynamics Facility at GSFC.

Science data from GRO's onboard instruments is collected and radiated at 32 Kb/s in spread spectrum S-Band mode to TDRS-1 located at 85 degrees east longitude. TDRS-1 receives the signal using either the 30-element phased array Multiple Access (MA) antenna or the 4.9 m S-Band Single Access (SSA) dish antenna. The S-Band return link from GRO is translated to Ku-Band onboard TDRS and transmitted to the ground at RGRT via the TDRS Space to Ground Link (SGL) antenna. The 32 kb/s data is received by either the MA or SSA receiving equipment, despread, demodulated and transmitted in real time via Intelsat to ETGT and then on to the GRO POCC.

Since the GRO POCC has ample opportunity for spacecraft commanding through the TDRS-E and TDRS-W, only return link user services are incorporated in the RGRT design.



RGRT IMPLEMENTATION - OVERVIEW

Since no equipment from the WSGT or Second TDRSS Ground Terminal (STGT) could be made available for this implementation, the system design relied on a mixture of commercial off-the-shelf (COTS) and GSFC custom designed equipment (both in-house and contractor-designed).

The RGRT design, like WSGT, is essentially two ground terminals in one - the TDRS TT&C system to control and monitor the TDRS and the User Service system to receive the science data from GRO.

The critical TDRS TT&C function was implemented using proven Ground Network (GN) receiver/exciter/ranging (RER) equipment - the same equipment used in the GN and the DSN 26 meter subnet for TDRS launch support and on-orbit TT&C backup. Redundant receivers and exciters and a single ranging equipment, switchable to either the S-Band or Ku-Band systems were used.

The User Service system was a more difficult problem because it required specialized TDRSS

beamformers, spread spectrum receivers and associated equipment. Fortunately, the STGT Project had just successfully developed and tested a new-generation multiple access beamformer. As a result, a subset of that equipment, modified for use at RGRT, was able to be procured from the same manufacturer in a timely fashion. The spread spectrum receiver used in both the SSA and MA systems was adapted from a recent GSFC in-house development - the TDRSS User RF Test Set (TURFTS). TURFTS was designed by Code 530 for use in TDRSS user transponder testing in the laboratory and for spacecraft project integration and test. Its design incorporates recent technology such as custom VLSI PN generator chips, numerically controlled oscillators (NCO's) for frequency generation and a digital signal processor (DSP)-based receiver. For use at RGRT, the TURFTS receiver sensitivity was improved and the controller software was modified to control a redundant set of hardware and to allow remote operation from WSGT. The transmitter portion of TURFTS was also adapted to provide the spread spectrum signal transmitted to TDRS at S-Band for MA beamformer calibration.

AlliedSignal Technical Services Corporation (ATSC), the principal technical support contractor for the GRTS Project, also designed a number of critical items of hardware and software essential to the implementation of the RGRT. These included the TT&C and User Service Processors, the Test Inject and Range Zeroset systems, the OMCS application software and components in the Common Time and Frequency Subsystem.

The third category of equipment, which filled in missing items in both the TT&C and User Service systems, was Commercial-off-the-shelf hardware and software. These were items available as "standard products" from manufacturers and included the 10-meter S-Band and 4.6-meter Ku-Band antennas and high power amplifiers, upconverters, downconverters, bit synchronizers, numerous rack-mounted PC's, monitor & control software, pilot signal generators, test equipment and many other items essential to the success of the project.

The critical procurement process of these items was handled by the Raytheon Service Company (RSC) as procurement agent to the government with GSFC/Code 530, supported by ATSC, providing technical oversight. The entire procurement process including specifications, solicitation and negotiation was completed by January 1993 - five months after project start. A 120-day delivery requirement was imposed on most of the vendors and critical long-lead components were incentivized to ensure on-time delivery. Monthly meetings were held at the plants of the more critical vendors for early detection and correction of technical and schedule problems.

RSC also developed an aggressive transportation strategy with a goal of 168 hours of transit time from the vendor's dock to CDSCC in Australia. Due to the excellent execution of the logistical planning and coordination between GSFC, RSC, ATSC and CDSCC, the transportation goal was achieved for about 95% of the 434 pieces/53 tons of equipment shipped, with essentially no damage enroute.

RGRT DETAILED DESIGN

The RGRT design philosophy was driven by the goal to deploy the station rapidly while keeping the costs reasonable. The need for rapid deployment forced the design to be modular, robust, and

conservative. The requirements were decomposed according to engineering disciplines and tasked to small design teams for execution. Another small group of system engineers was responsible for the overall architecture and insuring coordination between design teams.

All of the senior members of the design teams and many of the design engineers had extensive experience working with the GN, WSGT, or STGT. Many of the design decisions reflect lessons learned working with these programs.

The decision to locate the site at the Canberra Deep Space Communications Complex (CDSCC) saved approximately 6 to 12 months over development of a new site. CDSCC was chosen because it already had 60 Hz power, an available building, in-place NASA logistics, a NASCOM node, and personnel already familiar with some of the NASA equipment used at RGRT, while providing only 2 percent less GRO data when compared to a geometrically optimized site location over the Indian Ocean region.

The following station architecture decisions and system design guidelines were decided early and used throughout the project:

- The RGRT would be a remote front end for the White Sands Ground Terminal (WSGT). No command and telemetry data processing would be done at RGRT.
- Separate S and Ku-band antennas would be used to help make the station more robust.
- The design would minimize maintenance and operational personnel demands.
- Preference would be given to using equipment currently in NASA inventory
- Use of new designs was discouraged.
- Use of commercial-off-the-shelf (COTS) products was encouraged especially for equipment with a proven track record.
- The station design would incorporate redundant strings of equipment for TDRS command, TDRS telemetry, and GRO telemetry.
- The Operations Monitor and Control Subsystem (OMCS) would only do monitor and control. Other computationally intensive functions (such as ephemeris propagation, etc.) would be done by special purpose processors.

- The special purpose processors would use a common rack-mountable PC design.
- All equipment would have front panel or some type of local control for operation independent of the OMCS.
- Special attention would be given to avoiding self inflicted EMI and EMI with the CDSCC deep space communications activities.

Although many components required reconfiguration or minor modifications to support the RGRT mission, there were no new designs. NASA and contractor engineers worked very closely with the COTS vendors to insure that the technical requirements and delivery schedule were met. In many cases, monthly visits to vendors' plants were necessary during the development phase.

It was not possible to do full-up laboratory testing before the system was deployed to the field. All interfaces and specifications, however, were tested either in the lab or at the factory before shipment using specially configured test fixtures where necessary.

Almost all of the lead engineers responsible for the various subsystems and components traveled to the site to insure successful integration. Some surprises were found during integration, but these were primarily software related and mostly in the area of remote monitor and control. The monitor and control problems were mitigated by designing all of the components with front panel or local controls that could operate without the Operations Monitor and Control Subsystem (OMCS). This allowed the station's RF and data circuits to be tested independent of the OMCS. OMCS capabilities then incrementally came on line as the bugs were worked out.

The 16 hour time difference between CDSCC and the east coast of the US essentially forced split shifts between the integrators on-site and the supporting engineers in the GSFC area. This was used to advantage during software testing. The integrators tested software during the day shift in Australia and Emailed discrepancy reports back to the developers. The developers then made software updates during their day shift and put them on a file server. The next day, the integrators would get the updates from the server via Internet and load them on the target machines on site and

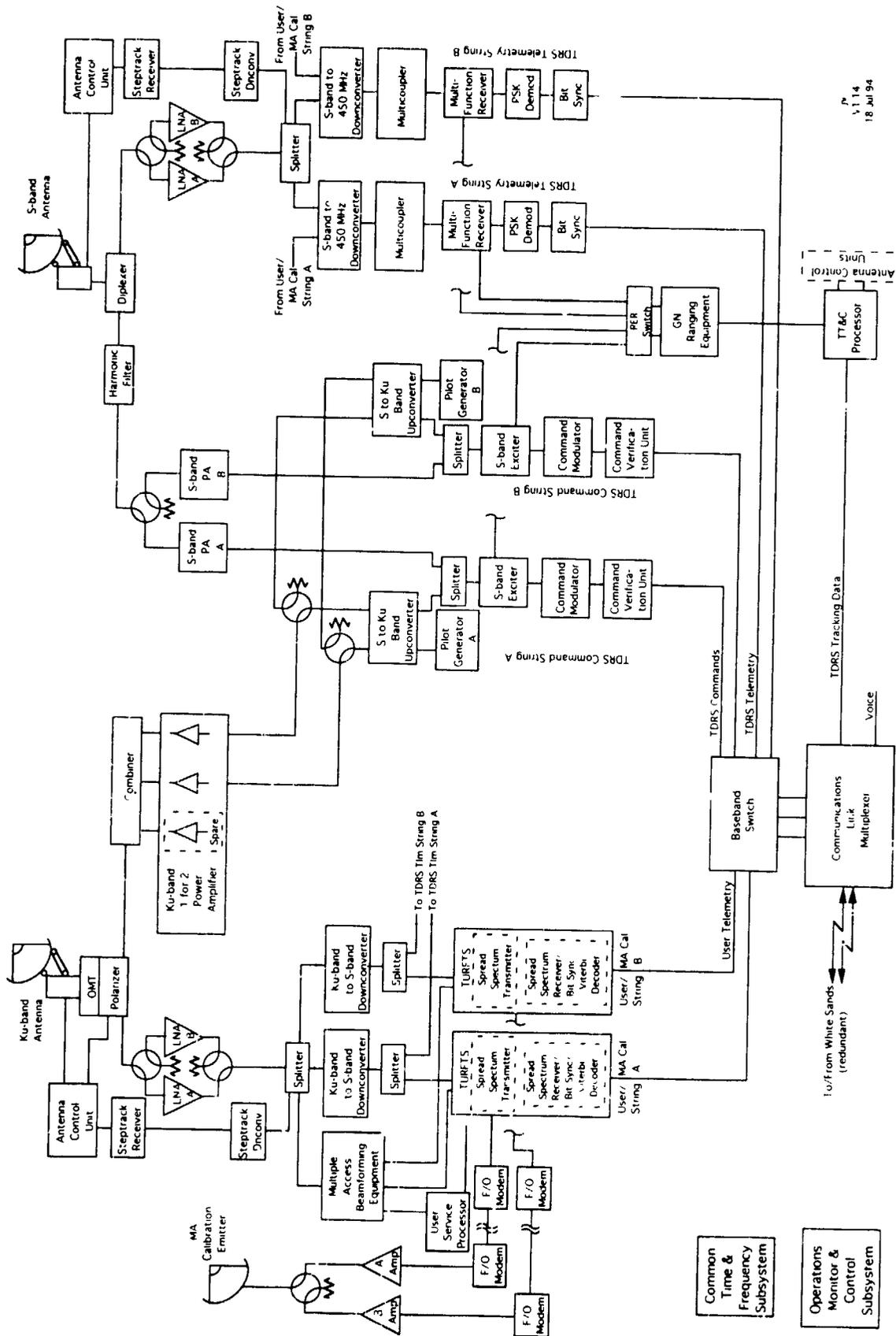
repeat the cycle. Good electronic data communications made this cycle very efficient.

Electromagnetic Interference (EMI) was also a major design concern. There was concern with both self inflicted RGRT EMI and the RGRT causing EMI to the sensitive deep space communications station at CDSCC. Because of the schedule, a detailed EMI analysis was not done. Instead, each subsystem was designed to minimize stray emissions and EMI shielding equipment racks were procured. The following steps were taken to minimize EMI:

- The high power amplifiers were located in shelters near the RGRT antennas and away from the main equipment room.
- Fiber optic connections were used for monitor and control and data communications between buildings wherever possible.
- NASA grounding specifications were rigorously applied.
- The TDRS Multiple Access Calibration Source was located approximately 2km away from the main site where there was no direct line-of-site to any of the DSN antennas.
- The MA Cal source is not scheduled to be used during any sensitive DSN S-band support periods.
- Equipment chassis suspected to having mutual interference problems were packaged in different racks.
- Racks with RF gaskets were used.
- Intermediate bars and RF gaskets for use between drawers in racks were procured, but were not used because the individual chassis were found to be sufficiently shielded.
- An extensive RF survey was performed.

Because of these steps, no major EMI problems were encountered during the implementation.

The RGRT has five major subsystems: the Antenna Subsystem, the TT&C Subsystem, the User Service Subsystem, the Common Time and Frequency Subsystem, and the Operations Monitor and Control Subsystem (See Figure 2 - RGRT Block Diagram). The salient features of each subsystem are discussed in the following paragraphs. The Common Carrier/Data Interface System, al-



P
1114
18 Jul 94

FIGURE 2. RGRT Block Diagram

though not technically part of the RGRT is also discussed below.

ANTENNA SUBSYSTEM

The Antenna Subsystem consists of the antennas, antenna controls, transmitters, and associated microwave circuitry. The RGRT has three antennas: the S-band antenna, the Ku-band antenna, and the Multiple Access Calibration (MA Cal) antenna. The S- and Ku-band antennas are used to support the TDRS space-to-ground links, while the MA Cal antenna is used periodically to calibrate the Multiple Access phased array antenna onboard the TDRS.

The S-band antenna is used to support command, telemetry, and ranging via the omnidirectional antenna on the TDRS. It provides a robust RF link and may be used regardless of the TDRS attitude. The S-band antenna uses a standard communications satellite limited-motion mount equipped with a 10 meter reflector and a single horn prime focus feed. TDRS-1's orbit currently has a 7.7 degree inclination, so the antenna moves substantially to cover the satellite's diurnal motion. Typically the antenna operates in program track mode. The pointing angles are computed by the TT&C processor. The mount, controls, reflector, feed, and microwave components, with the exception of the diplexer, are COTS products procured as a system from a single vendor. In order to meet the schedule, NASA furnished the manufacturer with an excess diplexer for incorporation into the delivered product.

The Ku-band antenna is used to support TDRS command, telemetry, and ranging via the SGL dish antenna on the TDRS. It is also used to uplink the pilot signal and receive the GRO return link data. It also has a limited-motion mount and has a 4.6 meter dish with Gregorian geometry. The mount, controls, reflectors, and microwave components are standard COTS items. The feed is also a COTS item that was re-tuned to operate in the TDRS Ku-band. The entire Ku-band antenna system was procured from a single vendor. It operates in a manner identical to the S-band antenna.

Both the S- and Ku-band antennas have identical azimuth coverage. The limited-motion mounts cover 180 degrees of azimuth in three overlapping sectors. This capability allowed these antennas to support the TDRS drift from 171 degrees west longitude to its present position of 85 degrees east longitude.

The decision to have two antennas rather than a single, dual-band antenna was made because this configuration inherently provides redundancy and two antennas had a faster delivery time. Two different manufacturers were used in order to spread the risk of late deliveries. As it turned out, both vendors were about a week behind the 120 day delivery schedule and either would have had a significant problem producing both antennas simultaneously.

TRACKING, TELEMETRY, AND COMMAND (TT&C) SUBSYSTEM

The TT&C Subsystem is responsible for handling the TDRS command and telemetry data. It also includes the equipment for generating the pilot signal and measuring the TDRS two-way range and Doppler. The TT&C Subsystem was built around a spare Receiver/Exciter/Ranging (RER) System identical to those used at the NASA Ground Network sites. Up- and downconverters are used to operate the S-band RER at Ku-band. New test inject and range-zero-set electronics were developed for RGRT, but incorporated COTS signal generators in the design.

The pilot signal is an unmodulated signal that together with the command uplink carrier is used by the TDRS on-board Master Frequency Generator to derive all of the local oscillator signals used by the TDRS user service RF processors. The pilot generators are COTS signal generators operating at S-band. A two-channel upconverter with a common local oscillator (LO) is used to upconvert the S-band command and pilot signals to Ku-band. The use of a common LO helps the RGRT meet the stringent TDRS phase noise specifications.

The TT&C processor is a special purpose PC that propagates the TDRS ephemeris data, interfaces with the COTS antenna controllers, and keeps the antennas on point. It also interfaces with the Ranging Equipment and generates tracking data that is used for TDRS-1 navigation. The ephemeris data processing and tracking data generation software are repackaged versions that were used on other GSFC projects.

The Command Verification Units (CVUs) are also special purpose PCs that verify that the TDRS command spacecraft address and parity bit are correct. The units insert an idle pattern for commands that

fail the simple verification tests and also keep command statistics. The CVU code was developed specifically for the RGRT application.

USER SERVICE SUBSYSTEM

The User Service Subsystem is responsible for receiving and demodulating the GRO data that is relayed through the TDRS. Only S-band Single Access and Multiple Access return link services are implemented at RGRT at this time.

The User Service Subsystem is built around the TDRS User RF Test Set. The TURFTS was originally developed by GSFC to test user transponders. The dual unit designed for RGRT includes two TDRS compatible spread spectrum receivers and transmitters. TURFTS firmware modifications were needed to meet RGRT mission requirements. Use of the TURFTS was the fastest and easiest way to get RGRT user service on-line. The TURFTS receiver is used to demodulate and decode the GRO data and the TURFTS transmitter is used to generate the test signal for TDRS multiple access phased array calibration.

The Multiple Access Beamforming Equipment (MABE) is used to steer the TDRS multiple access phased array for Multiple Access user support. RGRT uses a scaled-down version of the second generation MABE that was originally developed for the Second TDRS Ground Terminal program. The RGRT MABE only has two user channels while the STGT version has six. Two user channels allow for simultaneous GRO support and MA Calibration.

The User Service Processor (USP) propagates the TDRS and GRO ephemeris data. It provides direction cosines to the MABE for TDRS phased array steering and predicted Doppler data to the TURFTS for signal acquisition. The USP runs the identical ephemeris data propagation algorithm as the TT&C Processor.

OPERATIONS MONITOR AND CONTROL SUBSYSTEM

The RGRT Operations Monitor and Control Subsystem (OMCS) design capitalized on trade studies and market surveys completed by the GSFC Automated Ground Network Systems (AGNS) project some months earlier. The COTS monitor

and control system product selected was originally designed for applications such as chemical processing plant control and electrical power grid management. The choice to use a COTS monitor and control system allowed the designers to concentrate on the RGRT application rather than the OMCS infrastructure.

The COTS system is based on the use of industry standards, for example, POSIX compliant workstations are used for the operator interfaces, TCP/IP is used for interprocessor communications, and dBase IV is used for data base management. System utilities include graphical screen generation tools and device drivers.

The RGRT OMCS architecture consists of I/O Controllers and a workstation located in the RGRT equipment area, a workstation located at the CDSCC Signal Processing Center, two workstations at White Sands, and a workstation at the Network Control Center (NCC) at GSFC. All of the processing elements are connected via ethernet and COTS routers are used to connect the geographically dispersed LAN segments.

During RGRT installation, portions of the OMCS LAN were connected to the Internet for remote troubleshooting by the COTS vendor and engineers in the GSFC area. This proved to be very valuable. Now that RGRT is operational, there is no Internet connectivity because of security concerns. Software upgrade deliveries are done via the workstation at the NCC.

Additional information about the OMCS may be found in the SpaceOps '94 paper "GRTS Operations Monitor/Control System"

COMMON CARRIER/DATA INTERFACE SYSTEM

The Common Carrier/Data Interface System (CC/DIS) connects the RGRT in Australia to the TDRS data processing equipment in White Sands, New Mexico. The CC/DIS consists of COTS communications multiplexers and two 64 kb/s leased commercial communications lines. The 64 kb/s lines are physically diverse. They are leased from different companies and travel through different Intelsats.

The following data channels are multiplexed through the CC/DIS: 2 kb/s TDRS command data,

1 kb/s TDRS telemetry data, 32 kb/s GRO telemetry data, 12 kb/s RGRT monitor and control data, 110 b/s TDRS tracking data, and 6 kb/s digitized voice. The CC/DIS equipment is capable of multiplexing all of these channels onto a single 64 kb/s line, but typically they are divided with some channels on one 64 kb/s line and the remainder on the other. This provides faster failover time should one of the 64 kb/s lines fault.

CONCLUSIONS

The GRTS Project demonstrates NASA/GSFC's ability to quickly meet new spacecraft data acquisition and tracking challenges in a cost effective manner by inserting new technology and adapting exist-

ing space- and ground-based assets augmented with commercial off-the-shelf products.

The Space Network's communications support to GRO also illustrates one of the unique advantages of the synchronous Tracking Data Relay Satellite System. With the addition of GRTS, the now-global TDRS System is able to provide constant communications coverage for contingency support to NASA missions.

REFERENCES

Stocklin, F., (March 1994). The GRO Remote Terminal System - An Experience of a Lifetime. NASA Mission Operations and Data Systems Newsletter, Vol. 10, No.2

Data Management

2. Data Handling		Page 83
DM.2.a	Overview on METEOSAT Geometrical Image Data Processing <i>Frank J. Diekmann</i>	85-93
DM.2.b *	UARS CDHF Conversion to DEC AXP Platform <i>Stuart Frye</i>	95
DM.2.c	Use of a Multimission System for Cost Effective Support of Planetary Science Data Processing <i>William B. Green</i>	97-103
DM.2.d	A Second Generation 50 Mbps VLSI Level Zero Processing System Prototype <i>Jonathan C. Harris, Jeff Shi, Nick Speciale, Toby Bennett</i>	105-115
DM.2.e	A Corporate Memory for the GSFC Mission Operations Division <i>Beryl Hosack</i>	117
DM.2.f *	Fast Computational Scheme of Image Compression for 32-bit Microprocessors <i>Leonid Kasperovich</i>	119-123
DM.2.g	The Development and Operation of the International Solar-Terrestrial Physics Central Data Handling Facility <i>Kenneth Lehtonen</i>	125-132
DM.2.h	Economical Ground Data Delivery <i>Richard W. Markley, Russell H. Byrne, Daniel E. Bromberg</i>	133-138
DM.2.i	NASA Johnson Space Center Life Sciences Data System <i>Hasan Rahman, Jeffery Cardenas</i>	139-147
DM.2.j	A Processing Centre for the CNES CE-GPS Experimentation <i>Norbert Suard, Jean-Claude Durand</i>	149-154
DM.2.k	Single Stage Rocket Technology's Real Time Data System <i>Steven D. Voglewede</i>	155-168
DM.2.l	International Data Transfer for Space Very Long Baseline Interferometry <i>Alexandria B. Wiercigroch</i>	169-175
DM.2.m	ARACHNID: A Prototype Object-Oriented Database Tool for Distributed Systems <i>Herbert Younger, John O'Reilly, Bjorn Frogner</i>	177-183

* Presented in Poster Session

F-9

Overview on METEOSAT Geometrical Image Data Processing

Frank - J. Diekmann

ESA/ESOC
 METEOSAT Exploitation Project
 Robert-Bosch-Str. 5
 64293 Darmstadt
 Germany

ABSTRACT

Digital Images acquired from the geostationary METEOSAT satellites are processed and disseminated at ESA's European Space Operations Centre in Darmstadt, Germany. Their scientific value is mainly dependent on their radiometric quality and geometric stability. This paper will give an overview on the image processing activities performed at ESOC, concentrating on the geometrical restoration and quality evaluation. The performance of the rectification process for the various satellites over the past years will be presented and the impacts of external events as for instance the Pinatubo eruption in 1991 will be explained. Special developments both in hard- and software, necessary to cope with demanding tasks as new image resampling or to correct for spacecraft anomalies, are presented as well. The rotating lens of MET-5 causing severe geometrical image distortions is an example for the latter.

INTRODUCTION

The history of the METEOSAT system at ESA's European Space Operations Centre (ESOC) reaches back to the launch of the first satellite of the preoperational program in 1977. While this one was still of an experimental nature, its successor became fully operational in 1983 and was followed by

four further satellites since then. MET-3, launched in 1988, is the last of the preoperational series and serves to date at 75°W for the Extended Atlantic Data Coverage (XADC) mission (de Waard, 1993) on behalf of the National Oceanic and Atmospheric Administration (NOAA). MET-4 to MET-6, positioned at 0° and 10°W, already belong to the operational program (Mason, 1987) and are operated on behalf of EUMETSAT.

An enormous wealth of data has been provided by the METEOSAT satellites, with the image data being the dominant source of information. The images acquired from METEOSAT, processed and disseminated at ESOC, are used by a wide research community as a mean to gain a better understanding of atmospheric processes. This, however, depends largely on the radiometric quality and geometric stability of the image data (Diekmann, 1994). Raw (unprocessed) images transmitted from the satellite are distorted due to its various movements during the image taking process. Since this prevents an accurate geographical identification of image pixels, an evaluation of the image distortions and a following resampling of the image pixels is a necessary step to reduce these geometric errors. This is achieved during the onground data handling by an image correction process called rectification. The principles of this geometrical correction scheme is described by Wolff (1985) and Bos et al. (1990) summarize the changes necessary in order to perform the rectifica-

tion process in near real time. This paper gives a general overview on the various items related to the geometrical processing at ESOC.

METEOSAT IMAGE PROCESSING AT ESOC

The operational satellites of the METEOSAT series are spin-stabilized spacecrafts which scan the Earth once every half hour (a slot). The detector is a radiometer with a pointing direction stepping from south to north by rotating the telescope one step per image line. Four images are taken simultaneously in different frequency bands (infrared window, water vapour absorption band, two images in the visible spectral region), consisting of 8 bits per pixel and contain 2500*2500 pixels per image (2500*5000 for each of the VIS channels). Two of these data streams are currently processed in parallel at ESOC and a third one is possible for limited time periods (e.g. commissioning, anomaly investigations, etc.). A first preprocessing of the continuous raw image data stream consists predominantly of demultiplexing the data to form continuous pictures. The bulk of the following image data processing tasks is running operational on one of the two mainframe (MF) computers (COMPAREX 8/98). The second MF serves as backup and is normally used for other purposes. Both machines have undergone regular upgrades in order to cope with the growing need for computing power.

Figure 1 summarizes the main tasks and processes necessary to obtain rectified images and meteorological data as the final products to be provided to the user community.

After reception and preprocessing of the image and corresponding auxiliary data, the rectification of the images is the main and most time consuming task of the onground

processing. This term refers to the fact that images of the Earth's surface and its atmosphere transmitted from the satellite are distorted due to its various movements during the image taking process. Since this makes a geographical identification of an image element (pixel) almost impossible, an evaluation of the image distortion and a corresponding resampling of the image pixel is a way of reducing these geometric errors.

The true image signal sensed by the METEOSAT radiometer is additionally altered and degraded by various radiometric noise sources originating from the satellite and the transmission (both Gaussian- and periodic noise), by the sampling and digitization process, and also through the introduction of spatial shift errors (during the attempt to geometrically correct the images). A variety of parameters is determined during the operational image processing for assessing the radiometric quality of the METEOSAT images (Diekmann and Amans, 1990; Diekmann, 1994).

Various monitoring devices serve as indispensable tools for performance and quality controlling at the various stages of the image processing chain. Monitors connected to the front end processors display different raw image channels of the two operational satellites before any data processing. Different image display and processing devices allow an online control of raw and rectified images. In addition, a transputer augmented workstation (TAW) is available for fast monitoring and processing of images. They are transferred in real time via an FDDI interface from the mainframe computer, reduced to simple byte maps and stored on harddisks, which are each equipped with a dedicated transputer for fast I/O processing. This allows very fast and parallel zooming, scrolling and loops of up to 500 images in full resolution. A semi-automatic software system for quality controlling the image rectification (QCIR) is run on regular basis. The rectified images are finally subject to an advanced segmentation process, which is the basis for the

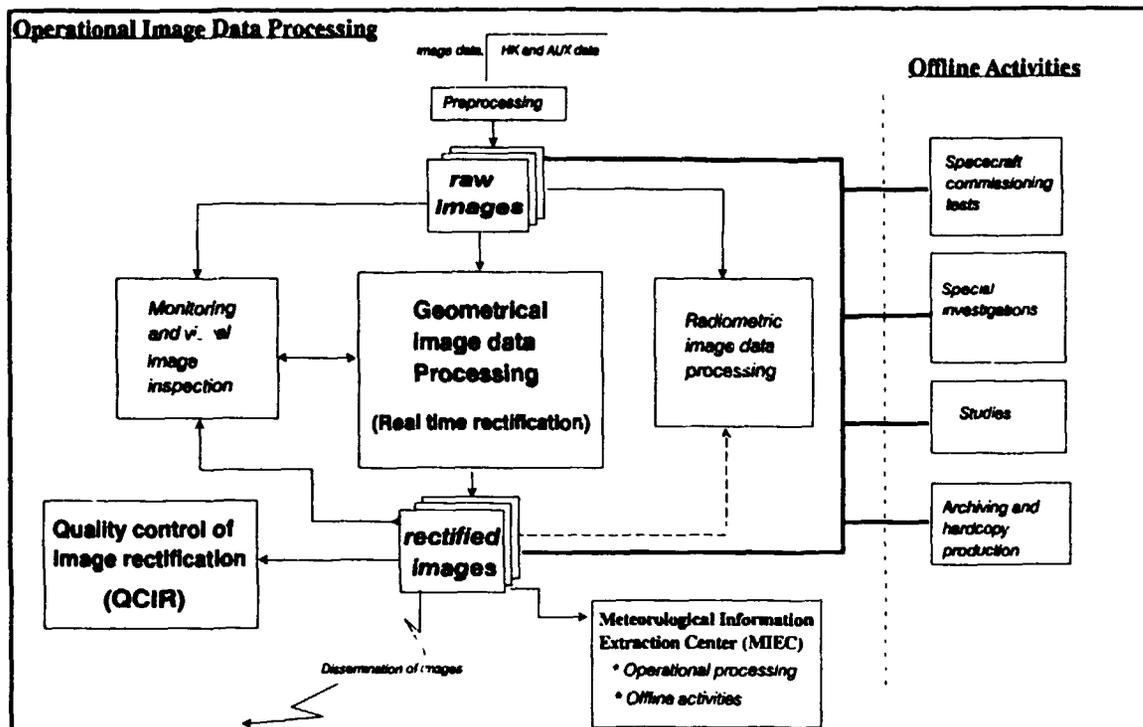


Figure 1 : METEOSAT image processing areas at ESOC

operational derivation of a number of meteorological parameters in the Meteorological Information Extraction Centre (MIEC).

Special software tools are finally developed for satellite commissioning operations and special investigations (e.g. satellite anomalies, end-of-life tests, measurement campaigns, etc.).

THE REAL TIME RECTIFICATION PROCESS

The purpose of the METEOSAT raw image rectification is to remove the geometric image distortions caused by non-nominal spacecraft orbit, attitude, spin and other effects. It basically involves the modelling of the distortions and the transformation of the raw images according to this deformation model so as to obtain a rectified image centred on the nominal sub-satellite point (Wolff, 1985). Besides those already mentioned, a number of parameters (radiometer

stepping parameters, vertical image centre positions, interchannel registrations, etc.) are used to calculate a pair of deformation matrices for the horizontal and vertical directions.

The calculation of some of the deformation model parameters is based on a continuous analysis of where the southern, northern, eastern and western horizons are located in the raw IR images. The polar horizon positions are necessary for determining parameters, such as the vertical image offset, the radiometer scanning parameters and for the determination of the refined spacecraft attitude. The equatorial horizons, on the other hand, are required for the calculation of the horizontal centring of the image, the sampling frequency and for general anomaly analysis. "Horizon" in this context is the point at which the IR sensor detects a change from space (count < 16) to atmosphere (count ≥ 16). This IR threshold corresponds to a temperature of approximately -90°C. It actually reflects a vertical atmosphere column integral and the temperature is the mean of this layer at the Earth's limb. Refin-

ed polar and equatorial horizons (in fractions of a pixel) are finally calculated by fitting a second order polynomial to a predefined number of lines containing (valid) horizons. A warming or cooling of the stratosphere consequently means a change of the determined horizon positions which in return degrades the rectification accuracy. This effect is observed during the normal seasonal temperature changes of the antarctic stratosphere, which allowed a modelling of this phenomenon, but also after a strong warming due to volcanic dust reaching the south polar stratosphere (Diekmann and Bowen, 1992).

Based on the two deformation matrices, the raw image is finally resampled to form the rectified image. This process is running since several years in near real time (Bos et al., 1990), with the old batch-system (starts the image rectification after reception of the whole image) providing a back-up. In the real-time rectification system some of the deformation parameters are predicted using a combination of information from previous IR images and measurements made in the first 350 image lines of the current IR image.

Another prerequisite was the use of the nearest neighbour resampling technique, because this fast and simple method was the only possible for the available data processing resources. After many years of experience with and improvements of the METEOSAT system, the residual geometrical errors in the image data are now mainly caused by the nearest neighbour resampling (Diekmann and de Waard, 1992). Since also the computer technologies have vastly stepped forward, the use of more sophisticated interpolation method within the METEOSAT rectification system has become possible. After a thorough study of various methods, the bicubic spline filter was selected for the image resampling. This CPU time consuming modification was first installed and tested on a dedicated hardware tool (see below) and 1993 installed on the operational mainframes, whose capacity had been more than doubled over the past three years.

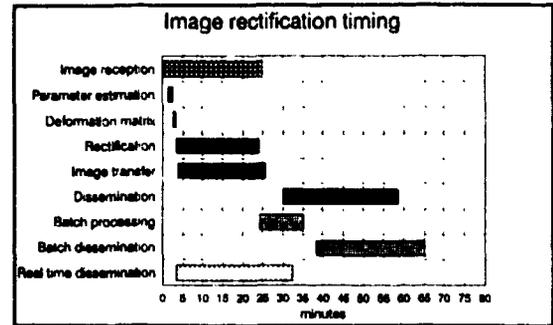


Figure 2 : Image rectification timing

The sequence of the main image rectification tasks is illustrated in Figure 2. The image is received within 25 minutes (5 minutes are needed for retrace and a standby period). Within the first ca. 4 minutes the deformation parameters are determined and the deformation matrices calculated. The re-sampling starts at that point and catches up with the incoming data stream fairly quickly. The processed image lines are transferred to the dissemination computer and divided into dissemination formats. The actual transmission of these formats starts with a small delay at the end of the slot. In the batch processing case this delay is at least eight minutes. A real time dissemination system with data compression and inclusion of other data products in the image data stream was developed and tested a few years ago, but not used in operations as it necessitates a change in the user reception equipment not covered by the present programme.

QUALITY CONTROL OF IMAGE RECTIFICATION

The method used to assess the rectification accuracy of METEOSAT images is described in detail by Adamson et al. (1988). The complex system called "Quality Control of Image Rectification" (QCIR) is based on about 120 reference landmarks (coastlines, islands, lakes) spread over the scanned Earth disk. They are extracted from rectified IR and VIS images and filtered by a simple automatic histogram and peak-identification process

to extract those landmarks with less than ca. 10% cloudiness. These landmarks are later subject of another automatic test (based on landmark specific correlation coefficients) to delete the remaining cloud contaminated landmarks collected during about one week. The cloudfree landmarks passing both tests are correlated with an accurate digital reference landmark data set. The landmark displacement is defined to be the displacement of the maximum of the correlation surface from nominal. Results for each landmark are presented in terms of line and pixel deviations as well as in absolute and relative rms errors of the sum of both. This process runs on a weekly basis. For the relative errors the landmark position is compared with the results of the previous slot, which gives an indication of the rectification stability.

Constant biases determined with this method are usually attributed to a set of registration

parameters describing the positions of the detectors onboard the satellite with respect to the detector optical axis. These important parameters are usually updated during the commissioning of a spacecraft and later optimized, if necessary.

Figure 3 summarizes the performance of the METEOSAT image rectification system since 1989. Large rms errors during the commissioning of MET-4 in 1989 were caused by imperfect detector registration values which could be corrected during the following weeks. The seasonal wave in the rectification errors can still be identified in 1990; a correction scheme based on a model of this oscillation was implemented in early 1991, resulting in a clear improvement of the quality. Volcanic dust in the lower stratosphere after the Pinatubo eruption in 1991 caused a warming of the east image horizons (July and August) and later of the antarctic stra-

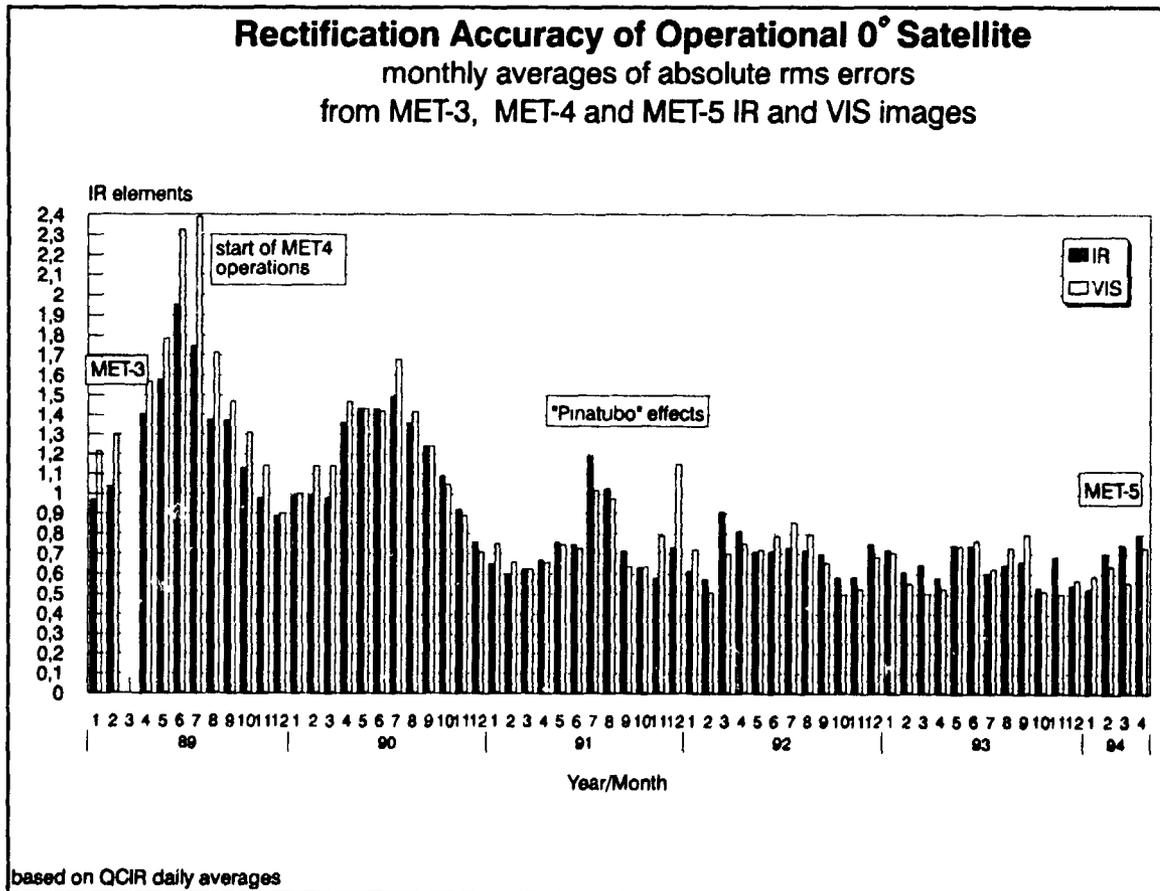


Figure 3 : Monthly means of absolute rms errors

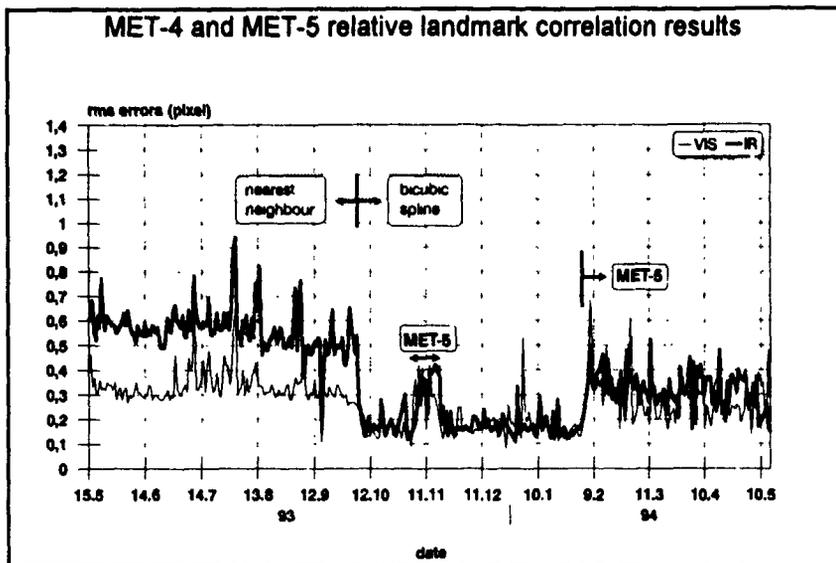


Figure 4 : Daily averages of relative rms errors, 1993 - 1994

better because of the higher spatial resolution in horizontal direction. After start of the bicubic spline filter, the geometrical errors went down to values around 0.2 IR pixels for the MET-4 satellite - an improvement of a factor 3. A detailed study of the impact of the new method on the radiometric contents proved that no series degradations lowered the overall quality of the rectified images.

tosphere. The resulting rectification degradations could be reduced by simple parametrizations of these temperature changes in the rectification software (Diekmann and Bowen, 1992). The remaining errors (around 0.6 IR elements rms) are predominantly inherent in the nearest neighbour resampling.

In addition to these rather predictable features, the rectification software had regularly to be extended in order to cope with unexpected anomalies originating from the spacecraft. For instance, the "fish" problem of MET-4 (electronic disturbances in all image channels, Baratelli et al., 1990), and the rotating lens of MET-5 (Olivier, 1991) had serious impacts on the radiometric and geometric image quality.

A significant improvement of the relative rectification was achieved in autumn 1993 with the initiation of the bicubic spline resampling technique in the operational rectification system. Figure 4 demonstrates this fact. The mean rms errors of landmark displacements from slot to slot (relative rms) was until October 1993 in the order of 0.6 IR pixels - approximately the statistical limit of the nearest neighbour technique. The results for the VIS channels were always slightly

THE MET-5 ROTATING LENS ANOMALY

MET-5 was launched in March 1991 and was expected to become the prime operational satellite after its successful commissioning. However, it was discovered that the rectification accuracy, in particular the relative rectification performance, was significantly worse than the results of other satellites. The reason was an unexpected movement of the Earth's disk in the image frame in the order of one IR pixel. Indicator for this anomaly were the east/west and north/south horizons. Only the IR and WV channels were affected, but not the VIS channels.

After extensive investigations, the cause of the MET-5 image anomaly was identified as being caused by a rotating lens (L3) inside the radiometer cold assembly just in front of the passively cooled longwave detectors (Olivier, 1991). This lens is not held firmly enough to prevent a rotation. The amplitude of this rotation corresponds to geometrical distortion of the Earth image of about 1.1 IR pixels, with a frequency between 2 and more than 10 slots. Even interruptions of the lens rotation have been observed. The more or less constant amplitude can be explained if one assumes that the optical and geometric

centre of the lens do not coincide. Such a constant lens rotation introduces a sinusoidal distortion into the IR and WV images (because L3 focuses the incoming radiation only in these detectors) in both horizontal and vertical direction.

As a consequence of this, the image rectification system has been extensively modified in two steps so as to enable it to minimize these geometrical distortions. The software version developed first was running in batch mode (after reception of the whole image) and separates the deformation modelling into two parts : the calculation of a base deformation that corresponds to the deformation model that would have been obtained in the absence of a lens rotation, and the evaluation of correction functions to compensate for the additional distortions (Hanson and Adamson, 1992). The latter is essentially achieved by an accurate determination of the horizontal Earth horizons and a representation of the east-west distortion curve by a sine wave model. This distortion curve is then translated into the corresponding south-north distortion curve using the assumption that the lens rotates uniformly during the course of a slot. This method gave satisfying results, which were, however, still somewhat worse than corresponding MET-4 results.

A real time correction independent on such assumptions developed under ESA contract is just being installed at ESOC. East and west horizons of each line delimit an Earth chord. Using datation data which are transmitted with each image line, the angle between the sun and the midpoint of each chord is measured and compared to a predicted value. This is based on a METEOSAT state vector model determined from observations over several days, such that the anomalous lens rotation effects are averaged out. When the lens motion displaces the line of sight in horizontal direction, the measured chord midpoint will be shifted with respect to the predicted midpoint, which allows a direct correction. When the lens rotation displaces the line-of-sight in vertical direction, the measured angular span of a chord will grow

or shrink when compared with the predicted value based on the state vector. This also allows a correction in south-north direction. First results have shown that with this method a real time rectification is possible with a rectification accuracy similar to non disturbed satellites.

A TRANSPUTER AUGMENTED WORKSTATION

In spite of various attempts to minimize the remaining errors in the real time rectification, the nearest neighbour resampling imposed a non-reducible limit to the final results. A variety of methods were studied which could possibly be used instead. All were based on interpolation of weighted pixel counts surrounding the pixel in question. Radiometric changes introduced by the finally chosen bicubic spline method are well within the accuracy of the radiance figures themselves. They could even lead to an improvement by reducing the impact of the satellite's radiometer and associated equipment on the original image.

The initial technical realization of this very time consuming resampling process was part of the development of a transputer augmented workstation (TAW) funded by the Austrian Authorities under ESA contract. The scope was a computer prototype in hard- and software for real time resampling using advanced filter techniques, product extraction as well as rapid image display and processing. The design of the TAW supports interfaces to modules for further applications. Besides the rectification module, a second component is connected for near real time water vapour wind vectors (WVWV) using an optimum pattern matching algorithm. Various automatic quality control tools are applied to the wind vectors, which were optimized for this special application beforehand (de Waard et al., 1994). Additional image processing tools are available on a connected workstation, which also allows the online calculation

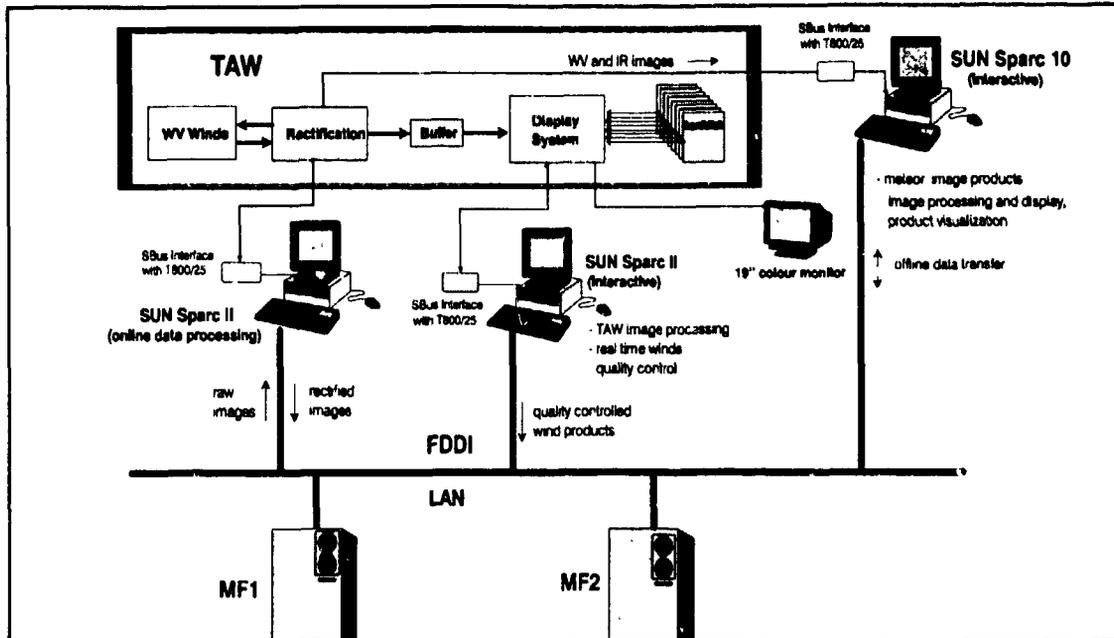


Figure 5 : TAW functional set-up

and display of various meteorological products.

The TAW is equipped with a total of 23 CPUs distributed over four different modules as sketched in Figure 5 and serves as a powerful state-of-the-art development and study environment. The capacity of this workstation is highlighted by its processing power which achieves a performance of almost 200 Mflops with the implemented applications (Scheiber, 1994).

The main components of the TAW are the resampling and WVWV modules, which both consist of T800 transputers and INTEL i860 processors. These functions are controlled via a Sun Sparc host computer, which also performs the image data transfer from and to the ESOC mainframes via an FDDI interface in real time mode. Raw and resampled images are also automatically transferred via a buffer module to the display system of the TAW. Eight harddisks with a total of 3.2 Gbyte, each equipped with a dedicated transputer for fast I/O processing, are available to store these images on a cyclic file system. This allows fast and parallel zooming, scrolling and loops of up to 500 IR im-

ages on a high resolution monitor. Additional functions of this module are pixel inspection, window extraction and grey value statistics, classifications and Fourier transformations, overlays and others.

SUMMARY

Digital images in three or four channels of two METEOSAT satellites (even three under certain circumstances) are received and processed at ESOC every half an hour - a maximum of 528 images per day. Before disseminated to the users, these images are geometrically corrected in near real time. The quality of this rectification process has constantly been improved over the past years due to better modelling of system inherent effects, anomaly handling and hardware upgrades. Monthly performance figures for perfectly processed and disseminated images of in most cases well above 98% and high quality cloud motion wind vectors deduced from METEOSAT images are good indications for a stable and reliable on-ground processing. Further developments are now coming to an end, since EUMETSAT is going to take over the whole operations in December 1995.

REFERENCES

- Adamson, J., Kerr, G.W., and Jacobs, G.H.P., 1988, Rectification quality assessment of METEOSAT images. *ESA Journal* 12, 467 - 482.
- Bos, A.M., Waard, J. de, and Adamson, J., 1990, Real time rectification of METEOSAT images. *ESA Journal* 14, 179 - 191.
- Diekmann, F.J. and Amans, V., 1990, On the radiometric and geometric quality of METEOSAT images. *Proceedings of the 8th METEOSAT Scientific Users' Meeting, Norrköping, Sweden, 28th-31st Aug. 1990*, EUMETSAT, Darmstadt, Germany, 17 - 22.
- Diekmann, F.J. and Waard, J. de, 1992, New concepts of METEOSAT image data resampling. *Proceedings of the 9th METEOSAT Scientific Users' Meeting, Locarno, Switzerland, 15th - 18th Sept. 1992*, EUMETSAT, Darmstadt, Germany, 401 - 408.
- Diekmann, F.J. and Bowen, R.A., 1992, Impacts of the Pinatubo eruption on METEOSAT images. *Proceedings of the 9th METEOSAT Scientific Users' Meeting, Locarno, Switzerland, 15th - 18th Sept. 1992*, EUMETSAT, Darmstadt, Germany, 33 - 40.
- Diekmann, F.J., 1994, An overview on applications and capabilities in the METEOSAT image processing system. Accepted by *International Journal of Remote Sensing*.
- Hanson, C. and Adamson, J., 1992, Software modifications to cope with the METEOSAT lens rotation anomaly. *Proceedings of the 9th METEOSAT Scientific Users' Meeting, Locarno, Switzerland, 15th - 18th Sept. 1992*, EUMETSAT, Darmstadt, Germany, 11 - 18.
- Mason, B., 1987, Introduction to the METEOSAT operational system. ESOC, METEOSAT Exploitation Project, ESA BR-32.
- Olivier, P., 1991, MOP-2 Anomaly 7. *ESA/ESTEC, Doc.No. MNAR0096, ESTEC, Noordwijk, The Netherlands*.
- Scheiber, J., 1994, Fast External Satellite Image Processing (FESIP). ESA Contract Final Report, Contract No. 9539/91/D/CN, GE.PAR.D., Vienna, Austria.
- Waard, J. de, 1993, Extended atlantic weather coverage provided by METEOSAT-3. *ESA Bulletin, No. 74, May 1993, 20 - 27*.
- Waard, J. de, Diekmann, F.J. and Holmlund, K., 1994, A new approach to real-time water-vapour wind extraction. *ESA Bulletin, No. 77, Feb. 1994, 110 - 116*.
- Wolff, T., 1985, An image geometry model for METEOSAT. *International Journal of Remote Sensing* 6, 1599-1606.

UARS CDHF CONVERSION TO DEC AXP PLATFORM

Stuart Frye
Mitre

Paper Not Available

PREVIOUS PAGE BLANK NOT FILMED

31747
1.7

USE OF A MULTIMISSION SYSTEM FOR COST EFFECTIVE SUPPORT OF PLANETARY SCIENCE DATA PROCESSING

William B. Green
California Institute of Technology
Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena CA 91109

ABSTRACT

JPL's Multimission Operations Systems Office (MOSO) provides a multimission facility at JPL for processing science instrument data from NASA's planetary missions. This facility, the Multimission Image Processing System (MIPS), is developed and maintained by MOSO to meet requirements that span the NASA family of planetary missions. Although the word "image" appears in the title, MIPS is used to process instrument data from a variety of science instruments. This paper describes the design of a new system architecture now being implemented within the MIPS to support future planetary mission activities at significantly reduced operations and maintenance cost.

INTRODUCTION

The MIPS configuration that has been used to support Voyager, and Magellan flight operations, and the Galileo Earth and Asteroid encounters, is a centralized system based on DEC VAX computing equipment running under the VMS operating system. The new system is a distributed system based on the Unix operating system, with significant support provided for international scientists operating remotely from JPL. Image and data display, data management, and production of archival data products exploit recently defined industry standards to insure hardware platform independence,

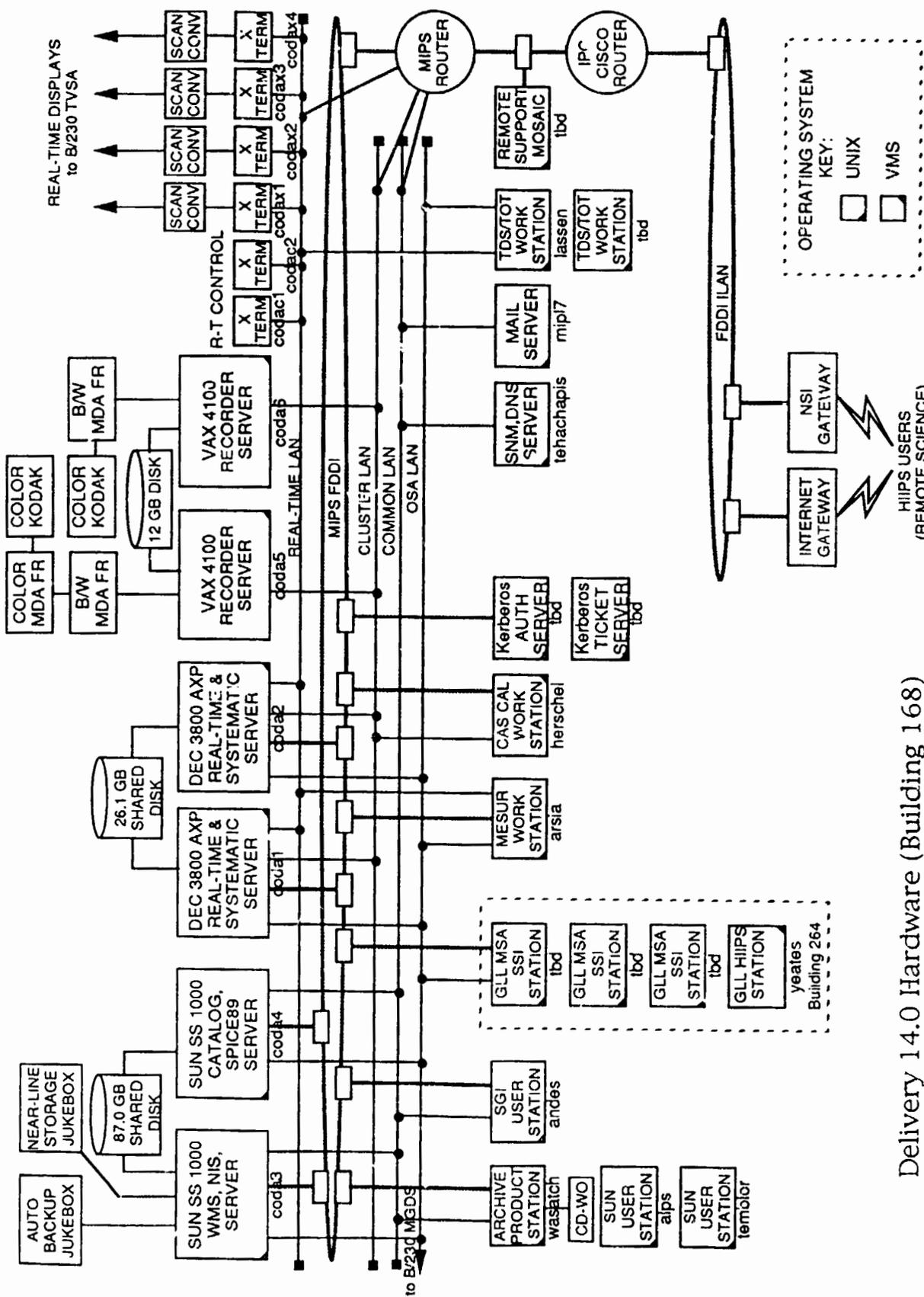
making it possible to evolve the system in the future on commercially available platforms at minimal cost. Significant support of science users not located at JPL is provided by the new system design. Operations and maintenance costs of the new system will be significantly less than the centralized system that has been in use for approximately ten years. The VICAR software system provides instrument data processing capabilities on the new system. Commercially developed software is also available, augmenting the VICAR capability that has evolved over the past 20 years to support specific requirements for planetary exploration data analysis.

HARDWARE SYSTEM DESIGN

Figure 1 shows the hardware configuration that will be in place in time to support flight operations of the Galileo spacecraft in late 1995. The system will also be supporting calibration and system level testing of the Imaging Science Subsystem (ISS) for the Cassini mission to Saturn and final ground data system testing for the Mars Pathfinder mission during that same period of time.

The main subsystems of the new system include the following:

- Dual DEC Alpha processors that support processing of telemetry data in real time as received from the spacecraft, and production of Experiment Data Records in near real time.



Delivery 14.0 Hardware (Building 168)
MDM, 6/2/94

Figure 1

- Dual Sun SparcStations that are used to host a data management system that develops and maintains catalogs that contain information regarding all versions of data processed by the system and the location of each version of data processed on the system. One of the servers also manages the system's mass data storage components.
- Dual VAXStations, that support CSI black and white and color film recorders. Image data can be forwarded to the VAXStations from any source in the system, or from remote sites, for film recording.
- A set of Unix workstations utilized by various projects for processing science instrument data from various missions.
- A set of dedicated workstations used to support single project requirements. Examples include one VAXstation used to support Cassini ISS calibration and a separate workstation that will be used to support Mars Pathfinder image processing.
- X Terminal displays, used to provide display of science instrument data in various formats during real time data acquisition.
- Network resources, including FDDI and Ethernet capability provided locally within the MIPS facility and connections to external network resources. Science teams operating remotely from JPL interface with the system through various networks and routers as shown. Real time data from

JPL's telemetry processing system is provided via an Ethernet connection.

SOFTWARE OVERVIEW

The VICAR software system developed by MIPS has been used to process planetary science data returned from NASA missions for over twenty years. The software is also used internationally by science team members involved in the NASA planetary program, and is made available to commercial organizations through NASA's COSMIC code distribution center. A modular design is used, where some general purpose software modules can be applied to data from a variety of instruments. The VICAR system is being modified to operate under Unix and will be transportable to a wide variety of hardware platforms.

There are several main components of the VICAR software system. The *executive* provides the user interface to the system, and links individual modules together to support specific data processing requirements. A *subroutine library* is available that provides a set of common routines optimized for performance when dealing with large scientific data sets. *Display software* provides support for interactive viewing and manipulation of image data.

Over 200 applications programs are available within VICAR. They include programs in the following categories:

- Arithmetic functions, including averaging, differencing, image summation, image statistics, etc.
- Instrument signature removal software, applied to data returned by instruments on NASA's planetary spacecraft
- Cartographic projection software, designed to interface

with ancillary data files containing navigation and spacecraft position data from the planetary missions and perform mapping projections as requested by the user.

- Atmospheric Feature Tracking software, providing derived velocity vectors based on observed planetary atmospheric motion.
- Data compression, providing a variety of lossless and lossy compression algorithms
- Color manipulation software, including algorithms for producing three color imagery from multispectral planetary imaging instruments.
- Filtering and mathematical transformation software.
- Georeferencing software, providing the capability of correlating remotely sensed imagery with other georeferenced data sets and map data.
- Format conversion, providing conversion between VICAR format and other popular image formats.
- Real time software, used to extract science instrument data records from telemetry data streams processed during receipt of spacecraft data.

VICAR software modules can be used as components in complex processing sequences. The system utilizes a common internal image format, and each program reads and writes data files in the common format. Examples of two processing sequences for the Galileo Solid State Imaging (SSI) and Near Infra-Red Mapping Spectrometer (NIMS) instruments are shown in Figures 2 and 3.

Figure 2 shows the sequence of processing used to produce color images from Galileo SSI image data. The SSI includes a set of spectral filters, and each image is exposed through a separate filter. The spacecraft moves between successive exposures, so it is necessary to register each of the component images to a common geometric reference to create a color composite image. The spectral filters used in the SSI do not correspond to the red-green-blue response of color film or video. It is necessary to perform radiometric processing to obtain transform instrument signal data into physical radiance coordinates, and to then generate a red-green-blue composite color image. Figure 2 indicates three possible products generated from this processing sequence, including color images with high frequency detail enhanced, color ratio images, and images containing the best estimate of radiometrically correct color based on instrument calibration.

Figure 3 shows the processing sequence used to produce a visualization film product of a NIMS data set. Here, software modules specifically designed to process data acquired by spectral instruments that record hundreds of spectral bands of data over a limited region of the surface are used to construct spectral plots and a photographic rendition of this type of data.

Both the processing sequences shown in Figures 2 and 3 can be utilized on other missions flying similar instruments. The only modifications necessary are those required to format the data for display to accommodate mission specific annotation, and any changes required to accommodate differences between specific instrument designs. These sequences illustrate the modular "building block" approach to VICAR design that enables construction of complex processing sequences using individual applications programs. They also illustrate the multimission nature of the software, where common modules can be used to process data acquired by different instruments from different missions.

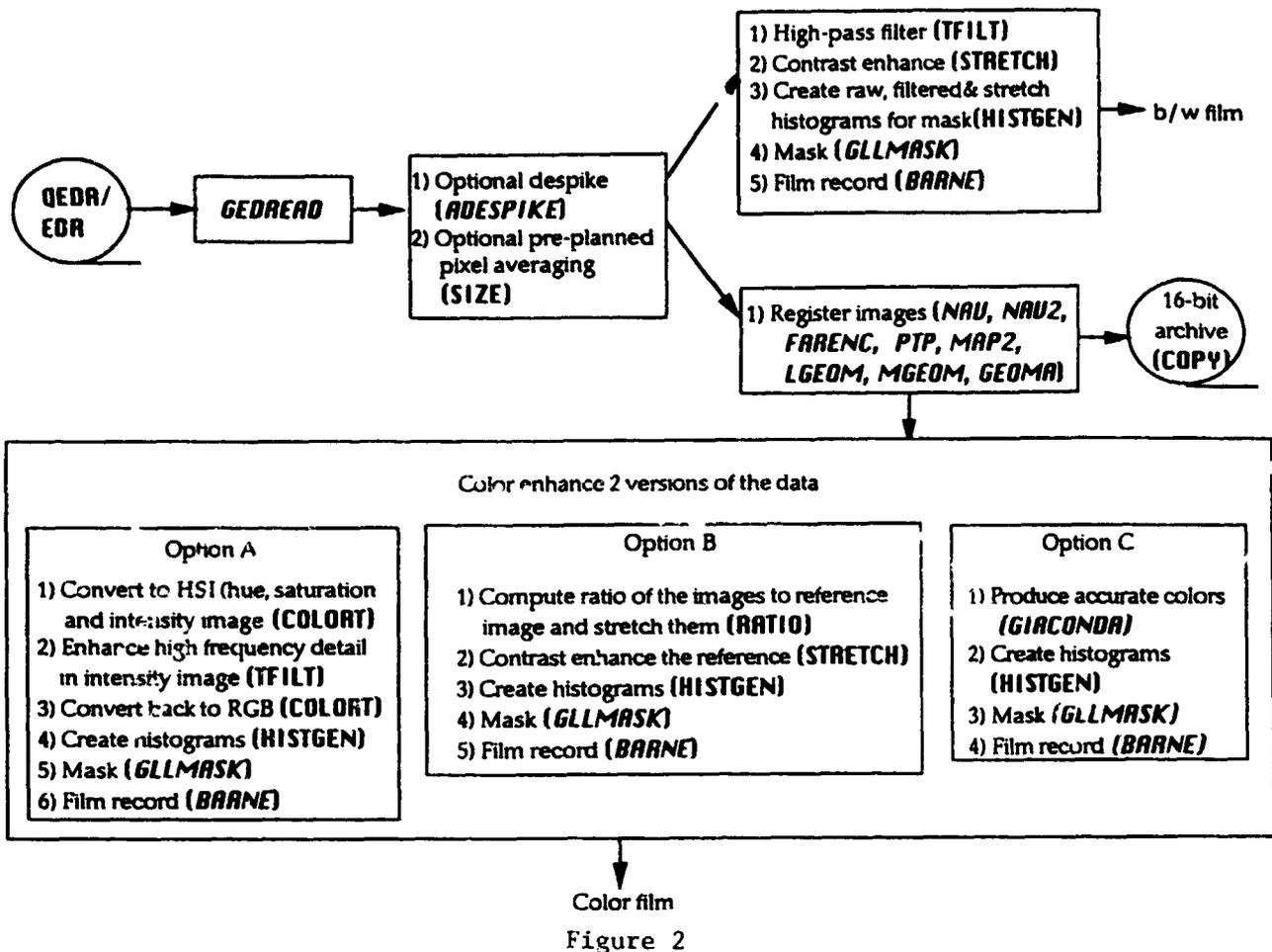


Figure 2

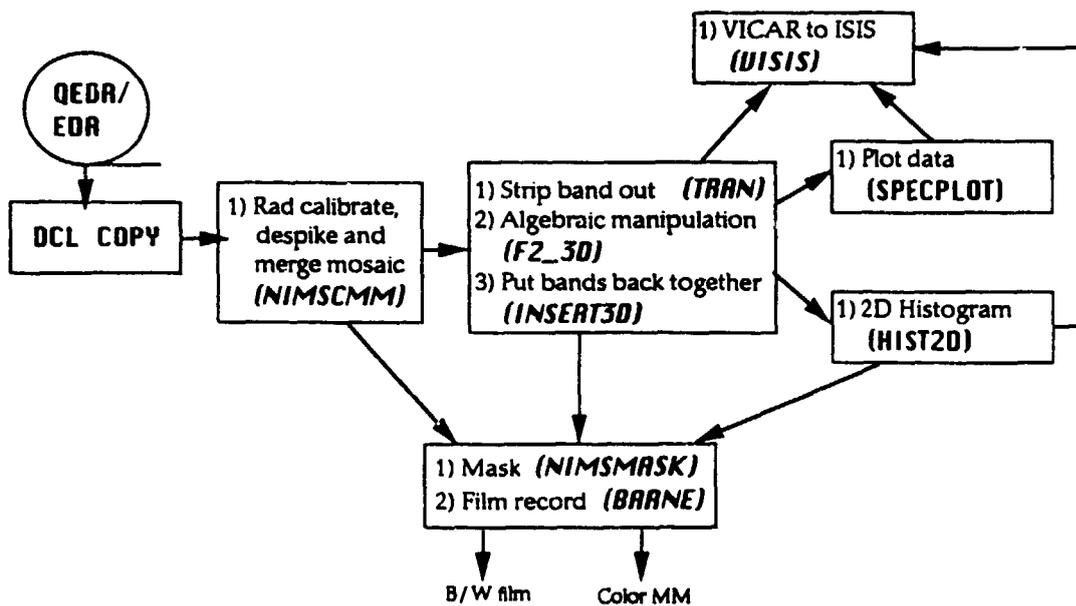


Figure 3

Figure 4 shows a typical standard Galileo black and white photoproduct generated using multimission software adapted for specific Galileo project needs. The Galileo science teams determine the format and content of the annotation information on the photoproducts. The data used to annotate photoproducts is obtained from ancillary data files (navigation data, for example) and from the engineering telemetry data stream. A multimission set of subroutines is used to create the photoproduct shown in Figure 4. Figure 5 shows one example of a photoproduct format being considered for use on Mars Pathfinder's Lander camera data. This is a preliminary format and is still undergoing change and modification based on interactions with the science team. The Mars Pathfinder photoproduct was generated using the same multimission subroutine library to build a mission specific format. With this approach, it is possible to develop prototype formats rapidly, and to complete development of photoproduct generation software with a minimum of effort for each new project.

SUMMARY

The new MIPS system provides a hardware and software system that is modular, flexible and adaptable to new requirements at minimal adaptation cost. The hardware configuration is modular and can be scaled up to handle major missions returning large quantities of data at high data rates, and provides the flexibility of accommodating missions with low data volumes through the use of dedicated workstations. The VICAR software system is also modular and adaptable to new mission requirements at low development cost.

Many principal investigators are finding it cost effective to utilize this multimission facility with established equipment, software, and interfaces with the telemetry processing system to generate first level data records for their instruments and to support other data processing requirements

using inherited software or the shared use of equipment and facilities at JPL.

The work described in this paper was carried out at the Jet Propulsion Laboratory/California Institute of Technology under a contract with the National Aeronautics and Space Administration.

A SECOND GENERATION 50 Mbps VLSI LEVEL ZERO PROCESSING SYSTEM PROTOTYPE

Jonathan C. Harris, Jeff Shi
RMS Technologies, Inc.
Code 520.9

Nick Speciale, Toby Bennett
Data Systems Technologies Division
Code 520

Mission Operation and Data Systems Directorate
NASA, Goddard Space Flight Center
Greenbelt, MD 20771

ABSTRACT

Level Zero Processing (LZP) generally refers to telemetry data processing functions performed at ground facilities to remove all communication artifacts from instrument data. These functions typically include frame synchronization, error detection and correction, packet reassembly and sorting, playback reversal, merging, time-ordering, overlap deletion, and production of annotated data sets. The Data Systems Technologies Division (DSTD) at Goddard Space Flight Center (GSFC) has been developing high-performance Very Large Scale Integration Level Zero Processing Systems (VLSI LZPS) since 1989. The first VLSI LZPS prototype demonstrated 20 Megabits per second (Mbps) capability in 1992. With a new generation of high-density Application-specific Integrated Circuits (ASIC) and a Mass Storage System (MSS) based on the High-performance Parallel Peripheral Interface (HiPPI), a second prototype has been built that achieves full 50 Mbps performance. This paper describes the second generation LZPS prototype based upon VLSI technologies.

1. INTRODUCTION

With the new Earth Observing System (EOS) era of satellites, telemetry downlink data rates will increase to 50 Mbps and beyond. Currently, most NASA missions operate at rates under 1 Mbps. These low data rates allowed ground system designers to use mainframes as well as workstation class computers to handle all the LZP with software, in near real-time. The ground system designers had little need to investigate hardware approaches to LZP.

The DSTD at GSFC saw the need for future high-rate ground telemetry systems, as well as the drawbacks to a full software implementations and began investigating VLSI technologies and their application to telemetry processing in 1989. The completion of the Consultative Committee for Space Data Systems (CCSDS) data format recommendations [1][2], made a combined hardware/software approach for performing LZP feasible. The hardware could be designed to understand the CCSDS data format and allow software to intervene for error condition handling or to handle non-standard data formats. The DSTD chose to implement a standard set of 2.0 Micron VLSI CMOS technology devices that would provide correlation, frame synchronization, frame buffering, packet sorting, and Central Processing Unit (CPU) support; all derived from the CCSDS recommendations. Using this set of VLSI components, the DSTD was able to build a set of processing modules based on the Versa Module Eurocard bus (VMEbus). Each processing module was responsible for one stage of telemetry processing, for example: frame synchronization, Reed-Solomon error detection and correction, or packet processing. With the use of these modules, the first VLSI LZP system prototype demonstrated sustained data rates up to 20 Mbps in the summer of 1992 [3].

The success of this prototype and the high data rate requirement from the Fast Auroral Snapshot Explorer (FAST) mission led to the development of FAST Packet Processing System (PPS). To

support high-resolution observation inside the auroral acceleration zone, the FAST satellite telemetry features downlink data rates up to 2.25 Mbps and data volume of 3.6 Gbytes per day. The project scientists also require all instrument data level zero processed and delivered within two hours of spacecraft downlink for their near real-time experiment. To meet these challenges, the architecture of VLSI LZPS prototype was chosen for science data processing. Within 15 months, the FAST Packet Processing System (PPS) was developed and delivered based on the VLSI LZPS prototype to support the FAST mission [4].

To continue the efforts of applying VLSI ASIC technologies to telemetry processing, the DSTD has migrated the original designs to new 0.6 and 0.8 micron ASICs capable of supporting data rates up to 300 Mbps. These new ASICs have been incorporated into a new set of processing modules ready for system integration. Using these new modules, and some Commercial Off-the-Shelf (COTS) boards, the DSTD has been able to design a second generation VLSI LZPS (VLSI LZPS 2) capable of 50 Mbps performance. This paper discusses the general architecture and functionality of the VLSI LZPS-2, with emphasis on the new elements and features, including an automated operations environment based on object-oriented design. Potential applications of this prototype in NASA's current and future missions are discussed as well.

2. SYSTEM FUNCTIONAL REQUIREMENTS

As a successor to the VLSI LZPS prototype phase 1 (VLSI LZPS-1), the VLSI LZPS-2 has not only continued to provide the functions implemented in VLSI LZPS-1, but has also added many new capabilities. The major performance breakthrough is the boost of sustained processing rate from 20 to 50 Mbps. The major functional enhancement is the support for CCSDS Advanced Orbiting System (AOS) data formats in addition to the packet telemetry formats. Services have been expanded from just Path service to others, including Virtual Channel Access (VCA), Virtual Channel Data Unit (VCDU), Bitstream, and Insert services.

The VLSI LZPS-2 will provide three types of data products: real-time data, quicklook data sets, and production data sets. The real-time data includes source packets received from selected instruments and data extracted from the insert zone, if desired. The data will be delivered to the users as soon as it is received. The quicklook data sets are generated for selected instruments. Each quicklook data set contains all packets received from an instrument in the order they were received. The production data sets are generated for all instruments, and may include data received from one or more passes or sessions. Packets in the production data sets are forward-time-ordered, with redundant ones removed from overlap regions. Data quality is checked; errors and gaps are annotated as a part of the data set.

Data distribution will be performed through standard networks such as Ethernet and Fiber Distributed Data Interface (FDDI), and standard protocols such as Transmission Control Protocol/Internet Protocol (TCP/IP) and File Transfer Protocol (FTP). With this suite of standards, real-time packets and production data sets can be sent to users directly from the VLSI LZPS-2 to simplify user interface and system operations. The processing latency is less than 5 ms for real-time data and 3 hours for production data sets.

In order to reduce operational staffing level and cost, the VLSI LZPS-2 emphasizes an automated operation environment. This environment will be able to setup system support automatically based on a master schedule. It will also allow users to locally or remotely setup and control system operations and monitor telemetry processing status. System events will be displayed,

annotated, and logged. Quality and accounting reports will be generated and logged for each processing session. The user interface will be graphically based and all commands will be menu-driven.

3. VLSI LZPS-2 SYSTEM ARCHITECTURE

The VLSI LZPS-2 is built upon the existing architecture of the 20 Mbps VLSI LZPS-1. This architecture emphasizes the utilization of VLSI technologies and industry standards. Over the past 8 years, the DSTD has developed a set of VLSI ASIC chips that perform standard telemetry processing functions. These chips are integrated into a set of custom-designed, highly reusable cards based on the industry standard VMEbus. Each card performs one or more generic telemetry processing functions. Through the high-level integration of these common telemetry processing functions into VLSI chips and cards, the system achieves high-performance, high reliability, low maintenance and cost.

To integrate these custom cards together with COTS VMEbus components into telemetry data processing systems, a modular software package has been developed that provides a generic software platform. With this platform, a system designer can select and configure a system based on various VMEbus processing cards depending on the given system processing requirements. Thus, the system based on this architecture offers high-configurability, reusability, and upgradability.

Automated operation is emphasized throughout the system design at all levels. The design of the VLSI LZPS ensures that all operations can be controlled by a remote host such as a Control Workstation, and that all status required for monitoring operations be collected and reported to the remote host. Once initialized for a pass, the VLSI LZPS requires no remote intervention to process data. The system will continue to operate even if the remote host fails during a pass.

The VLSI LZPS-2 rack, shown in Figure 1, contains a 21 slot VMEbus system, a 40 Gbytes super disk array system (super disk farm), and dual power supplies. The super disk farm takes up 1-1/4 standard 19 inch 6 foot racks. The remaining space in the second rack is used to house the VLSI LZPS VME Processing System. Figure 2 illustrates the system block diagram of the second generation VLSI LZPS, which contains four subsystems: the Control and Communication Subsystem (CCS), Frame Processing Subsystem (FPS), Data Set Processing Subsystem (DSPS), and MSS.

Each CPU within the rack runs its own copy of the VxWorks operating system. This is a UNIX like real time operating system that supports Network File System (NFS) protocols as well as FTP. Source code is developed and compiled on a separate platform, such as a SUN workstation and loaded dynamically across the network during operation. This seamless integration of a development platform and its application target provides a powerful real-time software development environment.

The CCS provides system base functions, including command and control, network interfacing, and system data storage. The FPS receives serial telemetry data, performs standard frame processing functions, and outputs synchronized frames to the DSPS. The DSPS extracts source packets out of the frames and delivers packets from each specified source to the user in real-time. It sorts all packets by source, merges real-time and playback data into data sets, and removes redundant data from the data sets. The output of the DSPS is quality annotated data sets. The

MSS serves as a large data buffer for data set processing and rate buffering. The detailed design of each subsystem is given in the following sections.

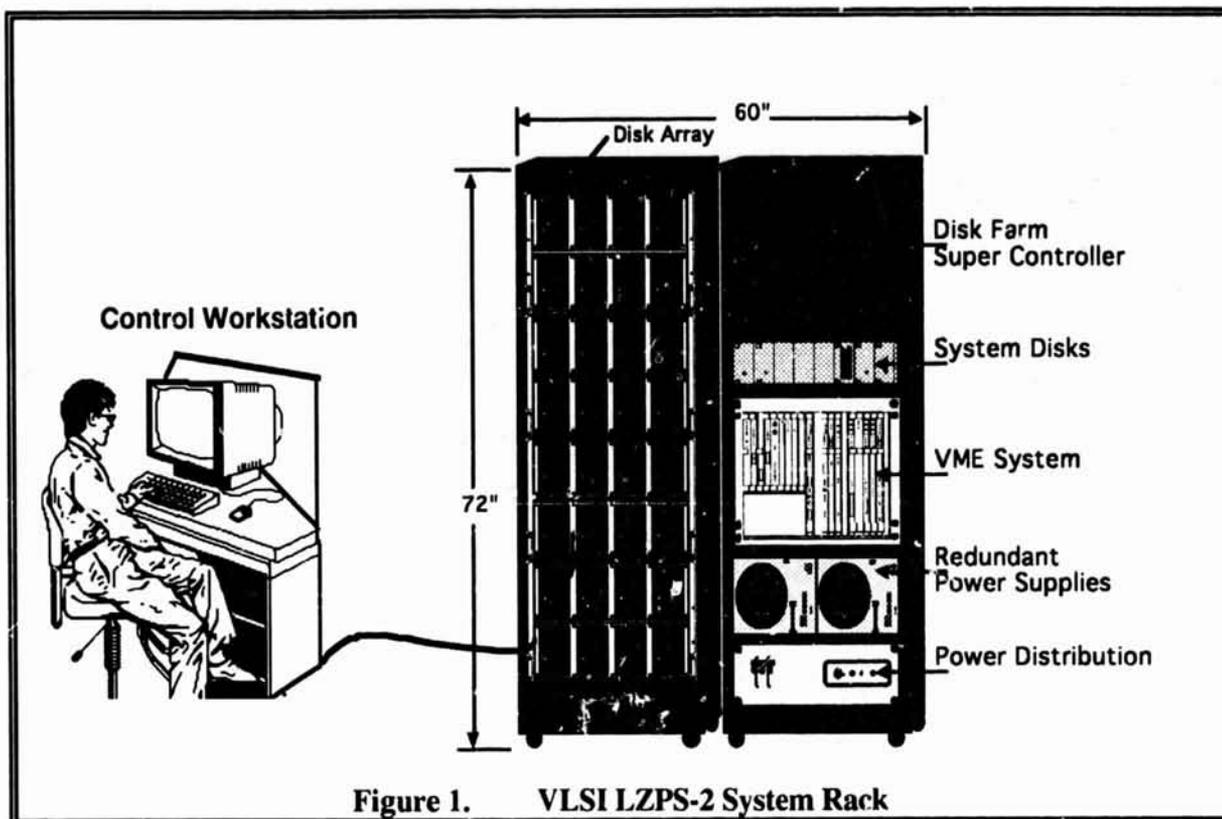


Figure 1. VLSI LZPS-2 System Rack

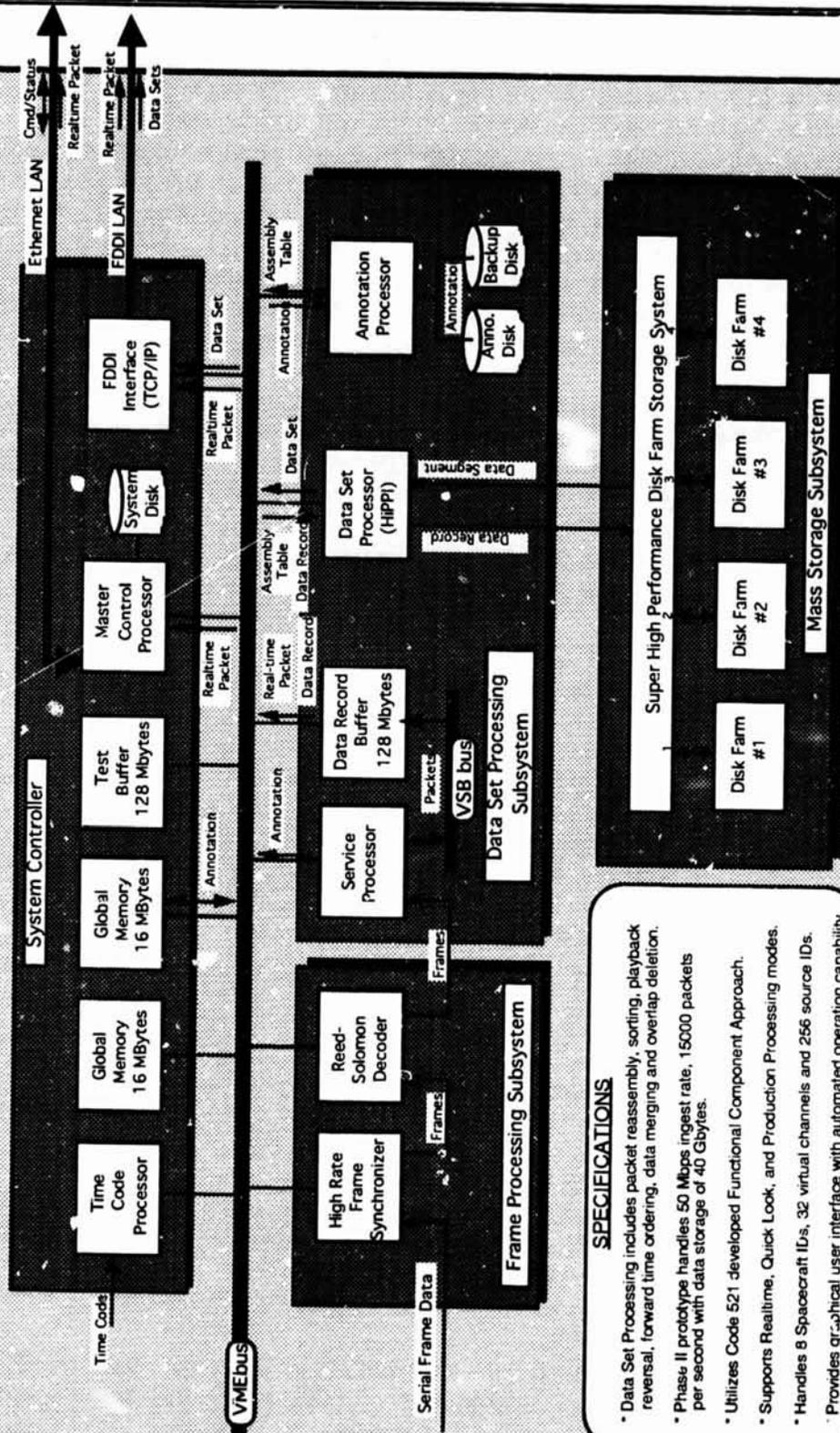
4. VLSI LZPS-2 SUBSYSTEM DESIGN

The VLSI LZPS-2 functional block diagram is depicted in Figure 2, which shows a set of commercial and custom-designed processing modules integrated in the VMEbus environment. These modules are grouped into the CCS, FPS, and DSPS subsystems, with the disk farm being in the MSS. The VMEbus is used for transferring command and status information among the modules. It is also used for high-performance 32-bit and 64-bit block data transfers to store and retrieve data to and from the MSS. High-speed telemetry data is transferred from one module to the other through the VME Subsystem Bus (VSB) and the custom telemetry pipeline implemented on the J3 backplane. Each subsystem will be described in detail in the following sections.

4.1 THE CONTROL AND COMMUNICATION SUBSYSTEM

The CCS consists of a Master Controller card, a FDDI Interface Processor, a Time Code Processor card, a 128 Mbytes Dynamic Random Access Memory (DRAM) buffer, two 16 Mbytes battery backed up Static RAM (SRAM) cards, and a Small Computer Systems Interface (SCSI) disk drive. All modules in the CCS are COTS products.

Code 521 VLSI LZP Phase 2 Prototype



- SPECIFICATIONS**
- Data Set Processing includes packet reassembly, sorting, playback reversal, forward time ordering, data merging and overlap deletion.
 - Phase II prototype handles 50 Mbps ingest rate, 15000 packets per second with data storage of 40 Gbytes.
 - Utilizes Code 521 developed Functional Component Approach.
 - Supports Realtime, Quick Look, and Production Processing modes.
 - Handles 8 Spacecraft IDs, 32 virtual channels and 256 source IDs.
 - Provides graphical user interface with automated operation capability.

Figure 2. VLSI LZP3-2 Functional Block Diagram

The Master Controller is based on a commercial VMEbus single board computer. It provides support for the Ethernet network and for the system disk. Through the use of the VxWorks operating system, both the Ethernet and the system disk can be shared by all CPUs on the VMEbus. The Master Controller accepts commands and configuration parameters from a Control Workstation, interprets the commands, and sends appropriate subcommands to the other system modules. Based on the commands, it configures the system for processing sessions. The Master Controller also gathers housekeeping and processing status and reports them to a remote Control Workstation. If any processing statistics exceed user-specified thresholds, the Master Controller can send event messages to the Control Workstation to alarm the operator. All interfacing to the Control Workstation is done using standard TCP/IP sockets on the Ethernet network.

The CCS provides interfaces to two networks: the Ethernet Local Area Network (LAN), and the FDDI LAN. The Ethernet interface is used for transferring command and status between the VLSI LZPS-2 and the Control Workstation. It may also be used for transferring real-time packets from the VLSI LZPS-2 directly to the user during real-time processing.

The FDDI LAN links the VLSI LZPS-2 directly to the user. Real-time packets can be sent out to the FDDI LAN during real-time processing. All production data sets are sent to the user via the FDDI LAN. As with the Ethernet, full TCP/IP support is provided for all data going out the FDDI port. The VLSI LZPS-2 will send each data set using FTP to designated users according to an operator-defined distribution table. This is a new feature that eliminates the need for an additional system to handle data distribution.

The 32 Mbytes of battery backed up SRAM serve as non-volatile ram disks used for maintaining a system database for high-speed access. The Time Code Processor inputs NASA36 time code and provides the current time to the FPS for time stamping of incoming frames. The 128 Mbytes DRAM buffer serves two purposes. During data set outputting, it provides rate buffering between the DSP and the FDDI network interface. The second use is during internal system testing. Test data is processed by the VLSI LZPS, and data sets are placed in the buffer memory for error checking. This allows the system to perform a full internal self test without extra equipment.

4.2 THE FRAME PROCESSING SUBSYSTEM

The FPS consists of a High-rate Frame Synchronizer (HRFS) card and a Reed-Solomon Decoder (RSD) card designed and built by the DSTD. Their functions are illustrated in Figure 3, together with modules from the DSPS.

The HRFS performs the frame synchronization functions. It receives serial telemetry data and clock through either a RS-422 interface, or a 100K Emitter Coupled Logic (ECL) interface. The card synchronizes the serial data to transfer frames according to a specified synchronization pattern and strategy. The card checks for Cyclic Redundancy Check (CRC) errors on each frame, if desired, and all results are reported in a quality trailer appended to each frame.

The RSD performs Reed-Solomon error detection and correction on the frame headers and frame data. The card is capable of 255-223 decoding on the frames and 10-6 decoding on the frame headers with interleaves 1 through 5. The results of all the error detection and correction are appended to each frame in a second quality trailer as it is sent the DSPS subsystem. The operator specifies the type of decoding desired and the filtering options for the RSD. A bypass option is provided for non-Reed Solomon encoded frames as well.

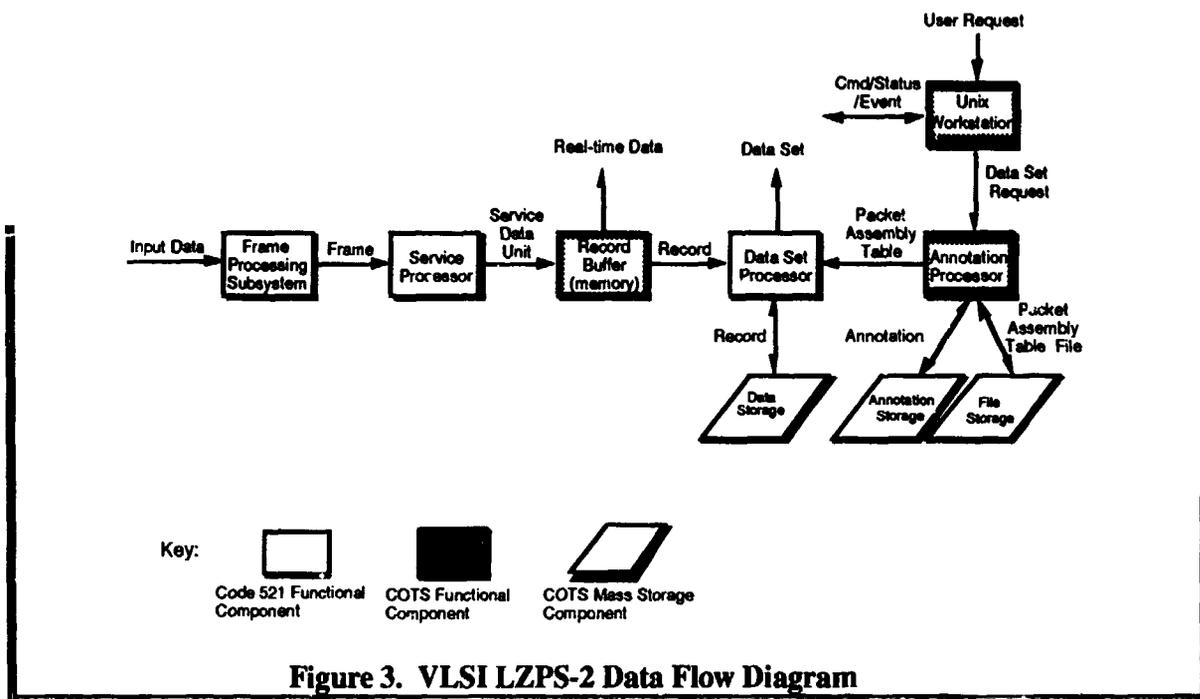


Figure 3. VLSI LZPS-2 Data Flow Diagram

4.3 THE DATA SET PROCESSING SUBSYSTEM

The DSPS consists of a Service Processor, a Data Set Processor, an Annotation Processor, a 128 Mbytes Data Record Buffer, and two SCSI disk drives. The Annotation Processor, Data Set Processor motherboard, and the Data Record Buffer are all COTS VMEbus products. The Service Processor and a mezzanine on the Data Set Processor are custom-designed and built by the DSTD and described in References 5 and 6. Their operations are also illustrated in Figure 3.

The Service Processor receives transfer frames from the RSD. It extracts packet data pieces from the frames, reassembles source packets, checks packet errors, and generates annotation for each packet. During a pass, packets from specified sources as specified by spacecraft ID, Virtual Channel Identifier (VCID), and Application Process Identifier (APID) are output to the user through the CCS as soon as they are received. The Service Processor also sorts packets by source and groups them into data records while outputting them to the Data Record Buffer (DRB) on the VSB. Packet time code is extracted, and sent to the Annotation Processor (AP) together with packet quality information as annotation data for storage in the annotation disks. Whenever a record is full, the Data Set Processor moves packets from the record buffer to the data disk through the VMEbus using the VME64 protocol.

When the pass is over, the AP examines the annotation data of each sensor, which consists of one or more sources, to determine how to merge real-time and playback data into a production data set, how to forward-time-order the packets, and where the overlap boundaries and redundant packets are. The result of this analysis will be stored in a data set assembly table file which will serve as an instruction set for assembling a data set. In addition to the assembly instruction sets, the AP generates quality annotation for each data set. The quality annotation indicates which packets have

errors and type of errors; for example: the packet came from a frame with CRC errors. The quality annotation also indicates the locations and sizes of gaps in the data set.

The Data Set Processor can begin output processing once each data set assembly file is finished by the AP. The Data Set Processor reads the assembly file, and begins retrieving data records from the MSS. The data records are received on the DSP from the HiPPI port, and locally Direct Memory Access (DMA) transferred to the Data Reassembly Unit Mezzanine (DRUM) for reassembly. The DRUM is a custom designed card by the DSTD and contains the Enhanced Ram Controller (ERC) ASIC also developed by the DSTD [7][8]. The ERC provides 4 Mbytes of data storage, with flexible output formatting based on instructions loaded into the chip. Once the ERC buffer is loaded, the DSP begins outputting the data sets from the DRUM, and DMAs the data to the FDDI interface using VME 32-bit block transfers. The FDDI interface then transfers the data using FTP to the user. This operation is repeated until the entire data set is output. The DSP then waits for the next assembly file from the AP. This direct FTP from the VLSI LZPS eliminates the need for another system to handle the data set transfer and maximizes the utilization of the MSS by using it as a short-term data storage device, not just a rate buffering device. The use of the DRUM and HiPPI card reduces the three 9u VME cards in the DSP subsystem of the first generation VLSI LZPS to one 6u card in the second generation system.

4.4 THE MASS STORAGE SUBSYSTEM

In telemetry level zero processing, data merging and overlap deletion functions can only be accomplished after all data has been received. Therefore, the VLSI LZPS system needs to store enough data to meet the users requirements for data set size. In addition to accumulated storage, rate buffering is required between the telemetry input and data set output. This data storage and rate buffering capability is provided by the MSS.

The MSS employs a Maximum Strategy HiPPI Super Disk Array system (super disk farm) with 40 Gbytes of disk space, configurable up to 320 Gbytes. This system is an enhanced version of the SP2 unit used in the first generation of VLSI LZPS. The SP2 model is a single unit capable of 160 Mbps data transfers and 10 Gbytes of data storage. The Super Disk Farm uses four SP2 units, and is capable of 640 Mbps continuous data transfers and 320 Gbytes of redundant storage. The super disk farm contains a super controller with a HiPPI interface and four custom ports to interface to SP2 disk farms. The super controller stripes the data across 4 of the SP2 units which in turns stripes the data across 8 disks with parity. This dual-level striping allows the super disk farm to operate at the full 640 Mbps continuous ingest rate. The DSPS interfaces to the super controller through the HiPPI network interface. This link is capable of 800 Mbps burst data transfers. Due to the DSP VMEbus interface, the maximum data transfer rate achievable is 408 Mbps from the Data Record Buffer to disk. This transfer speed far surpasses the system requirements of 50 Mbps and ensures maximum available bandwidth on the VMEbus for other operations. Information concerning the speed evaluation of the VMEbus to HiPPI to Disk farm link are available in reference 9.

Data integrity is an absolute must in an operations environment where serial data retransmission is either impossible, extremely difficult, or very expensive. This fact imposes the requirement on the VLSI LZPS that the MSS will function normally, without interruption or data loss, even if disk drives fail within the subsystem. The Strategy HiPPI Super Disk Farm achieves true fault tolerant operations with the use of a 48-bit Error Correction Code (ECC), parity disk drive, and stand-by disk drive on each SP2 unit; there are four SP2 units in the system. To further expand the fault

tolerance, additional SP2 units can be added to the Supper Controller to provide a second layer of Parity and Standby Disk Farms. With this scheme and ECC and parity protection, the Super Disk Farm can operate at full speed even if a disk drive is lost. Data integrity is preserved by the parity drives that can be used to reconstruct data that was on a lost drive. The drives are hot-swappable, and reconstruction is transparent, meaning it can be accomplished while data transfers are being performed.

5. AUTOMATED OPERATIONS ENVIRONMENT

One major goal of the whole VLSI LZPS development project was to provide fully automated operations of the system, from activity scheduling and remote setup to status gathering and data distribution. To accomplish this goal, the DSTD developed a UNIX-based software package called Telemetry Processing Control Environment (TPCE) [10]. The role of TPCE is to provide a Graphical User Interface (GUI) to make configuring and gathering status from the VLSI LZPS more user friendly. The system accepts an activity schedule from a file or network socket. TPCE will automatically initiate telemetry processing based upon activities identified in the schedule and can be edited by the local operator if necessary. Each activity in the schedule is associated with a pre-defined configuration set which is used for processing that particular telemetry session. Through the use of configuration sets, the VLSI LZPS/TPCE combination can support various types of telemetry processing scenarios. TPCE also provides the capability to edit all configuration sets. Data set distribution by the VLSI LZPS is also managed by TPCE. A log is kept of all data sets output, and any retransmission of individual data sets by user request. TPCE provides the link between the operator/user of the VLSI LZPS and the hardware, there-by keeping the user interface consistent, even after hardware upgrades.

6. POTENTIAL APPLICATIONS

The second generation VLSI LZPS is developed in anticipation of demands for high rate ground data processing systems in the 1990's and beyond. The selection of functional and performance specifications for the prototype has closely followed the requirement development of NASA's major missions such as Earth Observing System (EOS), Space Station, and Landsat-7. As a compact CCSDS telemetry processing system, the VLSI LZPS-2 can be used in many applications, including science data processing at permanent sites and at transportable ground stations, spacecraft Integration and Test, and ground data system testing and verification. Its modular architecture allows it to be configured as a stand alone system, or as a core processor in a large scale ground data system. Based on the FAST PPS development experience, the prototype system can be converted into a full production system in about 10-12 months.

7. SUMMARY

The design of the second generation VLSI LZPS has been discussed with the implementation of the particular subsystems covered in detail. Based on functional components and VLSI technologies, the VLSI LZPS supports CCSDS version 2 data processing at rates up to 50 Mbps with real-time and near real-time science data processing and fully automated data distribution. With the addition of a UNIX workstation, fully automated operation is achieved with the TPCE system. The fully automated operation allows projects to reduce operational staffing as well as operational costs. Because of extensive use of VLSI components and modular design, the system renders compact size, high reliability and high maintainability. The use of hardware and software functional components allows a full production system to be ready in less than a year.

8. REFERENCES

1. "Packet Telemetry," CCSDS 102.0-B-3, Blue Book, Consultative Committee for Space Data Systems, November, 1992.
2. "Advanced Orbiting Systems, Networks, And Data Links," CCSDS 701.0-B-2, Blue Book, Consultative Committee for Space Data Systems, November, 1992.
3. Shi, J., Horner, W., Grebowsky, G., Chesney, J., "A Prototype VLSI Level Zero Processing System Utilizing the Functional Component Approach," Proceedings of International Telemetry Conference, 1991, pp. 519-531.
4. Shi, J., Horner, W., Grebowsky, G., Chesney, J., "Fast Auroral Snapshot Explorer (FAST) Packet Processing System (PPS)," Proceedings of International Telemetry Conference, 1993, pp. 445-459.
5. "Service Processor Hardware Definition Document Revision B," Code 521, NASA Goddard Space Flight Center.
6. "Reed Solomon Hardware Definition Document, Revision C," Code 521, NASA Goddard Space Flight Center.
7. "Data Reassembly Unit Mezzanine Hardware Definition Document, Revision -," Code 521, NASA Goddard Space Flight Center.
8. "Enhanced Ram Controller," Code 521 ASIC Definition Document, Revision B, Code 521, NASA Goddard Space Flight Center.
9. "High-Performance Parallel Interface (HiPPI)-830/Disk Farm Test Report," Code 521, NASA Goddard Space Flight Center.
10. Costenbader, J., Thorn, K., "Reusable Software Components for Monitoring and Control of Telemetry Processing Systems," Proceedings of International Telemetry Conference, 1993, pp. 461 - 468.

9. NOMENCLATURE

AOS	Advanced Orbiting Systems
AP	Annotation Processor
APID	Application Process Identifier
ASIC	Application-specific Integrated Circuits
CCSDS	Consultative Committee for Space Data Systems
COTS	Commercial-Off-the-Shelf
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
DRAM	Dynamic Random Access Memory
DRUM	Data Reassembly Unit Mezzanine
DRB	Data Record Buffer
DSTD	Data Systems Technology Division
ECL	Emitter Coupled Logic
EOS	Earth Observing System
ERC	Enhanced Ram Controller
FAST	Fast Auroral Snapshot Explorer
FDDI	Fiber Distributed Data Interface
FPS	Frame Processing Subsystem
FTP	File Transfer Protocol
GSFC	Goddard Space Flight Center
GUI	Graphical User Interface
HiPPI	High-performance Parallel Peripheral Interface
HRFS	High-rate Frame Synchronizer
LAN	Local Area Network
LZP	Level Zero Processing
LZPS	Level Zero Processing System
Mbps	Mega bits per second
Mbytes	Megabytes
MSS	Mass Storage Subsystem
NFS	Network File System
RSD	Reed-Solomon Decoder
SCSI	Small Computer Systems Interface
SRAM	Static RAM
TCP/IP	Transmission Control Protocol/Internet Protocol
TPCE	Telemetry Processing Control Environment
VCA	Virtual Channel Access
VCDU	Virtual Channel Data Unit
VCID	Virtual Channel Identifier
VMEbus	Versa Module Eurocard bus
VLSI	Very Large Scale Integration

**A CORPORATE MEMORY FOR THE GSFC MISSION OPERATIONS
DIVISION**

Beryl Hosack
CSC/GSFC

Paper Not Available

FAST COMPUTATIONAL SCHEME OF IMAGE COMPRESSION FOR 32-BIT MICROPROCESSORS

Leonid Kasperovich
Space Research Institute
84/32 Profsoyuznaya St., Moscow, 117810, Russia
Fax: 7-095-310-7023

Abstract - This paper presents a new computational scheme of image compression based on the discrete cosine transform (DCT), underlying JPEG and MPEG International Standards. The algorithm for the 2-d DCT computation uses integer operations (register shifts and additions / subtractions only), its computational complexity is about 8 additions per image pixel. As a meaningful example of an on-board image compression application we consider the software implementation of the algorithm for the Mars Rover (Marsokhod, in Russian) imaging system being developed as a part of Mars-96 International Space Project. It's shown that fast software solution for 32-bit microprocessors may compete with the DCT-based image compression hardware.

INTRODUCTION

The discrete cosine transform (DCT) is widely applied in various fields including image data compression and was chosen as a basis of International JPEG (Joint Photographic Experts Group) and MPEG (Motion Pictures Experts Group) image / video Compression Standards. The DCT technique is applicable to the digital representations of natural scenes and other types of continuous tone gray-scale and color images.

An extensive research experience in the field of DCT studying has been summarized in the various publications and textbooks (e.g., Pennebaker et al., 1993). The most meaningful example of the 8x8 DCT implementation (Feig et al., 1992) uses 94 real multiplications and 454 additions, but only 54 multiplications and 162 additions in a scaled version, where the DCT computation is followed by normalizing and quantization.

Due to the rounding-off and truncation effects of the quantization process in image compression, one can carry out, in practice, all DCT calculations approximately, not increasing the overall computational error. Then making use of the floating-point multiplications is not necessarily. In this way, a technique based on generalized Chen transform for approximating with rational numbers the scales 8x8 DCT, has been developed that uses 608 additions per 8x8 image fragment (Allen et al., 1992).

This paper presents an improved algorithm on the basis of the scaled 8x8 DCT approximation method that has been previously published by the author in cooperation with Dr. V.F. Babkin (Kasperovich et al., 1993). The algorithm presented uses 530 additions (vs. 684 as before) per 8x8 block that is a little bit more than the overall number of arithmetical operations used in the Feig-Winograd algorithm, but considerably fewer than in the approximation algorithm by Allen and Bronstein.

Note that in the wide range of microprocessors a floating-point multiply execution takes ordinarily more processor clock cycles than a summation of integers, hence a

multiplication-free algorithm might be preferable in the applications. This paper consists of 3 sections discussing the algorithm, its accuracy and on-board implementation performance.

DCT DECOMPOSITION

The two-dimensional forward DCT (FDCT) of an input 8x8 block consisting of integers $x_{i,j}$, $i = 0, 1, \dots, 7$, $j = 0, 1, \dots, 7$ is defined by the following formula:

$$Y_{m,n} = \frac{1}{4} K(m) K(n) \sum_{i=0}^7 \sum_{j=0}^7 X_{i,j} \cos \frac{(2i+1)m\pi}{16} \cos \frac{(2j+1)n\pi}{16}$$

where:

$$K(t) = \begin{cases} 1/\sqrt{2}, & t = 0 \\ 1, & \text{otherwise} \end{cases}$$

$$m = 0, 1, \dots, 7, \quad n = 0, 1, \dots, 7.$$

The FDCT can be accomplished in row-column fashion using one-dimensional transform:

$$\begin{pmatrix} Y(0) \\ Y(1) \\ Y(2) \\ Y(3) \\ Y(4) \\ Y(5) \\ Y(6) \\ Y(7) \end{pmatrix} = \frac{1}{2} \begin{pmatrix} \gamma(4) & \gamma(4) \\ \gamma(1) & \gamma(3) & \gamma(5) & \gamma(7) & -\gamma(7) & -\gamma(5) & -\gamma(3) & -\gamma(1) \\ \gamma(2) & \gamma(6) & -\gamma(6) & -\gamma(2) & -\gamma(2) & -\gamma(6) & \gamma(6) & \gamma(2) \\ \gamma(3) & -\gamma(7) & -\gamma(1) & -\gamma(5) & \gamma(5) & \gamma(1) & \gamma(7) & -\gamma(3) \\ \gamma(4) & -\gamma(4) & -\gamma(4) & \gamma(4) & \gamma(4) & -\gamma(4) & -\gamma(4) & \gamma(4) \\ \gamma(5) & -\gamma(1) & \gamma(7) & \gamma(3) & -\gamma(3) & -\gamma(7) & \gamma(1) & -\gamma(5) \\ \gamma(6) & -\gamma(2) & \gamma(2) & -\gamma(6) & -\gamma(6) & \gamma(2) & -\gamma(2) & \gamma(6) \\ \gamma(7) & -\gamma(5) & \gamma(3) & -\gamma(1) & -\gamma(1) & -\gamma(3) & \gamma(5) & -\gamma(7) \end{pmatrix} \begin{pmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \\ X(4) \\ X(5) \\ X(6) \\ X(7) \end{pmatrix}$$

where: $\gamma(k) = \cos(2\pi k / 32)$.

Setting $\Psi = \sqrt{2}$, $C_1 \Psi = \gamma(1) / \gamma(7)$, $C_2 \Psi = \gamma(3) / \gamma(7)$, $C_3 \Psi = \gamma(5) / \gamma(7)$ and representing the transformed values $Y(i)$ in a "quasi-complex" form $R(i) + \Psi A(i)$, leads to the Kasperovich - Babkin FDCT algorithm mentioned above, in which the attends in the formula for the 2-d FDCT values $[R(R) + 2A(A)] + [A(R) + R(A)]\sqrt{2}$ are represented through the "basic" elements $A(A)$ by means of additions and subtractions. 64 multiplications by the

constants C_1, C_2, C_3 , which are closed to 5, 3 and 2, are sufficient for obtaining the basic elements. All multiply operations by these 3 constants are substituted in the DCT approximation by the additions and subtractions. Further,

$$2X_1 = [X_3 + X_5\Psi]$$

$$X_2 = (1 + \Psi)X_6$$

$$X_3\Psi = (X_1 + X_7)$$

where: $(a + \Psi b) = a - \Psi b$

Thus, only 30 of 60 multiplications by $\sqrt{2}$ should be computed, which are practically replaced

with a Taylor series approximation: $\sqrt{2} \approx 1 + \frac{1}{2} - \frac{1}{16}$.

Theorem. i) FDCT can be performed as an operator composition $FDCT = F \circ D \circ C \circ T$, where

T is a preliminary transform (192 preadditions),

C - computation of the basic elements,

D - deriving the output values,

F - pointwise factorization (scaling);

ii) C uses 64 multiplications by predefined constants, D calls 30 multiplications by

$\sqrt{2}$;

iii) Approximation of C uses 144 additions, approximation of D calls 194 additions;

iv) $IDCT = T^{-1} \circ F \circ D \circ C$;

The transformations T and C are separable (i.e. can be computed in row-column fashion) meanwhile D is non-separable 2-d transform. Generally speaking, the number of preadditions equals to 224 (as much as in Feig-Winograd algorithm), but the certain part of it is done while computing the basic elements (C transformation) in order to preserve the algorithm symmetry.

32-BIT IMPLEMENTATION

The DCT itself is parallelizable that makes it possible to group data elements in such a way, that DCT computation could be considered as sequential single-instruction/multiple-data process. In particular, two additions $a+b$ and $c+d$ can be achieved in one $(a,c) + (b,d)$, coupling the elements of an input 8×8 block into the pairs. Assuming that all computations can be done with 16-bit arithmetic, that observation is applicable to a single microprocessor taking the substantial advantages of a full-length processor word of 32-bit or newest 64-bit devices.

Since the image data precision is ordinarily 8-bit per sample and the average number of summation per point is $530/64 < 8.3$ in our algorithm, then in most case (48 of 64) the computations are done within 16-bit range and can be paralleled as mentioned above. However, this is worthy in a case of multiplication-free computational scheme, because a

fractional multiplying will destroy the least significant 16-bit word of a pair: .In turn, an additional error in most significant 16-bit word produced in our algorithm by a carry bit of addition/subtraction of the least significant 16-bit words can be neglected due to the scaling performed by the operator F and quantization.

The test of *LENA* standard image gives a good illustration of the tolerable computational accuracy, comparing the algorithm presented with a direct floating-point method. The maximum pint-size difference between the original and expanded pictures is identical for both methods, the mean arithmetic modules error is slightly different: 3.516 versus 3.501 in the direct computation.

APPLICATION TO THE ON-BOARD PROCESSING

In this section we consider the Mars Rover imaging system, that contains a panoramic camera along with 2 stereo cameras. Three compression modes are planned:

- Receiving the descent camera images compressed as the separate frames (specified data rate is 1 frame of size 512x512x8 bit per second).
- Compression of high resolution panoramic camera still images.
- Image sequence compression to create the virtual environment from real Martian surface data in order to control and navigate the rover manually.

In this way, an image compression module (ICM) based on JPEG compression chip set from Matra Maiconi Space (France) was supposed to be installed in Mars Rover as a hardware accelerator board. The ICM technical specifications are 3 watts consumption at 1 megapixel /sec; 12000 mm; 200 grams (see Mars-94 in the pictures, 1992). The chip set contains the two CMOS ASICs.

An alternative approach implementing in software a new algorithm to compute the DCT in multiplication-free 32-bit arithmetic seems to be more preferable. In order to provide the autonomy of movement, control and timing experiments, data collection and storing etc., the rover is equipped with a on-board computer based on the powerful 32-bit T805 transputer from INMOS Corporation (see Transputer Data Book, 1990), that can be regarded both as a special (i.e. image processing) and a general purpose processor. Major characteristics of IMS-T805 are:

- 32 bit internal and external architecture.
- 30 MIPS (peak) instruction rate.
- 4 Kbyte on-chip RAM direct addressable.
- Internal timers.
- 4 fast Serial Links (10 Mbit/sec).
- Less than 1 watt power consumption at 30 Mhz.

The heart transputer modules, which are the real copy of each other both electrically and even mechanically. There is no distinguished one among them as far as the access to the peripheral blocks concerned, but, and it is a substantial point, only two out of four transputer modules are powered at a time. Which two, it is determined by the actual state of the overswitch logic (Balazs et al., 1994).

The software implementation of the image compression algorithm for the on-board computer provides the same compression rate as ICM hardware, requiring no additional

weight and power consumption. Compression mode 3 gives a good illustration of the software solution flexibility, where DCT computation for intra- and interframe compression is combined with another algorithm (Motion Estimation) for the successive frame matching, that is a part of stereo-based autonomous navigation software.

CONCLUSION

The reliability and performance of the Mars Rover systems including on-board computer and the application software have been evaluated in the several tests with the real test site observation (e.g. Kamchatka, Far East, Russia, August 1993 and Mohave Desert, California, US, March 1994). The rover control as well as the compressed data transmission has been provided via satellite communication link. The results are quite good and show the possibility to use the software solution of the special tasks in various applications, in particular image processing and compression, where hardware assistance is currently required

REFERENCES

- Pennebaker, E., & Winograd, S. (September 1992). Fast algorithms for the discrete cosine transform. *IEEE Trans. Signal Processing*, 40(9), 2174-2193.
- Allen, J.D., & Blonsein, S. (July 1992). US patent # 5, 129,015.
- Kasperovich, L.V., & Babkin, V.F. (1993). Fast discrete cosine transform approximation for JPEG image compression. *Lecture Notes in Comp. Sci.*, 719, 98-104.
- Mars-94 in the picture. (July 1992). *News from Prospace*, 34, 48-50.
- Balazs, A. & Biro, J. & Szalai, S. (15-21 May 1994). Onboard computer for Mars-96 rover. *Proc. of 2-nd Int. Sympos. on Missions, Technologies and Design of Planetary Rovers*. Center National d'Etudes Spatiales(CNES, France)-Russian Space Agency (RKA), Moscow-St.-Petersburg (Russia) (to be published).
- Transputer Data Book. (1992). 2 nd Edition, London, UK:Prentice Hall.

THE DEVELOPMENT AND OPERATION OF THE INTERNATIONAL SOLAR-TERRESTRIAL PHYSICS CENTRAL DATA HANDLING FACILITY

Kenneth Lehtonen
 NASA/Goddard Space Flight Center
 Information Processing Division
 Greenbelt, MD 20771

ABSTRACT

The National Aeronautics and Space Administration (NASA) Goddard Space Flight Center (GSFC) International Solar-Terrestrial Physics (ISTP) Program is committed to the development of a comprehensive, multi-mission ground data system which will support a variety of national and international scientific missions in an effort to study the flow of energy from the sun through the Earth-space environment, known as the geospace.

A major component of the ISTP ground data system is an ISTP-dedicated Central Data Handling Facility (CDHF). Acquisition, development, and operation of the ISTP CDHF were delegated by the ISTP Project Office within the Flight Projects Directorate to the Information Processing Division (IPD) within the Mission Operations and Data Systems Directorate (MO&DSD). The ISTP CDHF supports the receipt, storage, and electronic access of the full complement of ISTP Level-zero science data; serves as the linchpin for the centralized processing and long-term storage of all key parameters generated either by the ISTP CDHF itself or received from external, ISTP Program-approved sources; and provides the required networking and "science-friendly" interfaces for the ISTP investigators. Once connected to the ISTP CDHF, the online catalog of key parameters can be browsed from their remote processing facilities for the immediate electronic receipt of selected key parameters using the NASA Science Internet (NSI), managed by NASA's Ames Research Center.

The purpose of this paper is twofold: (1) to describe how the ISTP CDHF was successfully implemented and operated to support initially the Japanese Geomagnetic Tail (GEOTAIL) mission and correlative

science investigations, and (2) to describe how the ISTP CDHF has been enhanced to support ongoing as well as future ISTP missions. Emphasis will be placed on how various project management approaches were undertaken that proved to be highly effective in delivering an operational ISTP CDHF to the Project on schedule and within budget. Examples to be discussed include: the development of superior teams; the use of Defect Causal Analysis (DCA) concepts to improve the software development process in a pilot Total Quality Management (TQM) initiative; and the implementation of a robust architecture that will be able to support the anticipated growth in the ISTP Program science requirements with only incremental upgrades to the baseline system. Further examples include the use of automated data management software and the implementation of Government and/or industry standards, whenever possible, into the hardware and software development life-cycle. Finally, the paper will also report on several new technologies (for example, the installation of a Fiber Data Distribution Interface network) that were successfully employed.

INTRODUCTION

NASA's spacecraft contribution to the ISTP Program includes the Interplanetary Physics Laboratory (WIND: 11/94 launch) and the Polar Plasma Laboratory (POLAR: 11/95 launch). The international contribution includes the GEOTAIL mission (successfully launched in July 1992) developed by the Japanese Institute for Space and Astronautical Science (ISAS) and the Solar and Heliospheric Observatory (SOHO: 7/95 launch) and Plasma Turbulence Laboratory (CLUSTER: 12/95 launch) being developed by the European Space Agency. In addition, scientific contributions are being provided by several ground-based radar investigations and

on-orbit correlative science missions such as the Los Alamos National Laboratory (LANL) spacecraft and the Geostationary Operational Environmental Satellites (GOES 6/7).

Within the framework of the ISTP Program objectives to combine resources and to promote cooperation in the scientific communities on an international scale, the primary function of the ISTP CDHF became one of computing summary parameter data ("*key parameters*") for every instrument on the GEOTAIL, WIND, and POLAR spacecraft, three instruments on SOHO, and the magnetic field instrument on the Interplanetary Monitoring Platform-8 (IMP-8); and to ingest and catalog key parameters from external sources such as the ground-based radars and other equatorial spacecraft missions that have been made an integral part of the overall ISTP Program. The key parameters provide for a quick, low resolution time series (on the order of one minute) survey of the global geospace. The major advantages of providing key parameters to the science community are their diversity of coverage over the geospace, timeliness and availability. The goal is to generate the key parameters within 6 hours of receipt of the corresponding Level-zero data.

The major functions of the ISTP CDHF are summarized as follows:

- Receive telemetry, orbit, attitude, and command history data from external ground system elements
- Receive and process near real-time data for WIND and POLAR
- Generate key parameter data for all instruments onboard GEOTAIL, WIND, and POLAR, and, selected instruments from the IMP-8 and SOHO spacecraft
- Receive key parameters from ground-based radar investigators and other correlative spacecraft such as LANL, GOES, and CLUSTER
- Store telemetry, orbit, attitude, command history, and key parameter data sets in online storage for user access and transfer to the

IPD's Data Distribution Facility (DDF) for subsequent distribution on Compact-Disk Read Only Memory (CD-ROM) media and to the National Space Science Data Center (NSSDC) for long-term archival purposes

- Manage, track, and account for all data flowing through the CDHF
- Provide interactive user services for catalog access, online data access, and data transfers

The remainder of this paper discusses several key programmatic and technical elements which were employed that directly led to the successful implementation and operation of the ISTP CDHF.

IMPLEMENTATION OF THE ISTP CDHF

Project Management Team

From the beginning of the implementation of the ISTP CDHF, a concerted effort was made to establish a solid project management team. This was accomplished by "matrixing" both technical and management staffs from three GSFC Directorates, namely, the Flight Projects Directorate (Code 400), the Space Sciences Directorate (Code 600), and the Mission Operations & Data Systems Directorate (Code 500). Once the implementation team was in place, several methods for conducting business expeditiously among the three Directorates were established and an excellent partnership evolved as a result.

The following summarizes some of the more important aspects of this partnership and the associated advantages that accrued, particularly from the perspective of the Information Processing Division:

- Requirements documents, Interface Control Documents, and other key documents were negotiated **directly** between the ISTP Project and the IPD which enabled the requirements to be captured in a high-fidelity manner.
- The IPD ISTP CDHF development team was given full responsibility to work directly

with the ISTP Principal Investigator (PI)/Co-Investigator (Co-I) teams both nationally and internationally and Code 600 personnel, when required, with Project oversight.

- The IPD ISTP CDHF development team was an integral part of the Project team and played a major role in the technical decision-making processes. The team was given broad latitude to make technical trade-offs and to suggest solutions, and as a result, a variety of solutions to improve system performance and to reduce on-going ISTP CDHF operations and maintenance costs were provided.

ISTP Science Management Team

The primary guiding force for the evolution of the ISTP CDHF as the key ISTP Program science facility was the ISTP Science Working Group (SWG) chaired by the Project Scientist. The SWG--which included the Project Scientist, Deputy Project Scientists and all of the Instrument Investigators PIs, Co-Is, Ground-Based Investigators, and Theory Investigators--established the ISTP science objectives in coordination with the national and international ISTP science community. The SWG was instrumental in developing a set of "Rules of the Road." This set of rules delineated how the ISTP science community shall "behave" with respect to data generation, data exchange, and data access rights (for example, proprietary data periods).

In order to use effectively the key parameters for collaborative science, several data formatting and exchange standards were jointly prepared by the ISTP science community and the ISTP CDHF development team. By working closely with the various science teams and actively soliciting their inputs, a very useful set of ISTP data standards and conventions was developed: first, the standard header used on all science files cataloged on the ISTP CDHF is the Standard Formatted Data Unit (SFDU). The SFDU standard is defined and operated under the auspices of the Consultative Committee for Space Data Systems. The ISTP Project selected SFDUs as a convenient yet standardized way of structuring, managing,

and tracking the multitude and variety of data products resident on the ISTP CDHF; second the SWG recommended the adoption of the NSSDC ISTP Common Data Format (CDF) as the common data format protocol for all key parameters generated within the ISTP Program. The adoption of the CDF, the SFDU concept, and other standards and conventions for the key parameters proved to be crucial to supporting multiple-instrument browsing and collaborative science. Also, the selection of the NSSDC's ISTP CDF provided the ISTP Program with the means to influence its future development.

One of the most significant scientific benefits to date of adopting the CDF and related standards has been the ability for the first time to review key parameter data ranging from 35 Earth Radii (Re) in "front" of the Earth to 200 Re "behind" the Earth in conjunction with geosynchronous orbit data at 6.2 Re and ground-based data.

ISTP CDHF Procurement Team

The ISTP Project Office delegated the procurement responsibility of the ISTP CDHF to the IPD. To that end, a Technical Evaluation Panel comprised of senior technical members of the three Directorates and chaired by the IPD Project Manager was formed. This team evaluated the vendor proposals with an emphasis on selecting a robust architecture amenable to the current ISTP requirements, one that could easily be expanded to accommodate future science requirements, and one with the ability to incorporate commercial off-the-shelf (COTS) hardware and software. In July of 1990, Digital Equipment Corporation (DEC) was selected to deliver, integrate, and test the hardware and operating system components of the ISTP CDHF; in September of 1990, this integrated system was turned over to the IPD for development of the ISTP core applications software.

ISTP CDHF Hardware Implementation

The selection of the DEC VMScluster architecture for the ISTP CDHF was significant because it enabled the ISTP CDHF to be configured as a scalable,

integrated system that provided robustness, stability, high availability, and access to a wide variety of computer processors and storage controllers. The initial configuration of the ISTP CDHF consisted of one VAX 6000-410, one VAX 6000-430, two Hierarchical Storage Controllers (HSC70s), twenty-four RA92 disk drives (36 Billion Bytes[GB]), a variety of terminals and workstations, local area and wide area networking interfaces, and the Virtual Memory System (VMS) operating system.

Not long after the initial configuration was installed, new ISTP Program requirements emerged that impacted the hardware baseline. Key among these were additional processing and storage requirements for the key parameters being generated and received at the ISTP CDHF, a requirement to generate three sets of WIND key parameters in near real-time for the Air Force, and expanded user/operator interface requirements. To satisfy the first requirement, a re-conditioned VAX 9000-210 computer and four RA73 disk drives (8 GB) were procured and integrated into the VMScluster; for the second requirement, a VAX 4000 Model 200 was purchased and connected to the existing internal Ethernet network; and, to address the third requirement, an Alpha 4000 Model 300 workstation was acquired. In each case, the VMScluster architecture provided the needed flexibility and ease in accommodating the new hardware elements. Refer to Figure 1 for a depiction of the ISTP CDHF hardware configuration and major external interfaces.

Another salient feature of the DEC architecture that contributed to the success of the ISTP Program was a robust electronic networking infrastructure that provided connectivity for the world-wide ISTP scientific community. Initially, only DECnet support was provided; however, with the rapid proliferation and emerging importance of scientific workstations running Unix, the need to provide connectivity for ISTP users of the Defense Department's Advanced Research Projects Agency Transmission Control Protocol/Internet Protocol (TCP/IP) became apparent. To meet this need, a VMS-based TCP/IP third-party package from TGV, Inc. called Multinet was acquired. The

Multinet software provided a full suite of TCP/IP services (for example, Telnet, File Transfer Protocol, Simple Network Mail Protocol) which enabled the implementation team to establish connectivity to the growing base of Unix-based users. The programmatic mechanism for achieving this overall global connectivity (DECnet and TCP/IP) was to connect the ISTP CDHF to the NSI.

Internal to GSFC, the ISTP community, particularly members of the WIND PI teams, obtained the advantage of high-speed access (1 Mbit/s) to the ISTP CDHF via the GSFC Local Area Communications Network.

Finally, in a proactive response to ISTP Program science requirements for increased bandwidth, reliability, and security, an ISTP-only Local Area Network (LAN) was installed that was based upon the American National Standards Institute-standard Fiber Distributed Data Interface (FDDI) specification. The resultant backbone FDDI LAN (100 Mbit/s) connected the IPD-funded Data Capture Facility (DCF) and DDF to the Project-funded ISTP CDHF, and has proved to be a very reliable platform for the transfer of large volumes of ISTP-only data products among the three facilities. Note: the ISTP FDDI network was one of the *first* operational FDDI LANs at the GSFC.

ISTP CDHF Software Implementation

The major software activity undertaken was the development of the ISTP CDHF core system or applications "umbrella." This core system was designated as the ISTP CDHF Software System or ICSS. The ICSS was developed by a combined team of civil servants from the IPD and Computer Sciences Corporation contractor personnel from the MO&DSD Systems, Engineering, and Analysis Support (SEAS) Contract.

The approach taken in developing the ICSS was based upon the SEAS System Development Methodology (SSDM). The SSDM represented a disciplined approach for developing software, consistent with the MO&DSD's System Management Policy. By rigorously applying field-proven software-development techniques, the ISTP

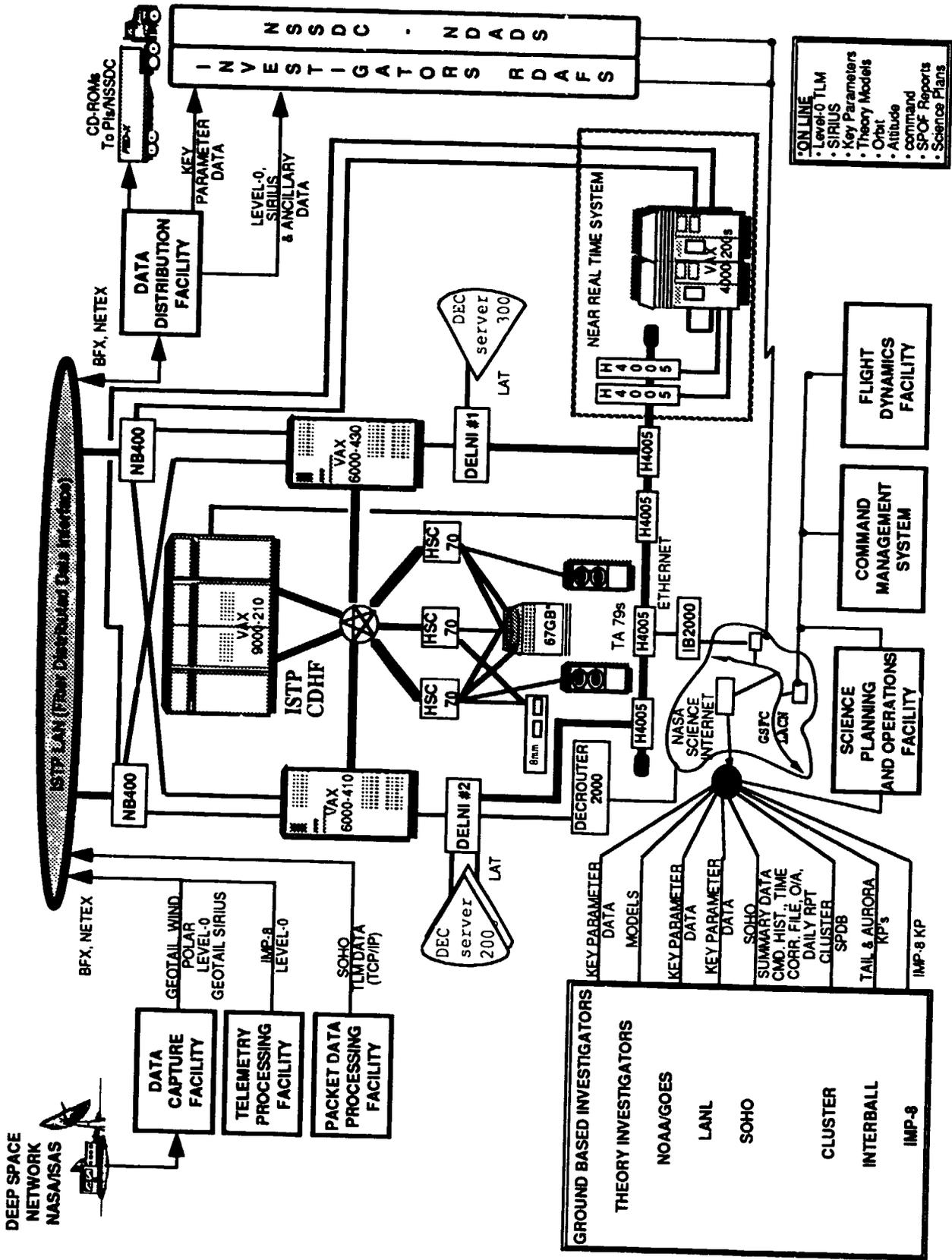


Figure 1: GSFC ISTP Ground Data Handling System

software development team was able to deliver the ICSS on schedule, within budget, and meeting all of the technical and operational requirements.

Another approach that was emphasized and proved invaluable was the use of COTS software to reduce the implementation time. Examples included the MO&DSD Transportable Applications Executive (TAE) software package for the operations interface and the selection of the Oracle relational data base management system (DBMS) to account for all of the ISTP data products. Extensive use of third-party VMS system management and networking software also helped in reducing the amount of new software that had to be developed and tested.

The following lists the major design drivers that were factored into the overall development of the ICSS:

- Must support network access to the ISTP Level-zero and science databases
- Must support multiple missions on different operational timelines
- Must support operations and development activities concurrently and on multiple computers
- Must provide for both manual and automated modes of operation
- Must produce key parameters within 24 hours for each 24 hours of Level-zero data
- Must be able to perform 100% reprocessing of key parameters
- Must process near real-time data within two minutes after receipt (WIND & POLAR only)

In order to implement the core system with these design drivers, the ICSS had to be designed with automation and flexibility in mind. Automation was needed to handle varying processing requirements from multiple missions and to support a variety of external electronic interfaces; flexibility was required so that the ICSS could execute on a variety of VAX computers: from VAX mainframes to Alpha workstations. To accomplish these two objectives, a scheduler concept coupled with the functionality of an Oracle relational DBMS was devised. The general concept was to develop an automated data ingest system which would verify and

catalog all files coming in (mainly unscheduled) from external sources (for example, the DCF, ground-based radar sites, and, ISAS), while at the same time providing the CDHF operations personnel with the means for scheduling, executing and monitoring the key parameter production jobs. This concept proved to be very successful as the ISTP CDHF operation runs "unattended" 75% of the time.

The ICSS was partitioned into two independent software environments: (1) a production environment which supported the daily ingest of Level-zero data, the receipt or generation of the key parameters, and the electronic access from the user community; and, (2) a development and test environment for the PIs' key parameter generation software (KPGS). The latter environment was established to assist the ISTP science teams in the development of their KPGS and was instrumental in the expeditious development, integration, and testing of the ISTP investigator software that executes as part of the ISTP CDHF's production stream. Indeed, one of the biggest challenges was integrating the operating system provided by DEC, the ICSS developed by the IPD, and the KPGS developed remotely at the various science facilities into an ISTP CDHF production environment.

To assist the ISTP science teams, a formal KPGS Integration and Test Team (KITT) was established. The KITT's charter was to work directly with the ISTP science community to provide a smooth transition of a PI's "bench" key parameter program to a full-fledged production version executing on the ISTP CDHF. One of the most important activities of the KITT was to provide "hands-on" training to each of the individual PI science teams. This extensive training, which often required domestic and foreign travel, significantly reduced the KPGS implementation schedule and was very instrumental in fostering an excellent working relationship between the KITT and the various ISTP science teams.

A very successful TQM initiative to emerge from the ICSS development phase was a pilot project to determine if the quality of delivered

software could be improved. This initiative involved applying DCA concepts to the ICSS development process. The basic premise of DCA is that the software developers making the errors have the insight into how those errors/defects were introduced and how to change the process to prevent them in the future. Early results indicated a reduction in error rates between Build 1 and Build 2 and which were also significantly lower than those documented for previous IPD projects.

In summary, to support the scientific objectives of the ISTP Program in general, and the specific objectives of the Japanese GEOTAIL mission, the ICSS implementation team delivered over 75,000 lines of source code on schedule and within budget. This achievement was attributed in large part to the excellent teamwork that was established among the project management teams, the ISTP scientific community (especially our Japanese colleagues at ISAS), and the IPD implementation and test teams.

OPERATIONS OF THE ISTP CDHF

On September 8, 1992, the ISTP CDHF became operational providing support for the GEOTAIL mission, several ground-based radar investigations, and the IMP-8, GOES, and LANL correlative science missions. The ISTP CDHF operations were provided by the MO&DSD Network Mission and Operations Support (NMOS) contract with RMS Technology Incorporated (RMS) responsible for providing daily mission operations and system management functions; AlliedSignal Technical Services Corporation was responsible for all hardware maintenance, sustaining engineering services, and ICSS acceptance testing.

In anticipation of technical and operational questions from the ISTP community, the operations staff was fully trained in all aspects of the ISTP CDHF and were thus able to provide immediate assistance, personally and electronically, to several members of the GEOTAIL science team located at ISAS in Japan, which made communications that much more difficult.

Another important function performed by the operations staff has been the timely re-processing of key parameter data, since it is not uncommon for the science teams to modify or enhance their key parameter science algorithms to reflect better the on-orbit performance of their instrument. The ISTP CDHF operations staff is responsible for accessing the relevant Level-zero data stored on the DDF's optical mass store system. Through a network link to this mass store, the Level-zero data can be expeditiously retrieved and the KPGS re-executed. The updated key parameters are then made available electronically at the ISTP CDHF and on CD-ROMs which are distributed later by the DDF.

In order to keep the ISTP science community informed of events at the ISTP CDHF, the operations staff publishes a bi-annual newsletter containing technical articles submitted from the development staff as well as the science community.

The ISTP CDHF is currently staffed to support a 5 days a week, 8 hours per day operation; because of cross-training of staff personnel, it is anticipated that the current staff will be adequate through the WIND mission, with some increase anticipated to support the SOHO and POLAR missions.

ENHANCEMENTS TO THE ISTP CDHF

In order to provide support for the upcoming WIND, SOHO, POLAR, and CLUSTER missions, additional software enhancements in the form of an ICSS release per mission will be delivered and tested over the upcoming months. In general, because the ICSS was designed from the beginning with multi-mission support in mind, each of the releases contains only minor enhancements. Most of the changes reflect mission-unique requirements and do not impact the existing functionality of the core ICSS. In addition, the KITT will be providing support to those PI teams who will be delivering their KPGS to the ISTP CDHF for integration into the operational environment.

Other noteworthy technical enhancements to be included are: online plotting of orbit data; a "quick-tour" guide for new users; access to Tsyganenko magnetospheric models and Theory Simulation modelling data; extraction of solar activity and magnetic indices from the National Oceanographic and Atmospheric Administration Space Environment Services Center; key parameter plotting using the Interactive Data Language software; generation of key parameters in near real-time during the WIND mission to support an Air Force early warning solar wind experiment; and an upgrade to the I/O subsystem.

CONCLUSIONS

The implementation and operation of the ISTP CDHF was a highly successful program because of several major factors. First and foremost, a strong management team matrixed together and comprised of key individuals from the Flight Projects, Sciences, and Mission Operations and Data Systems Directorates was instituted from the beginning of the Project life-cycle. Decision-making processes were streamlined so that the hardware and software procurement, implementation, and enhancements could proceed smoothly--this was due in large part to the ISTP Program managers' resistance to micro-manage the IPD development effort. In support of this streamlining, appropriate inter-Directorate status reporting and communication methods were devised. Second, by focusing the development of the ISTP CDHF on the science aspects and by working directly with the ISTP science community through the auspices of the SWG, the system that was delivered reflected the way the ISTP science community would operate. Third, the use of existing standards and the decision to adopt a common data format influenced to a large extent by the ISTP science community and to be used by all contributors within the ISTP scientific community have enabled the goal of collaborative science to be attained. And fourth, the development of a robust ISTP CDHF architecture along with the use of standards enabled the ISTP Program to accommodate in a cost-effective manner expanded scientific requirements that have

significantly improved the overall quality of the ISTP science return.

REFERENCES

- Ambardekar, P., Card, D., Dangerfield, O., Giblin, D., Paluzzi, P., "Defect Causal Analysis: A Report From The Field," *Software Quality Journal: Proceedings of the 2nd International Conference on Software Quality*, October 4-7, 1992.
- Kulkarni, Ravi and Whitman, William, *ISTP Data Format Study*, CSC/TR-91/6075, April 4, 1991.
- Mish, W.H. (Editor), *International Solar-Terrestrial Physics (ISTP) Key Parameter Generation Software (KPGS) Standards & Conventions*, Version 1.3, March 1994.
- Data Format Control Document (DFCD) Between The International Solar-Terrestrial Physics (ISTP) Program Information Processing Division (IPD) Ground Data Processing System and the ISTP Mission Investigators*, November 1993, CSC/TR-91/6014.
- International Solar-Terrestrial Physics (ISTP) Program Central Data Handling Facility (CDHF) Software System (ICSS) Operations Users Guide*, Release 4.0, January 1994, CSC/SD-92/6037R3.
- International Solar-Terrestrial Physics (ISTP) Central Data Handling Facility (CDHF) Software System (ICSS) Programmer's Guide to Key Parameter Generation Software*, Revision 3, December 1993, CSC/SD-92/6028.
- International Solar-Terrestrial Physics (ISTP) Central Data Handling Facility (CDHF) Key Parameter Generation Software (KPGS) Integration Test Plan*, July 1992, CSC/TM-91/6084.
- SEAS System Development Methodology (SSDM)*, Computer Sciences Corporation, July 1989.
- Software Requirements Specification for the International Solar-Terrestrial Physics (ISTP) Program Central Data Handling Facility (CDHF) Software System (ICSS)*, Revision 1, February 1993, CSC/SD-89/6017R1.
- System Requirements Specification for the International Solar-Terrestrial Physics (ISTP) Program Data Processing Segment (DPS)*, Revision 2, December 1993, CSC/SD-89/6181.

ECONOMICAL GROUND DATA DELIVERY**Richard W. Markley, Russell H. Byrne, and Daniel E. Bromberg****Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California, 91109****ABSTRACT**

Data delivery in the Deep Space Network (DSN) involves transmission of a small amount of constant, high-priority traffic and a large amount of bursty, low priority data. The bursty traffic may be initially buffered and then metered back slowly as bandwidth becomes available. Today both types of data are transmitted over dedicated leased circuits.

The authors investigated the potential of saving money by designing a hybrid communications architecture that uses leased circuits for high-priority network communications and dial-up circuits for low-priority traffic. Such an architecture may significantly reduce costs and provide an emergency backup. The architecture presented here may also be applied to any ground station-to-customer network within the range of a common carrier. The authors compare estimated costs for various scenarios and suggest security safeguards that should be considered.

INTRODUCTION

The DSN is a geographically distributed antenna network with antenna complexes in Canberra, Australia; Goldstone, California; and Madrid, Spain. The DSN is managed, technically directed, and operated for the National Aeronautics and Space Administration (NASA) by the Jet Propulsion Laboratory (JPL) of the California Institute of Technology in Pasadena, California. Data communications between the complexes and JPL include telemetry, command, tracking, radio science, and monitor and control information. Downlink telemetry data are usually acquired at the remote complexes and transmitted to JPL for further processing, and ultimately delivered to customers located anywhere in the world.

GROUND NETWORK TECHNOLOGY

Spacecraft data are usually delivered over carefully engineered data networks because of their high scientific value and irreplaceability. The DSN is in the midst of upgrading its ground networks to use the Transmission Control Protocol/Internet Protocol (TCP/IP) suite of networking standards, and intermediate buffers. This new architecture provides useful services such as automatic error detection, recovery, flow control, and fault-tolerance. This transition to TCP/IP makes it possible to use commercial, off-the-shelf network devices such as routers and bridges to interconnect local and wide area networks. In addition, the architecture enables NASA to potentially use emerging cost-saving technologies. One such technology that we have investigated provides dial-up bandwidth-on-demand. The enabling devices are dial-up routers and inverse multiplexers, which are an advancement of dial-up router technology.

Dial-up routers are very similar to traditional routers, only they include a network interface to a switched circuit. Whenever the user attempts to send data to a predefined

site, the router signals its interface to dial a dial-up router at the remote site and the connection is established. At the completion of the call, the connection is terminated. The user only pays for the time that the call takes place, plus a relatively small monthly fee (similar to telephone service).

Inverse multiplexers have the additional capability of aggregating multiple independent switched circuits to create a single higher-rate channel. An inverse multiplexer segments the data in the outgoing data stream and sends the streams out over the individual channels. At the receiving end, the inverse multiplexer accepts the data from these channels, reorders the segments, and compensates for variances in channel transit times. Inverse multiplexers can also add or remove channels from the aggregated connection without terminating the connection. This allows the total amount of bandwidth between the two sites to vary according to real-time bandwidth requirements—for economies of operation. This feature is sometimes referred to as dynamic bandwidth allocation. One of the penalties of using this approach is delay associated with establishing phone circuits (5-10 s for digital circuits and up to 30 s for analog circuits).

Interoperability is another important issue. Early inverse multiplexers implemented proprietary protocols to combine digital channels to form a transparent aggregate stream of data. Units had to be bought in pairs from the same vendor in order to achieve connectivity.

In September, 1991, the Bandwidth on Demand Interoperability Group (BONDING) was formed. Version 1.0 of the BONDING standard was published in September of 1992, and the first conformance event was held in April, 1993. The specification defines a frame structure and procedures for establishing an aggregate channel by combining multiple switched channels. It is now possible to implement networks using inverse multiplexers from several different vendors (there are 31 equipment manufacturers represented in the BONDING group).

The Integrated Services Digital Network (ISDN) is still unavailable in many areas, and just beginning to be supported by several of the BONDING manufacturers. An alternative technology, which is more widely available and supported, is the 56-kbps switched type (or "Switched-56") provided in most cities in the U.S. by commercial phone carriers. Since such circuits are entirely digital, they have low bit error rates and provide an economical, reasonably sized increment of bandwidth. Bandwidth-on-demand devices also work with analog modems. These modems can run wherever analog (Plain Old Telephone Service—POTS) phone service is available (i.e. almost anywhere in the world). There are several disadvantages: 1) circuit quality can vary widely, from virtually error-free to unacceptably noisy in which much bandwidth is wasted on error-correction, 2) the analog lines are only guaranteed for transmitting and receiving 4800 bps by the local service provider, and 3) calls take much longer to establish because of low-speed protocol negotiation and carrier detection. While compression standards such as V.42 *bis* create a virtual maximum throughput of 56 kbps, this maximum is rarely, if ever attained, and in practice throughput varies widely depending on the compressibility of the data.

While installation and monthly line costs are substantially cheaper on an analog phone line versus a Switched-56 digital line, the serious disadvantages discussed above make the analog option impractical except for (1) maintaining a single analog backup line should the digital system fail, and (2) in the event of a power outage, the analog system can be used during the period of time that access equipment is powered by

Uninterruptable Power Supplies (UPS). (The digital lines are powered by the customer, while the analog lines are powered by the service provider.)

3 GROUND ARCHITECTURES

3.1 Remote Antenna Complexes to Pasadena, CA

DSN ground communications from the antenna complexes to Pasadena are currently over dedicated satellite circuits that are exceptionally clean (error-free 99.5% of the day), secure, and dependable. The overseas links are very expensive because of the distance and generally higher cost of telecommunications in foreign countries.

An example of the nature of customer traffic can be deduced from the aggregated spacecraft downlinks at Goldstone, CA illustrated in Figure 1. These are the data that must be delivered to customers such as spacecraft teams, principal investigators, and non-NASA partners. Some of the traffic is "real-time," and must be delivered as quickly as possible. The real-time traffic from the stations to JPL usually totals less than 200 kbps. This traffic includes spacecraft engineering data, quick-look data, and other critical data. These data tolerate very little additional latency (over and above the expected 270 ms satellite propagation delays). They are not candidates for dial-up bandwidth, nor error correction techniques made possible by TCP/IP. This traffic requires dedicated circuits.

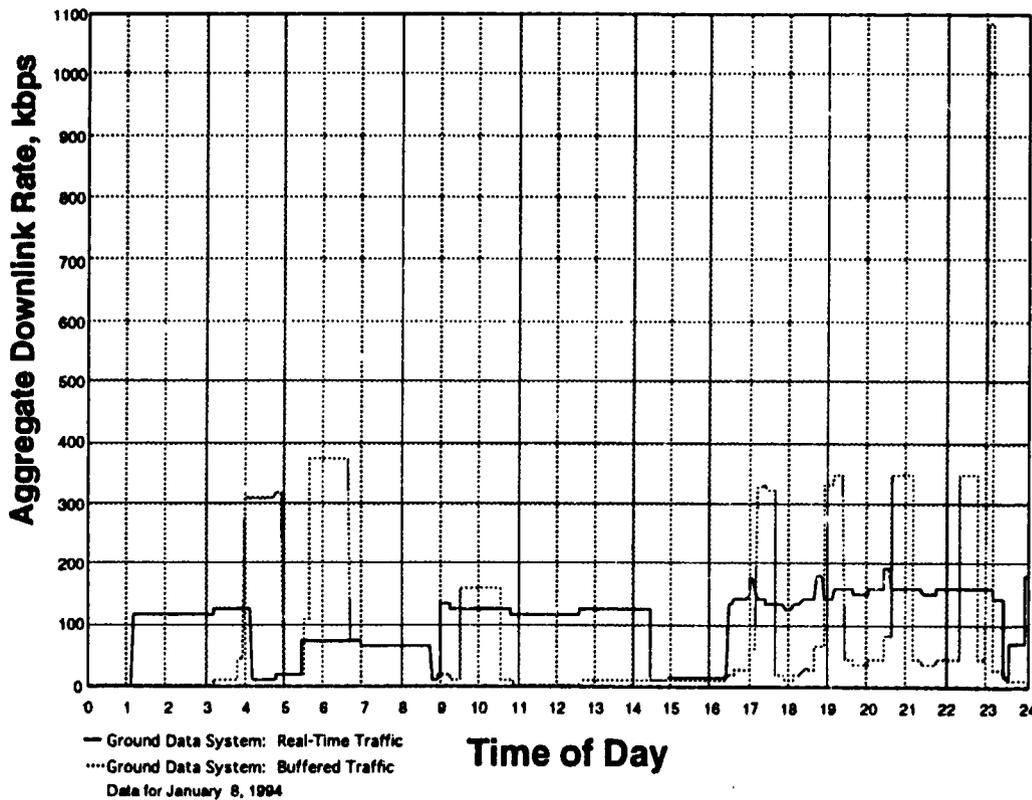


Figure 1 Aggregated Spacecraft Downlinks at Goldstone on January 8, 1994

The remainder of the traffic may be delayed to provide additional communications services such as automatic error correction or to balance the load on the ground circuits to Pasadena. The telemetry delivery system is capable of prioritizing the data and handling it appropriately.

This data may be buffered at the station or at Pasadena before being passed on to the customer. As shown in Figure 1, it is bursty (the result of brief spacecraft visibility for Earth orbiters). Switched circuits are ideal for delivering these bursty streams because: (1) the streams occur for brief periods of time, (2) there is no critical latency requirement, and (3) the streams are delivered with TCP/IP protocols, which provide appropriate flow control mechanisms and are compatible with inverse multiplexers.

Leased circuits make sense when the circuit is utilized most of the time. When considering the option of using leased versus dial-up lines, the monthly and per-hour cost of dial-up lines and the monthly cost of the leased lines can be expressed in an equation which is linear in dial-up hours. This can then be solved for the "break-even" number of hours per month between the two approaches. If the circuit will be used more than this value, it makes more sense to lease.

The costs involved depend on many things: the distance between the endpoints of the communications channel, whether or not this distance spans local service provider areas, the long-haul carrier (if any) used, the discount program used for leased lines (the longer the lease, the better the monthly rate), and whether or not data transmission can be scheduled to take advantage of lower evening and night time toll charges.

The resulting network architecture (Figure 2) has a limited amount of dedicated bandwidth for real-time traffic and optional "elastic" bandwidth for the lower-priority traffic. The traffic flows initially to the router where its priority is determined and the low priority traffic is shunted to the inverse multiplexer. The inverse multiplexer establishes circuits as required. In addition, in the event of losing the dedicated channel, the router may reroute high priority data to the inverse multiplexers to provide emergency communications channels.

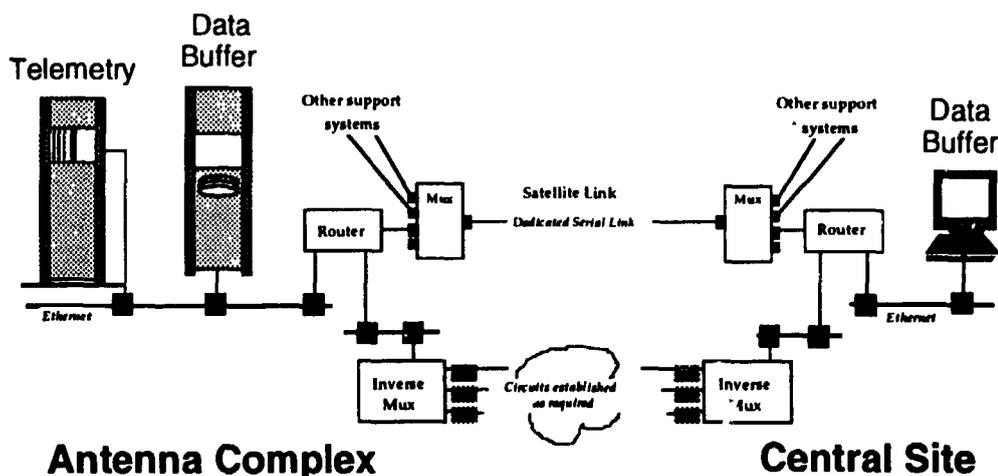


Figure 2 Complex-to-Central Site Network

3.2 Pasadena, CA to Customers

Once the non-real-time data is processed at JPL, it is transmitted to the individual customers. A typical example is illustrated in Table 1, which lists the preliminary plans for supporting the upcoming Cassini mission. Table 1 identifies the locations of the customers for the Cassini down-link data and the expected data rates. None of the traffic is in the real-time category.

Table 1 Candidate Cassini Customer Locations

Customer	Location	Required Bandwidth (kbps)
European Space Operation Center	Darmstadt, Germany	56
Goddard Space Flight Center	Greenbelt, MD	56
Southwest Research Institute	Phoenix, AZ	112
University of Arizona	Tucson, AZ	56
University of Heidelberg	Heidelberg, Germany	56
University of London	London, England	56
Johns Hopkins University	Baltimore, MD	56
University of Iowa	Ames, Iowa	56
University of Colorado	Boulder, Colorado	56

There are several options for data delivery. The first is to use traditional dedicated circuits, which may or not be cost-effective depending on the volume. The second is to transmit the data from JPL to the customer over the Internet since these particular customers are on the Internet. Security safeguards are necessary, such as secure local area networks at the customer sites for hosts that perform spacecraft data processing.

The third option is to use dial-up routers and inverse multiplexers, and establish dial-up circuits as required. Security safeguards available with inverse multiplexers include: (1) encrypted password protection, (2) dial-back features, and (3) data encryption.

3.3 Remote Antenna Sites

In addition to the DSN architecture, dial-up routers and inverse multiplexers may support remotely located antenna sites, assuming that there are common carrier services in the area. In this case there may be both monitor and control and telemetry data. If the station is used as a transmitter, there may also be command uplink data. The volume of data will determine the data rate required for the individual channels.

Assuming that the volume of data is relatively low, a low-cost architecture could involve one leased circuit and one dial-up circuit (Figure 3). We estimated communications costs for such a system between Goldstone and a customer site in Pasadena with 56-kbps circuits. Such a configuration could support volume up to 605 Mbytes per day over the dedicated circuit and cost-effectively support up to an average of 20.8 Mbytes per day (625 Mbytes/mo.) over the switched circuit. Above 605 Mbytes/mo. a second leased circuit would be more advantageous.

The details of the crossover volume of data calculation are as follows: A leased 56k line from Barstow to Pasadena costs \$538 per month. Switched-56 service is \$77 per month plus \$18.60 per hour in toll charges. So the equation gives a value of 24.8 hours

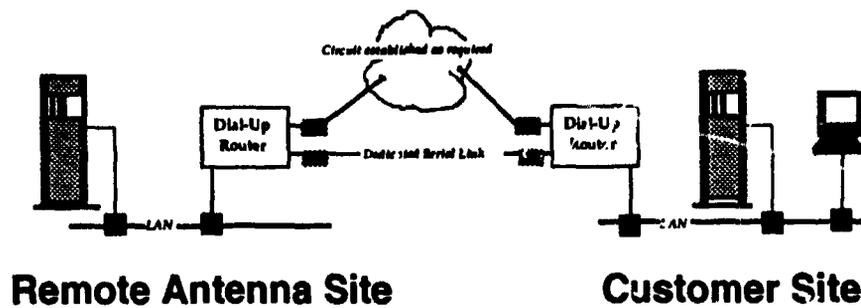


Figure 3 Remote Site-to-Customer Network

of connectivity per month as the crossover point. At a data rate of 56 kbps, this corresponds to an average volume of 20.8 Mbytes of data per day.

4 SUMMARY

This paper proposes a hybrid communications architecture that uses inverse multiplexers and dial-up circuits in addition to traditional leased circuits for spacecraft ground communications. Such an architecture may significantly reduce costs. In some cases it may significantly reduce the delivery time by providing additional bandwidth on demand. With appropriate security safeguards, non-critical data may be sent directly from the antenna complex to the end user. Therefore, the network architecture presented here may be applied not only to the DSN, but to any ground station within the range of a common carrier.

The research described in this paper was carried out by JPL, California Institute of Technology, under a contract with NASA.

**NASA JOHNSON SPACE CENTER
LIFE SCIENCES DATA SYSTEM**

**Hasan Rahman
Martin Marietta Services
Houston, Texas**

**Jeffery Cardenas
NASA/JSC
Houston, Texas**

ABSTRACT

The Life Sciences Project Division (LSPD) at JSC, which manages human life sciences flight experiments for the NASA Life Sciences Division, augmented its Life Sciences Data System (LSDS) in support of the Spacelab Life Sciences-2 (SLS-2) mission, October 1993. The LSDS is a portable ground system supporting Shuttle, Spacelab, and Mir based life sciences experiments.

The LSDS supports acquisition, processing, display, and storage of real-time experiment telemetry in a workstation environment. The system may acquire digital or analog data, storing the data in experiment packet format. Data packets from any acquisition source are archived and meta-parameters are derived through the application of mathematical and logical operators. Parameters may be displayed in text and/or graphical form, or output to analog devices. Experiment data packets may be retransmitted through the network interface and database applications may be developed to support virtually any data packet format. The user interface provides menu- and icon-driven program control and the LSDS system can be integrated with other workstations to perform a variety of

functions. The generic capabilities, adaptability, and ease of use make the LSDS a cost-effective solution to many experiment data processing requirements. The same system is used for experiment systems functional and integration tests, flight crew training sessions and mission simulations. In addition, the system has provided the infrastructure for the development of the JSC Life Sciences Data Archive System scheduled for completion in December 1994.

INTRODUCTION

Understanding the physiologic changes that occur in space requires detailed physiologic measurements before, during, and after spaceflight. This generates large amounts of pre-, in-, and post-flight data in a variety of data formats. Traditionally, different data systems have been implemented to deal with different phases of the investigation, each system responsible for inflight data and another (or perhaps several) systems handling pre- and post-flight data. As a result, there are varied data formats and computer systems, with little or no resulting standardization.

The LSDS provides a portable, self-contained unit capable of data acquisition,

monitoring, analysis, archival, playback, and network transmission. The LSDS can acquire RS449 synchronous serial Non-Return-to-Zero (NRZ) formatted RF-downlink data or Consultative Committee for Space Data Standards (CCSDS) packetized data from a network interface.

The LSDS system was designed to handle, in real-time, data from any phase of spaceflight and store it in a readily accessible and flight compatible format. The system allows for easy access to data files, implementation of mathematical functions, and portability to other support sites. The LSDS system has been tested and used on all Spacelab missions with a life sciences complement, and it has been adopted by university laboratories. In addition, the LSDS system will be used to store and display the continuous data entered into the NASA Life Science Data Archive.

Overall, the LSDS software has proven to be useful, practical, and powerful. Although the development of this software began to address problems specific to space research, the software is being used to solve ground-based command and control data issues.

DESIGN

Design Considerations

The original design specifications for the LSDS were created to support the NASA Life Sciences experiments and investigations. During the early design phase, it became clear that by selecting the proper hardware and using a modular, object-oriented approach to software development, the LSDS could become a flexible, powerful system capable of operating in a wide range of data acquisition environments. Some of the design goals implemented during the development of the LSDS are as follows:

- Ease of use
- Ease of maintenance and modification
- Real-time, multifunction capabilities
- Acquisition of data from several sources

- Support to multiple experiment data streams
- Generic data displays to handle a wide variety of data types
- Utilization of existing software and off-the-shelf (COTS) hardware where possible
- Real- and playback data analysis capabilities
- Distributed architecture
- Client/Server capability

Input Data Format

The original design specifications for the LSDS required that it support the acquisition and processing of a synchronous serial NRZ-formatted data stream generated by an experiment payload microcomputer at a minimum bandwidth of 256 kilobits per second. The data stream is formatted into NASA Standard 630 High-Rate Multiplexer (HRM) and NASCOM frames. In this format the data bits are formatted into 12 or 16 bit words, and the words are grouped into minor frames. A minimum of four minor frames are grouped into major frames. Each minor frame begins with a standard 6-byte header. The first 32 bits of the header are a 24 bit sync word and 8 bit minor frame number used for frame synchronization. Data parameters are stored in the remainder of the minor frames. Data parameters may or may not be major-frame repetitive; ones that are not repetitive are indicated by bits set to indicate the presence or absence of particular parameters in the major frame. Upon acquisition by a ground system, major frames are grouped into packets of one or more major frame per packet.

The LSDS system can be configured to acquire packetized digital data in other formats.

Up to sixteen analog channels from a single workstation can be input into the LSDS system. Samples of the signals are digitized and stored in digital data packets.

Hardware

The LSDS was initially developed on a VAX/VMS platform. The Server of the LSDS software will run on VAX or AXP systems. The Client application software is available in multiple platform architecture (e.g. Macintosh, MS-DOS and VAXStation).

Digital telemetry is supported with Commercial Off-The-Self (COTS) hardware manufactured by Simpact Associates. An ICP3222 and /or FreeWay I/O box is used to receive data directly from a telemetry interface at the JSC Mission Control Center. An ICP3222 provides up to 4 input channels with an aggregate rate of 1 MB/Sec.

The FreeWay I/O box is used to support data acquisition and transmission over Ethernet and FDDI using TCP/IP protocol. This methodology provides a complete open system distributed architecture to provide services within the Life Sciences user community. An external magneto-optical Disk is used for data storage and archival. A Graphtec 32-channel analog strip chart recorder can be used to provide hardcopy strip charts. Any printer connected on a Local Area Network (LAN) can be used for printed output.

See Figure 1 for a representation of the architecture of the LSDS.

Work Stations

The system environment provides a consistent user interface regardless of the underlying operating system or hardware platform. The LSDS II VAX workstation display subsystem is based on the X-Window System™ and the Motif Graphical User Interface (GUI). This system provides portability and interoperability of displays across various platforms (e.g., Macintosh, MS-DOS PC, and any X-Window System Terminal/Server).

The workstation environment supports a variety of functions, from initial loading of experiment requirements into a database.

acquisition and display of test and simulation data, timeline and activity plan development, real-time flight operations monitoring and control, to preflight and postflight biomedical baseline data collection. Applications software resident in the workstations is developed in-house from COTS applications. The system allows the user to define and build custom displays of processed experiment data.

The Macintosh-based workstation Microsystems Integrated Real-time Acquisition Ground Equipment (MIRAGE) was designed and developed to provide a portable, self-contained desktop unit capable of real-time acquisition, processing, monitoring, analysis and storage of experiment data. User Interface and Data Visualization and Representation tools were designed and developed on the MIRAGE workstation.

Software

The LSDS Server software was developed on VAX/VMS, AXP OpenVMS, and on the Macintosh platform using Apple's Macintosh Programmers' Workshop C. A modular, object-oriented approach to software design was used to assure ease of modifiability and maintainability. The LSDS Client software uses the X-Window (Motif) DataView™ toolbox, establishing the capability to transport the LSDS Client software directly to other systems such as UNIX, Macintosh, and DOS-based Personal Computers.

The LSDS Server software makes use of the VMS system services provided by the operating system to support real-time functionality. The LSDS Server software uses the VI Corporation's DataView™ library of function calls to control the display system. TCP/IP and DECnet network protocol is used to provide interface to the Ethernet and/or FDDI controller hardware.

The software uses a new feature of the OpenVMS and Macintosh system 7.0 operating system. In Macintosh environment

the application uses the Time Manager to create a multitasking environment within an interrupt-driven operating system. By using Time Manager, the user can archive, acquire and display data simultaneously. In addition, LSDS has built-in utilities that allow the user to manipulate archived data. With the "extract and convert" command the user can convert a section of data (or the entire archive) to a different data format (e.g. Motorola or Intel). With the "frame editor" command, the contents of a major frame or block of frames can be modified. The "packet lim scan" utility will scan the data for a particular event that occurred during data collection. The "gapper" utility generates a report about the gaps that occurred in the data.

See Figure 2 for an illustration of the LSDS software architecture.

Configuration Database

The LSDS configuration database is created using Microsoft Excel. The database consists of several tab-delimited text spreadsheets, arranged into folders on the disk the LSDS software resides on. The LSDS database is used to define the system defaults (fonts, colors, etc.), experiment-specific hardware configuration, data stream format (stream data rate, packet frequency, etc.), acquisition defaults, experiment parameter format (packet location, extract masking information, etc.), display format, and analog input and output characteristics.

Due to the design of the databases supporting the software, the type of data, format of the data, and displays for the data can be modified quickly and easily. By using the DataView Draw and Macintosh ResEdit 2.1 software, a display can be modified in a matter of minutes and new parameters can be added to the display by modifying a database.

Data Flow

Figure 3 gives a representation of the data flow through the LSDS system. There are three external sources: the Macintosh user

interface, the experiment database, and the experiment data source. The user enters experiment stream, parameter, and display data into the experiment database. Through the Macintosh interface to the LSDS Server application, the user controls at runtime the experiment data source and other application functions. The user can choose multiple acquisition functions. Data can also be played back from an archived disk file.

The data acquisition portion of the program reads data from the specified external data source and, based upon the contents of the stream database, extracts and stores the major frames in the primary buffers. If the user has the archival function turned on, the primary buffers are read by the archival process, a header is generated, and the data packet is written to the archive file.

The parameter extraction function then extracts the data values for each parameter specified in the display databases from the primary buffers. The parameter database is used to locate and process the data values. The extracted and processed data values are stored in the parameter buffers.

If analog output is enabled, the data values for the parameters to be output are extracted from the parameter buffers and stored in the analog output buffer. The buffer is then passed to the analog output process.

The LSDS Client software for each display and graph window defined in the display databases, the data values to be displayed are extracted from the parameter buffers and output in the specified window in textual or graphical form based upon input from the display databases.

IMPACT OF SOFTWARE

Value and Utility

The LSDS is used at NASA/JSC Life Sciences Project Division Science Monitoring Area (SMA) for a variety of applications. Real-time data from Spacelab missions are acquired, displayed and stored by this software. During acquisition and

playback, LSDS can display, archive, and transmit analog data to a strip chart recorder, or other display device. This allows investigators to view and get a hard copy of their data.

In addition, the Life Sciences Data Archive (LSDA) Project is underway at JSC. The archive will allow for the storage, preservation, and access to the biomedical information obtained during life science investigations. Data entering the LSDA will be stored, retrieved, and analyzed using LSDS.

At the Ames Research Center (ARC), the Space Life Sciences Project Office will be using LSDS Server system. A recent contract from Ames to Martin Marietta specifically called for four LSDS Server capable systems. At the Medical College of Virginia, Dr. Dwain Eckberg uses the system to help with the analysis of his studies on the carotid-cardiac baroreflex. At the University of Texas-Southwestern Medical Center, LSDS Server is also used for analysis of pre- and post flight biomedical data.

Recent work with the Russian Space Agency will also use LSDS Server configuration. The Macintosh systems in Houston and Moscow are being used to quality test the Standard Interface Rack Controller in support of the Shuttle-Mir Science Program joint missions. The LSDS Server is also being used to test the Mir Interface to Payload Systems (MIPS) Controller flight software. Four LSDS Server/Client systems will be established in Russia to support Mir-18.

Enhanced Productivity

Prior to LSDS Client/Server, biomedical data from spaceflight was collected on dedicated VAX-based systems. These systems were not portable and were only used during simulations or actual spaceflight. By using the portable LSDS Client/Server systems, data is collected during a mission or a simulation, and the same system is utilized to support baseline

data collection, science verification tests, data analysis, and data archival, both pre- and post-flight.

Data from life sciences missions and ground studies come in a variety of formats (analog, digital, High Rate Multiplex, CCSDS). Before the implementation of the LSDS Client/Server architecture, a separate system dealt with each kind of data. This required different software, with the attendant documentation, training, etc. Currently, only LSDS Client/Server need be used to deal with all the data and to provide a common format for retrieval. The ability to deal with multiple data types provides a savings to the LSDA, now under development, to handle the data from life sciences flights experiments. Rather than have multiple systems, LSDS will be used to view all continuous data from the flights, regardless of its initial format.

For future flights the time required to configure the data acquisition system and displays for new experiments and data has been drastically reduced. The LSDS Client/Server system uses spreadsheets to set the data acquisition parameters for a new experiment. In the spreadsheets the user can set LSDS system defaults (fonts, font sizes, display colors, etc.), experiment-specific hardware configuration, data stream format (stream data rate, packet frequency, etc.), acquisition defaults, experiment parameter format (packet location, extract masking information, etc.) display format, and analog input and output characteristics. Once this is complete, the spreadsheets are saved in a folder and can be retrieved upon demand.

Improved Efficiency, Reliability, Quality

The LSDS system provides real-time data quality checks and creates log files automatically indicating when LOS (loss of signal) periods occurred. In addition, data drop outs and the times when particular displays were activated, are also logged. Previously, the user had to wait until the disk was full (1-2 hours) before a data products analysis could be provided.

Formal Recognition

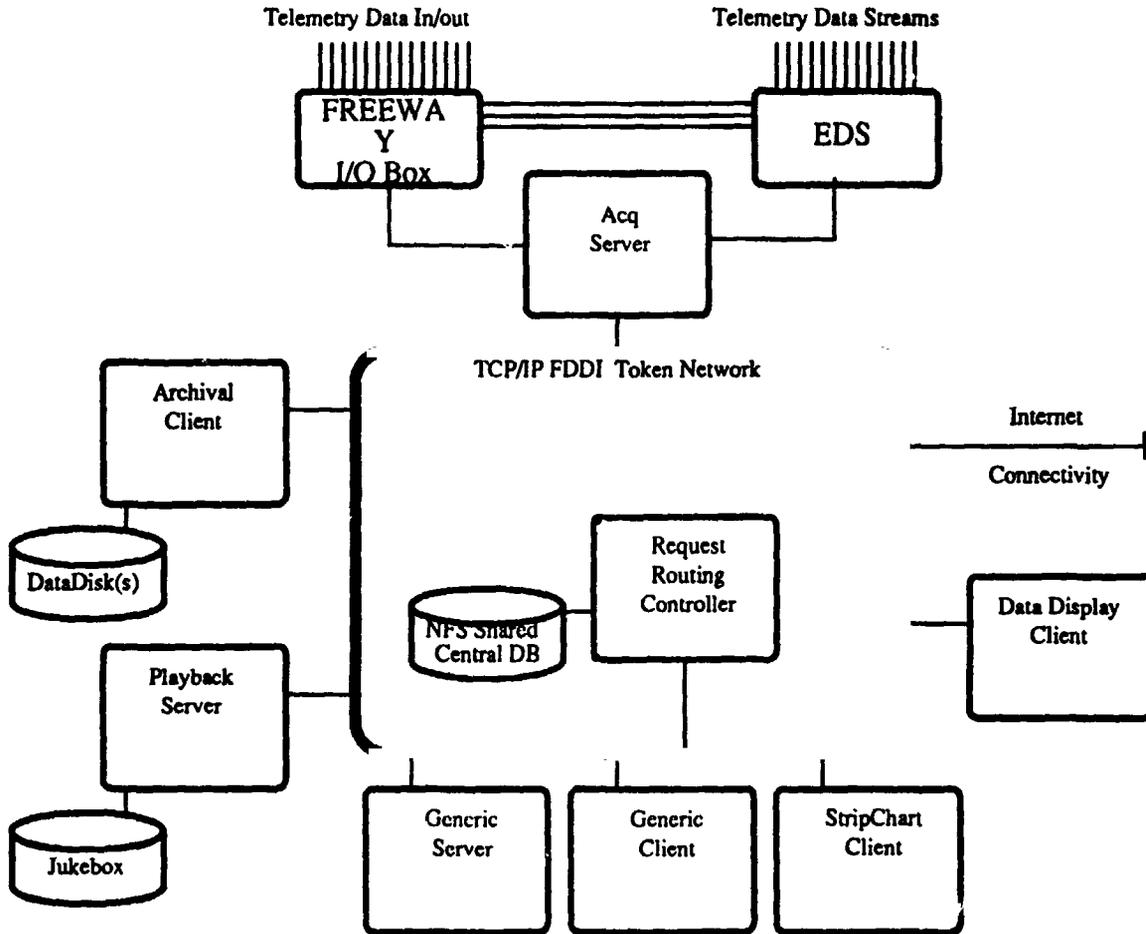
The LSDS Macintosh based Client/Server system has been presented at Technology 2002, the Third National Technology Transfer Conference and Exposition. The Macintosh based development team also won the NASA Public Service Group Achievement Award in March 1991, "... in recognition of their outstanding contribution to the design, integration, test and fabrication of the technologically advanced portable MIRAGE system."

Results/Conclusions

The LSDS II design provides data processing functions according to specifications set by users, in this particular case experiment investigators. The processed data will be distributed to locally- and remotely-based users and also stored on magnetic media for the future use of the investigators. The data processing functions use state-of-the-art workstation technologies, and analysis tools which may be developed by investigators and/or operations personnel at LSPD operations facilities or at remote locations.

Applications of the LSDS II and lessons learned by the JSC Project have already been made to the International Space Station Alpha Program, the Joint NASA RSA Shuttle MIR Science Program, and the NASA Life Sciences Flight Experiment Program. Utilization in the commercial and private sector has been coordinated through the NASA Technology Utilization Office.

LSDS SYSTEM OVERVIEW



Overall control starts in the middle of the diagram, with reliant tasks/resources branching off from their respective controllers. Connections are not necessarily representative of the physical layout.

Figure 1 Open Life Sciences Data System

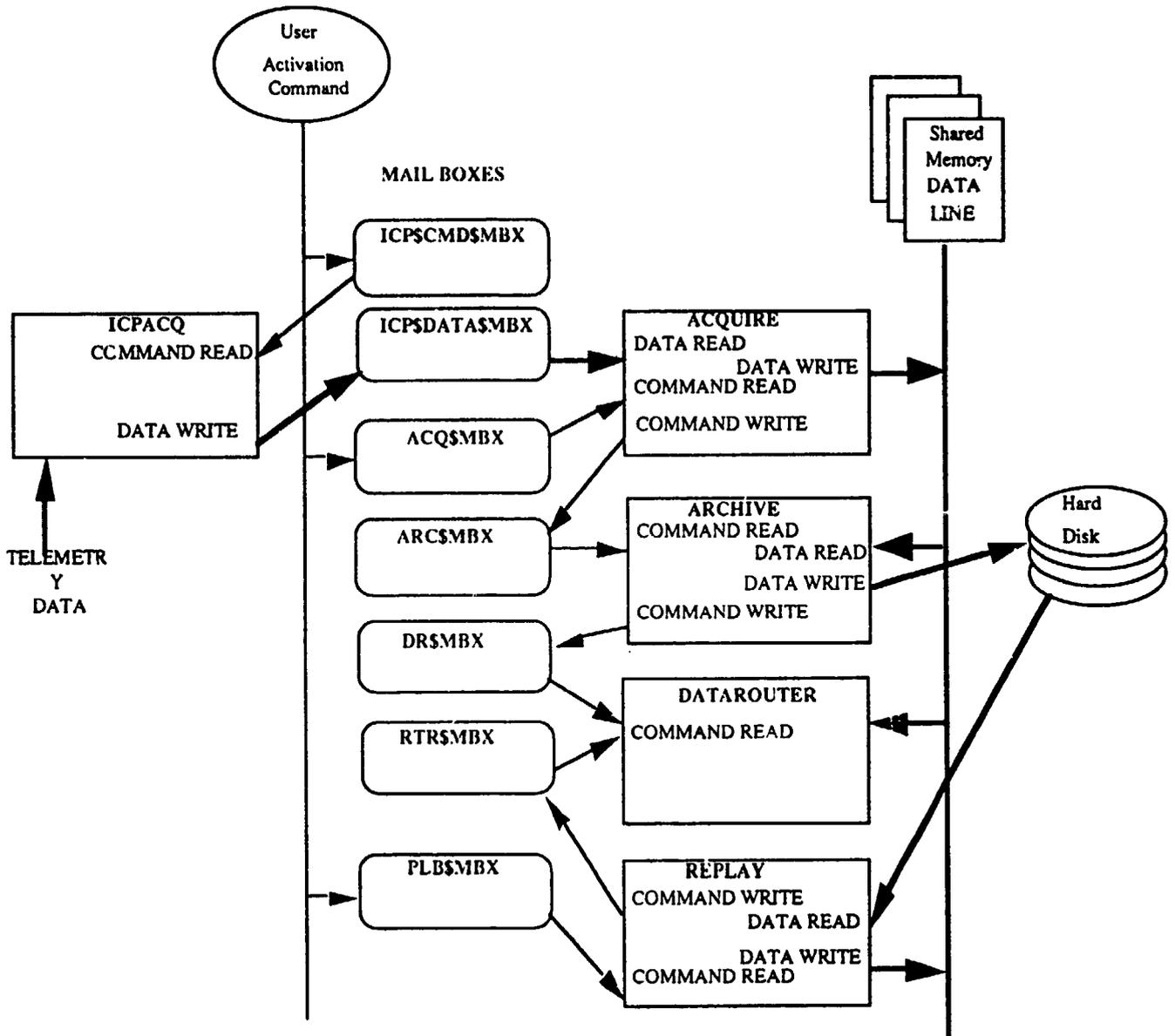


Figure 2 LSDS Software Architecture

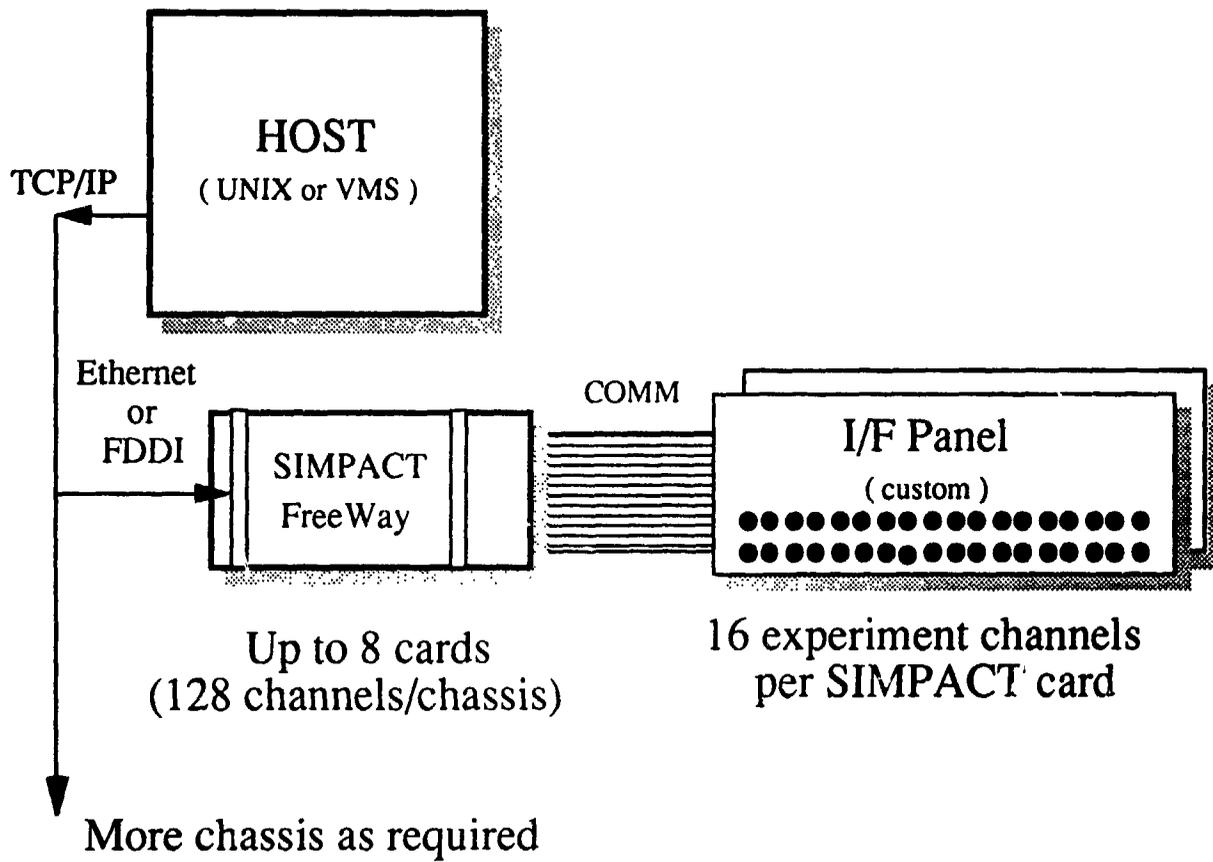


Figure 3 LSDS Data Flow

A PROCESSING CENTRE for the CNES CE-GPS experimentation

Norbert SUARD - Jean-Claude DURAND

C.N.E.S. - CT/TI/PS/AP & SDS/SM/ES
18 av. E. Belin
31055 TOULOUSE Cedex - FRANCE

1. INTRODUCTION

ABSTRACT

CNES is involved in a GPS (Global Positioning System) geostationary overlay experimentation. The purpose of this experimentation is to test various new techniques in order to select the optimal station synchronization method, as well as the geostationary spacecraft orbitography method. These new techniques are needed to develop the Ranging GPS Integrity Channel services.

The CNES experimentation includes three transmitting/receiving ground stations (manufactured by IN-SNEC), one INMARSAT II C/L band transponder and a processing centre named STE (Station de Traitements de l'Expérimentation).

Not all the techniques to be tested are implemented, but the experimental system has to include several functions; part of the future system simulation functions, such as a servo-loop function, and in particular a data collection function providing for rapid monitoring of system operation, analysis of existing ground station processes, and several weeks of data coverage for other scientific studies.

This paper discusses system architecture and some criteria used to its design, as well as the monitoring function, the approach used to develop a low-cost and short-life processing centre in collaboration with a CNES sub-contractor (ATT-DATAID), and some results.

Keywords : Ground System, Architecture, Software.

The GPS system offers exceptional qualities (accuracy and worldwide coverage). But for civil aviation (see (1)), this system has three major drawbacks :

- insufficient integrity,
- limited availability,
- voluntary spatio-temporal degradation.

Ranging GPS Integrity Channel services (RGIC) should enable GPS to be used by civil aviation.

The experimentation prepared by CNES (see (2)) is dedicated to the technical validation of the Ranging GPS Integrity Channel concept that always needs :

- station synchronization better than 10 ns,
- GPS-type signals transmitting,
- geostationary spacecraft orbitography better than 10 meters.

The CE-GPS (European complement to GPS) experimentation includes a master ground station transmitting a GPS-type signal to an INMARSAT 2 geostationary satellite. The repeater broadcasts this signal in L-band to the master station and to the other stations. These also have receiving and transmitting facilities for GPS-type signals.

Each ground station includes a computer and software to :

- a) record broadcasted and raw data from several facilities,
- b) process some of the data in a real time loop (0.6 seconds) to generate transmitting signals correctly,
- c) control and monitor equipment,
- d) make some of the data available to the processing centre.

Other data (such as orbital and some weather

parameters) required to drive the system or for various scientific studies are centralised at the processing centre.

The functions of the processing centre (STE) are to :

- prepare data for ground operation control station schedules,
- collect data from ground stations and other sources,
- archive and distribute these data to different scientific teams, sometimes after specific processing,
- monitor ground station operations.

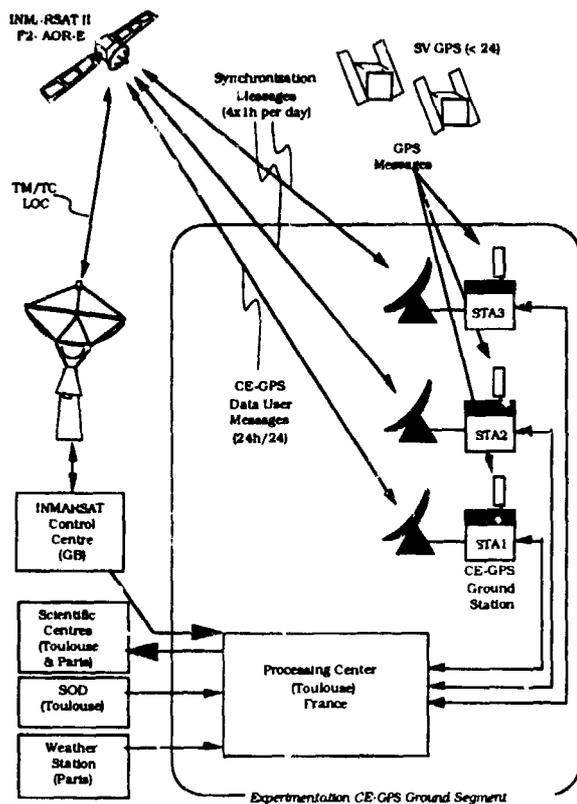


figure 1 : Experimentation CE-GPS overview

2. ARCHITECTURE

In this chapter, the architecture of the ground station and of the STE are presented. After, an overview of the exchange system is given, then criteria used to distribute software between ground station, processing centre and scientific centres are listed.

2.1 Ground stations architecture

Each ground station is composed with multiple equipments connected to a main computer. All software ground station functions are centralized at this computer

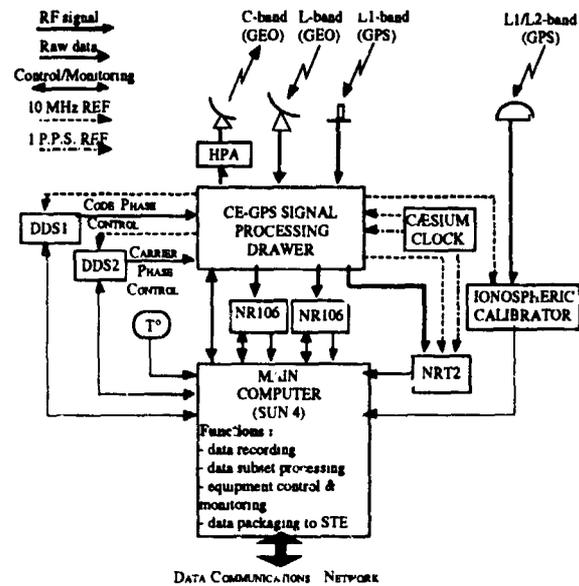


figure 2 : Ground station overview

2.2. processing centre architecture

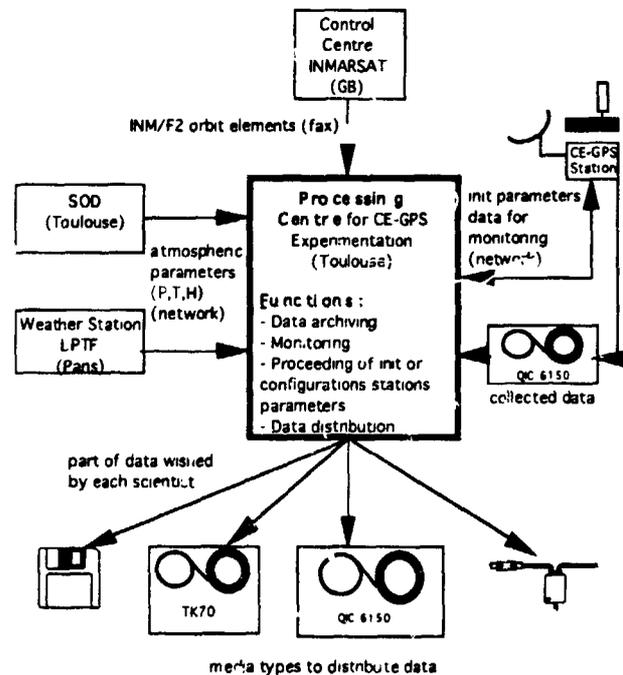


figure 3 : external links

The processing centre software architecture is decomposed in two blocks of functions (figures 4 & 5 below). These two blocks can be used together.

The diagram below shows functions that are only used when data are collected.

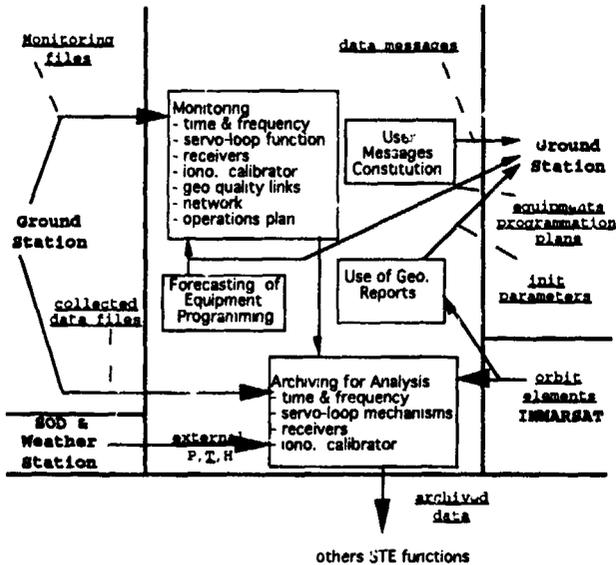


figure 4 : software architecture (1)

The diagram below shows functions that are only used when data are pre-processed and sent to a scientific centre.

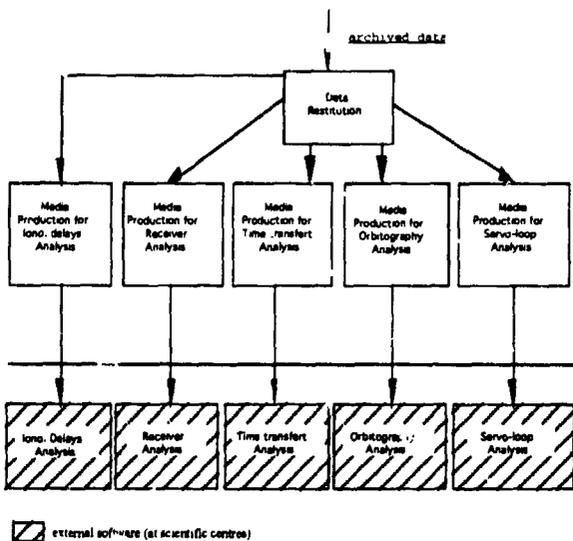


figure 5 : software architecture (2)

These different functions are activated and controlled through a man-machine interface displaying four main windows :

- 1) Function selection, after which a sub-function menu is displayed for input of various parameters.
- 2) Display of the number and type of tasks currently running.
- 3) Output of task messages.
- 4) System message output (console).

Other windows are available with OPENLOOK SUN system tools (file manager, calctool, cmdtool, etc.) and may be selected by the operator when required.

The diagram below shows the hardware architecture used to support the different functions.

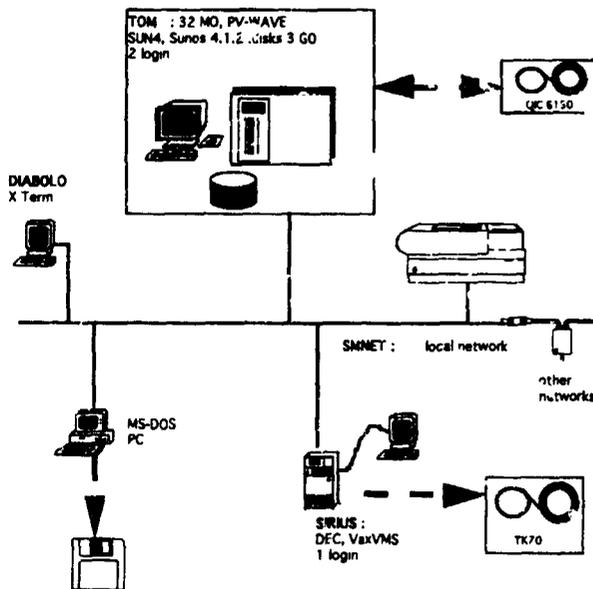


figure 6 : hardware architecture

PC/MS-DOS and DEC/VMS are only used to product adequate media for specific scientific centres.

2.3. exchange system

The system may use different configurations, with one, two or three ground stations. For example, this system was operating at the end of 1993 with two ground stations (Toulouse and Paris) but is operating with three in 1994

(Toulouse, Kourou in French Guyana and Hartebeesock in South Africa).

As ground station locations might vary according to use with a different network at each configuration, the interfaces between ground stations and the processing centre had to be **standardised and easily modified**, so data-files are used for exchange between the processing centre and a ground station.

Thus, when a network is available, FTP (File Transfer Protocol) is used to exchange subsets of data-files, QIC 6150 data-cartridges are used to transfer the remaining data-files.

Else (when no network is available : for example transfer from one development site to another) all data can be saved and easily transported by using only cartridges.

This rule was also applied for the other exchanges between the processing centre and the scientific centres. Computers are not homogeneous between the processing centre and the scientific centres, so data in files are in ASCII format.

Each file transfer is initiated by the STE processing centre.

2.4. Criteria for the distribution of software between ground stations and processing centre

In the operational phase, if one of the components were to fail (processing centre, ground station or network), the system would have to ensure that any data generated by the other components was not lost, but **without relying on any redundant facilities**.

As ground stations include a real-time loop to generate GPS-type signals, it was decided that the main processing system in each ground station should receive each function with **real-time constraint**, and thus should collect all data from all facilities through RS232 or IEEE links, even though the processing centre would be able to obtain data through a different link.

This provided an **uniform means of data exchange** between any ground station and the processing centre.

So, with the data-file exchange system, each ground station computer would then manage short-term file-saving (over a few days), while the processing centre would manage long-term

archiving for all ground stations and all system configurations used.

One of the aims was to cut down manual operations in ground stations so that they would not need to be staffed on a permanent basis.

All operations where data has to be **keyed in manually** are carried out in the processing centre, and the results file is then transmitted to the ground station (before the operational phase, data may be input with a text editor as the data in these files is in ASCII format).

NB : The only manual operation needed under normal station operating conditions is a twice-weekly cartridge change.

Another point we observed was to **avoid allocating to ground stations any processing occurring at irregular intervals** as site and azimuth angles processing, so that real-time loops would not be affected by a random load peak.

Any such processing is carried out at the processing centre and gives results files that are valid over the whole operational period and transmitted.

The ground station software only uses indirect time and date addressing to retrieve data when needed.

2.5. Criteria for the distribution of software between processing centre and scientific centres

The first criterion was to **avoid imposing specific types of equipment on scientists configuration**. Hardware for data exchange was defined for each scientific centre for its own data, STE processing centre which would be responsible for setting up a hardware and software configuration based on existing facilities at CNES.

The second criterion was to develop and operate at the processing centre any data **pre-processing software which would be common to at least two scientists**.

The third criterion was to **keep options open for specific software to be set up within the processing centre** to enable the operator to pre-process also scientific data, as the processing of raw data to obtain interpretable results can otherwise be very time-consuming for the scientists using them.

3. THE MONITORING FUNCTION

Station monitoring from the processing centre is not carried out in real time, for several reasons :

- 1) equipment is more and more reliable ;
- 2) operator at processing centre is only present 5 days per week, 8 hours per day even if each ground station is operational 7 days/7, 24 hours/24;
- 3) the loss of a few data-days is not a problem, but when the data collection function is operating, we have to be certain that the data is correct.

To meet this requirement, ground station software stores three types of data in monitoring files.

The first type is made up of raw data extracts, the second of extracts of equipment command data received and distributed by the servo-loop mechanism, while the third type consists of monitoring indicators generated on ground stations (watchdog function for the various flows of expected data, quality indicators for INMARSAT 2 satellite links as bit error rates, etc.).

These monitoring files are processed by dedicated software at the processing centre using simplified equations to describe observable phenomena. The operator can then display the resulting parameters in graph form. The curves change colour if values exceed monitoring thresholds, which take into account the simplifications in the equations.

The observable phenomena are :

- master or slave servo-loop,
- pseudorange and carrier phase measurements for pseudo random code (1 to 32 : GPS constellation, 33 to 36 : back up for GPS, used in the CE-GPS experimentation,
- pseudorange residues,
- vertical Total Electron Content,
- two-way time transfer through INMARSAT 2,
- INMARSAT2/ground station link indicators.

4. APPROACH USED IN STE DEVELOPMENT

The processing centre software was to be written

by a specialised firm, and to have it ready on schedule (development began in November 1992 for partial implementation at the end of March 1993), without the constraints involved in managing too large a team, the following considerations were applied.

1) As the processing centre would be operating for no more than 2 years, the **normal rules of management were made more flexible**. The alterations mainly concerned reviews at the end of each development phase and the documents to be managed within the configuration. For this processing centre, key-points with only the CNES and ATT-DATAID technical managers present were substituted for all reviews. This rule remained valid as long as no major differences arose between ATT-DATAID and CNES.

2) A study was carried out before the contract was signed, to assess the possibility of **including existing products** to meet requirements for all or some of the functions needed. Such products would be incorporated by adapting the new software packages to the interfaces. The functions delivered to ATT-DATAID thus included graph-plotting (developed in PV-WAVE command language), orbit computation functions for the geostationary and GPS satellites and computing routines for tropospheric delay factors.

3) Whenever existing **low-cost hardware could be used to resolve a particular problem**, this was acquired in preference to the development of specific software. For example, an additional 2 gigabit disk for file management was acquired instead of developing a file management system with existing compacting and decompacting tools, as the purchase price of the disk was equivalent to only a few days of software development.

4) File name specifications were set out from the start of the experimentation system definition phase, as well as the choice of the operating system for the main computers (UNIX) and a recommendation on the content of all files (ASCII). This enabled processing to be carried out with **tools which were incorporated within the operating system** and which were therefore easy to manipulate with "shell-scripts". The same reasoning was applied to the development of software for the ground stations.

5) A large number of parameters was incorporated into the processing centre software, either within configuration files to be handled by the text editor or as data to be keyed in through the man-machine interface. This last solution does not affect costs as the centre is permanently staffed (except at weekends) during system use.

6) In order to maintain autonomy between functions and to avoid over-automation of the processing centre, some data input is carried out by the operator even where such data can be deduced from available data in the processing centre.

7) The software for the processing centre was delivered in several stages :

- Stage 1 : man-machine interface ;
- Stage 2 : all data collection functions (see figure 4) ;
- Stage 3 : all data distribution with scientific data pre-processing ;
- Stage 4 : incorporation of specific processes when requested by a scientist.

This method enabled real progress in processing definition to be monitored without the need to program everything in advance.

Tasks were therefore not scheduled in the usual sequence for this type of development (definitions - specifications - realization).

This is not always advantageous (project management is more demanding), but the final product is better matched to the real needs of different users.

5. SOME RESULTS

5.1. about STE software

ATT-DATAID supplied 230 working days to write the processing centre software, starting on the 2nd November 1992 and ending with Stage 3 acceptance tests which were carried out on the 21st of April 1993.

For the STE project, the CNES work-load over the same period amounted to 70 days.

The software for this processing centre comprises 17 000 lines of source code without annotations (in FORTRAN, C, awk, shell), of which 4 000 were supplied by CNES. Certain functions were also directly supplied by CNES as binary codes.

Anomaly report number was :

- 16 after acceptance testing ;
- 29 after technical approval from the processing centre ;
- 34 at the beginning of the CE-GPS experimentation ground segment operational use with 2 ground stations (Toulouse and Paris) ;
- 47 on close of this operation.

Other scientists had access to the data collected, although they were not identified at the beginning of the project.

They required no specific processing. To enable the system to produce and distribute their data, declarative instructions were input into the parameter files then tested (1 day work load).

5.2. about STE/ground stations operations

The fact that STE is not staffed at weekends is just acceptable to detect a problem at a ground station, because the delay between the origin of a problem and its repair can be several days.

So, for a non experimental system, redundant facilities in ground station seem needful, operator should choose correct data. According the interrupt time acceptable by a mission of data collection, other solutions are possible (call by phone the operator after a detection of a problem, processing centre staffed at week-end, ...)

ACKNOWLEDGMENTS

The authors would like to express their thanks to the members of the CE-GPS team, to C. LAMANT, J. FOLIARD (CNES), H. JOUCLA and the ATT-DATAID team, and to F. SAFFRE (IN-SNEC) for their help during the development of the software.

BIBLIOGRAPHY

(1) Proposal for an interface between a GPS Integrity Channel with Ranging capability (RGIC) and GPS receivers
EUROCAE WG 28 Sub-Group 2, 13th May 1992

(2) *Définition détaillée de l'expérimentation CE-GPS* (Detailed definition of the CE-GPS experimentation)
J. Barbier, P. Brun, M. Brunet, J.L. Issler, N. Suard, T. Trémas, C. Valorge, C. Yven (CNES)
Ref note CNES/190/SDS/SM/ES - 15th april 1992

Single Stage Rocket Technology's Real Time Data System

By

Steven D. Voglewede
(407) 867-4310
(407) 867-2828 FAX
McDonnell Douglas Space Systems
Dept. F164
PO Box 21233
Kennedy Space Center, Florida 32815
Internet: sdv@grumpy.ksc.nasa.gov

ABSTRACT

The Single Stage Rocket Technology (SSRT) Delta Clipper Experimental (DC-X) Program is a United States Air Force Ballistic Missile Defense Organization (BMDO) rapid prototyping initiative that is currently demonstrating technology readiness for reusable suborbital rockets. The McDonnell Douglas DC-X rocket performed technology demonstrations at the U.S Army White Sands Missile Range in New Mexico from April-October in 1993.

The DC-X Flight Operations Control Center (FOCC) contains the ground control system that is used to monitor and control the DC-X vehicle and its' Ground Support Systems (GSS). The FOCC is operated by a flight crew of 3 operators. Two operators manage the DC-X Flight Systems and one operator is the Ground Systems Manager.

A group from McDonnell Douglas Aerospace at KSC developed the DC-X ground control system for the FOCC. This system is known as the Real Time Data System (RTDS).

The RTDS is a distributed real time control and monitoring system that utilizes the latest available COTS computer technology. The RTDS contains front end interfaces for the DC-X RF uplink/downlink and fiber optic interfaces to the GSS equipment. The FOCC operators run applications on the RTDS Unix workstations. Twenty one customized SSRT applications were developed for the FOCC RTDS. The application design was based on the programs "aircraft like" operability requirements.

The paper will contain descriptions of the RTDS architecture and FOCC layout. Detailed information on the 21 DC-X applications will be included. A section will include the DC-X ground operation philosophies and rapid prototyping techniques. The paper will describe the DC-X ground operations performed in the FOCC.

1. SSRT Introduction

McDonnell Douglas Space Systems Company, of Huntington Beach, California, was awarded the \$58.9 million Single Stage Rocket Technology Program contract to demonstrate single-stage rocket technology (SSRT) on the Delta Clipper Experimental vehicle, or DC-X, in August 1991. The DC-X is a Ballistic Missile Defense Organization (formerly the Strategic Defense Initiative Organization) rapid prototyping initiative that is demonstrating the technology readiness of SSRT. The DC-X, designed for vertical takeoff and landing, is an operational one-third scale experimental test vehicle of an actual reusable launch system. The reusable vehicle is propelled by liquid oxygen/liquid hydrogen rocket engines. A full-scale Delta Clipper, the DC-Y, will be capable of placing a 20,000-lb payload into low-earth orbit.

2. SSRT Real-Time Data System

The real-time data system (RTDS) was developed by the Kennedy Space Center division of McDonnell Douglas to meet the DC-X requirements for an advanced launch processing system that provided "aircraft-like" capabilities. The RTDS provided the ground monitoring, control, and data archival/reduction capabilities for the DC-X vehicle and its ground support systems (GSS). The RTDS was located at the White Sands Missile Range in the mobile DC-X ground operations base, a 40-foot trailer known as the Flight Operations Control Center (FOCC).

This paper presents a view of the DC-X program, then relates the RTDS to the program, and explains how the DC-X team was able to do its work quickly, cheaply, and successfully.

3. DC-X Program Summary

The DC-X is a rapid prototyping initiative that enabled the vehicle to be designed, built, and flown in two years. A highly motivated team of McDonnell Douglas employees from the company's Huntington Beach, Long Beach, St. Louis, and Kennedy Space Center divisions teamed with subcontractors from across the nation to design, build, and integrate the DC-X vehicle in 18 months. The vehicle was shipped to the NASA White Sands Test Facility in New Mexico in April 1993 for a series of static fire tests. The tests were successfully completed in June and the DC-X vehicle and its entire ground support system were packed and shipped to the flight test site at the U.S. Army White Sands Missile Range.

Two static fire tests were conducted at White Sands to verify system operation after the move from the test facility. These full-system tests were conducted successfully and the vehicle was prepared for flight. The first flight of the DC-X, a 60-second, 150-foot hover test to verify the operation of flight systems, was made August 18, 1993 at the White Sands Delta Clipper site. Two additional test flights, conducted at higher altitudes, with

increased pitch and roll maneuvers, were completed successfully on September 11th & September 30th. These tests demonstrated the following DC-X goals:

- Validate vertical takeoff and landing concepts
- Validate "aircraft-like" supportability and maintainability concepts
- Demonstrate rapid prototyping development approach
- Demonstrate rapid turnaround capabilities

Funding for the program was depleted at the end of October 1993, and the DC-X remained mothballed at White Sands awaiting additional funding. This funding was received in early May of 1994. The flight test program has completed two flights on June 20th and 27th of 1994. Additional DC-X flights which continue to expand the DC-X flight envelope and demonstrate the operability characteristics are planned in 1994.

4. DC-X - New Ways of Doing Things

The primary goal of the SSRT program is to provide inexpensive access to space into the next century to give this nation a low-cost advantage in space transportation. To meet this goal, the DC-X had to do things better, faster, and cheaper.

Rapid prototyping technologies were used extensively to allow the development team to complete the job on schedule. Automatic code-generating software aided in the rapid development of DC-X flight software, allowing the flight software to be changed and validated in hours instead of many days. Use of off-the-shelf technology with open system architecture was maximized through the program. The off-the-shelf products reduced development time while providing many of the necessary capabilities at much lower risk and costs.

The off-the-shelf products used extensively in the development of the RTDS for the FOCC included:

- UNIX system V
- ISO SC16 open systems interconnection protocol
- IEEE 802 network standards (includes Ethernet)
- DARPA TCP/IP networking protocol
- C-ISAM data structure
- ANSI X3.135-1986 SQL database interface
- IEEE 1014 (VME) bus interface
- IEEE 754 floating point number standard

The DC-X also took a new approach to operation of launch vehicles. The entire DC-X system was designed with aircraft-like operability and maintainability concepts. McDonnell Aircraft applied its experience in military aircraft design to develop the avionics systems for the DC-X vehicle, providing easy access to line-replaceable avionics

units from access bays, similar to aircraft. The avionics systems were designed with built-in test features and automated modes that allow for rapid checkout of the vehicle subsystems. Douglas Aircraft applied its experience in developing commercial aircraft supportability and maintainability features to help design these critical elements into the DC-X operating procedures. Douglas also contributed expertise from commercial aircraft cockpit controls and displays technology. Several of the commercial aircraft concepts were designed into the RTDS ground control system human-computer interfaces.

The RTDS was designed with many automated features that allowed the DC-X and GSS to be controlled and monitored with a crew of only three. The system was delivered and installed before the vehicle assembly was completed. This allowed the RTDS to be integrated and validated with vehicle subsystems as they were assembled and attached to the core vehicle structure. The parallel checkout of the RTDS interfaces with the actual hardware during assembly allowed for many real-time modifications and enhancements to the RTDS human-computer interfaces before the vehicle assembly was completed. The effective use of off-the-shelf software development packages allowed the RTDS changes to be made rapidly while integrating the vehicle components. This parallel effort allowed the entire vehicle and ground system to be fully integrated and ready for the static fire tests at White Sands on the same day that the vehicle completed final assembly.

The DC-X and GSS components and systems were all checked out with the same RTDS checkout system, which was also used for the integration component tests during vehicle assembly, avionics subsystem verifications, engine static fire tests, and the entire flight test series.

The reusability capabilities of the DC-X vehicle along with the new operability, maintainability, and supportability concepts have allowed the entire DC-X program to be conducted by approximately 35 persons. Thus, the DC-X has proved that low-cost programs are possible – today and for the future.

5. Real-Time Data System Background

The DC-X required a state-of-the-art automated ground control system that could implement customized real-time user monitoring and control interfaces, provide automated sequences and automatic reactive control functions, and contain capabilities for archiving, retrieving, and reducing flight test data. This system would have to be designed, developed, validated, and delivered within 10 months. The Kennedy Space Center division of McDonnell Douglas was asked to develop this ground control system because of its experience in this area. The RTDS was subsequently developed by the division's Automated Checkout Systems department based on a system it designed for space shuttle payload checkout, which is still in use. This baseline system, the partial payload checkout unit (PPCU), has been used in the Operations and Checkout Building at KSC since 1990.

PPCU is a generic, real-time data monitor and control system with front-end and back-end interface extensions and a distributed network of data processing and recording equipment. PPCU utilizes highly modular subsystems, industry standards, and commercial software and hardware, where practical, to provide a reliable, flexible, and continuously upgradable system at minimal cost.

6. Ground Systems Layout

The FOCC primarily consists of the equipment housed in a van and external interfaces to the DC-X vehicle GSS. The boundaries of the FOCC are illustrated in Figure 1.

The RTDS, the primary subsystem of the FOCC, serves as the operator interface for real-time monitor and control of both the DC-X vehicle and the GSS.

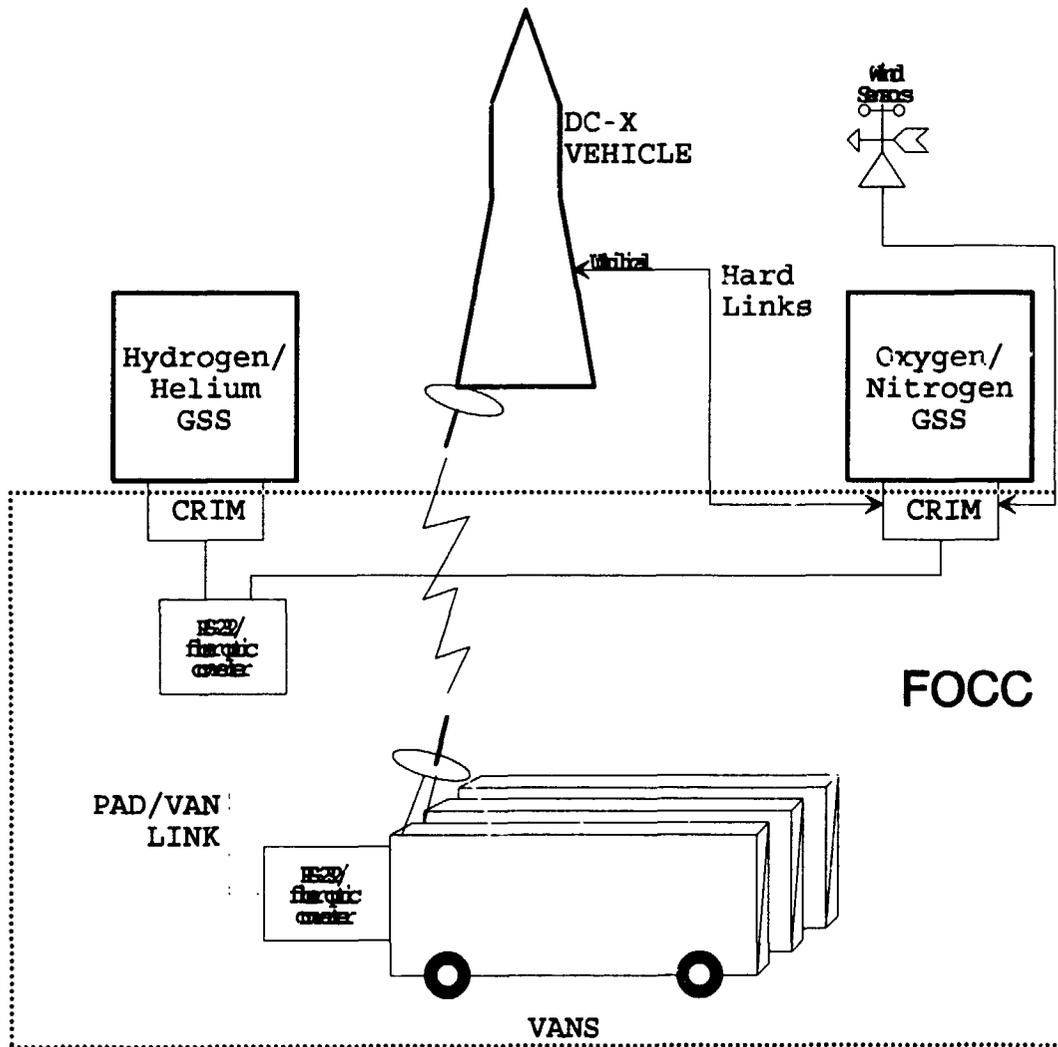


Figure 1. The Flight Operations Control Center

7. RTDS Architecture

The RTDS processing elements provide for monitoring and displaying real-time data. The recording elements provide independent real-time data recording. The architecture is implemented as shown in Figure 2 and consists of five major subsystems connected via Ethernet networks. Each subsystem consists of one or more processors in parallel or in clusters.

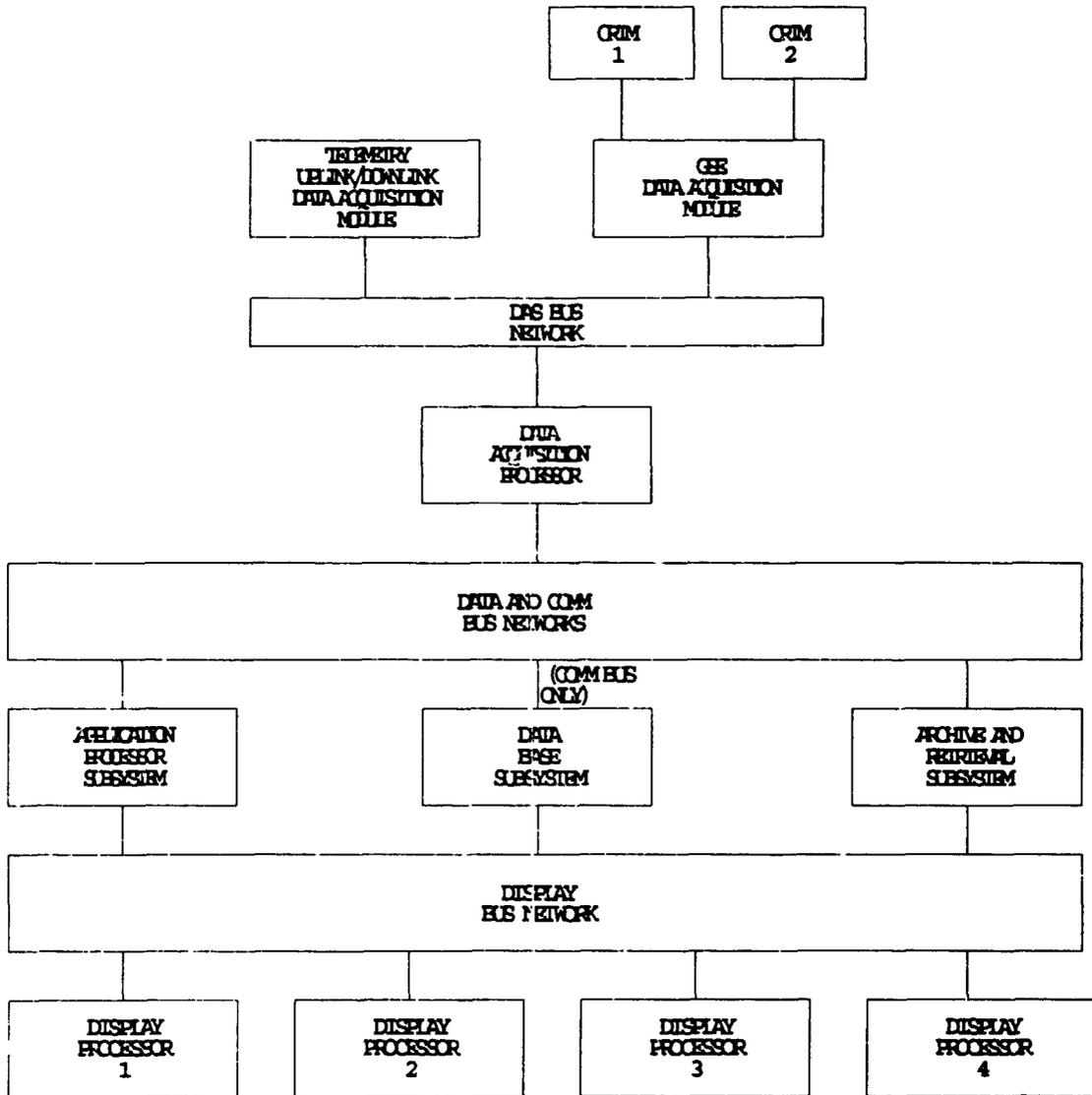


Figure 2. RTDS Architecture

Data Acquisition Modules

The interface from the RTDS to the DC-X vehicle and GSS is composed of a data acquisition subsystem that operates in an autonomous fashion. The subsystem features

three data acquisition modules connected to a data acquisition processor for data concentration. Each module is configured with hardware and software to preprocess a dedicated data link. The RTDS contains three data acquisition modules: telemetry downlink, telemetry uplink, and ground support equipment (GSE). For DC-X, these three modules' front-end interfaces are PCM telemetry downlink, RF RS-422 uplink, and GSE RS-232. They interface with fiber optic modems that connect the commercial remote interface modules (CRIMs) at the launchpad to the RTDS GSE data acquisition modules in the FOCC. This generic RTDS subsystem architecture provides flexibility to incorporate additional interface types in the future by simply adding an additional module to the data acquisition subsystem bus network in the RTDS.

Subsystem preprocessing consists of all the interface and data-dependent operations required to provide normalized, time-tagged data to the RTDS. The raw data from the vehicle and GSS is passed through a data acquisition module data filter. Each sample is compared with the last sample that changed significantly. If a significant change of value has occurred, then the new sample value is processed. Limit-checks are performed on the processed data in order to detect measurements that have violated any upper or lower limiting conditions. After limit-checking, the processed data is distributed throughout the rest of the system.

Commercial Remote Interface Module

The RTDS has a CRIM located on each side of the DC-X flight stand. The CRIM is a VME chassis that contains a microcontroller card and several discrete and analog input/output cards. The CRIM continuously monitors the analog and discrete status of the GSS equipment and sends the status back to the RTDS GSE data acquisition module. Commands from the RTDS GSE modules are received over the RS-232 communication lines and the appropriate analog and discrete channels are activated upon their receipt. Vehicle electrical power is also controlled through the CRIM.

Data Acquisition Processor

The concentrator in the data acquisition processor combines the data outputs from the modules into an integrated data stream broadcast to the other RTDS subsystems. The processor also receives system commands and end-item commands and routes them to the appropriate modules. Once loaded and initialized, the subsystem broadcasts data to the data acquisition processor for use by the application processor and the archive and retrieval subsystem.

Application Processor

The application processor is the RTDS real-time data processing element and provides for execution of the customized DC-X user application programs. It broadcasts measurement data to the display processors and processes commands from the users originating at the display processor. Commands issued from user applications are routed through a

command distribution manager on the application processor which verifies user permissions prior to issuing and routing commands to their proper destinations. All commands are recorded for posttest retrieval.

Archival and Retrieval Subsystem

The archival and retrieval subsystem contains the recording elements within the RTDS system. This subsystem records the digitized raw telemetry stream, as well as the processed telemetry and GSE data. Data are recorded to hard disk to support near-real-time retrieval of the vehicle and GSS information. Data can be retrieved and plotted in minutes using the hard disk-archived information. The data are also recorded on 4mm tape for the historical posttest retrieval archives. RTDS subsystem health, end-item commands, user "mouse" selections, and system messages are also recorded by the archival and retrieval subsystem.

Display Processors

Four color graphics workstations called display processors provide the user interface to control and monitor the DC-X vehicle and GSS. Operators send commands by a mouse and use the display processor multiwindow graphics capability to configure the system to their specific needs. The windows environment is an X-Window-based system; that is, it is implemented with off-the-shelf software tools to allow a continual upgrade path to future releases of hardware and software. The user interface is capable of being logically configured, based upon the user permission level, to support a range of capability from system configuration and monitor-only permissions to total control and monitoring permissions.

Database Subsystem

The database subsystem contains the RTDS data retrieval processing, configuration management utilities, and the RTDS generic measurement and command database. RTDS front-end interfaces and command data formats are defined in this generic database. The RTDS operator updates the telemetry uplink, and downlink, and GSE measurement and command information in the generic database using customized database forms. The subsystem then builds generic real-time tables for each of the RTDS subsystems. This generic database structure allowed for near-real-time modifications to the DC-X measurement and command information. This feature was critical in supporting the rapid pace of the DC-X program, where sensors were being added and modified frequently.

The generic format of the RTDS allows the system to contain multiple formats and provides the flexibility to easily support new systems in the future. New launch systems could be supported simply by defining the front-end telemetry and GSE information in the database and developing the customized user interface applications.

8. RTDS Human-Computer Interface

The DC-X ground operations procedures were developed using "aircraft-like" concepts to reduce ground operator workload. The RTDS was designed to allow a crew of only three operators to perform all the activities required for the DC-X preflight, flight, and postflight operations. Two operators, the flight manager and deputy flight manager, are assigned DC-X vehicle monitoring and control functions similar to pilot and co-pilot functions, while the third operator performs the duties of the ground systems manager.

Twenty-one customized applications were developed for the DC-X and its GSS as listed below:

Ground Support Systems (7)

- GSS propellant safing and master controls
- Liquid hydrogen
- Liquid oxygen
- Gaseous hydrogen
- Gaseous oxygen
- Gaseous nitrogen
- Gaseous helium

Vehicle Subsystems (9)

- Flight sequencer controls
- Vehicle hydraulics
- Vehicle main engines
- Vehicle reaction control system
- Vehicle propulsion system
- Vehicle avionics
- Vehicle rate accelerometer sensors and radar altimeter
- Vehicle electrical system
- Flight constants

Flight Displays (2)

- Flight profile
- Flight subsystem monitoring

RTDS Administration

- Pseudo measurement initialization
- Data acquisition module controls
- GSE-CRIM automated checkout

The RTDS applications were designed with many automated sequences and graphical monitoring features. The user interfaces maximize the DC-X operability features and keep the operator workload to a minimum. Electronic checklists record preflight steps to allow the operator to avoid use of paper manuals. Application displays are schematics of the actual GSS and DC-X vehicle subsystems to allow for rapid assessment of subsystem status and configuration. The displays rely heavily on the effective use of color coding, alarms, and positional dynamics to give the user both graphical and numerical representations of the vehicle and GSS information.

Ground Support Systems

The GSS contains seven applications for loading propellants and gases in the vehicle. The four gases have automatic topping modes that key off temperatures and pressures to control their flow. The liquid propellants have automated purging and loading sequences. The loading procedures monitor tank-level sensors to control the flow of the liquid oxygen and hydrogen. Automatic and manual safing features ensure that the vehicle can be quickly safed in the event of fires or leaks.

Vehicle Systems

The flight sequencer application contains the electronic checklist and controls to sequence through the 18 automated vehicle modes. The flight manager used this application to monitor events as the automated vehicle modes were commanded. The application has several screens that change as the flight manager commands the vehicle through its preflight built-in-tests and simulated flight modes. The deputy flight manager used the various subsystem screens to monitor the status of the built-in test equipment and subsystems throughout the preflight sequence.

Flight Screens

Two screens were developed to monitor the DC-X flight. The flight profile monitors the vehicle profile, latitude/longitude positions, altitude, and the pitch, roll, yaw information. The vehicle subsystem monitor displays engine performance graphs, propulsion, hydraulic, landing gear, flap, and engine gimbaling status.

Figures 1-4 contain black and white copies of four actual DC-X RTDS displays.

9. Conclusions

The DC-X program has proven that things can be accomplished quickly and cost-effectively by following a "build a little, test a little" rapid prototyping philosophy. The rapid prototyping technology was effective on this program, and could be used in the future as a means of achieving better, faster, and cheaper development.

The next step is to continue our "build a little, test a little" philosophy and move onto the two-third scale prototype known as the SX-2. This vehicle will be capable of higher altitude flight, higher velocities, and have greater maneuverability. The SX-2 will start to prototype and test lightweight material technology that will be required for the ultimate success of the full-scale single-stage-to-orbit vehicle known as the DC-Y.

FIGURE 3 - LIQUID HYDROGEN GROUND SUPPORT EQUIPMENT

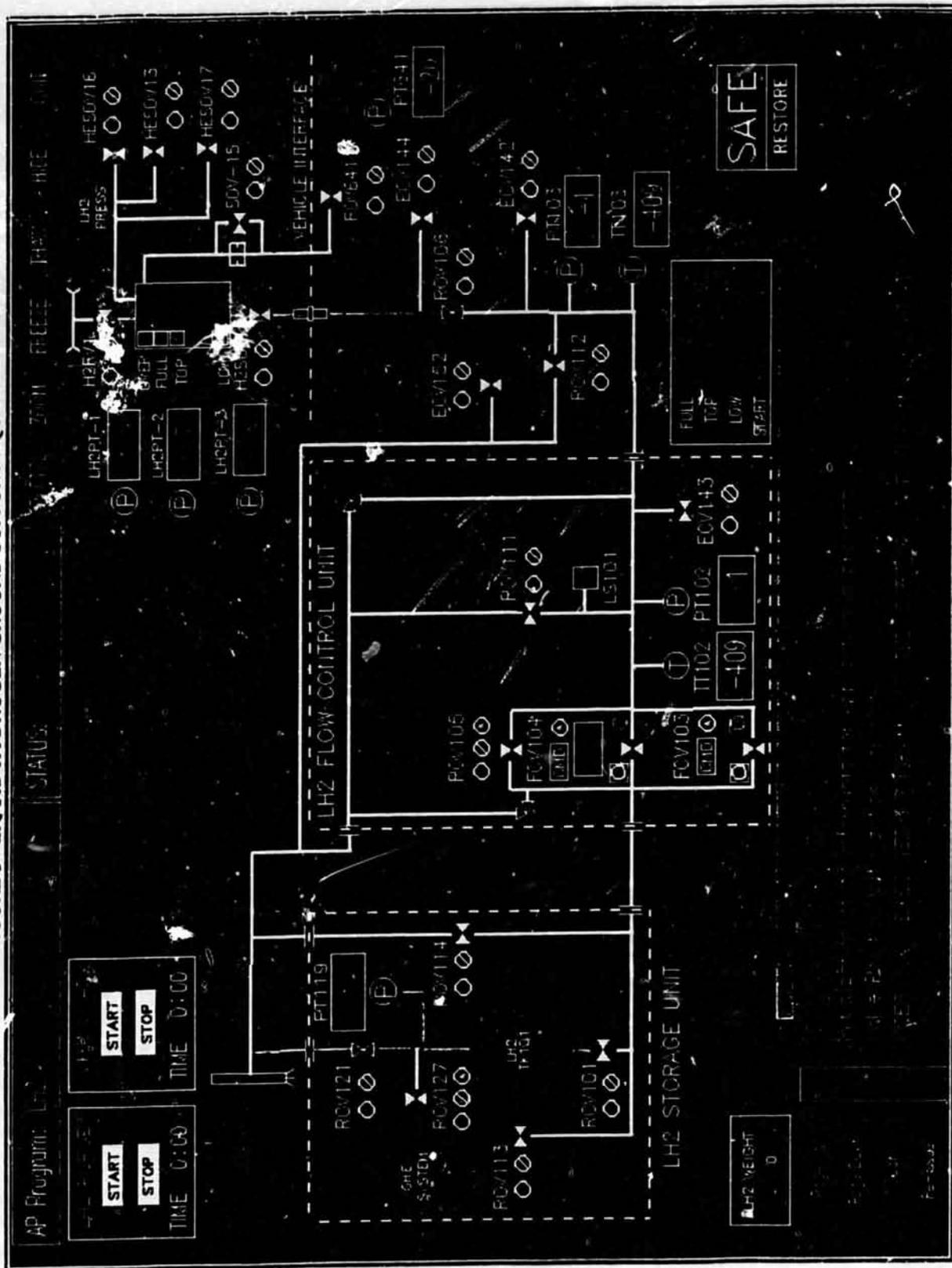


FIGURE 4 - DC-X VEHICLE SUBSYSTEM FLIGHT MONITORING

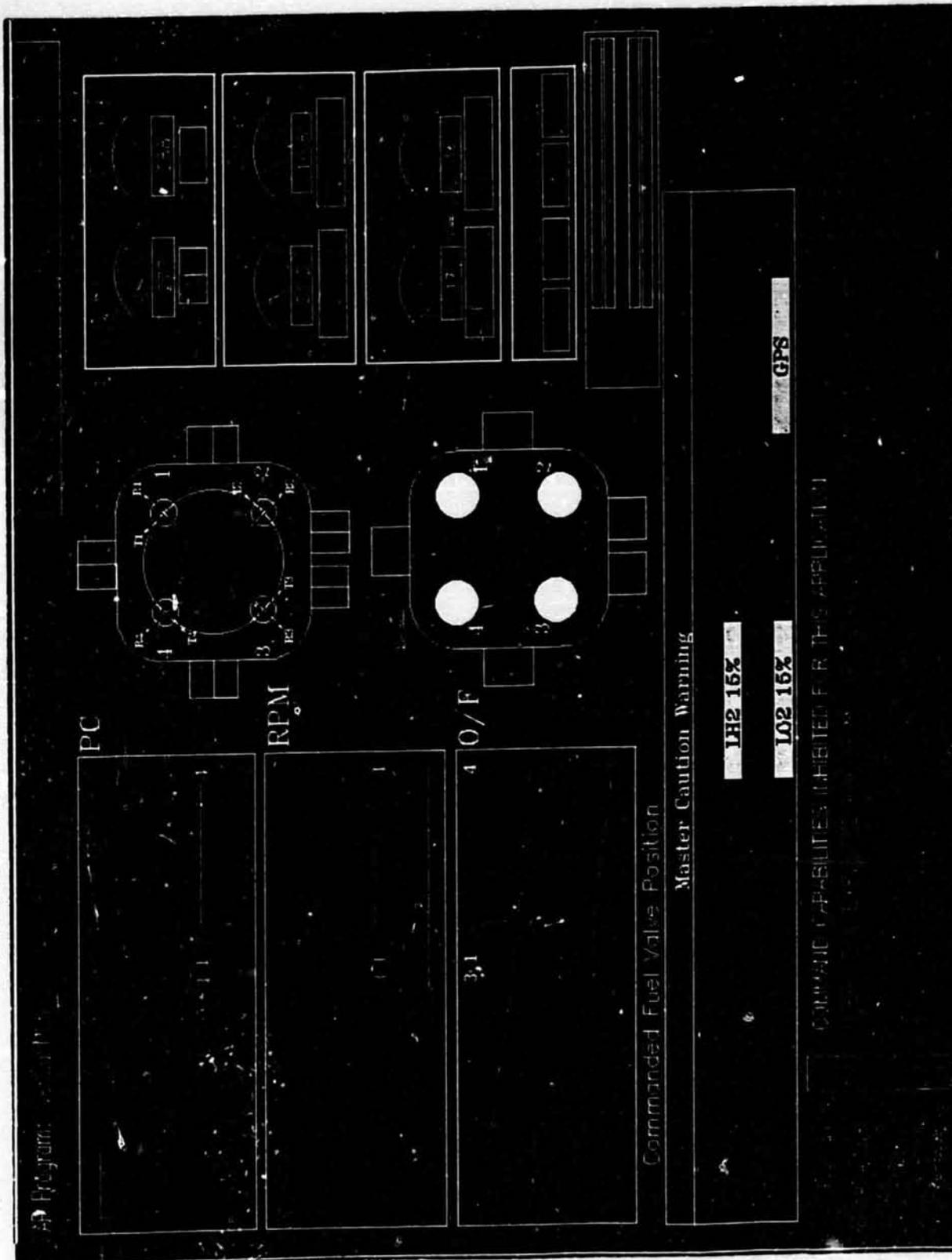


FIGURE 5 - DC-X FLIGHT PROFILE MONITORING

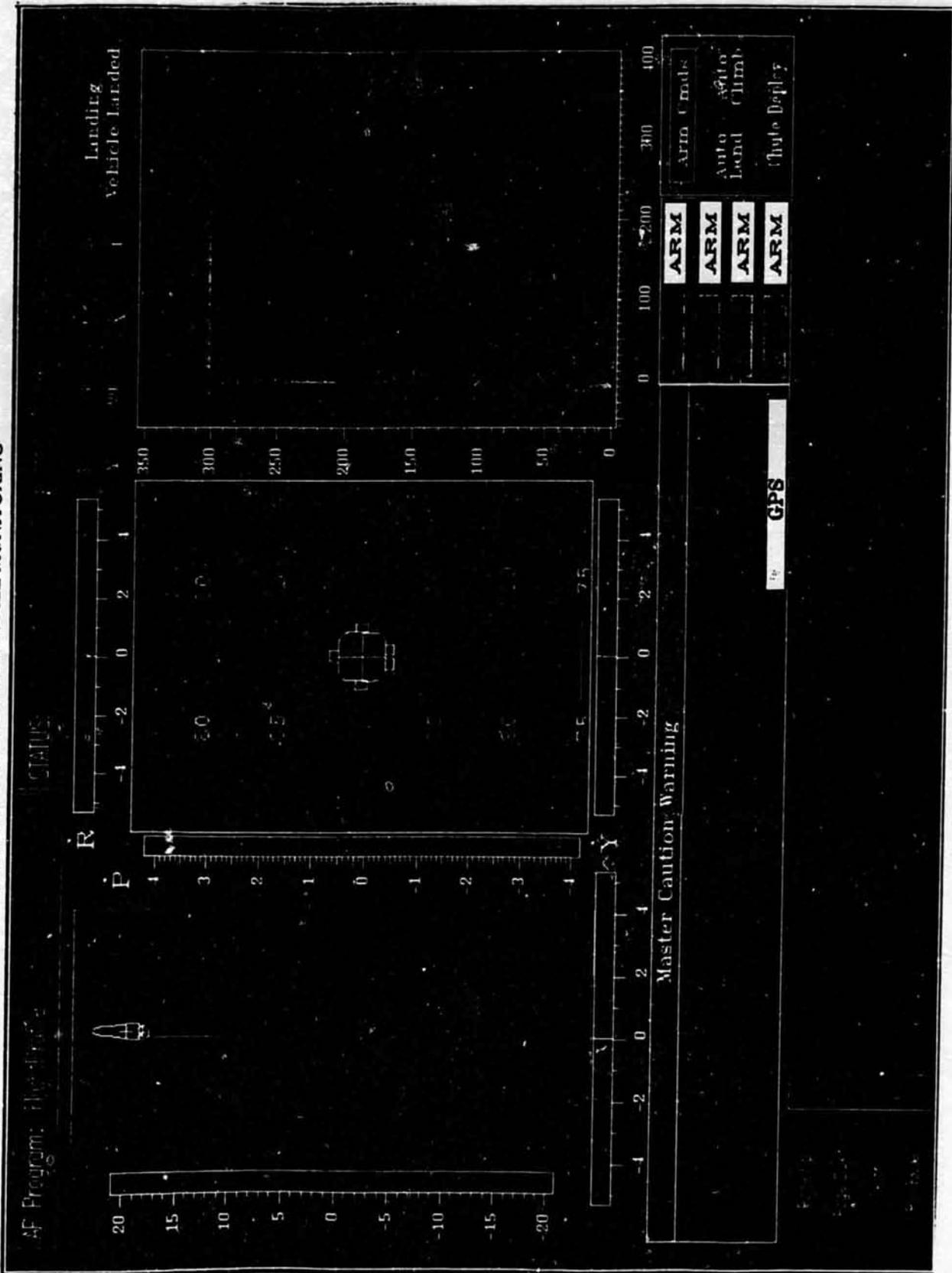
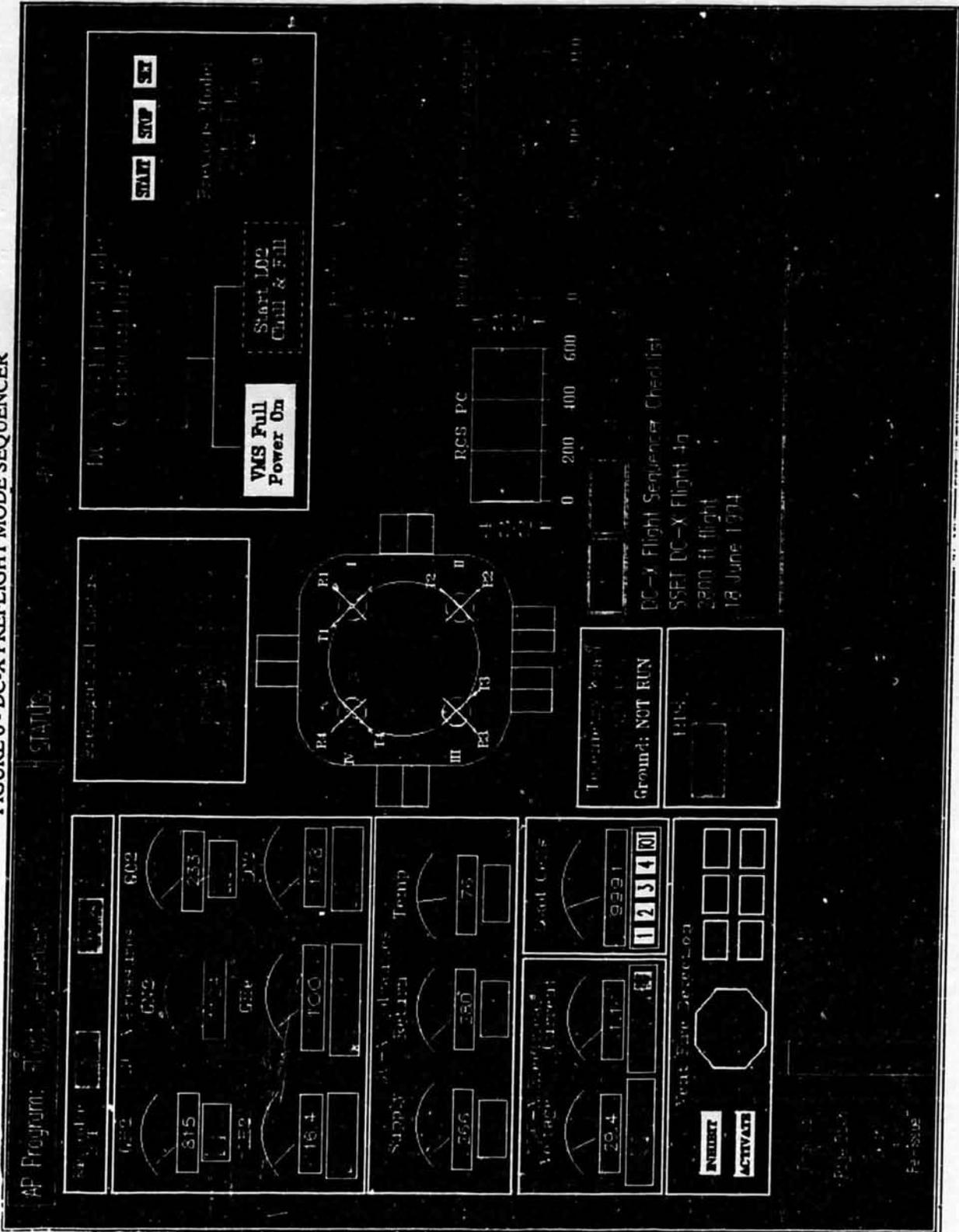


FIGURE 6 - DC-X PREFLIGHT MODE SEQUENCER



**INTERNATIONAL DATA TRANSFER FOR SPACE
VERY LONG BASELINE INTERFEROMETRY**21755
1-7

Alexandria B. Wiercigroch
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, CA 91109

ABSTRACT

Space Very Long Baseline Interferometry (SVLBI) experiments using a TDRSS satellite have successfully demonstrated the capability of using spacecraft to extend the effective baseline length of VLBI observations beyond the diameter of the Earth, thereby improving the resolution for imaging of active galactic nuclei at centimeter wavelengths. As a result, two spacecraft dedicated to SVLBI, VSOP (Japan) and RadioAstron (Russia), are scheduled to be launched into high Earth orbit in 1996 and 1997. The success of these missions depends on the cooperation of the international community in providing support from ground tracking stations, ground radio telescopes, and correlation facilities. The timely exchange and monitoring of data among the participants requires a well-designed and automated international data transfer system. In this paper, we will discuss the design requirements, data types and flows, and the operational responsibilities associated with the SVLBI data transfer system.

**INTRODUCTION TO THE SVLBI
DATA TRANSFER SYSTEM**

The number of facilities that play a role in any space mission is no longer restricted by institutional or national boundaries. A large majority of missions now involve international consortia, due to the immense cost and complexity of designing and operating a space mission and the desire to jointly benefit from space exploration. The ground-based technique of Very Long Baseline Interferometry (VLBI) has always required substantial international coordination, and the extension of VLBI to include one or more orbiting radio telescopes requires a similar effort.

Two spacecraft, VSOP (VLBI Space Observatory Programme) and RadioAstron, are being developed for use as orbiting observatories dedicated to making Space VLBI (SVLBI) observations at centimeter wavelengths. VSOP (Hirabayashi 1991; Hirosawa 1991), scheduled for launch in 1996, is a project of the Japanese Institute of Space and Astronautical Science (ISAS). RadioAstron (Kardashev and Slysh 1988), a project led by the Russian Astro Space Center (ASC), is due to be launched in 1997.

The success of the SVLBI missions depends strongly on the cooperation of a large number of international organizations, some of which are directly involved in radio astronomy, while others are contributing or funding hardware and software support. The international mission planning for SVLBI is presented in a paper by Ulvestad (1994) at this conference. The nominal operation and scheduling of experiments for the missions will take place at ISAS for VSOP and at ASC for RadioAstron. The tracking of the spacecraft will be performed by six tracking stations located worldwide and operated by four different organizations. They include three 11-m antennas under construction by NASA at the Deep Space Network (DSN) facilities located at Goldstone (United States), Madrid (Spain), and Tidbinbilla (Australia). A 14-m antenna will be operated by the National Radio Astronomy Observatory (NRAO) in West Virginia (United States). All four antennas will be capable of tracking either VSOP or RadioAstron. Additional tracking coverage will be provided by a 10-m antenna at Usuda, Japan (VSOP only) and a 32-m antenna at Ussuriisk, Russia (RadioAstron only). Tracking sessions at each site typically may occur two or three times a day and may last from a few minutes to more than 12 hours. Other ground radio telescopes that will provide co-observing support with RadioAstron and VSOP include

(but are not limited to) the Very Long Baseline Array (VLBA), the European VLBI Network, and the Australia Telescope National Facility. Correlation facilities located in Australia, Canada, Japan, the Netherlands, Russia, and the United States are being constructed or modified to accept the necessary files and data for correlation.

An integral part of the SVLBI missions will be the implementation of an international data transfer system that will retrieve and accept data files, monitor intermediate data products, and provide or construct the necessary files for final processing at the designated correlation facility. The major nodes of this data transfer system will be operated by ISAS, ASC, NRAO, and the Jet Propulsion Laboratory (JPL); these organizations are also responsible for all the tracking stations, and hence will provide all the space data. Processing of intermediate data products will be as automated as possible. Retrieval and redistribution of all data (except for the wideband VLBI data), will be via the Internet, utilizing standard file-transfer protocols.

The top-level requirements on the data transfer system are as follows:

1. Provide access to the detailed schedule files for all mission elements (principally ground telescopes and tracking stations) in a timely fashion.
2. Facilitate spacecraft performance monitoring by providing access to the telemetry data obtained from tracking stations.
3. Provide all spacecraft telemetry data required for the calibration of the space radio telescope.
4. Exchange navigation data required for high-accuracy orbit determination and subsequent VLBI data correlation.
5. Provide the VLBI correlators with all tracking-station information necessary for data processing.

UPLINK DATA TYPES

The types of data that are associated with the data transfer system can be grouped into two categories: files necessary to schedule and perform an experiment ("uplink") and the data necessary to correlate an experiment ("downlink"). Each data type (described below) may be the responsibility of disparate groups and, in many cases, depends on information provided by other organizations. In the following subsections, we will describe the files associated with the uplink processes necessary to prepare for a SVLBI experiment.

Schedule Files

A short-term schedule generated from the approved scientific program for VSOP and RadioAstron will be supplied by the VSOP Science Operations Group (VSOG) and RadioAstron Science Operations Group (RSOG) for VSOP and RadioAstron respectively. These two groups (known collectively as the SOGs) have the additional responsibility to provide a conflict-free schedule if both spacecraft are flying simultaneously. Each schedule will be divided into segments covering one week of mission operations and will describe all spacecraft and tracking-station activities in sufficient detail to allow each mission element to perform its required duties. A separate file following standard formats developed for ground VLBI will be made available to the ground radio telescopes. The initial schedule will be made available to the SVLBI data transfer system approximately five weeks in advance of the requested support period. It may be modified slightly up until a few days before the support is required.

The schedule file obtained from the SOGs will be maintained on multiple nodes of the data transfer system. These nodes will act as information servers to supporting ground radio telescopes and tracking stations. Negotiations are under way to finalize the contents and detailed formats of the schedule files and the method of procurement. It will be the responsibility of the personnel at the supporting facilities to retrieve each schedule file from the data transfer system, to extract all information needed to operate their facility, and to translate that information into the actual commands required for their mission element.

Predicted Orbit Files

An accurate predicted spacecraft trajectory is an important element of the uplink process. This trajectory is necessary not only for pointing ground tracking antennas, but also for generating an accurate frequency reference for VLBI observations. Predicted orbits will be generated by navigation teams associated with the DSN (for both spacecraft), ISAS (for VSOP only), and ASC (for RadioAstron only).

UPLINK DATA FLOW

Figure 1 shows a portion of the uplink data flow. Short-term schedule files covering a one-week time period are generated by the VSOG and RSOG for VSOP and RadioAstron, respectively. These schedule files must be conflict-free with respect to tracking and ground radio telescope support. They are made available to the SVLBI data transfer system nodes such as the one at JPL, where they may be accessed by supporting facilities. Information specific to ground tracking stations will be extracted and reformatted from the schedule file into the required operational commands. Ground radio telescopes will extract all the information needed to operate their facilities from a separate schedule file. In the case of the predicted spacecraft trajectory, navigation teams internal to each organization will construct the orbit and supply it directly to their own tracking stations. The predicted orbit also will be supplied from the JPL orbit determination group to the NRAO tracking station via the data-transfer node operated by the JPL Project. Figure 1 does not show all the details of the data flow within the data transfer system. Rather, the basic data types and pathways are indicated.

DOWNLINK DATA TYPES

The numerous data types created during and after an experiment, collectively known as "post-pass" data, include the VLBI data recorded on video cassettes or instrumentation tapes at each tracking station and telescope, near-real-time data used in monitoring the spacecraft and tracking station performance, and various other data supplied by the tracking stations following each tracking session. The latter data include two-way phase residuals from the link between the tracking stations and spacecraft, Doppler data, calibration informa-

tion from the downlink headers, and tracking-station logs which contain information about performance and recording parameters. These data must be extracted, processed, and supplied to the SOGs or the navigation teams on a regular (approximately daily) basis. Further processing, analysis, and combination of data is required to generate all the inputs needed for experiment processing at a correlation facility; all the final data products must be available to the correlators within 2-3 weeks after an observation. The logistics of a SVLBI experiment may involve mixtures of two spacecraft, four types of tracking station, three VLBI recording systems, a large number of ground telescopes, and up to six correlation facilities, providing an overwhelming number of possible data formats and combinations. Therefore, one of the biggest challenges for the international data transfer system is to define common procedures and interfaces for data handling. The subsections below discuss a few of the specific activities and problems associated with the different data types.

Wideband VLBI Data

The VLBI data are recorded in real time at each site and are later brought together at a special purpose processing correlation facility. The bulk of the tracking stations and telescopes will be equipped with VLBA-compatible recording systems. They will record data on tapes which can hold up to 10-12 hours of data at 128 Megabit/s and cost over \$1,000 per tape. Since a nominal SVLBI experiment may use 10 or more telescopes and may last 24 (or more) hours, the high cost makes it impossible to archive the raw data for all (or any) ground or SVLBI experiments. (A 1-month tape supply for 10 telescopes costs over \$600,000.) Instead, it is standard practice to correlate the experiments, provide a quality check of the data, and then to erase the tapes and distribute them into a general pool used by the world's radio telescopes. The data transfer system must provide all ancillary data in a timely fashion so that delays in processing do not lead to a requirement for purchasing additional tapes.

Science Headers

The header sections of the downlinked wideband VLBI data blocks contain information about the health and safety of the spacecraft,

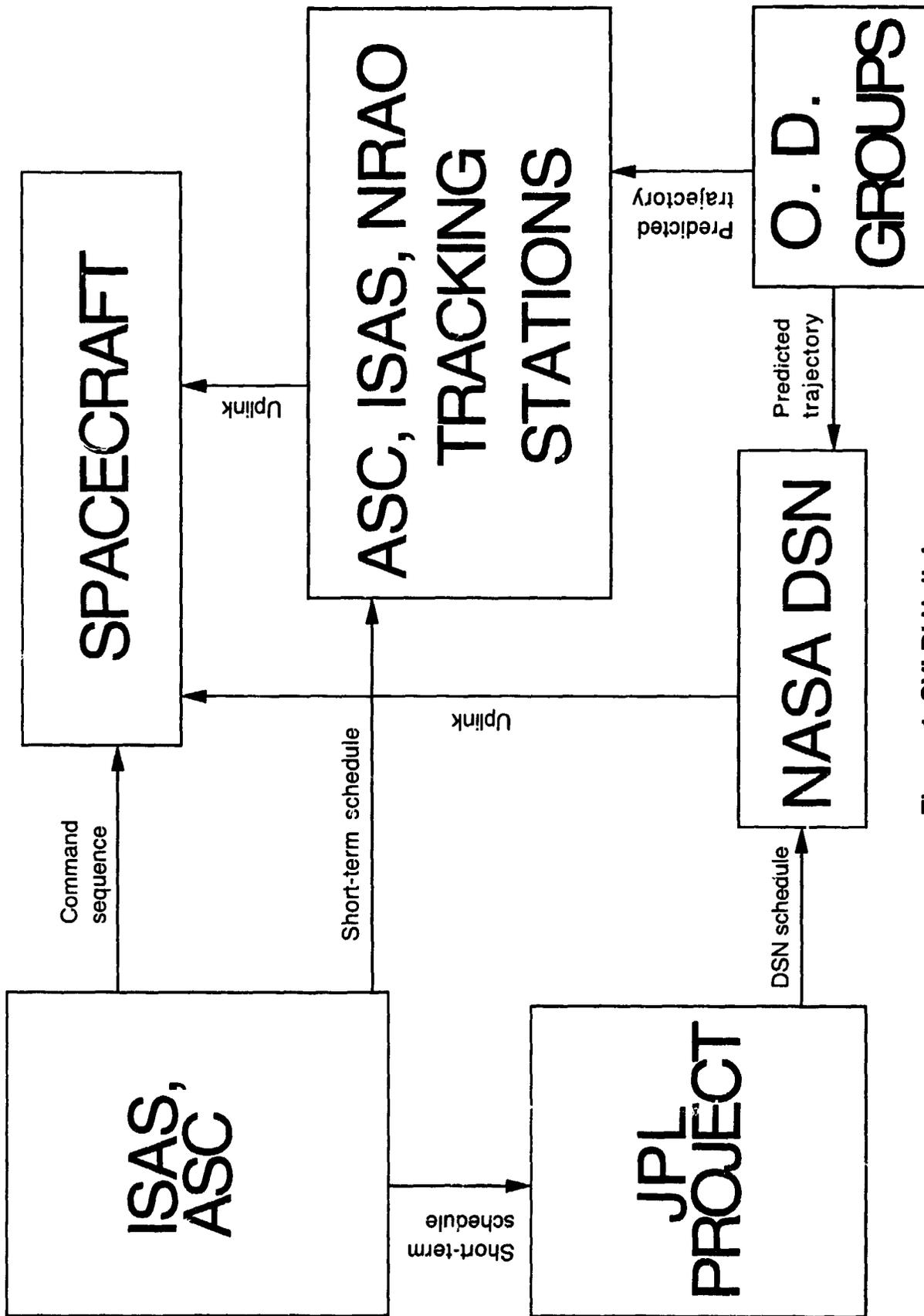


Figure 1. SVLBI Uplink

so some examination of these data in near-real-time is required. These headers will be extracted from the telemetry stream by each tracking station and made available to the spacecraft operators in a number of ways. The DSN tracking stations will extract portions of the headers and make them available to Russia or Japan in near-real-time, via the JPL data transfer node; these data will be available for monitoring of the health and safety of the spacecraft. The NRAO tracking station will check the values of some of the spacecraft parameters and report anomalous conditions to the SOGs in near-real-time. The Russian and Japanese tracking stations will extract the spacecraft monitor data and provide it directly to the appropriate spacecraft control teams.

The science headers also contain data critical to space radio telescope calibration. These data will be extracted and made available to the SOGs in a manner similar to that for spacecraft health data, either during (DSN) or after (NRAO) each tracking session. Extracted calibration data supplied by the Russian and Japanese tracking stations will be delivered to the appropriate SOG.

Observing Logs

Station log files describing the performance of the tracking stations and events that occurred during each session will be delivered to the data transfer system. These logs contain information on how the VLBI data were recorded, and are essential to the correlation process. The data will be assembled by the RSOG and VSOG and will be merged into the correlator input files (see below).

Phase Residuals and Reconstructed Orbit Files

A phase link between the tracking station and spacecraft is needed to provide an accurate on-board frequency reference (e.g. Edwards 1987; Levy et al. 1989; D'Addario 1991). Phase residuals from the two-way link must be tabulated at a rate of 10 Hz or greater to correct for orbit errors and the propagation of the signal through the Earth's troposphere and ionosphere. They are necessary for correcting the time and frequency information used by the correlator. Phase residuals from each type of tracking station are derived in a different man-

ner, but each station must supply equivalent information. The phase residuals will be supplied to the data transfer system, and retrieved by the SOGs for provision to the correlators. A common interface required for the different tracking stations and correlators is under development.

The phase link also will be used to generate two-way Doppler data. Each tracking station will deliver Doppler data in near-real time to the appropriate navigation teams, who will derive a final reconstructed orbit. The reconstructed orbit must meet stringent accuracy requirements for VLBI correlation. It may be delivered to the data transfer system as much as 3 weeks after an observation.

Correlator Input Files

In order to process a SVLBI experiment, a correlator must receive the following via the data transfer system: (1) VLBI data, (2) the reconstructed spacecraft orbit, (3) phase residuals, and (4) the correlator input log file. The last file is created by the SOGs based on the tracking-station logs, calibration data, and spacecraft performance information. A model of delay and delay-rate is used to search for the location of interference fringes (if any) in the cross-correlation data for each antenna pair. The correlator output data is then delivered to the principal investigator and may be used to derive visibility functions and various astrophysical parameters. This data is archived at the correlator facility.

DOWNLINK DATA FLOW

Figure 2 shows a portion of the downlink data flow associated with a SVLBI experiment. As with Figure 1, this diagram over-simplifies the data paths within the data transfer system. Near-real-time and post-pass data generated by various agencies must ultimately be collected and processed by the SOGs before being passed to the correlator. The operation of tracking stations by different agencies necessitates independent flow of data to the SOGs and correlators. Due to the complex data flow, the final correlation of data may begin only when all the necessary data has been supplied to the correlator, which may take as long as 3-4 weeks after a SVLBI experiment.

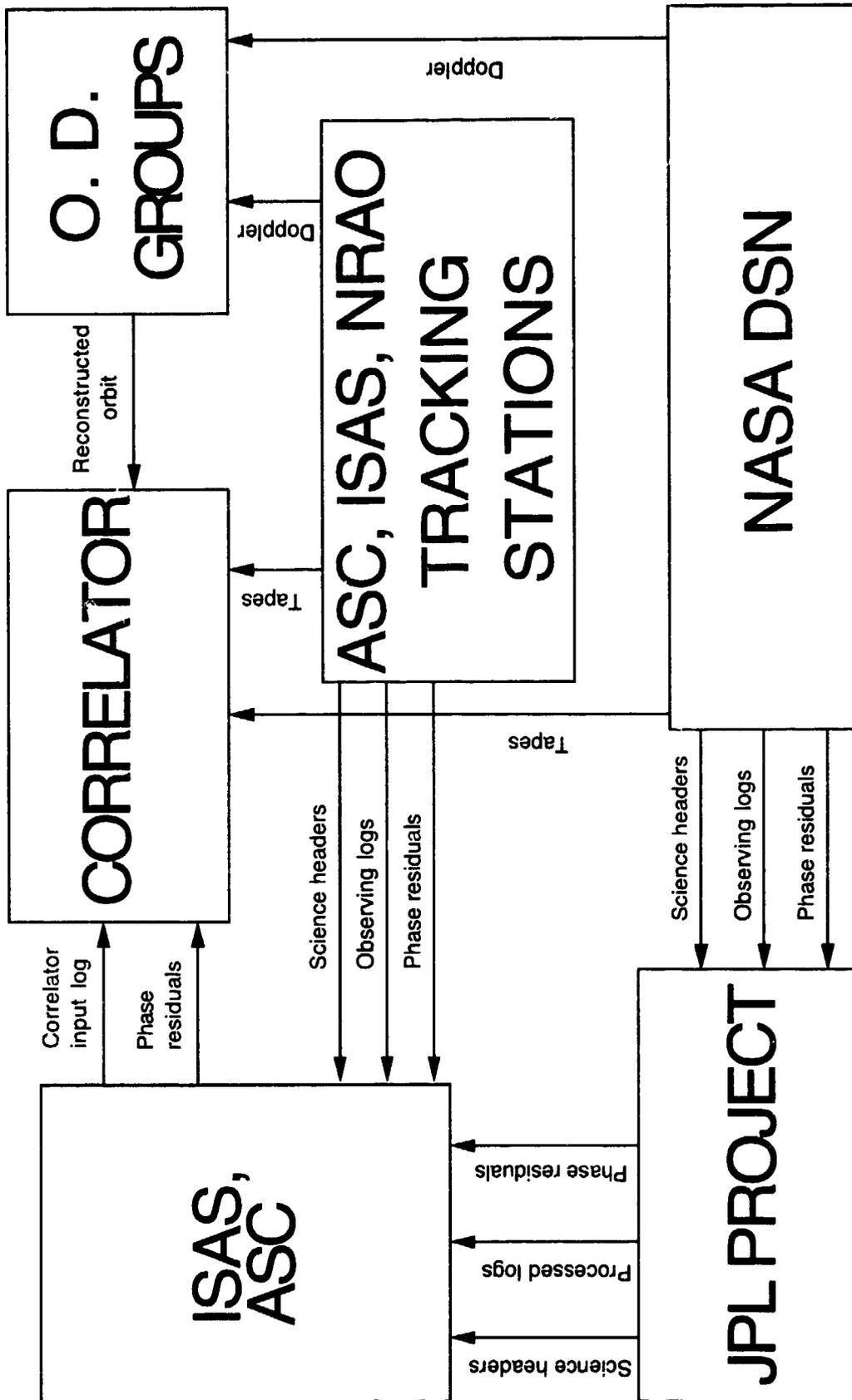


Figure 2. SVLBI Downlink

ACKNOWLEDGMENTS

The author would like to thank the following individuals who have contributed to the development of the international data transfer system: M. Artyukhov, L. D'Addario, H. Hirabayashi, S. Kamenon, N. Kawaguchi, H. Kobayashi, G. Langston, Y. Murata, M. Popov, J. Romney, A. Sheikhet, J. Smith, J. Ulvestad, and V. Yakimov. In addition, the following individuals at JPL have contributed greatly to the design and implementation of the DSN tracking stations and the data interfaces between those stations and the JPL Space VLBI Project: A. Berman, F. Chen, S. Davis, J. Ellis, J. Hops, K. Liewer, B. McLemore, A. Short, and J. Springett.

A portion of this work was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under contract with NASA.

REFERENCES

- D'Addario, L. R. (1991). Time Synchronization in Orbiting VLBI. *IEEE Transactions on Instrumentation and Measurement*, Vol. 40, 584-590.
- Edwards, C. D. (1986). Link Calibrations for the TDRSS Orbiting VLBI Experiment. *Radio Astronomy from Space*. Proceedings of NRAO Workshop No. 18, 125-129, Green Bank: NRAO.
- Hirabayashi, H. (1991). Initial VSOP Astronomical Requirements. *Frontiers of VLBI*, 15-19, Tokyo: Universal Academic Press.
- Hirosawa, H. (1991). VSOP Satellite System Overview. *Frontiers of VLBI*, 21-25, Tokyo: Universal Academic Press.
- Kardashev, N. S., and Slysh, V. I. (1988). The RadioAstron Project. *The Impact of VLBI on Astrophysics and Geophysics*. Proceedings of IAU Symposium 129, 433-440, Dordrecht: Kluwer Academic Publishers.
- Levy, G. S. et al. (1989). VLBI Using a Telescope in Earth Orbit. I. The Observations. *Astrophysical Journal*, 336, 1098-1104.
- Ulvestad, J. S. (1994). International Mission Planning for Space Very Long Baseline Interferometry (these proceedings).

ARACHNID:**A PROTOYPE OBJECT-ORIENTED DATABASE TOOL
FOR DISTRIBUTED SYSTEMS**

Herbert Younger
Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, CA 91109

John O'Reilly, PhD
Bjorn Frogner, PhD
MIMD Systems, Inc
3145 Porter Drive
Palo Alto, CA 94304

ABSTRACT

This paper discusses the results of a Phase II SBIR project sponsored by NASA and performed by MIMD Systems, Inc. A major objective of this project was to develop specific concepts for improved performance in accessing large databases. An object-oriented and distributed approach was used for the general design, while a geographical decomposition was used as a specific solution. The resulting software framework is called ARACHNID.

The Faint Source Catalog developed by NASA was the initial database testbed. This is a database of many giga-bytes, where an order of magnitude improvement in query speed is being sought. This database contains faint infrared point sources obtained from telescope measurements of the sky. A geographical decomposition of this database is an attractive approach to dividing it into pieces. Each piece can then be searched on individual processors

with only a weak data linkage between the processors being required.

As a further demonstration of the concepts implemented in ARACHNID, a tourist information system is discussed. This version of ARACHNID is the commercial result of the project. It is a distributed, networked, database application where speed, maintenance, and reliability are important considerations.

This paper focuses on the design concepts and technologies that form the basis for ARACHNID.

INTRODUCTION

Progress in the field of software for multiple processors is lagging behind the progress made in development of the processors themselves. A key issue is development of effective algorithms that can distribute the load among the processors in a manner that is transparent to the application developer.

Significant R&D progress has been made throughout the industry during the last few years¹, but the distance between actual versus potential throughput is still very large. For these reasons, it is an exciting field since the technical issues are challenging and the business opportunities are numerous.

The use of multiple processors for searching databases spans the field from massively parallel processors, where each CPU is very low cost, to distributed systems where each CPU essentially is a separate computer (e.g., PC); e.g., see refs^{2,3}

PROJECT OBJECTIVES

Phase I of the project started with the goal of using parallel processors to achieve dramatic speed improvements. However, at the start of Phase II, the focus was changed to distributed systems as an arena where more cost effective solutions could be found. The potential gain in this arena was judged to have a larger commercial potential since networked PCs are so widely available.

The specific objectives of Phase II were as follows:

- Develop algorithms and software for distributed access to large databases.
- Demonstrate and test the software.
- Develop a commercialization plan.

Phase II proceeded through a progression of tasks that are typical for a software R&D project: design specifications, implementation, testing, evaluation, and documentation.

When this paper was written, the focus was on the commercialization of the technology.

RELEVANT TECHNOLOGIES

Resource Allocation

Using two or more processors to solve one computational problem (e.g., search through a database looking for specified aggregate results) can be treated as a resource allocation problem⁴. The issue is how to distribute the load onto the available CPUs considering their performance, the speed of communication between them and the coupling between the sub-problems solved by the processors.

Thus, this project did a substantial review of mathematical programming methods (e.g., Dynamic Programming) to determine their possible contribution towards solving this resource allocation problem. It was finally determined that a heuristic approach derived from an understanding of the unique nature of the problem had to be developed.

A geographic decomposition approach was found to be of sufficient general utility for ARACHNID. Although we cannot quantify the extent to which this is a sub-optimal solution, there is a sufficiently large number of problems for which this is an attractive approach.

Object-Oriented Databases

Organizing data in terms of logical units (e.g., records in a database) is well proven. Using an object-oriented approach⁵⁻⁸ is relatively new and in many ways very appealing. Thus, this project made a review of the available object-oriented databases that were available at the time the project started.

Leading products (e.g., Vbase⁹, GemStone¹⁰, and Iris¹¹) were evaluated with respect to their relevance to ARACHNID. Although, it was found that these products would give high flexibility and powerful data representations,

they would be cumbersome for a distributed configuration and difficult to optimize for computationally intensive problems.

The final solution was to use a product called C Data Manager (CDM), which is a C-callable library. It has a low-level, object-oriented, database engine with a high degree of flexibility for the developer. However, it does not have the robustness for concurrent access as we all have become accustomed to in relational databases. Thus, CDM was used for storage of local data, mostly in support of the user interface and temporary results.

SYSTEM REQUIREMENTS

Functional Requirements

ARACHNID is based on object-oriented and distributed technologies. Users and application developers are provided with an object-oriented environment, including encapsulation, object identity, persistence, and inheritance.

ARACHNID is designed to integrate with existing databases to provide object persistence. The design allows for interfacing with multiple database engines.

ARACHNID's major technology features are summarized as follows:

- *Object-Oriented.* It uses an object-oriented database for storage of its own local data.
- *Local Autonomy.* Once initiated, it does not rely on external processes for its operation. Furthermore, if a particular server fails, only clients associated with the server objects will be hampered.
- *Object-Oriented Interface.* It provides an object-oriented interface and a set of

support utilities for both the database administrator and application developer.

- *Fragmentation Transparency.* It hides the storage fragmentation of an object. Hence, it manages aggregate object types (sometimes multimedia data) and presents the entity as a single object.
- *Distributed Transaction Management.* At the internal level, it relies on the underlying database subsystem for object recovery control.
- *Operating System Independence.* The design assumes that the underlying operating system supports a message-passing paradigm and a client/server architecture.

DESIGN

ARACHNID is based on a distributed client-server model (see Figure 1) in which an interconnected set of servers uses an object-oriented approach to provide a high-level database facility. ARACHNID interfaces to other system-level and application-level software that can access arbitrary data structures across the network.

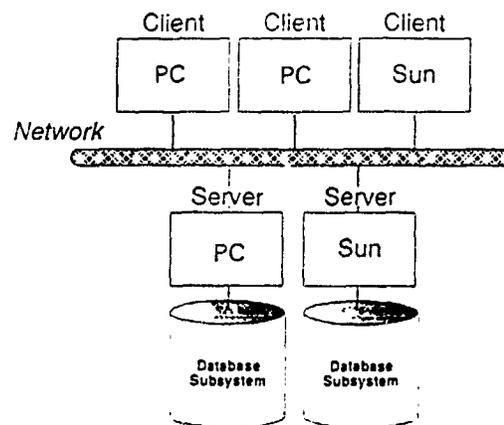


Figure 1: Client/Server Configuration

Although ARACHNID is an object-oriented tool for accessing databases, it is not an object-oriented database management system. Instead, it accesses existing databases that have been organized (and are managed by) other DBMSs, such as Sybase, Paradox, and Oracle. These databases are stored on computers networked with ARACHNID. The system's services are connected in a web-like fashion, thereby logically leading to its name: ARACHNID.

ARACHNID was designed to realize the potential of multiple processors for complex operations on databases located on geographically distributed, heterogeneous computers. ARACHNID accesses data from these databases and transports it to its client computer when needed.

In ARACHNID, a query is divided into small components ("fragments") and each fragment is allocated to a computer on the network. A query module resides on each network computer and controls the execution of fragments on that computer. A control node coordinates the distribution of fragments to network computers, as well as the collection of fragment queries implemented on those computers.

As an example of fragments and objects, consider a form with several fields of which some have methods that are activated when selected by the user. Assume that some of these methods will bring up other forms with data from other databases. In ARACHNID, each field in the form is an object while one or more forms can be classified as a fragment if it represents a natural unit to be executed as one entity.

The user interface for ARACHNID employs the latest in Graphical User Interface (GUI) methods. The ARACHNID user screens has a

"point-and-click" environment that makes the creation of user requests quick and easy. ARACHNID is available on both MS Windows 3.1 and UNIX. Figure 2 is an example of an input screen for generation of a query to the Faint Source Catalog on a SUN computer.

Variable Name	Element	Relation	Value
fcat	1	<=	3

cntr	<	1
name	=	2
fcat	<=	3
rah	>	4
ram	>=	5
fcat [1]	<	7

Figure-2: Example of a Query Input Screen

DATABASE DECOMPOSITION

Decomposition can be used to divide a domain into a number of non-overlapping subdomains. This approach has seen many applications in large matrix problems, and it is particularly appropriate for the decomposition of large celestial regions into smaller regions.

Databases often have a high degree of independence between subdomains. When a subdomain is retrieved or updated, other domains are often not involved. This independence is not only a product of the distribution of data -- it is a common occurrence in databases because the designer typically created the database schema that way.

The distribution of astronomical sources can be represented in a spherical coordinate system, with each source described by two arguments: equinoctial α and ecliptic β , as shown in Figure 3. It is natural and

convenient to search concurrently all stars within a given distance from a fixed position.

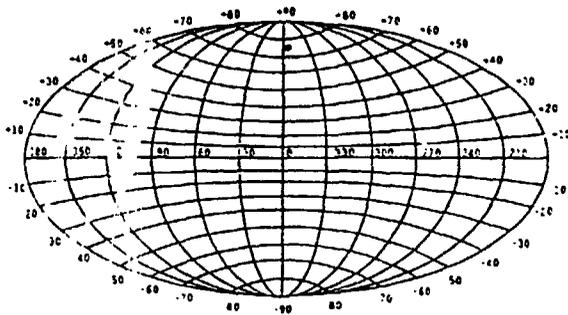


Figure -3: Decomposition for a Sphere

The following two factors were not considered in the above discussion. First, the spherical meshes are not uniform if the decomposition is made according to the method shown in Figure 3. Second, the distribution of the stars in the sky is not uniform. In fact, the distribution is quite irregular, as can be seen in Figure 4.

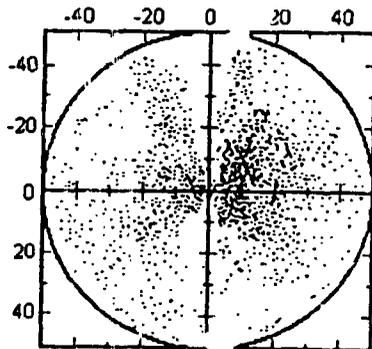


Figure 4: Distribution of Stars

Thus, if sparse star domains, a coarser net is better than a fine net. The allocation problem is then to divide the small group into even

smaller groups and to distribute those components in a balanced manner.

Most computer systems provide efficient file managers that allow multiple users simultaneous access to reading a file. Thus, by distributing subsets of the database across multiple processors and each processor only having occasional need to read data residing on another processor, a decomposition method can be effective for finding (for example) objects having a given brightness, a given color, or location within a certain angular separation of a given point.

COMMERCIAL APPLICATION

The objective of SBIR projects is to develop technology that can benefit the sponsor and result in commercial products for the contractor. This project very much followed this path.

The concepts of geographic partitioning of databases developed for ARACHNID has been the basis for a key element of a new tourist information system. Consider such a partitioning in the context of the Miami - Fort Lauderdale area. A system installed in Miami covers that part of Florida that is closest to Miami. The same statement is correspondingly true for Fort Lauderdale. Now consider a user that needs directions from Fort Lauderdale to locations near Miami.

To solve this problem, it is necessary to make a decision with respect to the location of the database which contains the driving directions. There are three simple but unattractive choices:

- The database can be restricted to only cover a local area. This represents the simplest design. However, it is not attractive since there will be a wide

network of fairly closely located ARACHNID systems.

- The entire database could reside in one central location. This is a relatively simple design but it has several performance related problems; e.g., less than timely response to the queries and unacceptable down-time when the network fails.
- The database can have overlap between the various locations. This also represents an easy design; however, from a maintenance point of view it is unattractive to keep duplicate copies of portions of the database.

Thus, ARACHNID is designed to access databases at remote locations for long distance driving directions. Since the directions are stored as a set of connected nodes, a database query can easily result in "hits" from more than one location. It is expected that the users of this system will find it acceptable to wait a little longer for long distance directions than local directions.

RESULTS AND CONCLUSIONS

This paper has described a Phase II SBIR project for NASA performed by MIMD Systems. The objective of this project was to develop algorithms for distributed access to a certain class of large databases.

Much of the ARACHNID software has been implemented on both 486-based PCs and SUN SparcStations. Most of the NASA-specific development and testing was done on SUN computers, while the commercial version is primarily for the PC.

NASA's Faint Source Catalog was used as the initial testbed for a geographical decomposition of the database. Although more work is

needed to harness the potential gain, there are good indications that the established goal can be achieved. The limiting factor is in the speed of communication between the processors and the cost involved in implementing the distributed hardware and software configuration.

For the commercial version of ARACHNID, the distributed approach has proven to be very valuable in terms of providing competitive performance, reliability and maintenance. This tourist information implementation of ARACHNID is now being installed on a nationwide basis for a major tourism industry company.

REFERENCES

1. DeWitt, D.J., Gray, J.: "Parallel Database Systems: The Future of High Performance Database Systems." Comm. ACM 35 (6), 85-98, 1992.
2. Sakti, P.; Ho, K.M., "Generalized Parallel Processing Models for Database Systems," Proc. of the 1988 Inter. Conf. on Parallel Processing, 1988.
3. Fox, G. et al., Solving Problems on Concurrent Processors: Volume 1, Prentice Hall, 1988.
4. McEntire, P.L.; and Larson, R.E., "Optimal Resource Allocation in Sparse Networks", IFAC/81, 1981.
5. Blaha, M.R.; Premerlani, W.J.; and Rumbaugh, J.E., "Relational Database Design Using an Object-Oriented Methodology", CACM, Vol 31, No. 4, April 1988.
6. Deppisch, U. P.; and Schek, H.J., "A Storage System for Complex Objects", Proc. of the 1986 International Workshop on Object-Oriented Database Systems, pp. 183-195, 1986.

7. Dittrich, K. R., "Object-Oriented Database Systems: the Notion and the Issues," Proc. of the 1986 International Workshop on Object-Oriented Database Systems, pp. 2-4, 1986.
8. Meyer, B., Object-Oriented Software Construction, Prentice-Hall, 1988.
9. Ontologic Corporation, Vbase Functional Specification for Release 1.0, 1987.
10. Maier, R., et. al., "The GemStone Data Management System", Object-Oriented Concepts, Database, and Applications, ACM Press, 1989.
11. Fisherman, D.H. et. al., "Overview of the Iris DBMS", Technical Report HPL-SAL-89-15, Hewlett-Packard Laboratories, 1989.

Acknowledgements - The work presented in this paper was partially sponsored by NASA Phase II SBIR Contract # NAS7-1156. While the authors have the sole responsibility for the contents of this paper, we wish to recognize the contributions of: Dr. Joe Mazzarella at NASA/JPL and the following individuals at MIMD Systems: Dr. Robert E. Larson and Dr. Paul L. McEntire.

3. Telemetry Processing		Page 185
DM.3.a	A Low-Cost Transportable Ground Station for Capture and Processing of Direct Broadcast EOS Satellite Data <i>Don Davis, Toby Bennett, Nicholas M. Short, Jr.</i>	187-195 -21
DM.3.b	Applications of Massively Parallel Computers in Telemetry Processing <i>Tarek A. El-Ghazawi, Jim Pritchard, Gordon Knoble</i>	197-204 -22
DM.3.c	Test Telemetry and Command System (TTACS) <i>Alvin J. Fogel</i>	205-212 -23
DM.3.d	Ground Equipment for the Support of Packet Telemetry and Telecommand <i>Wolfgang Hell</i>	213-224 -24
DM.3.e	Process and Methodology of Developing Cassini G&C Telemetry Dictionary <i>Edwin P. Kan</i>	225-232 -25
DM.3.f	Multimission Telemetry Visualization (MTV) System: A Mission Applications Project From JPL's Multimedia Communications Laboratory <i>Ernest Koeberlein, III, Shaw Exum Pender</i>	233-240 -26
DM.3.g	VLSI Technology for Smaller, Cheaper, Faster Return Link Systems <i>Kathy Nanzetta, Parminder Ghuman, Toby Bennett, Jeff Solomon, Jason Dowling, John Welling</i>	241-248 -27
DM.3.h	Telemetry Distribution and Processing for the Second German Spacelab Mission D-2 <i>E. Rabenau, W. Kruse</i>	249-256 -28
DM.3.i	Lessons Learned Supporting Onboard Solid-State Recorders <i>Jeff Shi, Tony Mao, Tim Clotworthy, Gerald Grebowsky</i>	257-264 -29
DM.3.j	Experience With the EURECA Packet Telemetry and Packet Telecommand System <i>Erik Mose Sørensen, Paolo Ferri</i>	265-272 -30
DM.3.k	A Modular, Software Reprogrammable, Telemetry Preprocessor for Open Systems Backplane Architectures <i>Steve Talabac</i>	273

* Presented in Poster Session

A Low-Cost Transportable Ground Station for Capture and Processing of Direct Broadcast EOS Satellite Data

Don Davis, Toby Bennett
Microelectronics Systems Branch, Code 521
Mission Operations and Data Systems Directorate
NASA, Goddard Space Flight Center
Greenbelt, Maryland 20771

Nicholas M. Short, Jr.
Information Science and Technology Branch, Code 935
Earth Sciences Directorate
NASA, Goddard Space Flight Center
Greenbelt, Maryland 20771

ABSTRACT

The Earth Observing System (EOS), part of a cohesive national effort to study global change, will deploy a constellation of remote sensing spacecraft over a 15 year period. Science data from the EOS spacecraft will be processed and made available to a large community of earth scientists via NASA institutional facilities. A number of these spacecraft are also providing an additional interface to broadcast data directly to users. Direct broadcast of real-time science data from overhead spacecraft has valuable applications including validation of field measurements, planning science campaigns, and science and engineering education.

The success and usefulness of EOS direct broadcast depends largely on the end-user cost of receiving the data. To extend this capability to the largest possible user base, the cost of receiving ground stations must be as low as possible. To achieve this goal, NASA Goddard Space Flight Center is developing a prototype low-cost transportable ground station for EOS direct broadcast data based on Very Large Scale Integration (VLSI) components and pipelined, multi-processing architectures. The targeted reproduction cost of this system is less than \$200K. This paper describes a prototype ground station and its constituent components.

1. INTRODUCTION

For a number of years, any organization equipped with a relatively low-cost weather ground station has been able to acquire imaging data directly from weather satellites. For these spacecraft, direct broadcast of data has been necessary to service a widely distributed and diverse user community. This method of delivering data to users has allowed weather data to be used worldwide in a variety of applications. NASA science missions, however, have generally lacked this method of data delivery. A number of factors have precluded the use of direct broadcast techniques in NASA science missions. Among these factors are past policies restricting public access to science data and the prohibitive cost of the ground station and processing equipment required for science missions.

For the Earth Observing System (EOS), NASA has adopted an open door policy allowing wider access to science data products. Over a 15 year period, EOS will deploy a constellation of low-earth orbiting remote sensing spacecraft to monitor the Earth's environment. Although all EOS instrument data will be captured, processed and made available through centralized NASA facilities, the data of select instruments will also be available through direct broadcast from EOS spacecraft. The first EOS spacecraft, EOS-AM1 which is planned for launch in 1998, will broadcast data from the MODIS instrument. Through this capability, users will be able to capture real-time MODIS images of their geographic region taken while the spacecraft is flying overhead.

The EOS direct broadcast capability can be very valuable to many different organizations. Scientists can use this capability to conduct or validate field measurements, plan corroborative campaigns, and observe rapidly changing conditions in the field. International meteorological and environmental agencies could take real-time measurements of the atmosphere, storm and flood status, water temperature and vegetation stress. International science partners will have the ability to perform engineering quality checks and scientific studies at their own analysis centers. [1] Academic institutions could use the ground station and the data collected by it to illustrate important topics in science and engineering education and to provide students with hands-on experience. In addition, all users could get a "quick look" at the data until they received it from NASA EOS institutional processing facilities.

The number of organizations that will have access to EOS direct broadcast depends largely on the end-user cost of the equipment required to acquire and process the data. In order to maximize the number of potential users, the cost of receiving ground stations must be made as low as possible. Major cost reductions can be gained by developing high integration components to perform the major system processing functions. Further cost reduction can be gained by recognizing that this class of ground station is not necessarily constrained by the same stringent requirements imposed on NASA institutional systems. By relaxing availability, fault tolerance, and redundancy requirements, additional cost reductions may be realized.

2. A LOW COST SOLUTION

NASA Goddard Space Flight Center is currently developing a low-cost solution to provide low-cost direct data access. This solution will demonstrate a prototype low-cost transportable ground acquisition and processing station for EOS direct broadcast data. This system will include four elements: an antenna, Radio Frequency (RF) processing equipment, Consultative Committee for Space Data Systems (CCSDS) protocol processing equipment, and a UNIX workstation containing an automated schedule-driven software package for system control and science data processing. The initial targeted reproduction cost for this direct broadcast acquisition system will be less than \$200K with further cost reductions to be realized as the technology matures. Through the use of this system, users will have the ability to receive direct broadcast data for any capable spacecraft, track the spacecraft, acquire and process science data and analyze the data on a local workstation.

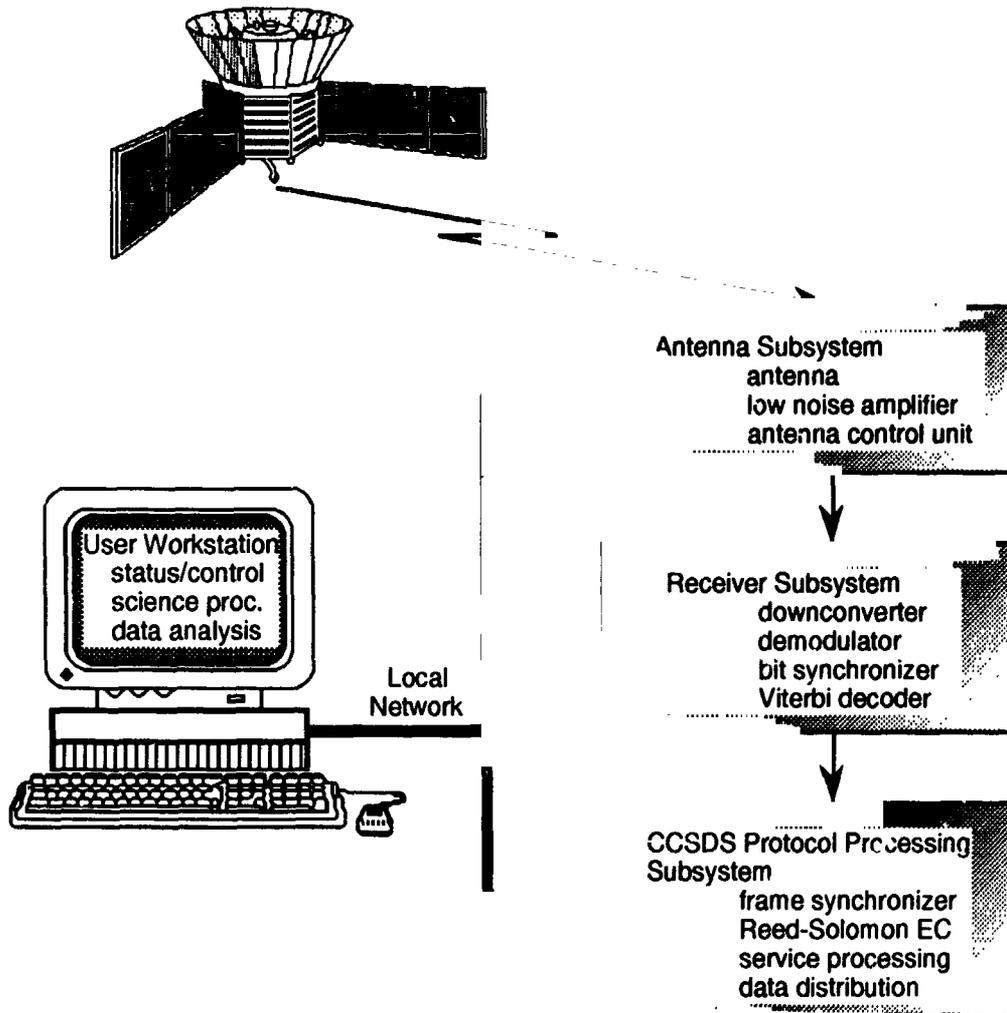


Figure 1. Subsystem Breakdown

The data rate for the EOS-AM spacecraft direct broadcast is approximately 13 Mbps. Based on current link budget calculations for this spacecraft, the minimum size dish antenna required to acquire this data will be about 2.5 meters. To keep the cost of the system low, the antenna will use programmed tracking. For transportability, the antenna will be easy to disassemble and store in a compact crate.

Future development will explore the use of phased array antennas to provide a smaller form factor, more reliable acquisition, and simpler setup and use. Initially a small, low complexity phased array antenna may be used to provide ephemeris data to the dish antenna. In the future, if sufficient levels of integration in RF processing components can be achieved at low cost, a phased array may be used to acquire science data. If this significant technical obstacle can be overcome, the use of a phased array antenna could present a superior method of acquiring science data. Phased arrays have a higher efficiency than a standard dish antenna and require no mechanical movement to track the spacecraft. Their planar structure also allows them more potential installation locations.

The RF processing equipment will down-convert the signal to Intermediate Frequency (IF) at which point the data is demodulated, bit synchronized, and error corrected using Viterbi decoding. The RF processing equipment outputs the digital data to the CCSDS protocol processing equipment.

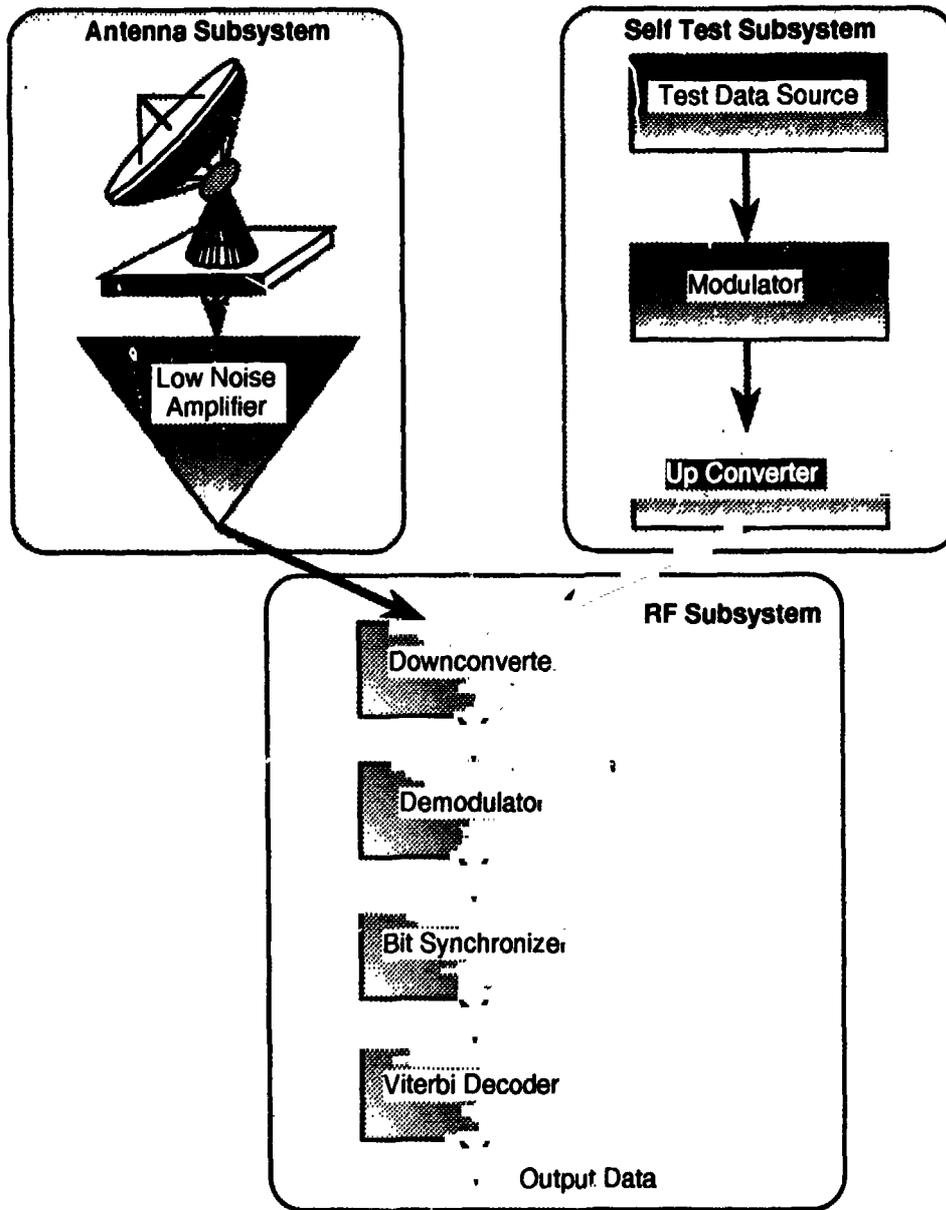


Figure 2. RF Processing Equipment Block Diagram

The CCSDS protocol processing equipment provides the following functionality: frame synchronization, Reed-Solomon error correction, CCSDS services processing and data routing to a user network. This equipment is a single printed circuit board containing three high performance ASICs to provide the major functionality of board. In addition, a high performance CPU is used to control the system and Peripheral Component Interface (PCI) bus slots are provided to allow the

user to expand the system with commercially available cards. The figure below shows a high level block diagram of the internal and external interfaces for the CCSDS Protocol Processor subsystem.

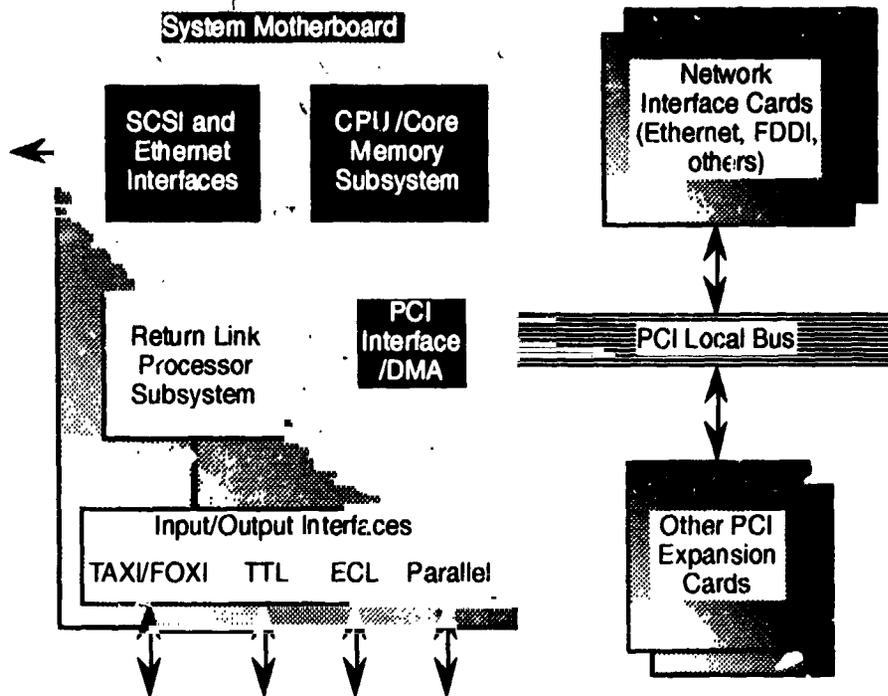


Figure 3. CCSDS Protocol Processing Subsystem

Figure 4 shows a more detailed block diagram of the functional blocks involved in CCSDS protocol processing. The data from the digital receiver subsystem is first synchronized using the frame sync, then corrected using Reed-Solomon error detection and correction. Finally, CCSDS service processing is performed on the synchronized frames. The data products can then be routed to a user network for further (Level 0 and up) processing.

The CCSDS Protocol Processing subsystem also accumulates information on the data quality and maintains statistics on such parameters as number of frames processed, data rate, number of packets processed, etc. Additionally, the subsystem has the ability to perform an automated self-test that can isolate a subsystem error to a functional block.

An important capability of the CCSDS Protocol Processing subsystem is the ability to accept commercially available PCI card for system expansion. A typical application may involve the installation of a high performance network card (e.g. ATM) into the system to allow interface to a user network. In this manner, the system is able to interface to many different network installations. This allows users to receive the data products and perform higher level processing on them locally. The CCSDS Protocol Processing subsystem has the ability to route data to a number of users based on specific criteria (SCID/VCID, APID, etc.) allowing it to handle multiple users requirements. Also note that a ethernet interface is included on the motherboard for lower rate data requirements and for system status and control functions.

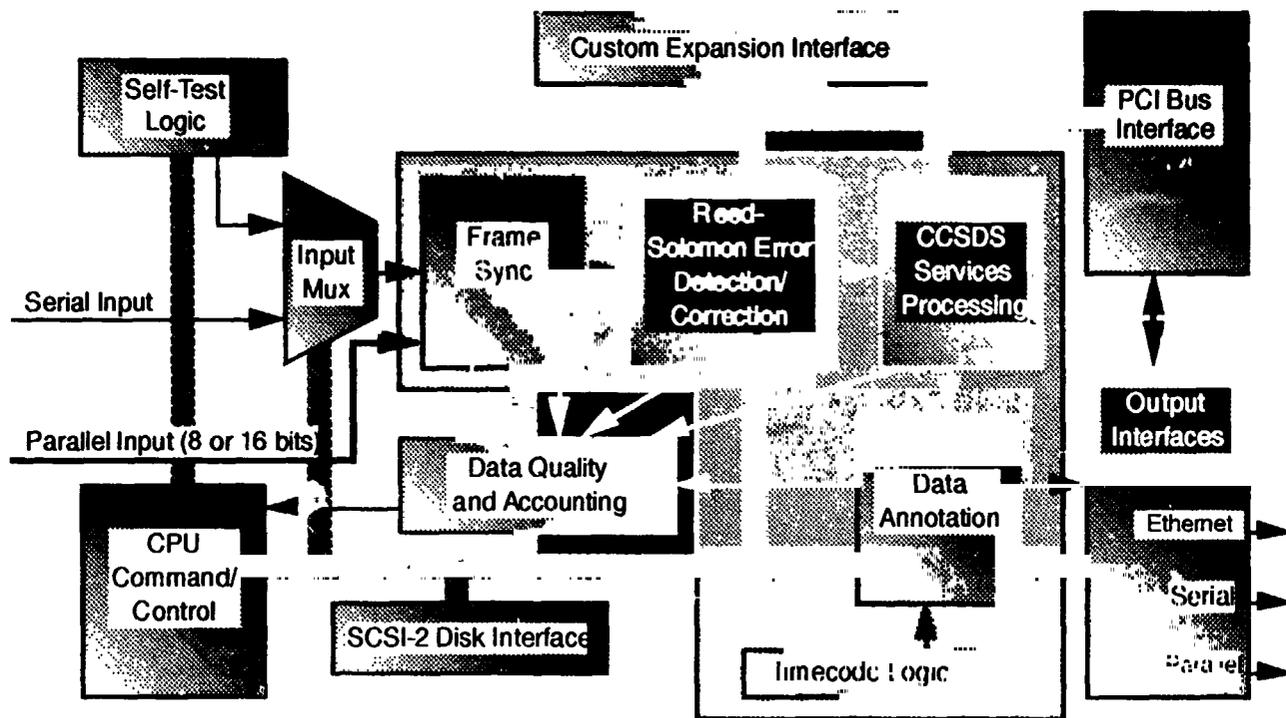


Figure 4. CCSDS Protocol Processing Subsystem
Detailed Block Diagram

The automated workstation-based operation and control software provides an interface to control the antenna, RF and CCSDS Protocol Processing system elements. It communicates with these elements through an ethernet port. The command and control software has a graphical user interface and on-line help to allow a non-expert (in data systems) to configure and operate the system elements with minimal training. Any number of users may be configured to receive data products and status from the CCSDS Protocol Processing subsystem.

3. SYSTEM DEVELOPMENT

Development of the system is already under way. The development plan calls for a three stage approach.

Stage 1 consists of a Commercial Off-the-Shelf (COTS) antenna, COTS Virtual Module Eurocard (VME)-based RF receiver, a highly integrated single motherboard CCSDS protocol processing subsystem and Version 1.0 of the command and control software running under VxWorks. This software will include limited science processing capabilities including level zero processing. Targeted release date is May 1995.

Stage 2 consists of a COTS antenna with a phased array antenna for generating ephemeris data, VLSI digital receiver, a highly integrated single motherboard CCSDS protocol processing subsystem and version 2.0 of the command and control software. This software will include expanded science processing capabilities including limited browse product generation. Targeted release date is January 1996.

Stage 3 will explore the use of a phased array antenna to acquire data while including all of the functionality of Stage 2. In addition, version 3.0 of the command and control software will include hardware support for accelerating higher level processing (levels 0 and 1) and refined software for generation of browse data products. The targeted release date is October 1996.

4. TYPICAL OPERATIONAL SCENARIO

The complete direct broadcast acquisition system, including antenna, will be small enough to fit into a conversion van for transportation. Once setup and operational, all equipment except the antenna can be contained in the same van.

Operationally, the system can easily be configured to handle different data formats and missions. Upon startup, the system performs an end to end self-test. This is done by generating a typical data stream and processing it through the system. System statistics are then accumulated and compared to the expected results and a dump of selected frames/packets are compared on a bit by bit basis to the expected results. Any differences are flagged and the user is notified of the error detected, the subsystems which are affected and the probable cause of the error.

After the system has passed self-test, it may be configured to process telemetry data. A configuration file that has been previously generated may be downloaded to the system or the user may generate a new configuration file. This file contains the high level mission information that the system needs to be aware of to recognize the format of the telemetry stream. These parameters include but are not limited to the frame sync pattern, the frame length, the type of error detection/correction used and the types of services to be performed. This high level information is translated by the operation and control software to low level hardware commands to correctly configure the system.

The system is now ready to process data. The antenna acquires a signal as the EOS spacecraft is passing overhead. The signal passes through a low noise amplifier and is then down-converted to an IF. At this point, the data is digitally sampled and the demodulation is done using digital processing algorithms. The bit synchronizer takes the output of the demodulator and produces a clock and serial data bitstream. Viterbi error correction then takes place and the data is sent to the frame synchronizer which recognizes the frame sync pattern and delimits the data based on a user defined acquisition strategy. The data is then optionally bit transition density decoded.

The delimited frames are then sent to the error detection and correction subsystem. This subsystem provides deinterleaving and block error correction using the Reed-Solomon code on either the entire frame only, the header only, or both.

After error detection and correction, the frames are sent to the CCSDS service processing subsystem which allows any of the various CCSDS specified services to be performed on the frames on a Virtual Channel Identifier (VCID) basis. These services include Virtual Channel Data Unit (VCDU) service, Insert service, VCA service, Bitstream service, Path packet service and Encapsulation service. These data products are then routed to the user over standard user network interfaces.

All quality and statistics information is accumulated by the system and provided to the workstation for display to the user. Various flags and limits may be set up which trigger system events. For example, if the system sees many frames which have uncorrectable data passing through the system, false synchronization could have occurred, in this case the system could be set to respond by forcing a back-to-sync signal that would allow the system to re-acquire sync and disregard the false sync.

5. SCIENCE PROCESSING

After the low-level CCSDS Processing (e.g. packet extraction) has been performed, this raw sensor and ancillary data must be transformed into working models of the complex whole-Earth systems called standard products. In addition, summary information about the data called browse products may be generated to allow the scientist determine which data may be of interest. The process of going from raw sensor data to browse products to integrated Earth models involves several levels of processing.

Level 0 processing involves reconstructing complete sensor scenes and engineering data from data packets, including packet resequencing and transmission error detection and correction. Level 1 processing involves radiometrically and geometrically calibrating the data due to atmospheric anomalies, sensor noise, and spacecraft attitude or orientation. Level 2 processing involves transforming the data into their intended sensor units (e.g., radar backscatter cross section, brightness, or temperature). Then, depending on the target application, the data are mapped into a set of scientifically meaningful features. This classification can be as simple as taking the ratio of two channels to quantify biomass or as complex as statistical classification of the spectral signatures to known features such as land use categories. Level 3 processing takes into consideration dynamic issues by mapping the data to a uniform space-time coordinate system. This often involves the interpolation of missing values due to orbital track characteristics and the mosaicking of multiple orbits. Processing at higher levels becomes highly application-specific involving various types of numerical code.

The EOS Direct Broadcast Acquisition System described in this paper will be capable of performing the various levels of science processing at limited data rates. In addition, it will be capable of producing summary metadata known as browse products. These browse products provide a low-resolution, low-accuracy view of the data to assist in determining which data may be of interest to the scientist. Giving an earth scientist the ability to acquire raw data directly and perform these data processing algorithms to generate browse products on a local machine will allow quicker data validation and refinement of models and simulations.

6. CONCLUSION

Low cost acquisition of EOS direct broadcast data is soon to be a reality. NASA Goddard Space Flight Center is using high-performance Complementary Metal Oxide Semiconductor (CMOS) VLSI circuitry and parallel processing algorithms to provide a transportable acquisition station at unprecedented levels of performance versus price. The integration of this station with intelligent software that allows a non-expert to configure the system and process data will allow not only a large community of earth scientists access to real-time science data but also extend the capability to a whole new class of users.

7. REFERENCES

1. Asrar, G., Dokken, D., "EOS Reference Handbook", 1993.

8. NOMENCLATURE

CCSDS	Consultative Committee for Space Data Systems
CMOS	Complementary Metal Oxide Semiconductor
COTS	Commercial Off-the-Shelf
EOS	Earth Observing System
GSFC	Goddard Space Flight Center
IF	Intermediate Frequency
NASA	National Aeronautics and Space Administration
PCI	Peripheral Component Interface
RF	Radio Frequency
VCID	Virtual Channel Identifier
VCA	Virtual Channel Access
VCDU	Virtual Channel Data Unit
VME	Versa Module Eurocard

APPLICATIONS OF MASSIVELY PARALLEL COMPUTERS IN TELEMETRY PROCESSING

Tarek A. El-Ghazawi*, Jim Pritchard**, and Gordon Knoble

*Department of Electrical Engineering and Computer Science
The George Washington University
Washington, D.C. 20052

**Information Processing Division, Code 560
Mission Operations and Data Systems Directorate
Goddard Space Flight Center
Greenbelt, MD 20771

ABSTRACT

Telemetry processing refers to the reconstruction of full resolution raw instrumentation data with artifacts, of space and ground recording and transmission, removed. Being the first processing phase of satellite data, this process is also referred to as level-zero processing.

This study is aimed at investigating the use of massively parallel computing technology in providing level-zero processing to spaceflights that adhere to the recommendations of the consultative committee on space data systems (CCSDS). The workload characteristics, of level-zero processing, are used to identify processing requirements in high-performance computing systems. An example of level-zero functions on a SIMD MPP, such as the MasPar, is discussed. The requirements in this paper are based in part on the Earth Observing System (EOS) Data and Operation System (EDOS).

1. Introduction

Telemetry processing refers to end-to-end delivery for satellite systems adhering to the Consultative Committee for Space Data Systems (CCSDS) recommendations. This involves link processing as well as production data processing. Return link processing

includes: Data Capture, real-time processing, playback processing, and rate buffering. On the other hand, production data handling includes production data processing, quick-look data processing, and level-0 backup archiving.

In return link processing: data capture refers to receiving all unprocessed telemetry data, including fill data, and storing it for a predetermined period of time for use in recovery processing. Real-time processing entails receiving and processing return link data of urgent nature, such as data pertaining to health, and delivering it to its sinks (earth ground system units) with minimal delay, as required. Playback processing, on the other hand, restores the "as recorded order" as data is originally stored on magnetic tape recording devices. This playback processing starts after the completion of one communication session during which telemetry data is received, a TDRSS session in EDOS. Finally, rate-buffering in which data from a spacecraft is received at one rate and transmitted to its final ground destination at another rate.

In production data handling, production data processing of CCSDS packets is the process in which packets from one or more communications sessions with the spacecraft, TSS in EDOS, are sorted by application process identifier (APID), quality checked, and forward ordered by sequence counter. Further, redundant and previously processed packets are deleted and a production data set is formed. Quick-look data processing is similar to production data processing except it is limited to packets from one communication session, one EDOS TSS. It could include all packets of that session or only those that have the same APID. Level-0 data archiving is for storing the production data sets created by the above processes.

Recent advances in high-performance computing have demonstrated that supercomputers based on massively parallel scalable architectures have the potential to offer a much higher performance/cost than traditional vector supercomputers. In fact machines with performance approaching 10 GFLOPS can be now purchased at less than \$1M. Combining all this with the flexibility offered by off-the-shelf general-purpose computers creates a great potential for the use of such technology in telemetry processing.

2. Telemetry Processing and Massively Parallel Computers

Some of the concerns that arise from suggesting a massively parallel high-performance computing architectures for telemetry processing are (1) can the massive hardware parallelism in such architectures be exploited in the processing, (2) can the I/O cope up with the tremendous data rates and adequately balance storage/retrieval and processing activities?, (3) what are the classes of parallel computers that best suite such kind of processing?, (4) is the cost reasonable?.

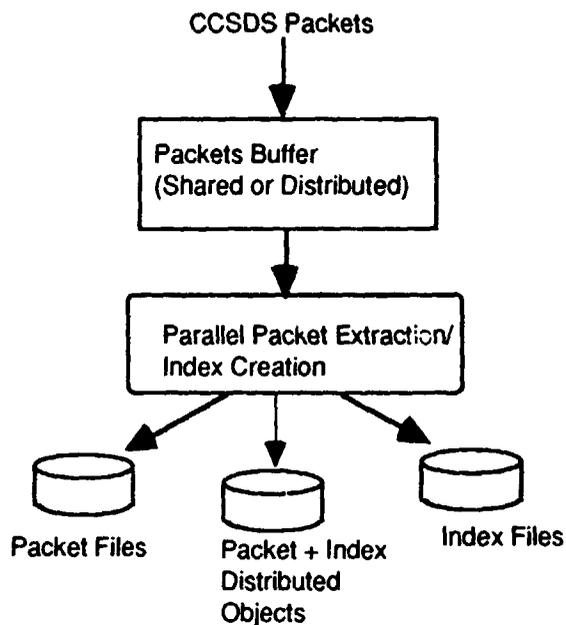


Figure 1

CCSDS Parallel Packet Extraction and Generation

Telemetry processing has inherent parallelism which is proportional to the rate of packet arrival. This is because such packets need to be indexed and the indices have to be kept sorted. Sorting the indices can be performed efficiently on such parallel machines when large volumes of data are involved, due to the large degree of data parallelism. This is exactly the case in near future telemetry processing systems. The rate of packet telemetry in such near future systems as the Earth Observing Systems (EOS) Data and Information Systems (EDOS) is projected at a range of 70K to a 120K packets per second [EDOS92C].

Modern massively parallel systems are designed with I/O scalability in mind. This basically means that as more processors participate in a parallel I/O operation, more I/O bandwidth become available. With the real-time nature and the massive data parallelism present in telemetry processing, parallel I/O can be coordinated to take full advantage of the scalability of the I/O systems. Sorting and storing indices are examples of scenarios that give rise to utilizing the power of the massive hardware parallelism and the scalability of the I/O systems. In such processing, similar operations needs to be performed on the data items in these large data sets. This massive data parallelism could be exploited efficiently with the hardware parallelism of massively parallel processing (MPP) computers. Since such data sets could be orders of magnitude larger than the available number of processors and their memory in an MPP, parallel I/O could be used to distribute data onto the

processors. Processing and I/O have to be coordinated in order to minimize I/O and amortize I/O overhead using largest possible data block sizes.

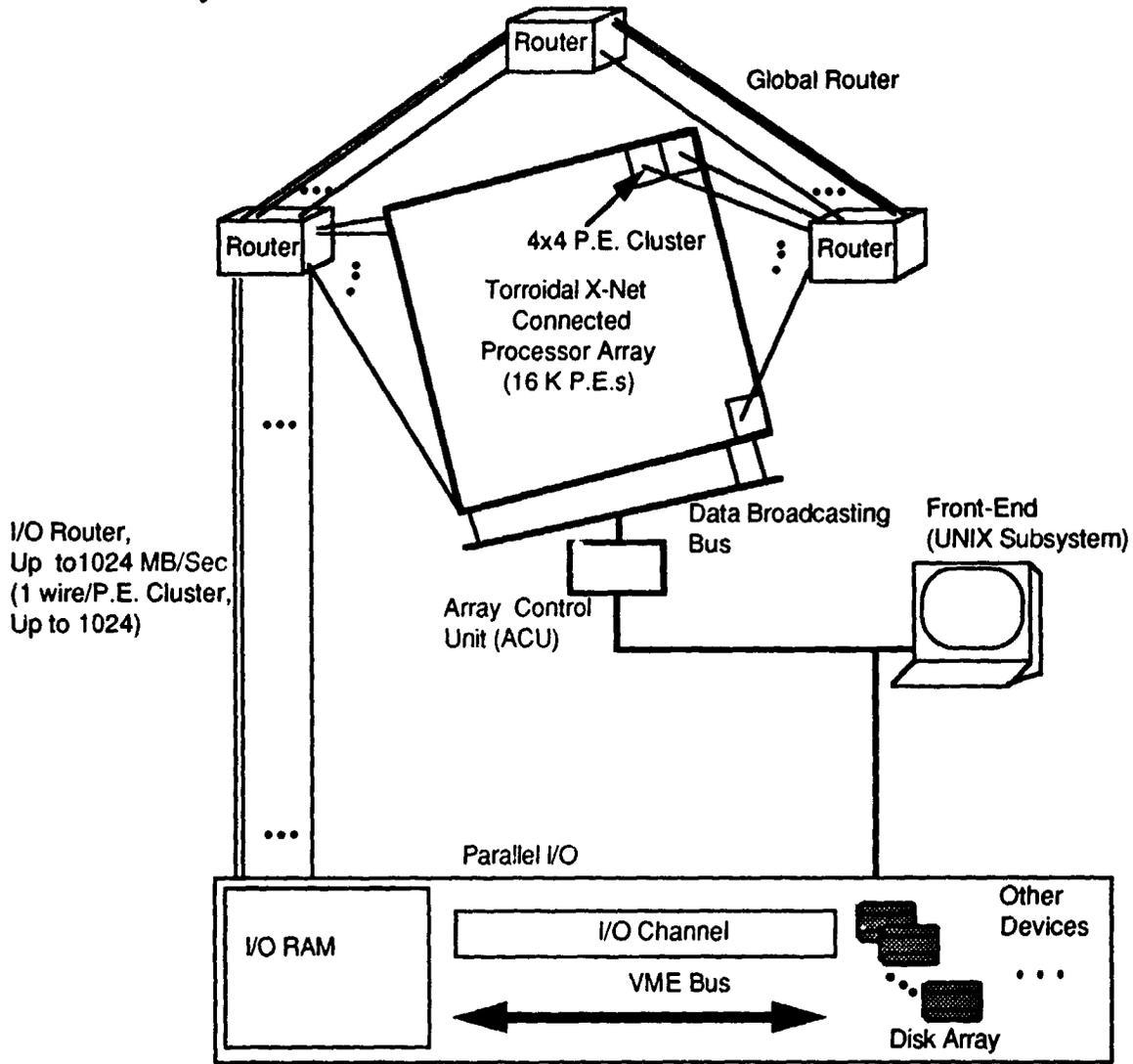
Figure 1 outlines some of the processing to be initially performed on arriving CCSDS packets. As a one load arrives, it will be partitioned among the systems processors through a parallel read. The exact size of a load depends on the specifics of the MPP system and the telemetry system parameters. All processors scan their data, searching for the first packet boundary in their respective data block. Thereafter, processing could jump over data extracting only index information from packet headers. Index information could be then added into index files, or used to design standardized objects which can contain both indices and packet data. Such object can be distributed in non RAID systems [Katz89], where the mass storage is distributed. The packet data could be gathered and stored at that point into packet files. In preparation for data production, indices must be kept sorted by APID and packet sequence numbers. This means that indices from one load need to be sorted, then merged with the previously sorted indices. Sorting and merging can be done efficiently with MPPs and many parallel algorithms for these operations already exist, [Akl85] and [Akl89]. It must be noted, however, that index files could be very large and hard to accommodate in processors memory. External sorting and/or caching of portions of such Redundant packets can be eliminated during indices sorting or at data sets productions. Doing so at sorting time could cost additional memory accesses or interprocessor communication steps for compacting the data.

3. A Case Study

3.1 A Case for SIMD Architectures

SIMD (Single Instruction Multiple Data) architectures are designed mainly to exploit data parallelism. In such class of architectures, all processors (also called processing elements or PEs) execute the same instruction synchronously and under the guidance of a centralized control unit. Due to the synchronous central control, such machines are very cost effective. With the real-time nature of telemetry processing and the massive data parallelism in this domain SIMD machines have the potential for delivering high performance/cost in telemetry processing. Thus, we are currently developing a telemetry processing architecture based on that technology and creating a scaled-down benchmark for such architecture using the MasPar SIMD MPP.

3.2 MasPar System Overview



The MasPar Architecture

Figure 2

MasPar Computer Corporation currently produces two families of massively parallel-processor computers, namely the MP-1 and the MP-2. Both systems are essentially similar, except that the second generation (MP-2) uses 32-bit RISC processors instead of the 4-bit processors used in MP-1. The MasPar MP-1 (MP-2) is a fine-grained, massively parallel computer with Single Instruction Multiple Data (SIMD) architecture. The MasPar has up to 16,384 parallel processing elements (PEs) arranged in a 128x128 array, operating under the control of a central array control unit (ACU), see figure 3. The processors are interconnected via the X-net into a 2-D mesh with diagonal and toroidal connections. In

addition, a multistage interconnection network called the global router (GR) uses circuit switching for fast point-to-point and permutation transactions between distant processors. A data broadcasting facility is also provided between the ACU and the PEs. Every 4x4 grid of PEs constitutes a cluster which shares a serial connection into the global router. Using these shared wires, array I/O is performed via the global router, which is directly connected to the I/O RAM as shown in figure 2. The number of these wires, thus, grows as the number of PEs to provide for scalable I/O bandwidth. Data is striped across the MasPar disk array (MPDA), which uses a RAID-3 configuration. For more information on the MasPar, the reader can consult the MasPar references cited at the end of this study [Bla90], [Mas92], [Nic90].

4.3 MPP Telemetry Processing on the MasPar

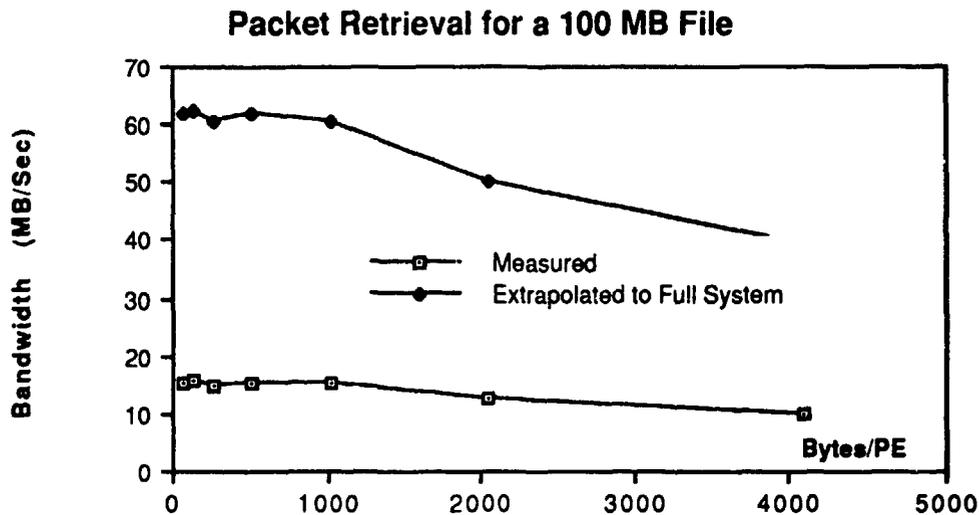


Figure 3

Retrieval Rates for Different Packet Block Sizes at a 100 MB Load

On the MasPar, blocks of packet data streams can be given to each processor. All processors can proceed simultaneously searching for the packet headers and forming the corresponding indices. Retrieving and storage of such packets data need to be performed at a sustained rate at least equal to the incoming data rate. At 70 packets per second and using the EDOS average of 819 bytes per packet [EDOS 92C], data can typically arrive at a 55MB/Sec rate. In EDOS, however, data speed is bottlenecked by ground communications and no more than 150 Mb/Sec arrival rate will be needed. The MasPar seems to be able of handling the I/O rates required by EDOS, but it could have some problems handling the 55

MB/Sec data rate. Measurements were collected on a MasPar system whose parallel disk I/O has a published sustained performance rate of 16 MB/Sec. A full-blown MasPar I/O system, however, has a published sustained performance of about 64 MB/Sec. Our measurements indicate that these published rates are achievable when I/O is amortized over sufficiently large files, more 100 MB, see figure 3. Using the same 819 B average packet and the 70K Packets/Sec rate, a 100 MB is accumulated in a little less than 2 sec.

5. Conclusions and Future Directions

This study supports the belief that SIMD MPPs could provide cost efficient solutions for today's telemetry processing. The initial results demonstrate that although at certain points such systems could not be completely adequate some customization could be done to satisfy the requirements, such as adding more disks or clustering more than one of these systems. We are currently proceeding with a benchmark that will demonstrate the know how and the performance constraints of using these SIMD architectures. Further, our future work will also include novel ways of indexing telemetry packets and applying temporal database concepts on parallel systems in general. Our work will also include performance comparisons of using such a SIMD machine versus the other classes of high-performance computer architectures.

References

- [Akl89] S. Akl, The design and Analysis of Parallel Algorithms. Prentice-Hall, 1989.
- [Akl85] S. Akl, Parallel Sorting Algorithms. Academic Press, 1985.
- [Biswas93] P. Biswas, K.K. Ramakrishnan, and D. Towsley. Trace Analysis of Write Caching Policies for Disks. Proc. of 1993 ACM SIGMETRICS, pages 13-23, 1993.
- [Brau89] C.K. Baru and O. Frieder, "Database Operations in a Cube -Connected Multicomputer System, "IEEE Transactions on Computers, 38(6), pp 920-927, June 1989.
- [EDOS92A] Earth Observing System (EOS) Data and Operations System (EDOS) Operations Concept. 560-EDOS-0106.0002. IPD, Code 560, NASA GSFC. Greenbelt, December 1992.
- [EDOS92B] Earth Observing System (EOS) Data and Operations System (EDOS) Functional and Performance Specifications. 560-EDOS-0202.0004. IPD, Code 560, NASA GSFC. Greenbelt, November 1992.

- [EDOS92C] Earth Observing System (EOS) Data and Operations System (EDOS) Traffic Model. 560-EDOS-0106.0002. IPD, Code 560, NASA GSFC. Greenbelt, Dec. 1992.
- [El-Ghazawi94B] Tarek A. El-Ghazawi. "I/O Performance of the MasPar MP-1 Testbed". TR-94-111. CESDIS, Code 930.5, NASA-GSFC, January 1994.
- [Frieder90] O. Frieder, Multiprocessor Algorithms for Relational Databases Operators on Hypercube Systems, "IEEE Computer, 23(11), November 1990.
- [Hwang93] K. Hwang. Advanced Computer Architecture: Parallelism, Scalability, and Programmability. McGraw-Hill, 1993.
- [Johnson 93] Theodore Johnson, "Supporting Insertions and Deletions in Striped Parallel File Systems,"7th IPPS, April 1993.
- [Katz89] R. H. Katz, G. A. Gibson, and D. A. Patterson, "Disk System Architecture for High Performance Computing", Proc. of the IEEE, Dec.1989, 1842-1858.
- [Kotz91] D. Kotz and C. Eillis, "Practical Prefetching Techniques for Parallel File Systems". Proceedings of the IEEE DCS'91.
- [Miller 91] E. L. Miller and R. H. Katz, "Input/Output Behavior of Supercomputing Applications", Proc. of Supercomputing'91, 567-576.
- [Patterson92] R. Patterson, G. Gibson, M. Stayanarayanan, "Using Transparent Informed Prefetching (TIP) to reduce File Read Latency", Goddard Conference on Mass Storage Technologies, September 1992.
- [Staelin88] C. Staelin. File Access Patterns. Tech. Rep. CS-TR-179-88, Department of Computer Science. Princeton University, Sept. 1988.
- [Tansel93] Tansel, Clifford, Gadia, Jajodia, and Seguv. Temporal Databases: Theory, Design, and Implementation. Benjamin/Cummings, 1993.

TEST TELEMETRY AND COMMAND SYSTEM (TTACS)

Alvin J. Fogel
 Jet Propulsion Laboratory
 4800 Oak Grove Drive, M/S 525-3970
 Pasadena, California 91109-8099

Abstract - The Jet Propulsion Laboratory has developed a multimission Test Telemetry and Command System (TTACS) which provides a multimission telemetry and command data system in a spacecraft test environment. TTACS reuses, in the spacecraft test environment, components of the same data system used for flight operations; no new software is developed for the spacecraft test environment. Additionally, the TTACS is transportable to any spacecraft test site, including the launch site. The TTACS currently supports projects at the Jet Propulsion Laboratory involved in the unmanned exploration of deep space. The TTACS is currently operational in the Galileo spacecraft testbed; it is also being provided to support the Cassini and Mars Surveyor Program projects.

TTACS usage results in lower cost planetary missions since no new software is developed for the spacecraft test environment. Also, minimal personnel data system training is required in the transition from pre-launch spacecraft test to post-launch flight operations since test personnel are already familiar with the data system's operation. Additionally, data system components, e.g. data display, can be reused to support spacecraft software development; and the same data system components will again be reused during the spacecraft integration and system test phases. TTACS usage also results in early availability of spacecraft data to data system development and, as a result, early data system development feedback to spacecraft system developers.

The TTACS consists of a multimission spacecraft support equipment interface and components of the multimission telemetry and command software adapted for a specific project. The TTACS interfaces to the spacecraft, e.g., Command Data System (CDS), support equipment. The TTACS telemetry interface to the CDS support equipment performs serial (RS-422)-to-ethernet conversion at rates between 1 bps and 1 mbps, telemetry data blocking and header generation, guaranteed data transmission to the telemetry data system, and graphical downlink routing summary and control. The TTACS command interface to the CDS support equipment is nominally a command file transferred in non-real-time via ethernet. The CDS support equipment is responsible for metering the commands to the CDS; additionally for Galileo, TTACS includes a real-time-interface to the CDS support equipment.

The TTACS provides the basic functionality of the multimission telemetry and command data system used during flight operations. TTACS telemetry capabilities include frame synchronization, Reed-Solomon decoding, packet extraction and channelization, and data storage/queue. Multimission data display capabilities are also available. TTACS command capabilities include command generation, verification, and storage.

TTACS CAPABILITIES

TTACS provides the operational interface between the spacecraft support equipment and the end-user workstation in the spacecraft system test environment. The TTACS, together with the end-user workstation, provide the primary system test visibility. Figure 1 describes TTACS functions in the context of the spacecraft testbed data flow. For spacecraft subsystem test, TTACS has limited functionality; subsystem personnel determine which TTACS functions are appropriate for that subsystem test.

A basic supporting element of the TTACS concept is the existence of multimission software which

can be adapted as necessary for each project. This allows the early availability of ground data system (GDS) software in support of spacecraft development, which in turn supports early availability of spacecraft data for ground system development. The multimission infrastructure which supports each project's TTACS is provided through the Multimission Operations System Office (MOSO). MOSO was established at JPL to provide those operational capabilities, services, and tools that are common to all missions/projects thereby realizing a cost avoidance to the projects and a cost savings to NASA HQ.

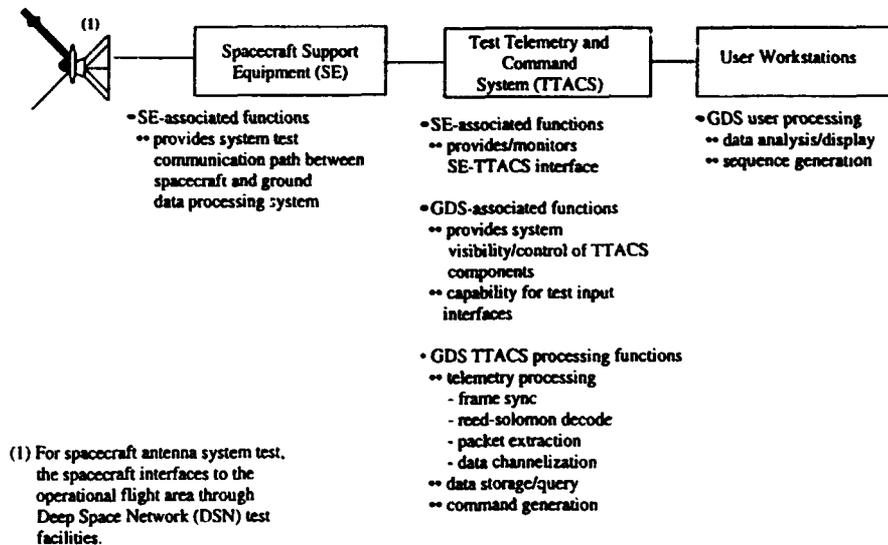


Figure 1 - TTACS Functions Within the Spacecraft Testbed Data Flow

TTACS COST BENEFIT SUMMARY

TTACS results in lower cost planetary missions as summarized below:

- Minimize software costs through software reuse
 - TTACS based on reusable multimission infrastructure
 - TTACS reuses project flight operations software to support spacecraft system test and spacecraft simulator development/operations
 - TTACS reuses multimission SE-TTACS interface
- Enhance spacecraft-GDS system test through early/continued use of GDS in spacecraft development/test
 - GDS test with actual spacecraft data
 - GDS test in operational environment
 - Spacecraft test in GDS operational environment
 - Spacecraft subsystem test use of multimission display software
- Minimize post-launch ground system training by using ground system pre-launch in spacecraft system test

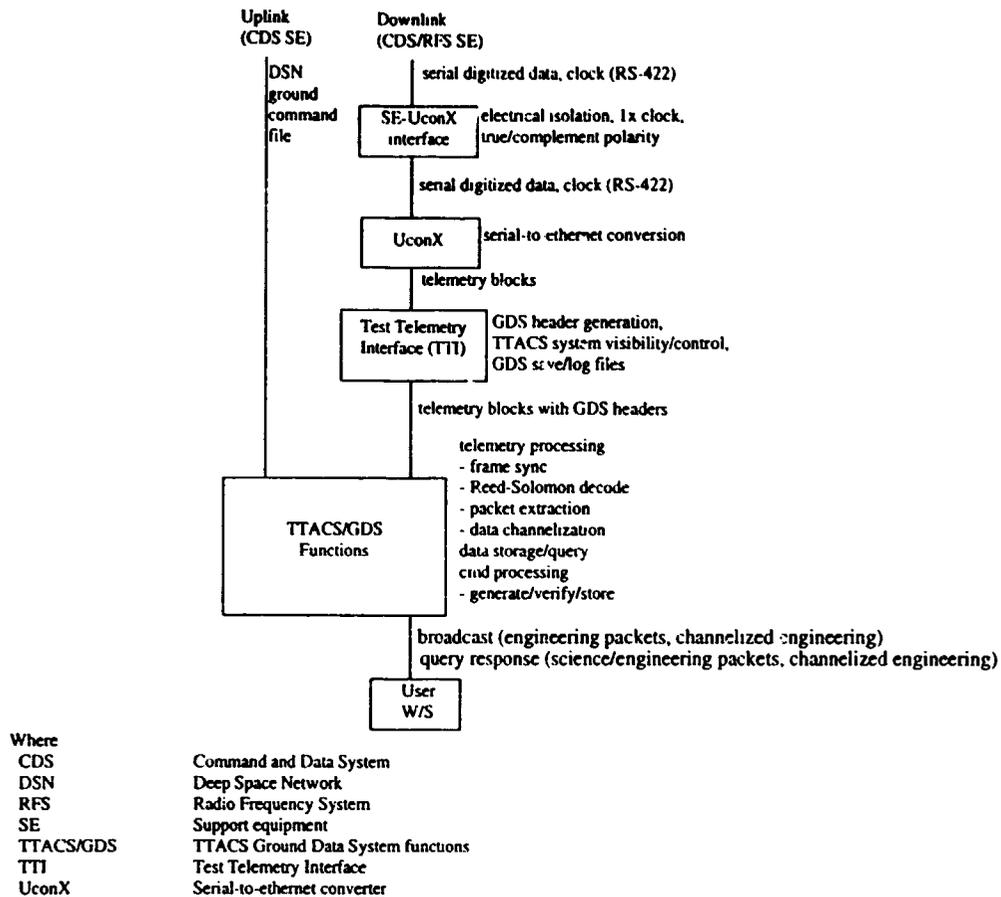
- Support the extension of commercial, low-cost products, e.g., UconX Communique¹; share functionality growth of those products as featured by other customers.
- Maximize support of testbed/launch sites by enhancing transportability, e.g., TTACS is sized to fit customer-specific needs

TTACS ARCHITECTURE/COMPONENT DETAILED DESCRIPTION

The TTACS detailed data flow/component description is presented in Figure 2. Key points include:

- TTACS interfaces only with spacecraft support equipment, not spacecraft (which has extensive interface requirements).
- TTACS downlink processing includes a support equipment - TTACS interface which optionally can provide electrical isolation, biphasic clock reduction, and true/complement polarity.
- TTACS downlink processing also includes a serial-to-ethernet conversion function. This function resides in the UconX Communique, a multi-protocol communication server for LANs. UconX developed, in coordination with JPL, a Synchronous Bit Stream Interface (SBSI) protocol to process the RS-422 NRZ-L stream.
- TTACS uplink processing provides DSN ground command files to spacecraft support equipment. The support equipment is responsible for extracting the included command bits and metering those bits to the spacecraft.
- The opportunity also exists for TTACS to transmit only command bits to the spacecraft support equipment. This architecture would utilize the newly developed transmit protocol.

¹UconX is a trademark of UconX Corporation, San Diego, CA



*Figure 2 - TTACS Detailed Data Flow/
Component Description*

The TTACS generic testbed architecture is presented in Figure 3. Key points include:

- The ground test configuration which supports spacecraft system test is interconnected via a test LAN. A router provides external security/connectivity for the ground test configuration.
- TTACS provides computer clock synchronization via Network Time Protocol (NTP) on the test LAN. The master NTP server resides on the TTI node and utilizes a DATUM² Time Code Translator (TCT). DATUM support is now a standard feature of the freely available NTP code; JPL provided the DATUM support code to the NTP maintainers.
- The UconX Communique is configured to provide four serial (RS-422) input ports; serial-to-ethernet conversion may be performed concurrently on any of these input ports. The SE-UconX interface also provides four input/output ports to match the UconX capability.

²DATUM is a trademark of DATUM INC., Anaheim, CA

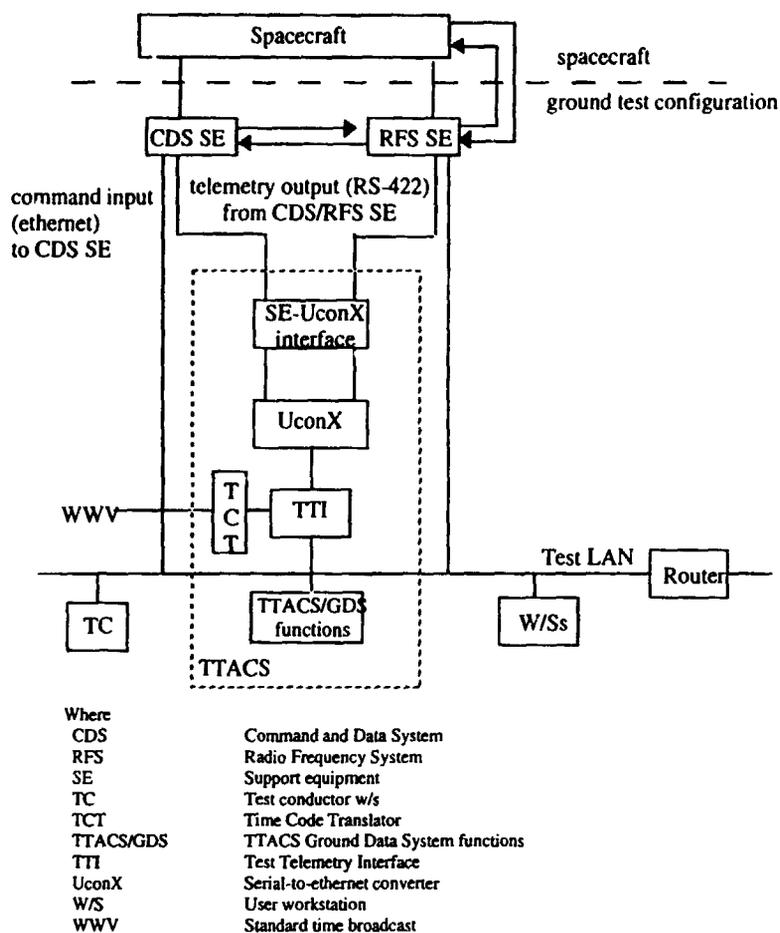


Figure 3 - Spacecraft Generic Testbed Configuration

PROJECT-SPECIFIC TTACS EXAMPLES

Projects may implement TTACS in its entirety, (i.e., SE hardware interface, GDS compatibility/visibility/control, TTACS GDS functions), or a subset of TTACS functionality. Projects which currently plan/implement the total TTACS functionality include Cassini, Galileo, and Mars Global Surveyor. Projects which currently plan/implement partial TTACS functionality include High-Speed Spacecraft Simulation, Mars Pathfinder, and Disaster Recovery Facility (DRF).

Cassini plans to use TTACS for both spacecraft integration and spacecraft system test. Key points follow:

- Figure 3 is the architecture for Cassini spacecraft system test. The TTACS/GDS component of Figure 3 becomes two computers to process the high input data rate; one computer performs telemetry/command processing, the other performs data load/query.
- Cassini TTACS will process test rates, i.e., up to 249 kbps, which are higher than flight rates, i.e., up to 140 kbps.

Galileo has implemented TTACS in their spacecraft testbed which is used for spacecraft integration and sequence checkout. Key points follow:

- Figure 4 is the architecture for the Galileo TTACS. This architecture accommodates the Galileo testbed which predates TTACS, e.g., multiple concurrent spacecraft outputs, pre-existing support equipment configuration.
- Galileo TTACS provides multiple (1 data, 2 test) concurrent telemetry data streams (RS-422) from spacecraft support equipment to the end-user.
- Galileo TTACS provides real-time (RS-232) DSN ground command file transmission to the spacecraft support equipment which, in turn, meters the commands to the spacecraft.
- Galileo TTACS processes test rates, i.e., up to 134 kbps, which are much higher than flight rates, i.e., up to 160 bps.

Mars Global Surveyor plans to use TTACS; details will be a function of spacecraft contractor discussions. TTACS downlink capability has been demonstrated using the Mars Observer Verification Test Lab (VTL).

The High-Speed Spacecraft Simulation project has implemented TTACS partial functionality to support the development and operation of its spacecraft bit simulation. Multimission simulations are planned; the Galileo simulation is currently being implemented. Key points follow:

- Figure 5 is the architecture for the High-Speed Spacecraft Simulation TTACS. Since the output of the simulation is a TCP/IP stream on an ethernet LAN, the TTI test input interface is used instead of the UconX (RS-422) interface.
- The High-Speed Spacecraft Simulation TTACS processes simulated rates higher than flight rates. The current Galileo simulation uses one computer as both TTACS and its user workstation.

The Mars Pathfinder project plans to implement partial TTACS capability. Key points follow:

- Figure 6 is the architecture for the Mars Pathfinder TTACS. The project will closely integrate the support equipment and the GDS via a TCP/IP stream on an ethernet LAN. Additionally, the SE will generate GDS headers for output telemetry blocks. In this architecture, the SE replaces the UconX/TTI function.
- The AIM SE is the single TTACS interface, i.e., RFS SE downlink is provided to TTACS via the AIM SE.

The Disaster Recovery Facility (DRF) project plans to implement partial TTACS capability for a disaster recovery operational facility which supports all projects. Key points follow:

- Figure 7 is the architecture for the Disaster Recovery Facility TTACS. The facility will interface to the Deep Space Network (DSN) via an IP endpoint; additionally, DSNI must process DSN telemetry/command protocols. In this architecture, DSNI replaces the UconX/TTI function.
- The Disaster Recovery Facility will support one spacecraft at a time, performing housekeeping/safing via real-time low rate (1200 bps maximum) telemetry reception and smaller command file transmission. Science data is processed after JPL comes back on-line (six months maximum) using DSN station recordings.

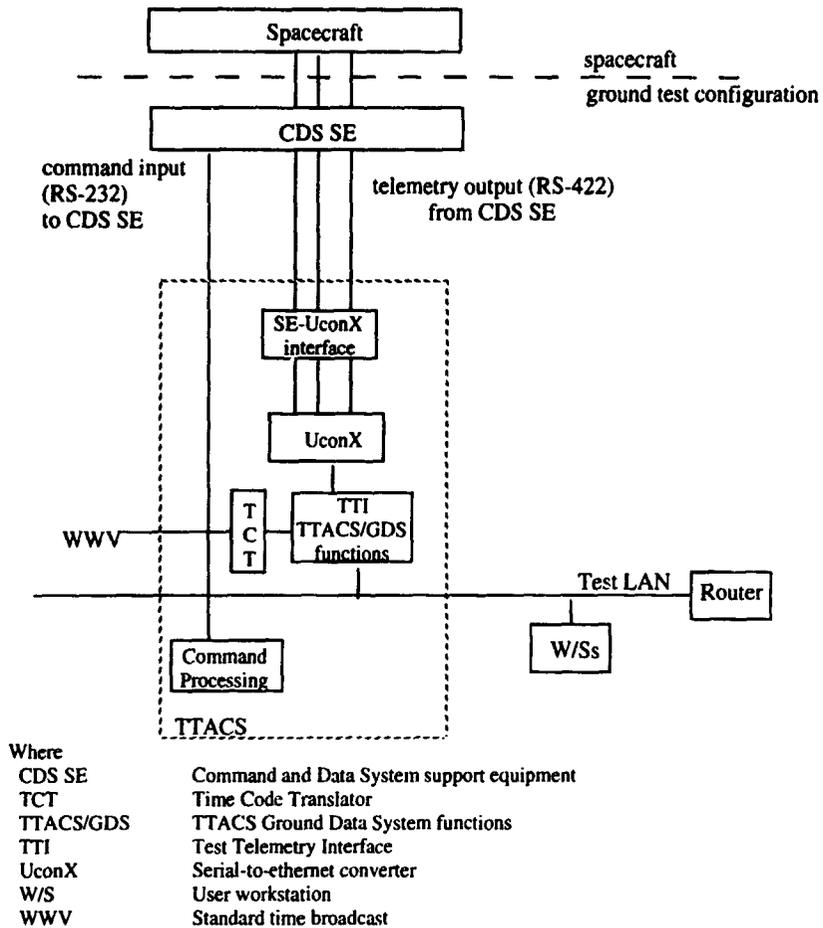


Figure 4 - Galileo Spacecraft Testbed Configuration

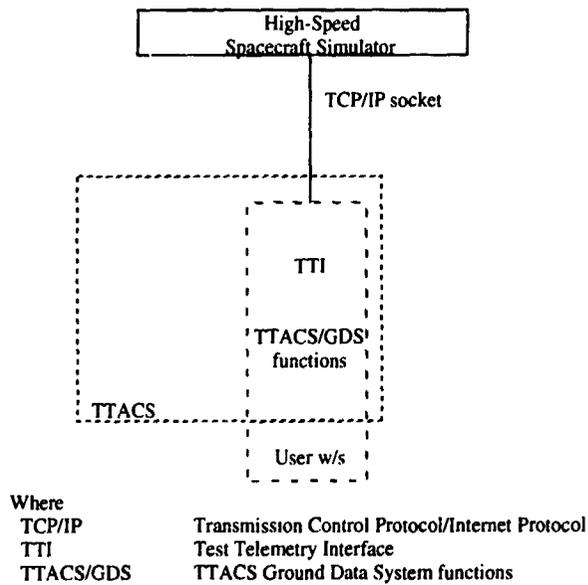
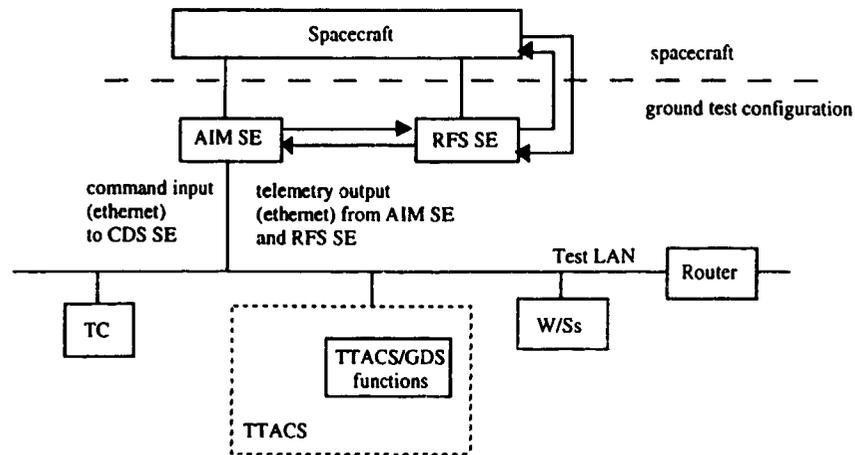
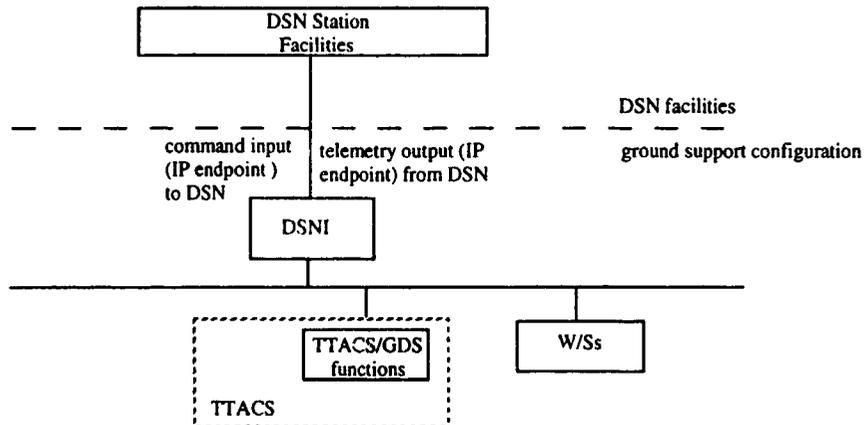


Figure 5 - High-Speed Spacecraft Simulation Test Configuration



Where
 AIM Attitude Information System
 RFS Radio Frequency System
 SE Support equipment
 TC Test conductor w/s
 TTACS/GDS TTACS Ground Data System functions
 W/S User workstation

Figure 6 - Mars Pathfinder Spacecraft Testbed Configuration



Where
 DSN Deep Space Network
 DSNI Deep Space Network Interface
 IP Internet Protocol
 TTACS/GDS TTACS Ground Data System functions
 W/S User workstation

Figure 7 - Disaster Recovery Facility (DRF) Operational Configuration

31-5

1-12

Ground Equipment for the Support of Packet Telemetry and Telecommand

Wolfgang Hell

Abstract - This paper describes ground equipment for packet telemetry and telecommand which has been recently developed by industry for the European Space Agency (ESA). The architectural concept for this type of equipment is outlined and the actual implementation is presented. Focus is put on issues related to cross support and telecommand as far as they affect the design of the interfaces to the users of the services provided by the equipment and to the management entities in charge of equipment control and monitoring.

Introduction

This paper describes the telemetry and telecommand sub-systems which have recently been developed by European industry on behalf of the European Space Agency (ESA) and are presently being deployed to the Agency's ground station network. On the one hand, the design of these subsystems has been driven by ESA's Packet Telemetry and Telecommand standards (*PSS-04-106, PSS-04-107*) which in turn have been derived from the related "Blue Books" produced by Panel 1 of the Consultative Committee for Space Data Systems (CCSDS) (*CCSDS 102, CCSDS 201, CCSDS 202, CCSDS 203*). These standards in essence determine the functionality to be provided by the sub-systems. On the other hand, although final results are not yet available, also Panel 3 activities aiming at a standardisation of the services made available by ground segment entities have been taken into account. Specifically in the design of the subsystem interfaces care has been taken to cleanly separate services accessible to users from management issues. This approach and usage of the full OSI protocol suite will facilitate cross support between space flight agencies. In order to ensure appropriate growth potential and life time of the architecture, the design took also into account CCSDS work on Advanced Orbiting Systems (AOS) (*CCSDS 701*).

From this comprehensive set of requirements initially an "ideal" architecture has been derived which subsequently has been modified to accommodate constraints in terms of available hardware and software as well as cost.

The "Back-end" Architectural Concept

In ESA terminology, the Back-end of a ground station encompasses all equipment connecting the intermediate frequency equipment of the front-end to the ground communication network in order to provide the remote users (normally control centres) with the services required to operate the spacecraft and to acquire payload data. Set-up and monitoring of the back-end subsystems is done via a "management" interface. Figure 1 presents the back-end's context.

Wolfgang Hell is member of the Technology Development Division in the Stations and Communications Engineering Department of the European Space Operations Centre, Robert-Bosch-Str. 5, D-64293 Darmstadt, Germany

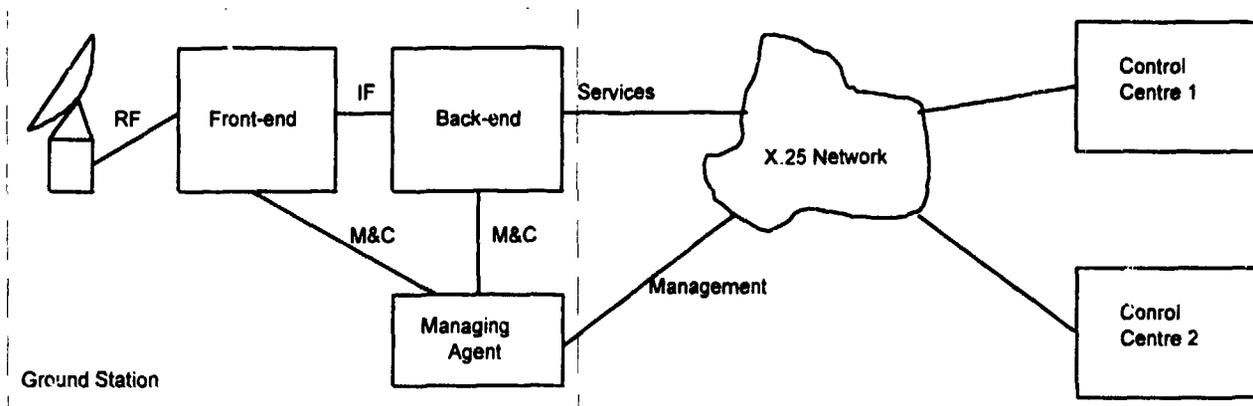


Figure 1: Back-end Context

On the return link, the back-end receives the demodulated symbol stream from the front-end which is either immediately processed and forwarded to the user (on-line service) or stored in the back-end and forwarded later on user request (off-line). Regarding the forward link, the back-end receives data from the user, i.e. the control centre, via the communication network and, depending on the user request, either forwards them immediately to the front-end (on-line) or stores them for uplink at the specified time (off-line service). Conventional telecommanding can be considered as a special type of on-line forward link service.

The back-end shall support the space data standards for spacecraft operation listed below:

- PCM Telemetry and Telecommand (*PSS-46, PSS-45*),
- Packet Telemetry and Telecommand (*PSS-04-106, PSS-04-107*),
- a subset of AOS (*CCSDS 701*).

A prime design objective has been to relieve the users from the need to be aware of configuration details which are internal to the back-end. In other words, the user should only need to know about the service requested, but not how the back-end manages to provide this service and how the required availability figure (e.g. by means of redundancy) is attained. These design goals have led to the internal back-end structure depicted in figure 2.

The various Functional Units (FU) presented in figure 2 are physically interconnected by means of a dual ring (i.e. redundant) FDDI (Fibre Distributed Data Interface) Local Area Network which allows to set up so-called Functional Processing Chains (FPC) by establishing virtual circuits between the Functional Units as required for the provision of the service enabled by station management. For the Storage FU no redundancy is shown, as it is assumed that the required availability and performance for concurrent provision of multiple service instances is attained by means of internal redundancy (e.g. a RAID (Redundant Array of Inexpensive Disks) architecture). FUs drawn with dashed lines are not part of the initial implementation presented below, elements drawn with dash-dotted lines are only partly implemented. The concept depicted in figure 2 provides for a high degree of scalability in terms of attainable throughput (limited by the FDDI bandwidth) and availability. Typical station integration

problems where a high level of redundancy entails complex cabling and switching units which in turn have a negative effect on the actual availability are avoided.

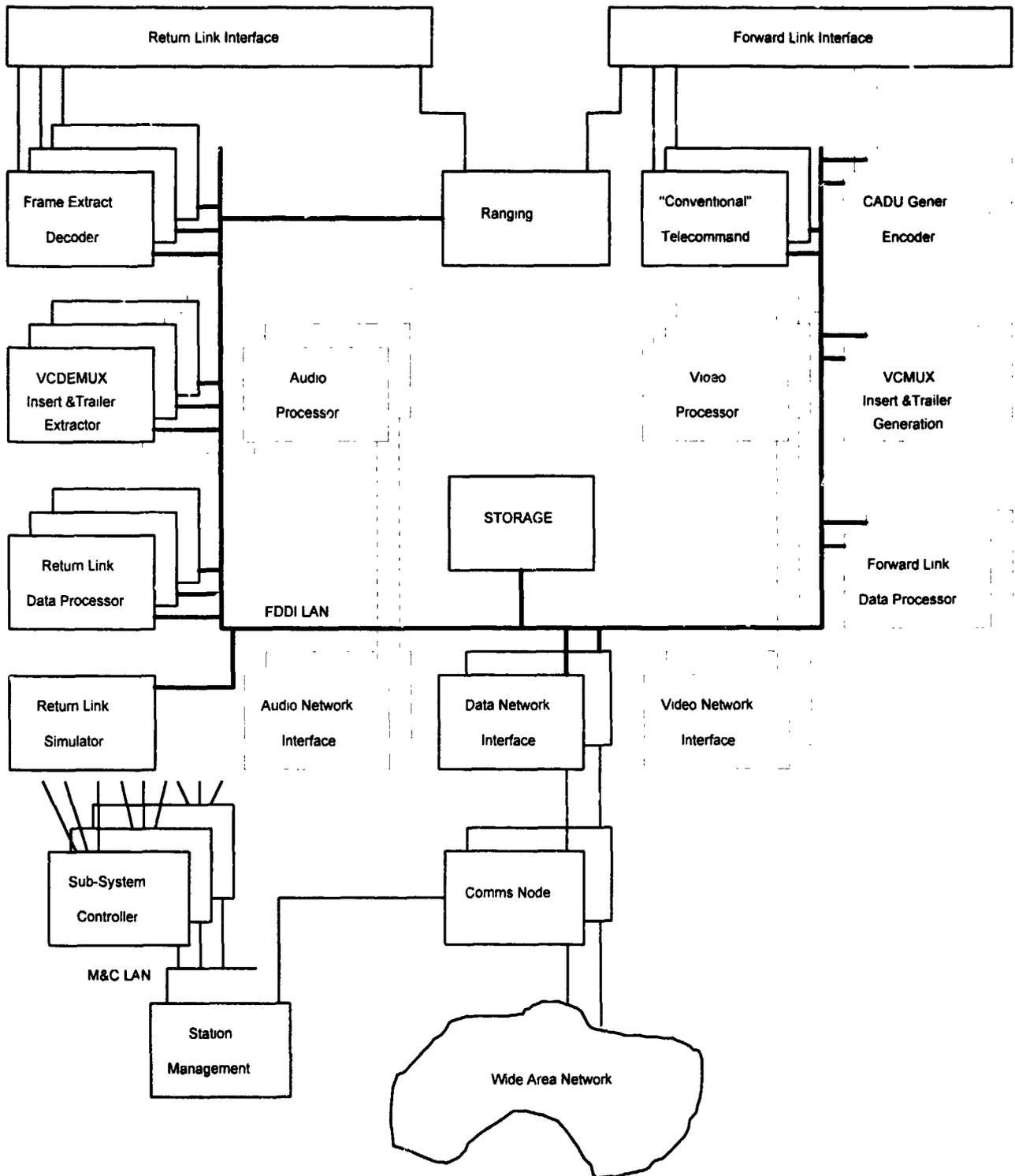


Figure 2: Back-end Architectural Concept

Actual implementation

Unfortunately, in practice the above outlined concept had to be somewhat diluted since the actual implementation has been constrained by the availability of suitable hardware (and software) and cost. Furthermore, for some interfaces compatibility with existing installations had to be ensured such that the newly developed equipment could serve as an in situ replacement of existing, but obsolescent equipment. Figure 3 shows the resulting block diagram. It is intended to upgrade the subsystems in an evolutionary process to the concept outlined in figure 2.

As regards the return link handling part, the actual implementation does not (yet) support audio and video as defined for the AOS environment (components drawn with dashed lines in figure 2). For reasons of interface compatibility with existing stations, the Frame Extractor/Decoder (FED) functional units do not yet support the FDDI interface, but deliver the synchronised and decoded transfer frames to the VCDEMUX units via IEEE-488 point-to-point connections. Similarly, the VCDEMUX units provide the extracted Command Link Control Word (CLCW) messages via dedicated serial links rather than the FDDI LAN to the telecommand encoders. The Storage system had to be split into individual storage processors as a dual port RAID architecture was found to be by far too costly for the time being. This limitation unfortunately implies that in order to achieve redundant storage of telemetry, the data have to be duplicated by the VCDEMUX and sent twice through the FDDI network since the protocol (see below) does not allow "broadcasting".

The forward link handling part has been confined to the "conventional" (i.e. PCM and packet) telecommand function. The AOS type of forward link is not (yet) supported. A limited capability to generate and encode CLTUs for the forward link is (for test purposes only) available in form of the return link simulator and the encoding capability of the FED units. Therefore, the CADU (Channel Access Data Unit) Generator/Encoder units are drawn in figure 2 with dash-dotted lines.

Although the FDDI network provides for 100 Mbps bandwidth, the sustained throughput on a virtual circuit connecting two functional units was found to be less than 6.4 Mbps even with large block sizes. This is in part due to the fact that off-the-shelf only TCP/IP as protocol is available. This protocol is designed for a very unreliable network service and therefore introduces considerable overhead which in an FDDI environment is superfluous. The problem is further aggravated by the fact that the bit manipulations required for the TCP/IP error control field calculation are carried out on the CPUs of the connected units. From the introduction of a suitable "light weight" protocol such as XTP one can expect a substantial throughput improvement, in particular when in addition to point-to-point connections "broadcasting" is supported.

The Return Link Protocol Handling System (R-PHS)

Since the R-PHS must be usable as an in situ replacement for the presently deployed telemetry system, it must emulate the existing equipment as regards the PCM telemetry standard in order to ensure continuing support to on-going missions without any impact on the control centre system. For PCM

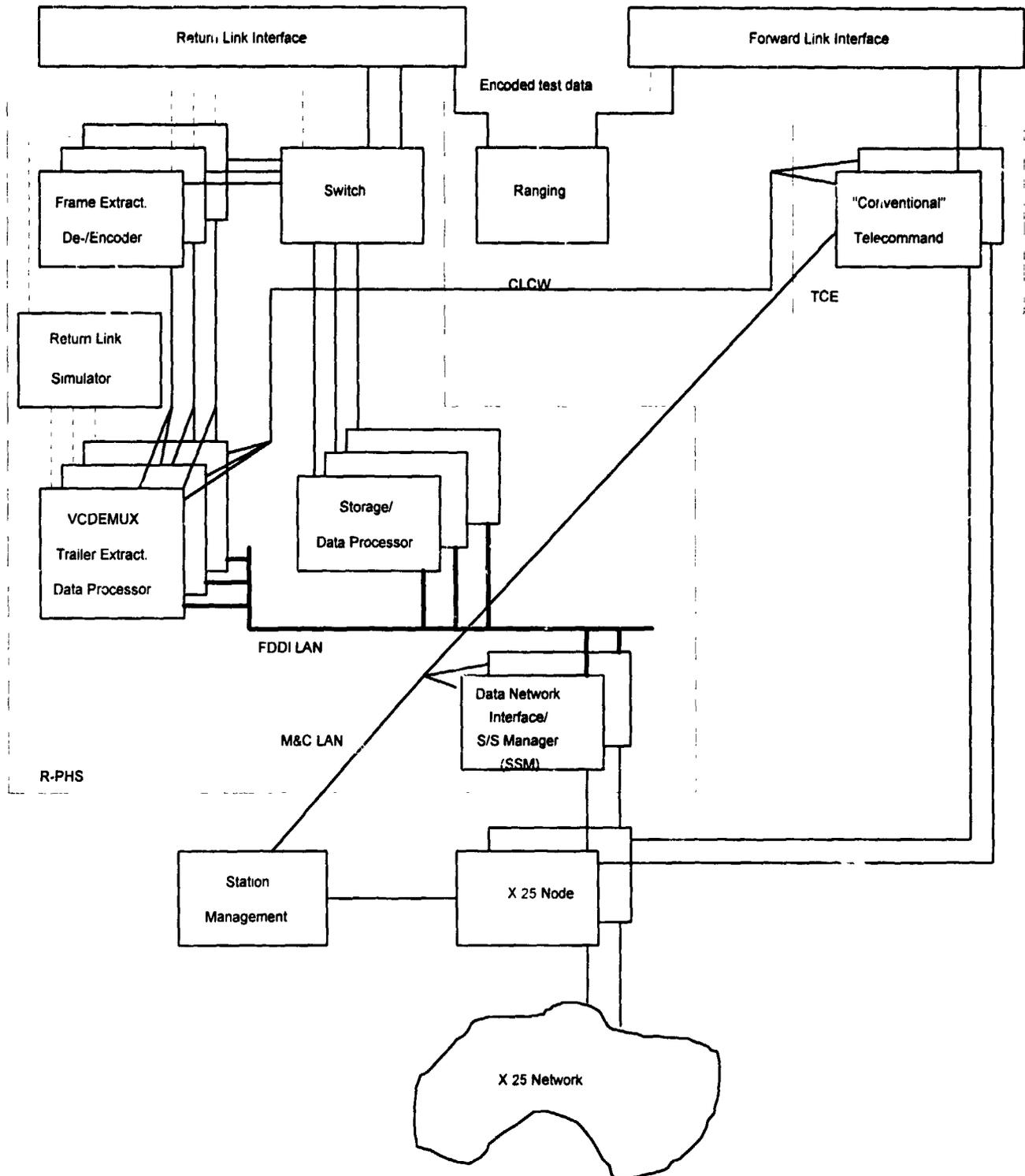


Figure 3: Back-end Implementation

mode, the R-PHS thus supports the private "ESA Message Protocol" built directly on top of the X.25 network layer and provides both on-line and off-line telemetry data delivery.

For the services related to packet and AOS telemetry, a prime objective of the project has been to develop a system which due to the services and protocols provided facilitates telescience and cross support. Application layer (i.e. layer 7 in the OSI reference model) protocols such as FTAM and CMIP will give the least problems in terms of interoperability between the R-PHS and the user which will generally require inter-operating of computer systems from different vendors. However, in most cases the telemetry system is connected to the user via an X.25 network which provides limited bandwidth only. Therefore, the efficiency of the bandwidth usage is a critical issue in particular for the transfer of (high-volume) payload telemetry, as long as high-speed wide area network technology like Frame Relay or ATM (Asynchronous Transfer Mode) are not yet widely available. For the R-PHS the link to the user has been split into a control channel, on which the layer 7 Common Management Information Protocol (CMIP) is used, and a data channel for bulk data transfer, on which for efficiency reasons the OSI protocol suite has been limited to session layer.

Before a user can connect to the R-PHS, the system must be configured via the Sub-System Manager (SSM) such that the Functional Processing Chains providing the services to be granted to the user are established by connecting the various functional units in the appropriate way. The correct functioning of the established chains is checked by means of built-in test facilities and, in case a functional unit is found to be faulty, alternative chains excluding the defective unit are set up by the SSM. The Data Network Interface units are notified of the final set-up in terms of service providers and user access rights such that incoming requests can be validated and routed to the unit providing the service requested by the user. In this way, the user is relieved from the need to be aware of the internal R-PHS set-up. The R-PHS appears to the user as a telemetry server which can be accessed by using the network addresses of the DNI units. Which actual Functional Processing Chain (i.e. which physical FUs) provides the requested service is transparent to the user.

As for PCM telemetry, for packet and AOS telemetry the R-PHS provides on-line and "off-line" services. In on-line mode the data are delivered without flow control, but with overflow management. When the volume of data requested by the user exceeds the available bandwidth of the connection to the user, data are discarded in a controlled way such that minimum size blocks of contiguous (as far as successfully reconstructed from the incoming symbol stream) telemetry are delivered. The block size is user selectable. In addition, a user controlled release timer warrants a worst case latency of the telemetry delivery.

In terms of data selection, the R-PHS supports these options:

- Space Link Channel SLC (i.e. all frames)
- Master Channel MC (i.e. all (good) frames of the specified S/C ID and Version ID)
- Virtual Channel VC (i.e. all good frames of the specified VC in the specified MC)
- MC Secondary Header (i.e. the Primary and Secondary Headers extracted from the frames received on the specified MC; Packet Telemetry only)
- VC Secondary Header (i.e. the Primary and Secondary Headers extracted from the frames received on the specified MC/VC; Packet Telemetry only)

- VC Access (i.e. the VCDU Data Zone extracted from the frames received on the specified MC/VC; AOS only)
- VC Bitstream (i.e. the Data Field Status (Packet Telemetry only) and the Data Field/Data Unit Zone (without fill data) extracted from the frames of the specified MC/VC)
- MC Control Field (i.e. the Operational Control Field extracted from the frames received on the specified MC)
- VC Control Field (i.e. the Operational Control Field extracted from the frames received on the specified MC/VC)
- Source/Path Packets (i.e. the reconstructed source/path packets with the specified AP-IDs received on the specified MC/VC; the AP-ID list can be modified on-line; synchronisation markers inserted into the data transferred to the user indicate when the new selection has become effective)
- Time Calibration (i.e. the Time Calibration Packet constructed for the specified VC; Packet Telemetry only)
- Space Link Status (on user request, the R-PHS monitors the space link status and reports any status changes detected)

The other service class is the so-called "immediate data access" (IDA), which as opposed to the on-line service delivers the selected data with full flow control. This means that the selected data will be delivered to the user as fast as the available link bandwidth allows, but due to the applied flow control no data will be lost. This service class is not called "off-line", since it allows the user in a single selection not only to request data already stored by the R-PHS, but even telemetry still to be acquired. This means that, available communications bandwidth permitting, the IDA service class can also be used for near real-time telemetry delivery, in case flow control is essential.

The data selection options are mostly identical to the on-line service class with the following exceptions:

- Information on the presently stored telemetry can be retrieved in the form of directories
- Data selections can be further refined by specifying
 - the start and end time and or counter range
 - the start time or counter and the number of data units to be delivered
- List of AP-IDs can only be changed when a data transfer invoked earlier has been terminated
- Space link status reports are not available

If the real-time telemetry received by the R-PHS contains also a Virtual Channel conveying so-called tape dump data, the transfer frames of this virtual channel will initially be stored as any other VC. Under control of the SSM, the data zones of the transfer frames of the tape dump VC are extracted and, if required, in reverse order, serialised and forwarded to a Frame Extractor Decoder (FED) unit which performs frame synchronisation and decoding. As real time telemetry, the annotated frames are forwarded via the VCDEMUX to the Storage unit which stores them in the "tape-dump" directory rather than the real-time directory. By means of the IDA services the user has access both to real-time and tape-dump telemetry by selecting the appropriate directory.

The Telecommand Encoder (TCE)

The "conventional" telecommand function is implemented in a single physical unit encompassing the communication network interface for connection to the user, the telecommand engine proper and the PSK modulator. As opposed to the R-PHS, the Functional Processing Chain providing the telecommand service cannot be dynamically built from a pool of functional units interconnected by virtual circuits over a LAN. Therefore, for the telecommand function the "server" concept has not yet been implemented. By selecting the network address, the user also specifies implicitly the physical resources which will provide the requested service.

Otherwise, as regards the protocols, the same considerations as presented for the R-PHS have been applied. Since the present modulation standard limits the maximum throughput on the space link for telecommands to 4 kbps, and, in general, compared to telemetry, the throughput requirements are moderate, for telecommanding a single seven layer protocol architecture (i.e. not split into control and data channel) is used.

The TCE provides three types of services:

- the PCM Telecommand service,
- the Packet Telecommand service,
- the Physical Layer Interface service.

Before a user connects to the TCE, the service to be provided is enabled through the management interface and the access rights for the user(s) are set.

The PCM telecommand service has been implemented in order to ensure the continuation of support to on-going missions, whenever the new TCE has to be installed as an in situ replacement of the equipment presently deployed. To avoid the need for any modifications on the control centre side, the related private ESA message protocol implemented on top of the X.25 network layer has been implemented for accessing the PCM telecommand service.

The Packet Telecommand service, which can be accessed using the Common Management Information Protocol (CMIP), supports telepresence applications by allowing payload control centres to connect in parallel with the flight agency's control centre. To safeguard the mission, only the flight agency's control centre has control over the telecommand session, i.e. the establishment and release of the radio link to the spacecraft. Only this control centre has access to the Bypass Control (BC) service and is allowed to send directives affecting the state of the telecommand protocol engine in the TCE. It also determines the uplink bandwidth allocation by specifying the MAP (Multiplexor Access Point) multiplexing scheme. Should an emergency require to do so, the flight agency's control centre can at any time lock out payload control centres. The TCE will only accept telecommand requests as long as the user access rights encompass the selected MAP and Application Identifier(s) (AP-ID). In order to facilitate cross support, the TCE implements, in addition to the ESA packet telecommand standard, also the CCSDS Blue Book, where the latter, in contrast to the ESA standard, allows Bypass-Control (BC) services to be supported without first terminating the Acceptance-Data (AD) service.

The Physical Layer Interface service is intended for support of spacecraft which adhere neither to the PCM nor to the Packet Telecommand standard. In this service, the TCE is practically transparent and enables uplinking of CLTUs (Command Link Transfer Unit) as submitted by the user. In addition, the user has control over the insertion of acquisition and idle sequences, where in case the TCE is requested to insert them automatically the bit patterns can be defined.

The Monitoring & Control Concept

The objective of the new M&C concept has been to establish a clean management hierarchy (ground station network, individual ground station, individual subsystem) to allow control of the services made available to users by the various ground segment entities. Furthermore, the implementation of the concept should exploit more advanced technology to replace the mostly IEEE-488 bus-based station internal M&C infrastructure which is cumbersome and expensive to maintain because of the lack of truly standardised protocols, data types, data presentation and bandwidth limitations.

Within the scope of this paper, it is not possible to present the entire M&C concept. What is presented below, addresses the sub-system related aspects of this concept.

Also in the area of ground station equipment, considerations of cost and user friendliness have led to the introduction of Graphical User Interfaces (GUI), replacing the expensive, individually designed front panels. The availability of powerful GUI building packages in the UNIX world resulted in the introduction of UNIX in embedded systems which traditionally had been built exclusively around real-time kernels. UNIX also made the LAN technology readily available which enabled to place the GUI infrastructure, which then is shared between individual sub-systems, at the stations operator's normal working position, relieving him from having to walk to the individual sub-system to control it.

This evolved environment also facilitates the introduction of a modern M&C infrastructure, where the expensive and cumbersome IEEE-488 infrastructure is replaced with the LAN and where due to UNIX suitable protocols available as off-the-shelf products can be introduced providing for truly standardised data exchange. Candidate protocols have been evaluated and CMIS/CMIP has been chosen. The initial implementation only uses a subset of the service elements provided by this protocol, in particular scoping and filtering are not used. In order to obtain good adaptation to the application and to avoid unnecessary complexity, privately defined managed objects (*M&C-ID-0*) are used rather than the Generic Managed Objects defined in ISO 10165. By means of these objects, a "conceptual" view of the sub-system which is then available to the managing entity can be modelled. These objects are briefly described below.

Any sub-system is assumed to arrange the M&C related resources in a tree structure in line with the hierarchy depicted in figure 4, where this tree has three levels: the sub-system proper, individual subsystem units, and function blocks (function blocks can however exist at sub-system level). These elements are not only structuring the resource tree, but they are also Managed Objects which support generic sub-system administration like state transitions from set-up to operable, control mode (local or remote) and the like. The tree structure determines the scope of visibility of resources. At any node within the tree only those resources which belong to that node or a branch below that node are visible.

Associated with each node of the tree, different types of resources like Variable Lists (VL), and tasks may exist. These resources can be mapped to the different managed objects accessible through the management interface of the subsystem. This resource structure as well as the mapping to the managed objects is specified in the so-called Management Information Base (MIB) description file (*M&C-ID-1*), which is evaluated at start-up of the subsystem. If the particular site or the mission to be supported require a modification of the view presented to the managing entity, the MIB description file is updated accordingly. The generation of a modified view, i.e. a different mapping of resources to the managed objects, does not require any change to the software of the sub-system. At start-up, the MIB is built according to the MIB description file. Obviously, since the resource structure is implemented in the sub-system's application software, this part of the MIB description file is fixed.

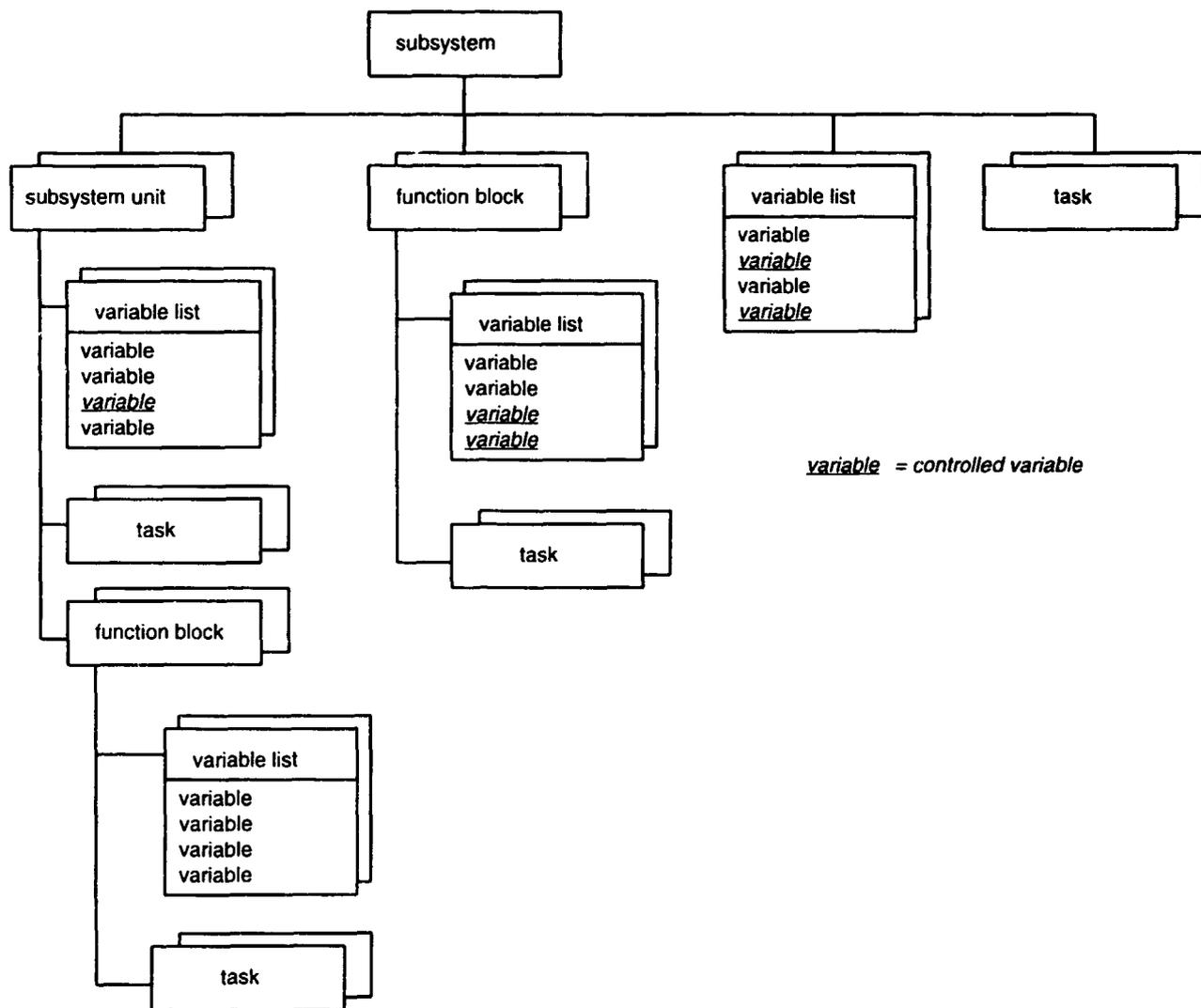
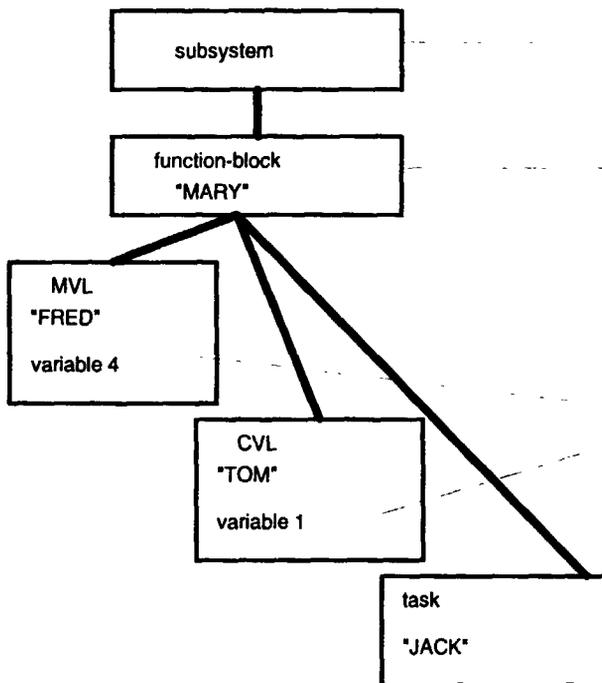


Figure 4: Sub-system Resource Tree Structure

The mechanism by which the resources are mapped to the Managed Objects accessible to external entities is illustrated by means of a very simple example in figure 5. The purpose of the Managed

Objects "sub-system", "sub-system unit", and "function block" is explained above. The "monitored variable list" provides read access to subsystem internal variables, where the manager can choose to receive a report either cyclically, on change of (at least) one variable contained in the list, or only on request. The "controlled variable list" enables the manager to set subsystem variables to either the values specified in the request or to default. Any subset of the variables contained in the Managed Object is accessible. The "task" object is used to invoke, stop, or abort the execution of specific functions in the sub-system, where the object is used both to convey any arguments as well as for monitoring of the function execution. The "event handler" objects allow the detection of sub-system internal events and the automatic triggering of associated actions. The manager can switch on or off the complete event detection as well as the individual associated actions. The "log" object is used to copy the specified subset of the sub-system log into a "public" file store from which it can then be retrieved by the manager. As a future extension, it is intended to implement a Managed Object for the administration of sub-system schedules. Another set of Managed Objects is used for control of inter-subsystem communication. This feature is used e.g. by the TCE which connects to the Front-End Controller for checking the front-end status.

CONCEPTUAL M&C DATA BASE



PHYSICAL M&C DATA BASE

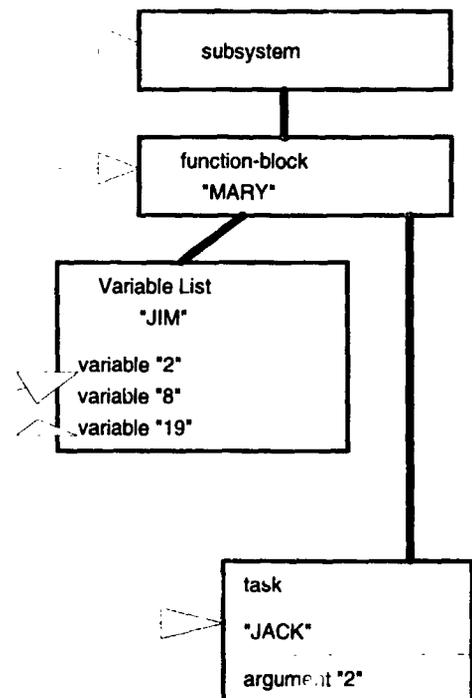


Figure 5: Mapping of Resources to Managed Objects

Conclusion and outlook

Starting from the architectural concept for the ground station back-end equipment, this paper has described the actual implementation as constrained by presently available hardware and software, cost and the need for backward compatibility. The growth path towards full implementation of the CCSDS AOS recommendations has been outlined.

The features designed into the equipment to facilitate cross-support and to promote telescience in terms of available services and management concept have been high-lighted.

Further system enhancements of the described sub-systems will be driven by mission needs. Hardware modifications will aim at getting closer to the architectural concept, in particular as regards the Frame Extractor/Decoder component. Mid-term extensions of functionality are expected in the area of further refined services resulting from the introduction of the Packet Utilisation Standard (PUS). Another activity which has already been started is the development of a "low-end" telemetry system which while retaining the functionality and user interface will be based on much simpler (and therefore cheaper) hardware. The considerably lower performance of this system is still sufficient to cover a wide range of TT&C applications such as geostationary communications satellites.

References

- | | |
|------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| PSS-04-106 | Packet Telemetry Standard, ESA PSS-04-106, Issue 1, January 1988 |
| PSS-04-107 | Packet Telecommand Standard, ESA PSS-04-107, Issue 2, April 1992 |
| CCSDS 102 | Packet Telemetry, CCSDS 102.0-B-2, Blue Book, January 1987 |
| CCSDS 201 | Telecommand: Part 1 Channel Service, Architectural Specifications, CCSDS 201.0-B-1, Blue Book, January 1987 |
| CCSDS 202 | Telecommand: Part 2 Data Routing Service, Architectural Specifications, CCSDS 202.0-B-1, Blue Book, January 1987 |
| CCSDS 203 | Telecommand: Part 3 Data Management Service, CCSDS 203.0-B-1, Blue Book, January 1987 |
| CCSDS 701 | Advanced Orbiting Systems, Networks and Data Links, Architectural Specification, CCSDS 701.0-B-1, October 1989 |
| PSS-46 | PCM Telemetry Standard, ESA PSS-46 (TTC.A.02), Issue 1, April 1978 |
| PSS-45 | PCM Telecommand Standard, ESA PSS-45 (TTC.A.01) Issue 1, April 1978 |
| M&C-ID-0 | M&C Interface between the Station Computer II and the Sub-System Controllers, ESA, M&C-ID-PTS-0000 (SCED/TDD/27318/PTS), Issue 1.3, July 1994 |
| M&C-ID-1 | ESTRACK Standard for Management Information Base Definition Files, ESA, M&C-ID-PTS-0001, Issue 1.1, July 1994 |

Acknowledgements - While the author has sole responsibility for the contents of this paper, I wish to extend special thanks to the following LSA personnel for their substantial contributions to the projects reported on in this paper: G.P. Calzolari, Y. Doat, F. Nunez-Mendoza, L. Proehl-Hansen, and G. Theis. A complete list of credits to the individual staff members of our industrial partners would be impossible; however, the author wishes to thank CRI's (Denmark), Dornier's (Germany), and Softlab's (Germany and Austria) staff for their contribution to the concept and implementation of the equipment presented in this paper.

PROCESS AND METHODOLOGY OF DEVELOPING CASSINI G&C TELEMETRY DICTIONARY

Edwin P. Kan

Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, Ca 91109

ABSTRACT

While the Cassini spacecraft telemetry design had taken on the new approach of "packetized telemetry", the AACS (Attitude and Articulation Subsystem) had further extended into the design of "mini-packets" in its telemetry system. Such telemetry packet and mini-packet design produced the AACS Telemetry Dictionary, iterations of the latter in turn provided changes to the former. The ultimate goals were to achieve maximum telemetry packing density, optimize the "freshness" of more time-critical data, and to effectuate flexibility, i.e. multiple AACS data collection schemes without needing to change the overall spacecraft telemetry mode. This paper describes such a systematic process and methodology, evidenced by various design products related to, or as part of, the AACS Telemetry Dictionary.

INTRODUCTION

An efficient ground data system and effective telemetry data processing / analysis system stem from good engineering design with respect to timeliness, frequency, accuracy, and sufficiency of the data contents in the telemetry stream. The human interaction with the data, thence consumption of the data, can also be enhanced by human-engineered telemetry displays and systematic organization of the telemetry measurements.

Such objectives can be achieved, in part, by an up front design of a flexible and efficient telemetry handling system on board the spacecraft, and of an equally efficient ground data analysis system. A common thread between the flight and ground systems is the Telemetry Dictionary.

In the present context, the Telemetry Dictionary is more than just a collection of telemetry

measurements with their descriptions, arranged in some alphabetical ordering. The development process of the Dictionary is intertwined and iterative with the design process of the telemetry system. In fact, the Dictionary is not simply the child-of-the-parent of the telemetry design; it is also the parent-of-the-child. The Dictionary evolves from the telemetry design process; and through iterations, the Dictionary development in turn provides improvement and optimization to the telemetry design.

This iterative process was particularly necessary for the Cassini AACS (Attitude and Articulation Subsystem) because of its new approach of using a "packetized telemetry" system versus the widely used "time division multiplex" (TDM) system. The AACS further extended the packet design to include the "mini-packet" design.

The ultimate goals of the mini-packet and packet telemetry design were to achieve maximum telemetry packing density, optimize the "freshness" of more time-critical data, and to effectuate flexibility, i.e. multiple AACS data collection schemes without needing to change the overall spacecraft telemetry mode.

The Cassini AACS telemetry design also responded to the object-oriented design approach of the AACS flight software. The fundamental entity of telemetry collection was to be based on each software object. A bottoms-up approach was used to assemble and analyze the telemetry measurements per software object. A database was constructed in which each measurement (i.e. record) was associated with attributes including measurement-number (E-numbers in Cassini), mini-packet, software object, channel¹ type, bit assignment, scale factor etc.

¹ "Channels" are herein used synonymously with telemetry "measurements", and should not be confused with "telecommunication channel, bandwidth".

Through iterative analysis, the collection of measurements was screened, organized, and assigned to the fundamental unit of a telemetry mini-packet. Mini-packets were created that grouped measurements by similar functions and/or similar collection periods. A systematic optimization of mini-packet assignments led to the consolidation of the database, from which statistics were synthesized and analyzed. AACS telemetry modes were designed corresponding to the overall spacecraft telemetry modes - a virtue of the flexibility of a mini-packet packetized telemetry system. Telemetry maps specifying the periodicity of telemetry mini-packets were designed, satisfying overall spacecraft telemetry bandwidth allocation requirements.

This paper describes such a systematic process and methodology, evidenced by various design products related to, or as part of, the AACS Telemetry Dictionary. This work was performed during the first part of Fiscal Year 1994, and was completed before the AACS Flight Software Critical Design Review.

FEATURES OF A TELEMETRY DICTIONARY

References to the AACS Telemetry Dictionary of Galileo (ref. 1), Mars Observer (ref. 2), and Cassini (ref. 3) reveal the common features of a telemetry dictionary of a major-size spacecraft. Putting aside those spacecraft-specific design features that should always be documented, the following list shows the major features to be included in the telemetry dictionary:

- Spacecraft telemetry system description
- Subsystem (e.g. AACS) telemetry system description
- Telemetry design: data acquisition, processing, storing, and transmission; telemetry maps, rates, modes (overall spacecraft mode versus subsystem mode)
- Telemetry detailed design: data format, headers, trailers, fillers, engineering "transfer frames", major frames
- Telemetry packets, mini-packets
- Special telemetry modes
- Telemetry Indices: by channel number, display mnemonics, data type, subsystem association, flight software name, and frequency (periodicity)
- Telemetry data sheet (by channel number)

- Telemetry subcommutation map (for TDM design; packet and mini-packet tables (for "packetized" design)
- Telemetry modes, transitions, relationship between spacecraft mode and subsystem (telemetry / operation) mode
- Parent-to-child relationship between channels (child-channels are usually derived in Ground Data System in order to relieve spacecraft downlink burden)

Spreadsheet or database documentation of channel data is ideal not only for sorting / indexing purposes, but also invaluable in the analysis / synthesis of telemetry modes, rate (periodicity) association, decommutation and mini-packet / packet design. Spreadsheet columns, i.e. attributes, should at least include channel number, display mnemonics, data type, subsystem association, flight software name, and frequency (periodicity).

In fact, the basis of the Cassini AACS Telemetry Dictionary used for the mini-packet / packet design, rate group association, and overall downlink channel bandwidth optimization, was a spreadsheet documentation of all telemetry channels.

Additional attributes included in the Cassini AACS Telemetry Dictionary spreadsheets were associations to software object, hardware unit, and mini-packet function (hence mini-packet name). Desired data frequency (periodicity) was a very important attribute, used in the iterative design of the mini-packets. The desired periodicity expressed the "freshness" requirement, and was represented by cardinal ratings of F, FM, M, MS, and S (i.e. fast, fast-medium, medium, medium-slow, and slow). Attributes of data types (signed integer, unsigned integer, floating-point, digital, state and ASCII) and number of data bits were included for channel bandwidth optimization and statistics summarization.

PACKET / MINI-PACKET DESIGN vs TDM (Time Division Multiplex) DESIGN

The gist of the design differences between packet / mini-packet design versus TDM design is the absence vs presence of a "telemetry decommutation map".

In a TDM design, a channel will be included in the telemetry stream (regardless of whether the stream is to be downlinked or stored on-board) at a fixed location according to the decommutation map. A map covers all locations of a complete unit of telemetry stream (also known in Galileo as Major Frame, in Mars Observer as Engineering Transfer Frame). At a given bit rate, the "frame" always spans the same duration of time. (Hence, the scheme is called TDM.)

Within a decommutation map, the same channel can appear once or multiple times. In the former case, the channel is said to be in the "slow deck"; in the latter, "medium" or "fast" deck, depending on the repetition rate. In Galileo, there are basically three rates, the "ninety-one-deck", "thirteen-deck", and "zero-deck", ranging from slow to fast. For 1200 bps telemetry rate, the periods are 60 $\frac{2}{3}$ sec., 8 $\frac{2}{3}$ sec., and $\frac{2}{3}$ sec. In Mars Observer, in the 2000 bps Engineering Mode, there are the 32 sec., 8 sec., 1 sec. "-decks" for the flight computer processed data.

Decommutation maps are large. There can be multiple maps, one for each Spacecraft "mission" mode. In Mars Observer, there are four modes: Engineering, Mission, Emergency and Safe Mode; with different bit rates ranging from fast to slow, respectively. In Galileo, even though bit rate can change from 1200 bps down to 8 bps, the same decommutation map still applies; however, there is an extra "Variable Telemetry Map" that can be selected from four choices. All Variable Telemetry Maps provide 22.5 (16-bit) words, equivalently 18 plus 9 one-half channels at the zero-deck rate.

Changes to decommutation maps are possible normally via memory loads at specific memory addresses. Such a change process is labor-intensive.

For Cassini, if TDM were used, the maps would be even larger (about five times as large as Galileo, and one-and-a-half times larger than Mars Observer). This is not simply due to complexity of the spacecraft, i.e. number of subsystems, but is due to increase of computation power of the on-board computers.

Without using the packet / mini-packet design, Cassini would suffer excessive sluggishness in AACS telemetry - where the fastest allocation

downlink rate was at 1896 bps, with 576 bps allocated to AACS.

The mini-packet design provides AACS with total freedom to assign desired / appropriate mini-packets to the fixed packet size allocated to AACS. Each Spacecraft Subsystem is allocated a certain packet size. Multiple (not necessarily integral number of) packets can be included in an "engineering transfer frame".

Flexibility is achieved by associating AACS Telemetry Modes for certain AACS Operation Modes, and against all Spacecraft Mode. Instead of having the TDM decommutation map(s), maps of telemetry channels in mini-packets (regardless of modes), and maps of mini-packets in packets (per AACS Telemetry Mode) are stored. The first set of maps are much smaller than a TDM decommutation map. The second set of maps are basically tables of "(m,n) frequency" allocation of mini-packets to packets.

"(m,n)" frequency in Cassini means that, for that AACS Telemetry Mode, m mini-packets will be contained in n packets. E.g. (8,1) is the fastest rate and (1,64) is the slowest rate in Cassini. At an AACS packet period of 8 sec., they represent mini-packet periods of 1 sec and 512 sec.

For more details on TDM, mini-packets, guaranteed delivery of mini-packets in packets, see (ref.1 - 4)

CASSINI PROCESS & METHODOLOGY of Telemetry Dictionary Development

The Cassini AACS telemetry design and Telemetry Dictionary development was an interactive and iterative process. Using project organization terminology, it was a cooperative task performed between the AACS Subsystem Group, Control Analysis Group, Flight Software Group, Hardware & Electronics Group, and the Ground Data Systems / Mission Operations Group.

While generic telemetry channel requirements were synthesized by the Subsystem Group, specific candidates were proposed by the Hardware Group, Analysis Group, and the Software Group. Inheritance from the Galileo and Mars Observer designs was duly observed. In fact, a one-to-one comparison was made

between the Galileo AACS Telemetry Dictionary and the candidate Cassini Dictionary, revealing potential omissions and confirming completeness.

From the respective AACS Groups, requirements for candidate telemetry channel, periodicity, data bits (resolution, precision), and format were drawn on hardware (sensors, actuators, hardware-to-electronics interfaces); control states, intermediate and observable variables; flight computer hardware data, hardware configuration and overall fault protection data. The Ground System Group was consulted regarding mission operations requirements and channel bandwidth optimization. Human engineered mnemonics and channel type assignment were prescribed to all measurements, conforming with JPL's AMMOS (Advanced Multi-Mission Operations System) ground software standards.

The object-oriented software design of the AACS flight software design (some 20 objects) (ref. 5) provided an easy association of telemetry to software objects. The list of object names and their statistics are given in Table 1. (The Telemetry Manager is one such object.) Table 2 is a sample of this initial compilation of telemetry dictionary, for the Software Object of "Accelerometer_Telemetry_Manager". Since object-oriented software design has distinct input output data flow, the same telemetry can be tapped from either the source or destination. A rule of thumb was adopted to tap the telemetry from the source, unless certain functional groupings made it more desirable to tap from the destination.

A spreadsheet for all telemetry channels was then composed, where all attributes were entered, including their cardinal ordering of periodicity.

At that point, mini-packets were designed which attempted to group telemetry by

- functionality
- similarity in periodicity requirement
- manageable size of mini-packet.

The number of mini-packets were kept to a minimum, compromising with the uniformity (diversity) of the functionality and periodicity of the channels grouped within the same mini-packet.

The mini-packet attribute was then added to the spreadsheet. With each iteration, new packet / mini-packet design was synthesized and their statistics analyzed. Iterations on the spreadsheet, good engineering practice, and negotiations with the engineer(s) requiring the specific channels (and other requirements), then led to a compromised mini-packet design.

While the design work was approaching completion, bandwidth allocation had yet to be analyzed. This was when the cardinal ordering of mini-packet periodicity was translated into ordinal (m,n) association.

New spreadsheets were prepared (Table 3), which were linked to the Telemetry Dictionary spreadsheet, linked for channel attributes such as data bit size and mini-packet association. An iterative analysis and synthesis further led to optimized (m,n) periodicity associations, addition/deletion/merging of mini-packets, and final assignment of channels to mini-packets.

Finally, an overall design of AACS Telemetry Modes, corresponding to all AACS Operation Modes and Spacecraft "Mission" Modes led to more rounds of iterations and finalization of the telemetry design, mini-packet / packet design, and, above all, the AACS Telemetry Dictionary.

Samples of the Final Dictionary (as of Jan., 94) are given in Table 4 and 5, where the telemetry channels are ordered by channel-numbers (i.e. "E-numbers", also by Software Objects), and by mini-packets.

All in all, 1088 channels in 67 mini-packets were assembled in the AACS Telemetry Dictionary. Out of these 67 mini-packets, 6 contained the less used off-diagonal covariance and Kalman gain elements (161 measurements), which are non-essential during normal mission operations. Eliminating those left 947 measurements in 61 mini-packets. A total of seven telemetry maps corresponding to 7 AACS telemetry modes were constructed. These modes are: (1) Record; (2) Nominal Cruise; (3) Medium Slow Cruise; (4) Slow Cruise; (5) Orbital Ops; (6) Δv ; (7) ATE (Attitude Estimator) Calibration. These 7 maps cover all spacecraft telemetry modes. For further information about mode transitions, and for details of the AACS Telemetry Dictionary, refer to (ref. 3 and 6.)

CONCLUSION

The process of bottoms-up development, use of human engineering skills, and the construction of the database had permitted a systematic way of sorting, synthesizing and analyzing all Cassini AACS telemetry measurements. Maximizing the use of database formulas and linking databases also permitted expedient parametric variation and analysis of bottom-line figures; examples of the latter were dictionary statistics, and bandwidth consumption (vs allocation) for specific telemetry modes. Hence, an effective and flexible packet / mini-packet design scheme.

This process of developing the packet / mini-packet design and the establishment of the AACS Telemetry Dictionary had proven to be closely intertwined and cross-productive. The end result also provided the design for the "Telemetry Manager" flight software object. The process helped to bind a contract, i.e. interface specification of telemetry measurement between software objects. It further provided important feedback to software control algorithm designers for finalizing design parameters.

In conclusion, not only was this Cassini process a means to an end - the Telemetry Dictionary, it was also a team-player in the overall AACS flight software design.

ACKNOWLEDGEMENT

This work was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under contract to the National Aeronautics and Space Administration. The author would also like to acknowledge the efforts of D. Bernard, M. Brown, and R. Rasmussen, who laid the foundations to this present work.

REFERENCES

1. Anon., "Galileo AACS Telemetry Dictionary," Document GLL-OET-91-393, Jet Propulsion Laboratory, Pasadena, Ca., USA, 10/30/91.
2. Anon., "Mars Observer: Engineering Telemetry & Command Dictionary," CTD-3271152 Vol. 1 & 2, Contract #957444, prepared by Astro-Space Division, General Electric Company for Jet Propulsion Laboratory, Pasadena, Ca., USA, 9/11/92.
3. Anon., "Cassini AACS Telemetry Dictionary, Vol. 1 and 2," JPL Document #D-11796, Jet Propulsion Laboratory, Pasadena, Ca., USA, 5/9/94.
4. Anon., "Cassini Functional Requirements: Telemetry Format and Measurements," JPL Document #699-3-281, Jet Propulsion Laboratory, Pasadena, Ca., USA, Apr. 1994.
5. Hackney, J., D. Bernard & R. Rasmussen, "Cassini Spacecraft Object Oriented Flight Control Software," Paper 93-033, Proc. AIAA AAS Conf., Jan. 1993.
6. Kan, E. P., "The Cassini AACS Telemetry Dictionary - Development Process and Results," presentation on 1/5/94, also IOM# GLL-OET-93-667, Jet Propulsion Laboratory, Pasadena, Ca., USA, 12/28/93.

TABLE 1. Summary Statistics - # of Channels vs Software Object

Software Object	Hdwe ass'n	# channels	Notes
ACL	Attitude Control	Software	29
ACM	Attitude Commander	Software	43
ADC	Attitude Determinat'n Commander	Software	1
ATE	Attitude Estimator	Software	244
CFG	Configuration Manager	XXX	124
CMD	Command Manager	Software	15
CMT	Constraint Manager	Software	13
FPA	Fault Protection & Analyzer	XXX	280
FPR	Fault Protection Recovery	Software	3
FSX	Flight Software Executive	EFC	71
IOUmgr	Input/Output Unit Manager	XXX	54
IYP	Inertial Vector Propagator	Software	1
MOD	Mode Commander	Software	5
PROM	PROM_Control	Software	6
SID	Star ID (identification)	Software	64
TLM	Telemetry Manager	Software	2
XBA	Cross-string Bus Adapter	Software	24
ACC	Accelerometer Manager	Hdwe_Mgr	7
EGA	Engine Gimbal Actuator	Hdwe_Mgr	10
IRU	Inertial Reference Unit	Hdwe_Mgr	12
PMS	Propulsion Module System	Hdwe_Mgr	10
RWA	Reaction Wheel Assembly	Hdwe_Mgr	48
SRU	Stellar Reference Unit	Hdwe_Mgr	18
SSA	Sun Sensor Assembly	Hdwe_Mgr	10
			TOTAL # ch.'s = 1094
			less non-ess. ATE = 933
			less TBD FPA ch = 677

Table 2. Telemetry List for Accelerometer_Manager Software Object

Ch	Ch#	Mnemonics	Mini	Attribute	Hardware	Software	Rate	Notes	Type	Bit	Scale
	(new#)		Pkt#	prime	associat'n	object	:cruise*				Factor
E	1001	ACC__state	4	Sfwe_State2	ACC	HdweMgr	M			S	4
E	1002	ACC_calBIAS	21	deltaV	ACC	HdweMgr	Z	"driftDelta" - cal'b prior to deltaV	I	16	
E	1003	deltaV_ACC	21	deltaV	ACC	HdweMgr	Z	"deltaV" - after scale factor compensatic	I	16	
E	1004	ACC_totBIAS	21	deltaV	ACC	HdweMgr	MS	drift"= driftNominal*driftDelta	I	16	
E	1005	deltaV_tmtg	21	deltaV	ACC	HdweMgr	Z	"deltaVTimeTag", used to compute "diffTin	U	16	
E	1006	ACC_tmtg	21	deltaV	ACC	HdweMgr	Z	"timeTag"; 8+8 bits	U	16	
E	1007	raw_ACC_CT	22	IRU/ACC_data	ACC	HdweMgr	M	"accumDeltaV" - before scale factor compe	I	16	
Legend: Rate (cruiseF = fast; M = medium; S= slow; FM = medium fast; MS = medium slow; Z = zero, except in special mode											
E	1394	ACC_cycle	26	IRU/ACC_stat	ACC	CFG	S		U	8	
E	1395	ACC_ONtime	26	IRU/ACC_stat	ACC	CFG	S	unit in second	U	16	
E	1650	ACC_RESEtct	37	Reset_count	ACC	IOU_mgr	S		U	8	
E	1665	ACC_ERR	38	Bus_error	ACC	IOU_mgr	S		U	8	
E	1666	ACC_ERR_ct	38	Bus_error	ACC	IOU_mgr	S		U	8	
E	2035	ACC_val1_ERR	46	Anomaly_st2	ACC	FPA	MS	ACC_mgr: "accTooBigFault"	D	4	
E	2036	ACC_drf_ERR	46	Anomaly_st2	ACC	FPA	MS	ACC_mgr: "driftTooBigFault"	D	4	

Table 4. AACS Telemetry Dictionary - sorted by Channel# and Software Object (page 1 of XX)

Ch	Ch#	Mnemonics	Mini	Attribute	Hardware	Software	Rate	Notes	Type	Bit	Scale
	(new#)		Pkt#	prime	associat'n	object	:cruise*				Factor
E	1001	ACC__state	4	Sfwe_State2	ACC	HdweMgr	M		S	2	1
E	1002	ACC_calBIAS	21	deltaV	ACC	HdweMgr	Z	ACC_mgr: "driftDelta" - calib prior to	I	16	5.0 E6
E	1003	deltaV_ACC	21	deltaV	ACC	HdweMgr	Z	ACC_mgr: "deltaV" - after scale factor cc	I	20	500
E	1004	ACC_totBIAS	21	deltaV	ACC	HdweMgr	Z	ACC_mgr: "drift"= driftNominal*driftDelta	I	16	5.0 E6
E	1005	deltaV_tmtg	21	deltaV	ACC	HdweMgr	Z	ACC_mgr: "deltaVTimeTag", used to compute	U	20	BITcrop
E	1006	ACC_tmtg	21	deltaV	ACC	HdweMgr	Z	ACC_mgr: "timeTag"; 12+8 bits	U	20	BITcrop
E	1007	raw_ACC_CT	22	IRU/ACC_data	ACC	HdweMgr	MS	ACC_mgr: "accumDeltaV" before scale fac	I	24	1
E	1021	momUNLOADst	3	Sfwe_State	Sfwe	ACL	FM	RCS/ACL: "inactive/THRUSTR_WARMUP/UNload;	S	2	1
E	1022	MANUVR_st	3	Sfwe_State	Sfwe	ACL	FM	TVC/RCS_deltaV/ACL: off/TVC_enabled/RCS	S	2	1
E	1023	ATT_CNTR_st	4	Sfwe_State2	Sfwe	ACL	M		S	4	1
E	1024	POSdadBND_X	6	SC_pointg2	Sfwe	ACL	S	RCS/ACL: changes by +/-20% during cruise;	U	16	1.0 E5
E	1025	POSdadBND_Y	6	SC_pointg2	Sfwe	ACL	S	RCS/ACL: changes by +/-20% during cruise;	U	16	1.0 E5
E	1026	POSdadBND_Z	6	SC_pointg2	Sfwe	ACL	S	RCS/ACL: changes by +/-20% during cruise;	U	16	1.0 E5
E	1027	RATEddBND_X	6	SC_pointg2	Sfwe	ACL	S	RCS/ACL: constants	U	16	1.0 E6
E	1028	RATEddBND_Y	6	SC_pointg2	Sfwe	ACL	S	RCS/ACL: constants	U	16	1.0 E6
E	1029	RATEddBND_Z	6	SC_pointg2	Sfwe	ACL	S	RCS/ACL: constants	U	16	1.0 E6
E	1030	dBND_IbitX	6	SC_pointg2	Sfwe	ACL	S	RCS/ACL: impulse bang-bang ctrl S/C att c	U	16	TBD
E	1031	dBND_IbitY	6	SC_pointg2	Sfwe	ACL	S	RCS/ACL: impulse bang-bang ctrl S/C att c	U	16	TBD
E	1032	dBND_IbitZ	6	SC_pointg2	Sfwe	ACL	S	RCS/ACL: impulse bang-bang ctrl S/C att c	U	16	TBD
E	1033	POSerr_X	8	Att_cntrl	Sfwe	ACL	FM	Same measurement for RCS,RWA,TVC. RCS: de	I	16	2.0 E5
E	1034	POSerr_Y	8	Att_cntrl	Sfwe	ACL	FM	Same measurement for RCS,RWA,TVC. RCS: de	I	16	2.0 E5
E	1035	POSerr_Z	8	Att_cntrl	Sfwe	ACL	FM	Same measurement for RCS,RWA,TVC. RCS: de	I	16	2.0 E5
E	1036	RATErr_X	8	Att_cntrl	Sfwe	ACL	FM	Same measurement for R S/RWA/TVC. RCS: de	I	16	1.0 E6
E	1037	RATErr_Y	8	Att_cntrl	Sfwe	ACL	FM	ditto [R]U res of 0.25µrad/pulse/0.25 sec	I	16	1.0 E6
E	1038	RATErr_Z	8	Att_cntrl	Sfwe	ACL	FM	ditto [R]U res of 0.25µrad/pulse/0.25 sec	I	16	1.0 E6
E	1039	cmdTORQUE_X	11	RWA_cntrl	Sfwe	ACL	Z	RWA/ACL: Different form RWA_TQ's. Here:	I	16	2.0 E5
E	1040	cmdTORQUE_Y	11	RWA_cntrl	Sfwe	ACL	Z	RWA/ACL: Different form RWA_TQ's. Here:	I	16	2.0 E5
E	1041	cmdTORQUE_Z	11	RWA_cntrl	Sfwe	ACL	Z	RWA/ACL: Different form RWA_TQ's. Here:	I	16	2.0 E5
E	1042	cmd_S/C_H_X	11	RWA_cntrl	Sfwe	ACL	Z	RWA/ACL: Different from ATE's momemntum	I	16	1.0 E3
E	1043	cmd_S/C_H_Y	11	RWA_cntrl	Sfwe	ACL	Z	RWA/ACL: Different from ATE's momemntum	I	16	1.0 E3
E	1044	cmd_S/C_H_Z	11	RWA_cntrl	Sfwe	ACL	Z	RWA/ACL: Different from ATE's momemntum	I	16	1.0 E3
E	1045	cmd_thrustX	21	deltaV	Sfwe	ACL	Z	TVC/ACL: "TsubC"	I	16	TBD
E	1046	cmd_thrustY	21	deltaV	Sfwe	ACL	Z	TVC/ACL: "TsubC"	I	16	TBD
E	1047	cmd_thrustZ	21	deltaV	Sfwe	ACL	Z	TVC/ACL: "TsubC"	I	16	TBD
E	1048	BURN_time	21	deltaV	Sfwe	ACL	Z	16+8 bits. 0.004 sec res 2^16 = 65536 sec	U	24	BITcrop
E	1049	deltaV_pred	21	deltaV	Sfwe	ACL	Z	TVC/ACL: predict of the time profile of 1	U	20	500
E	1061	TJRN_status	3	Sfwe_State	Sfwe	ACM	FM	"Completed/Rate_Matching/POS_matching/COA	S	2	1
E	1062	ATT_CMD_st	4	Sfwe_State2	Sfwe	ACM	M		S	4	1
E	1063	cmdSC_Q1	5	SC_pointing	Sfwe	ACM	MS	"base_attitude"	I	16	32768
E	etc	etc	etc	etc.	etc.	etc.	etc.	etc.	etc	etc	etc

Table 3: AACS "Record" Telemetry Mode* - Mini-Packet Assignment Map
(576 bps AACS allocation)

		AACS Packet Size (bits) = 4608				AACS Packet Period (sec) = 6			
		Header Size (bits) = 96				Assignable Bit Rate (bps) = 564			
		Assignable Space (bits) = 4512							
Mini-Pkt#	Mini-Packet Name	# channels	Size (bits)	n	m	Period (sec) ^A	bps	% Bandwidth	
1	Estimated Attitude	6	132	6	1	1	132.00	23%	
2	Hardware Configuration	13	184	1	4	32	5.75	1%	
3	Software State	8	69	2	1	4	17.25	3%	
4	Software State_2	30	144	1	2	16	9.00	2%	
5	Spacecraft Pointing	15	264	1	2	16	16.50	3%	
6	Spacecraft Pointing_2	20	528	1	32	256	2.06	0%	
7	Turn telemetry	19	328	1	32	256	1.28	0%	
8	Attitude Controller	6	120	4	1	2	60.00	11%	
9	Constraint Attitude Control	10	184	1	64	512	0.36	0%	
10	RWA General Data	8	120	1	16	128	0.94	0%	
11	RWA Controller	10	184	1	16	128	1.44	0%	
12	Attitude Estimator (ATE) Metric	9	168	4	1	2	84.00	15%	
13	Attitude Estimator (ATE) Data	22	376	1	2	16	23.50	4%	
14	ATE Auto-Calibration Data	12	216	1	8	64	3.38	1%	
15	ATE Star Pre-Filter Data	22	376	1	8	64	5.88	1%	
16	ATE Covariance Data	12	216	1	64	512	0.42	0%	
17	ATE Kalman Gain Data	9	168	1	64	512	0.33	0%	
18	SID: Star 1 & 2 Data	10	184	2	1	4	46.00	8%	
19	SID: Star 3,4 & 5 Data	19	308	1	2	16	19.2	3%	
20	SID: Calibration-Type Data	34	632	1	16	128	4.54	1%	
21	ΔV Maneuver Data	12	248	1	32	256	0.97	0%	
22	IRU & ACC Output Data	9	176	1	4	32	5.50	1%	
23	SSA & SRU Output Data	22	264	1	4	32	8.25	1%	
24	RWA Output Data	28	440	1	32	256	1.72	0%	
25	VDE & EGE Data	14	200	1	4	32	6.25	1%	
26	IRU & ACC Statistics	6	96	1	64	512	0.19	0%	
27	SSA & SRU Statistics	8	120	1	64	512	0.23	0%	
28	EGA Statistics	12	184	1	64	512	0.36	0%	
29	RWA Statistics	16	248	1	64	512	0.48	0%	
30	PMS Latch/Valve & ME Statistics	15	176	1	64	512	0.34	0%	
31	PMS Thruster Cycles	16	280	1	16	128	2.19	0%	
32	PMS_A FireTime	8	216	1	16	128	1.69	0%	
33	PMS_B FireTime	8	216	1	16	128	1.69	0%	
34	PMS CatBer/Heater Cycles	16	280	1	16	128	2.19	0%	
35	AFC Errors	13	288	1	64	512	0.56	0%	
36	EFC Errors	28	540	1	64	512	1.05	0%	
37	Reset Counters	15	144	1	64	512	0.28	0%	
38	AACS Bus Errors	30	264	1	64	512	0.52	0%	
39	Prime AFC Hardware Status	20	404	1	8	64	6.31	1%	
40	Backup AFC Errors	9	184	1	64	512	0.36	0%	
41	Backup EFC Errors	20	328	1	64	512	0.64	0%	
42	Backup AFC Hardware Status	20	372	1	64	512	0.73	0%	
43	Bus Message Status	18	256	1	1	8	32.00	6%	
44	Fault Protection Status	17	412	1	4	32	12.88	2%	
45	Anomaly Status	13	176	1	8	64	2.75	0%	
46	Anomaly Status 2	10	76	1	8	64	1.19	0%	
47-54	High Water Marks	128	280 x 8	1	64	512	4.38	1%	
55-62	Persistence High Water Marks	128	280 x 8	1	64	512	4.38	1%	
102/.15	ATE Covariance Data 2/.15	105	1776	0	64	#DIV/0!	0.00	0%	
106	ATE Kalman Gain Data 2	36	600	0	64	#DIV/0!	0.00	0%	
		Total #ch's	Total #bits			Total bps	Total bwdth		
		1094	13865			534.33	95%		
* This "Record" Telemetry Mode is one out of 7 modes: (1) Record; (2) Nominal Cruise; (3) Medium Slow Cruise; (4) Slow Cruise; (5) Orbital Ops; (6) delta_v; (7) ATE (Attitude Estimator) Calibration.									
^A Cardinal vs Ordinal Rating of Periodicity: F=(8,1),(4,1) FM=(2,1),(1,1) M=(1,8),(1,16) S=(1,32),(1,64)									

Table 5. AACS Telemetry Dictionary - sorted by Mini_Packet# (page 1 of XX)

Ch	Ch# (new#)	Mnemonics	Mini Pkt#	Attribute prime	Hardware associat'n	Software object	Rate :ruse*	Notes	Type	Bit	Scale Factor
E	1121	BODY_Z_RA	1	Est_Att	Sfwe	ATE	F	20 bit: 6 μrad resolution	I	20	2 ¹⁹ /R
E	1122	BODY_Z_DEC	1	Est_Att	Sfwe	ATE	F	20 bit: 6 μrad resolution	I	20	2 ¹⁹ /R
E	1123	BODY_Z_TWIS	1	Est_Att	Sfwe	ATE	F	20 bit: 6 μrad resolution	I	20	2 ¹⁹ /R
E	1124	X_rate	1	Est_Att	Sfwe	ATE	F		I	16	2.0 E5
E	1125	Y_rate	1	Est_Att	Sfwe	ATE	F		I	16	2.0 E5
E	1126	Z_rate	1	Est_Att	Sfwe	ATE	F		I	16	2.0 E5
E	1381	BUS_prime	2	Hdwe_config	EFC	CFG	M		D	16	1
E	1382	SNSR_pwr	2	Hdwe_config	Hdwe	CFG	M		D	8	1
E	1383	ACTP_pwr	2	Hdwe_config	Hdwe	CFG	M		D	8	1
E	1384	SNSR_prime	2	Hdwe_config	Hdwe	CFG	M		D	4	1
E	1385	ACTR_prime	2	Hdwe_config	Hdwe	CFG	M		D	4	1
E	1386	SNSR_hlth	2	Hdwe_config	Hdwe	CFG	M		D	16	1
E	1387	ACTR_hlth	2	Hdwe_config	Hdwe	CFG	M		D	16	1
E	1388	VDE_pwr	2	Hdwe_config	PMS	CFG	M		D	12	1
E	1389	VDE_prime	2	Hdwe_config	PMS	CFG	M		D	12	1
E	1390	vDE_hlth	2	Hdwe_config	PMS	CFG	M		D	20	1
E	1391	RCS_prime	2	Hdwe_config	PMS	CFG	M		D	8	1
E	1392	RCS_A_hlth	2	Hdwe_config	PMS	CFG	M		D	16	1
E	1393	RCS_B_hlth	2	Hdwe_config	PMS	CFG	M		D	16	1
E	1021	monUNLOADst	3	Sfwe_State	Sfwe	ACL	FM	RCS/ACL: "inactive/THRUSTP_WARMUP/UNloadi	S	2	1
E	1022	MANJVR_st	3	Sfwe_State	Sfwe	ACL	FM	TVC/RCS_deltaV/ACL: "off/TVC_enabled/RCS	S	2	1
E	1061	TURN_status	3	Sfwe_State	Sfwe	ACM	FM	"Completed/Rate_Matching/POS_matching/COA	S	2	1
E	1127	SunEphm_chk	3	Sfwe_State	Sfwe	ATE	FM	SSA sun_vect not equal (with tolerance) t	S	1	1
E	1541	CMT_status	3	Sfwe_State	Sfwe	CMT	FM	"Nominal/noJ2000/withJ2000/timeout"	S	2	1
E	1741	AACS_mode	3	Sfwe_State	Sfwe	MOD	FM		S	4	1
E	1742	AACS_stat1	3	Sfwe_State	Sfwe	MOD	FM		D	16	1
E	1743	AACS_stat2	3	Sfwe_State	Sfwe	MOD	FM		D	16	1
E	1023	ATT_CNTR_st	4	Sfwe_State2	Sfwe	ACL	M		S	4	1
E	1062	ATT_CMD_st	4	Sfwe_State2	Sfwe	ACM	M		S	4	1
E	1111	ADC_state	4	Sfwe_State2	Sfwe	ADC	M		S	4	1
E	1128	ATT_EST_st	4	Sfwe_State2	Sfwe	ATE	M		S	4	1
E	1129	deltaV_ESst	4	Sfwe_State2	Sfwe	ATE	M	TVC/RCS_delta_V/ACL: "idle/acc/timer/imp	S	2	1
E	1542	AVOID_state	4	Sfwe_State2	Sfwe	CMT	M	"Celestial_vect/body_vect"	S	4	TBD
E	1543	PTGviolatST	4	Sfwe_State2	Sfwe	CMT	M	"body_vect/thermal violation_duration"	S	4	TBD
E	1001	ACC_state	4	Sfwe_State2	ACC	HdweMgr	M		S	2	1
E	1561	EGAA_state	4	Sfwe_State2	EGA	HdweMgr	M		S	2	1
E	1562	EGAB_state	4	Sfwe_State2	EGA	HdweMgr	M		S	2	1
E	1711	IRUA_state	4	Sfwe_State2	IRU	HdweMgr	M	"on/off"	S	2	1
E	1712	IRUB_state	4	Sfwe_State2	IRU	HdweMgr	M	"on/off"	S	2	1
E	1713	IRUA_status	4	Sfwe_State2	IRU	HdweMgr	M	"max_pulse_viol:max_acc_viol;A&B_consiste	D	8	1
E	1714	IRUB_status	4	Sfwe_State2	IRU	HdweMgr	M	"max_pulse_viol:max_acc_viol;A&B_consiste	D	8	1
E	1761	PMSA_state	4	Sfwe_State2	PMS	HdweMgr	M	"on/off;idle;ME_critical_enabled;ME_pulse	D	8	1
E	1762	PMSB_state	4	Sfwe_State2	PMS	HdweMgr	M	"on/off;idle;ME_critical_enabled;ME_pulse	D	8	1
E	1791	RWA1_state	4	Sfwe_State2	RWA	HdweMgr	M		S	3	1
E	1792	RWA2_state	4	Sfwe_State2	RWA	HdweMgr	M		S	3	1
E	1793	RWA3_state	4	Sfwe_State2	RWA	HdweMgr	M		S	3	1
E	1794	RWA4_state	4	Sfwe_State2	RWA	HdweMgr	M		S	3	1
E	1931	SRUA_state	4	Sfwe_State2	SRU	HdweMgr	M		S	2	1
E	1932	SRUB_state	4	Sfwe_State2	SRU	HdweMgr	M		S	2	1
E	1961	SSAA_state	4	Sfwe_State2	SSA	HdweMgr	M	"on/off"	S	2	1
E	1962	SSAB_state	4	Sfwe_State2	SSA	HdweMgr	M	"on/off"	S	2	1
E	1963	SSAA_status	4	Sfwe_State2	SSA	HdweMgr	M	"auto/grd_cmd d_thrshld;sun_there;sun_sta	D	8	1
E	1964	SSAB_status	4	Sfwe_State2	SSA	HdweMgr	M	"auto/grd_cmd d_thrshld;sun_there;sun_sta	D	8	1
E	1731	IVP_status	4	Sfwe_State2	Sfwe	IVP	M	ala GLL IVP_Stat	S	4	1
E	1851	SID_state	4	Sfwe_State2	Sfwe	SID	M	states in transition diagram	S	4	1
E	1981	AACStlmMode	4	Sfwe_State2	Sfwe	TLM	M		S	4	1
E	1982	SCTlm_mode	4	Sfwe_State2	Sfwe	TLM	M		S	4	1
E	1063	cmdSC_Q1	5	SC_pointing	Sfwe	ACM	MS	"base_attitude"	I	16	32768
E	1064	cmdSC_Q2	5	SC_pointing	Sfwe	ACM	MS	"base_attitude"	I	16	32768
E	1065	cmdSC_Q3	5	SC_pointing	Sfwe	ACM	MS	"base_attitude"	I	16	32768
E	1066	cmdSC_Q4	5	SC_pointing	Sfwe	ACM	MS	"base_attitude"	I	16	32768
E	etc.	etc.	etc.	etc.	etc.	etc.	etc.	etc.	etc	etc	etc

E: 762
198

Multimission Telemetry Visualization (MTV) System
A Mission Applications Project from JPL's Multimedia Communications Laboratory

Ernest Koeberlein, III and Shaw Exum Pender

California Institute of Technology
 Jet Propulsion Laboratory

I. ABSTRACT

This paper describes the Multimission Telemetry Visualization (MTV) data acquisition/distribution system. MTV was developed by JPL's Multimedia Communications Laboratory (MCL) and designed to process and display digital, real-time, science and engineering data from JPL's Mission Control Center. The MTV system can be accessed using UNIX workstations and PCs over common datacom and telecom networks from worldwide locations. It is designed to lower data distribution costs while increasing data analysis functionality by integrating low-cost, off-the-shelf desktop hardware and software. MTV is expected to significantly lower the cost of real-time data display, processing, distribution, and allow for greater spacecraft safety and mission data access.

II. INTRODUCTION

As the leading NASA center involved in unmanned space missions, JPL has a long and distinguished scientific record of achievement in collecting, analyzing, distributing, and archiving data and images from planetary exploration missions. To manage its multi-scientific and engineering operations connected with the exploration of the Solar System, JPL has developed extensive local area communication networks for linking its user community in clusters of cooperative workgroups. Utilizing this infrastructure of networked desktop workstations and PCs as display platforms, MTV was developed to provide an easy, plug-in access to real-time mission data using local and wide area net-

works. Registered MTV users now have convenient access to key telemetry data channels from a variety of platforms and graphical user interfaces (GUIs) from almost any location. MTV data distribution and display ergonomics have increased the electronic exchange of engineering and science data by allowing principal investigators, scientists, engineers, and managers worldwide access to real-time data, anywhere, any time and seamless transport of data into other analysis, spreadsheet, word processing, or other software tools.

Recent technology advances in multimedia communications hardware and software have provided MTV users with a wide range of concurrent processes beyond telemetry viewing. Now audio and video services can be requested during MTV sessions, thus providing mission operations personnel with new processes for distributing and displaying high resolution photographs, conducting point-to-point video conferencing, shareware, and digital television monitoring/capturing. This paper also briefly outlines the role of MCL in developing low-cost multimedia communication tools for JPL and NASA scientists, engineers, and managers on a wide variety of projects.

The following MCL capabilities will be discussed:

(1) Prototyping and demonstrating network distribution of real-time mission data using networking, i.e. Institutional Local Area Networks, TCP/IP, FDDI, and telecom, i.e. standard 9600 baud telecom lines, Switched 56 and ISDN. This activity serves as a proof-of-concept function for the MTV project.

(2) Testing and evaluating promising technologies, applications and implementation strategies associated with distribution of bandwidth-intensive multimedia mission data types which are compressed and distributed over networks currently installed or planned at JPL, i.e. desktop video teleconferencing, groupware, image and video servers, multimedia electronic mail and remote telepresence over Ethernet, ATM and FDDI optical fiber interfaces.

(3) Analyzing and predicting the productivity impact of multimedia computing and communications on organizational effectiveness, and communications within and between scientific and engineering workgroups including multilingual communication for international spaceflight workgroups.

(4) Developing a five-year institutional strategy and implementation plan for integrating multimedia workstations with networked supercomputers, the National Information Infrastructure (NII), and High Definition Digital Television (HDTV) for space mission applications.

(5) Designing, developing, and implementing interactive, digital applications using interoperable workstations and PCs for supporting technical and management presentations, large group video teleconferencing, and on-line, interactive training for ground and mission operations.

(6) Development of multimedia productions for Internet Mosaic Home Pages including hypertext, full-motion video, and interactive CD-ROMs.

As the technologies of multimedia platforms, software, and subsystems enter mainstream computing and communications, the JPL MCL team evaluates promising commercial, off-the-shelf (COTS) technologies and products as they are released from developers. Those products which, after test and evaluation in the MCL, are found to contribute to cost-effective mission operations and add value to JPL's institutional processes, will be considered for service within our flight operations groups. MTV

was the first such product. As with any technology involving the widespread distribution of images and audio over networks, there is potential to reshape the way spacecraft data is viewed and shared. Multimedia communications is opening many new avenues for cost-effective, innovative processes which support the national space program. Further, it is expected that MTV will find application as a dual-use system in the commercial sector. Real-time medical monitoring, industrial and environmental monitoring and process control are a few promising applications under consideration for technology transfer.

III. ANALOG VERSUS DIGITAL SYSTEM

For twenty years JPL has relied upon an analog TV telemetry distribution system for viewing up to 3500 possible telemetry channels. After telemetry data is received by the Deep Space Network (DSN) and decommutated at JPL, it is converted to an NTSC video signal and distributed to a large switch for delivery to video monitors scattered throughout JPL's primary flight operation facilities. The system allows only viewing of the desired channels which the user may select for his/her mission. The Digital Television System (DTV) as it is called is not a digital system in the true sense, but was given this designation presumably because it displayed digits! The system has served JPL's telemetry data analysis users well and still has many proponents. But with the advent of desktop computers, the DTV system became an antiquated liability with little flexibility in the era of cheaper, better, and faster.

After reviewing the costs vs. capability of the DTV, it became clear that use of desktop PCs and workstations connected to the JPL Institutional Local Area Network (ILAN) and Internet could perform the primary DTV functions with greater flexibility. Enhanced telemetry visualization, a rich set of data analysis tools, including automated alarming of data streams by use of set points, were compelling reasons for a new system. Further, users could access the system from remote sites-globally. This feature is attractive for missions involving domestic and international partners with remote command centers

IV. MTV PROTOTYPE DESCRIPTION

The development of an MTV prototype was started by the Multimedia Communications Laboratory (MCL) in January 1993. After quickly abandoning the concept of continued broadband distribution of the DTV analog signals, except for video display on desktop platforms, it was decided by the designers to interface a Unix server with the Galileo spacecraft data stream, separate and condition the data channels, and distribute the data to remote PC clients and workstations using JPL's ILAN.

The initial prototype, which used a 486 PC running MS Windows 3.1 with a network interface, was demonstrated to several of JPL's mission teams. From this demonstration, experimental users were identified for testing MTV and the system was installed at several sites. These experimental sites were used to debug the system and to gain insight into user requirements. User's suggestions and comments were solicited and the design team made several enhancements to the MTV (GUI) as a result of this prototyping. The architecture of the MTV prototype is shown in Figure 1.0 below.

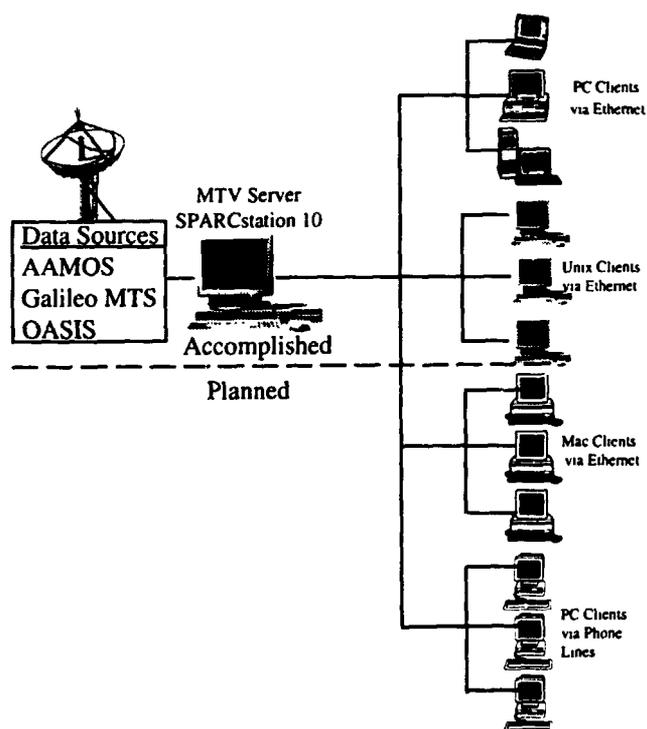


Figure 1.0 MTV Network Architecture

Samples of MTV display windows are shown in Figures 2.0 and 3.0 below.

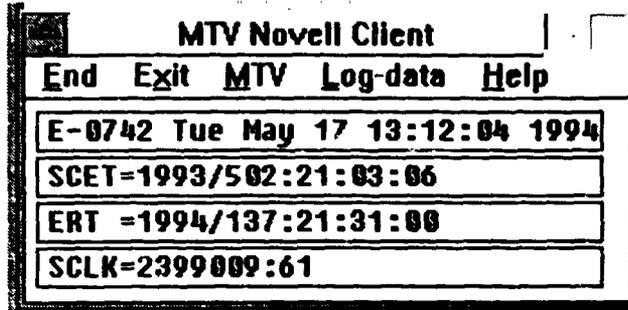


Figure 2.0 MTV Client ID Display Window

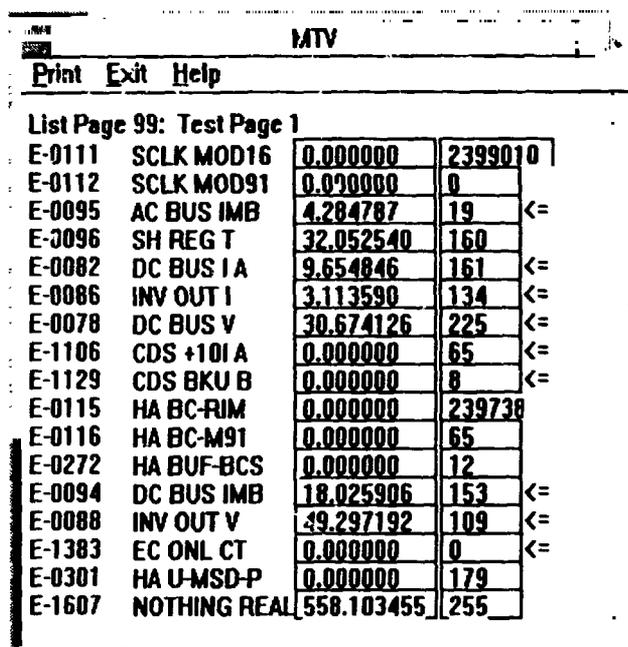


Figure 3.0 MTV Sample List Page Window

These windows can be scaled and sized by the user and seamlessly cut and pasted into other popular Windows applications like Excel™, or Word™. The MTV user can also configure the system to analyze the selected data streams for anomalous science or engineering data outside the range of setpoints. When such conditions are encountered, the MTV system sends a message and alarm to the user at their location of choice, i.e. home, office, or on travel. This feature reduces the time of notification over using the DTV-operator-in-the-loop method. The system interface was also designed to use color in discriminating conditional cues for data states.

Network Interface

MTV currently operates with Novell's LAN Workplace for DOS™ (future adaptations for Microsoft's LAN Manager™ and PC-NFS™ are planned) as the principal network transport interface.

After an initial connection has been made with the Unix server, data is transported across Ethernet (and soon, telecom lines) on a point-to-point basis. If the requesting PC has the proper security registration, it will begin to receive the requested data as a background process that will be ongoing on the PC. On 33 Mhz or faster PCs, the degradation of CPU power by this process is virtually imperceptible. When a user wishes to view a particular page of spacecraft data (a page being a select group of engineering channels-see Figure 3.0), all he or she is required to do is open the MTV window that shows the available pages, mouse-click the desired icon button, and immediately the list page with the latest available data is displayed.

Data Synchronization

One of the significant features in the design of MTV is the functional requirement that *all* users of MTV have access to and view exactly the same data. This aspect was inherent in the earlier analog DTV system. Users who tuned into channel 23 viewed consistent channel 23 data. But with the advent of client-server, distributed computing architectures and custom GUIs, synchronized data views are no longer guaranteed.

With spacecraft alarms capable of being changed at will and independently on all platforms, no one user has the same data viewpoint. This becomes very apparent with data plots. For instance, at time t_1 , a mission controller, monitoring a temperature value, notices the value oscillating in and out of an alarm setpoint over a period of time. When the controller notifies a spacecraft engineer of the condition and requests that they investigate it, the engineer displays his plot at time t_2 and views the data which appears nominal. With MTV, all users

would be seeing the same data/plots, independent of when or where they started the request.

Remote Monitoring and Alarming

Recent tests with portable PCs and Personal Digital Assistants (PDA) with Beeper Notification Systems have demonstrated that MTV can be transmitted over telecom lines via modems. This feature will allow a mission controller almost anywhere in the world to be promptly alerted to data anomalies in science experiments, or failure of a spacecraft component. The MTV server will automatically contact the mission manager's MTV laptop, notebook, or PDA for alarming and immediate access to the required data channel, thus providing continuous 24-hr., 7-days-a-week monitoring of critical data setpoints.

V. FUTURE PLANS FOR MTV

As MTV evolves from the prototype, proof-of-concept stage into a fully-supported system product, several enhancements are planned. The Prototyping Phase has been very useful in developing a solid set of user requirements and continuous product improvement strategy. Currently, there are twenty registered prototype users. Requests for connectivity are increasing daily. The potential exists for over 600 users at JPL, and probably 300 more at remote locations. To further aid in the widespread distribution of information about MTV, a JPL Mosaic Home Page is planned for Internet. Diffusion to other platforms include Macintosh™ versions in Fall '94. When this task is completed, all major desktop platform types at JPL can be supported by MTV.

Specific enhancements and modifications planned for FY '95 include the following:

- ◆ Help System support with spacecraft telemetry data dictionary.
- ◆ Automatic software configuration control and download of current versions and data.

- ◆ DSN tracking, sequence of events, and readiness status reports on demand.
- ◆ Whiteboard shareware, video teleconferencing, and E-mail between cooperating MTV users.
- ◆ System-defined global alarms.
- ◆ Communications port for MS LAN Manager™ for NASA Headquarters.
- ◆ Communications port for PC-NFS networking protocols.
- ◆ Customize GUI for each supported mission.
- ◆ Expand system to include status monitoring of other critical JPL infrastructure-related systems, such as the IBM™ mainframes, Cray™ super-computers, and TV display of the NASA Select and CNN broadcast channels.

These improvements will provide MTV with the capabilities to support an expanding user population and provide data and information when and where they need it at the lowest possible cost. Figure 4.0 shows the MTV prototype suite as it is installed in the Mission Control Center of JPL.

Dual-Use Commercial Applications

The MTV system has commercial applications in manufacturing process control, medical monitoring, and other critical real-time systems requiring automatic feedback loops and adaptive control. Potential commercial projects may be found in the medical, chemical, energy, and process industries.

Physicians, plant managers, researchers, and other decision makers could be instantly notified of critical conditions and monitor key industrial process parameters on their MTV systems. The MTV team is currently in technology transfer discussions with several outside sponsors. Most process control systems are site-localized. MTV, in contrast, is based on the concept of remote monitoring and control.



Figure 4.0 MTV Prototype in Mission Control

MTV Development Strategy

As MTV becomes a fully-supported, mature system at JPL and moves from development to an operational status, the strategy for its continued improvement and success will be contingent upon the following five ongoing activities:

1. Listening and understanding the user's needs.
2. Sponsoring Lab-wide technical demonstrations and communications for potential users.
3. Developing efficient processes—small technical teams, minimum bureaucracy, recognize and foster innovation and new technologies which promote generic multimission designs.
4. Promoting the effective use of outside technology, i.e. integrate COTS technologies and external scientific and engineering innovations.
5. Gaining sponsorship from senior managers and key mission operations teams.

As new missions are planned and costs become key factors in funding decisions, mission planners will be searching for new ways to deliver scientific objectives for less. MTV was designed to provide JPL with a low-cost, flexible alternative by focusing on the ubiquitous PC with its declining cost and increasing power. MTV is proving to be a cost-effective solution to s/c data distribution.

VI. MULTIMEDIA COMMUNICATIONS LABORATORY (MCL) DESCRIPTION

This section describes the technical core competencies and product incubating features of JPL's MCL. The MCL is becoming a focal point in transitioning emerging institutional requirements into high-quality multimedia products such as MTV at the lowest possible cost to users. To realize this goal, the MCL has developed an advanced prototyping center to facilitate the test, evaluation, and insertion of off-the-shelf, interactive, multimedia technology into multimission applications for use by JPL science and engineering teams.

The MCL includes a platform triad of (1) Apple Macintosh Quadra™ 950, (2) Sun Microsystems SPARCstation™ 2, and (3) IBM 486 Ultimedia™ PC. Each of these platforms is equipped with state-of-the-art multimedia subsystems and software libraries for managing full-motion video playback, capture, and editing; graphics; animation and 3-D rendering; and interactive authoring applications.

These systems are connected to Ethernet, ISDN, and FDDI networks to aid in the investigation of local and global multimedia compression and transmission requirements. Each system is also linked to a central, multichannel, high-resolution, large screen RGB projector for technical evaluation demos and briefings involving MCL developments. Figure 5.0 shows an overview of the MCL architecture.

In addition to serving multimission requirements at JPL, other institutional applications include TCP/IP multimedia electronic mail, scientific visualization, technical and executive-level presentation, interactive CD-ROM/video-disk training and educational authoring, photographic image and video storage and retrieval, video teleconferencing/groupware, and management of engineering data libraries. The JPL MCL was also designed to investigate bandwidth requirements, packet video transmission, compression effects on visual quality, user ergonomics, storage requirements, productivity, and feasibility of network distribution of

video, images, and HDTV broadcasts to NASA science centers worldwide.

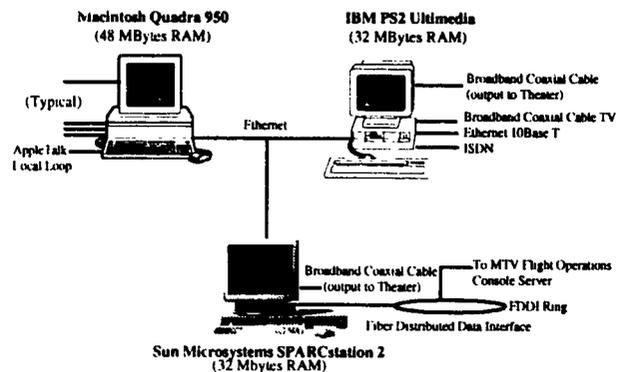


Figure 5.0 MCL Desktop Architecture

VII. MCL PLATFORM DESCRIPTIONS

Macintosh Quadra 950-This system is a high-end PC workstation which has a variety of multimedia devices integrated for playback, capturing, editing, and producing full-motion video from any National Television Standards Committee (NTSC) Source. This system has the primary task of multimedia content production and visual data base management. Once the content material is developed, it is converted to target file formats and transmitted to other multimedia workstations. It is capable of producing video tapes, 35mm slides, CD-ROMs, high-resolution photographs, and technical presentations in a Science Conference Center. The system has 48 Mbytes of RAM and 6.5 Gbytes of magnetic and optical disk storage. It is equipped with a Radius VideoVision/Studio™ high performance A-D converter and compression/decompression subsystem. The system has a professional flatbed scanner and film recorder capable of 4000 lines resolution output. Audio is sampled at 16-bits and the system has a built-in speaker and microphone. Video teleconferencing is accomplished through a special interface card which is connected to an ISDN switch. Ethernet and AppleTalk™ are used for local communication, and connection to the FDDI ring is planned.

Desktop display is accomplished by dual-16 and 13-inch Apple monitors. Full-screen, full-motion 640x480 digital video is displayed on the smaller monitor while desktop applications are displayed on the larger unit.

Sun Microsystem SPARCstation 2- This RISC-based workstation is used for video teleconferencing and groupware test and evaluation. The system has a X-Video™ A-D video converter (S-bus subsystem) which allows the simultaneous display of two video sources using the Joint Photographic Experts Group (JPEG) compressor. The system is equipped with a dedicated CD-ROM drive and video camera for video teleconferencing. Broadcast TV and other NTSC sources can also be displayed, captured, and stored. A built-in microphone, speaker, and headset jack allows audio input and output. The system runs BSD Unix version 4.1.2 with Open Windows™ 3.0 GUI. A high resolution 21" monitor allows very detailed pixel manipulation of graphical imagery in 16 million colors. The system has 32 Mbyte of RAM and 1.5 Gbyte of magnetic storage. The system has a frame buffer which allows 3-D modelling and rendering at 310 vectors/sec. The communications interface includes an FDDI dual attach connection which is on a subnet fiber ring.

IBM Ultimeia M77 486 DX2- This system is the most powerful desktop multimedia system in IBM's product line. The system comes equipped with a 66/33 Mhz cached microprocessor and includes a math coprocessor. Its features include 32-bit bus architecture, XGA non-interlaced graphics adapter, and 32 Mbyte of RAM. The system has a built-in CD-ROM Drive, and a 212 Mbyte capacity hard drive. Multimedia content is displayed at 640x480 pixels with a pallet of 65K colors, or 1024x768 at 256 colors. Audio is processed through the M-Audio Capture and Playback Adapter with analog conversion to and from a digital PCM data format at 8- and 16- bit stereo with sampling rates up to 44KHz. Digital audio processing is 16-bit ADPCM compression, CD-extended architecture audio decompression, mix line in with PCM audio. The system is equipped with an Action Media II™ display adapter which uses Intel's

proprietary i750 Digital Video Interactive (DVI/Indio)™ chip. This subsystem allows 72 minutes of full-motion, compressed video to be recorded and played back on a standard CD-ROM, in addition to a variety of other input devices. This platform is connected to JPL's ILAN and shares many of the Quadra's video and desktop multimedia systems.

VIII. MCL PROJECT PORTFOLIO

The MCL has been under development since 1992 and recently achieved operational status. It now supports several JPL projects including MTV. Other projects are described below:

Science Conference Center- The MCL controls a complete digital theater projection environment from any of its platforms. Support includes real-time visualization episodes, executive and technical presentations and technology demonstrations, or group video teleconferencing.

DSN Archiving Project- As the construction of the next generation of advanced tracking antennas progresses, video footage is collected for each stage of construction. This video has been stored on laser disk. MCL software for rapidly retrieving analog video clips by DSN personnel was provided. Currently a series of digital, interactive CD-ROMs is under development for rapidly navigating and displaying digital source material, i.e. video clips, narration, still photographs, CAD drawings, etc.

Video Conference Center Design, T&E- To effect better communications among suppliers and subcontractors and to lower travel costs, a low-cost video teleconferencing center is being designed for a major JPL instrument project. The MCL is designing and integrating the subsystems for this project. International and domestic connectivity is being provided through ISDN switched technology.

Robotic Vehicle Communications, T&E- The MCL recently tested the remote control of a lunar rover using an Asynchronous Transfer Mode (ATM) switch and FDDI ring for transmission of JPEG compressed video telepresence signals.

Macintosh and Quadra are registered trademarks of Apple Computer Corporation.

AppleTalk is a registered trademark of Apple Computer Corporation.

Cray is a registered trademark of Cray Research, Inc.

IBM and Ultimedia are registered trademarks of IBM Corporation

Action Media II is a registered trademark of Intel Corporation.

DVI is a registered trademark of Intel Corporation.

Indio is a registered trademark of Intel Corporation.

Windows is a registered trademark of Microsoft Corporation.

Excel is a registered trademark of Microsoft Corporation.

LAN Manager is a registered trademark of Microsoft Corporation.

Word is a registered trademark of Microsoft Corporation.

LAN Workplace for DOS is a registered trademark of Novell Corporation.

X-Video is a registered trademark of Parallax Corporation.

VideoVision Studio is a registered trademark of Radius Corporation.

PC-NFS is a registered trademark of Sun Microsystems Computer Corporation.

SPARCstation is a registered trademark of Sun Microsystems Computer Corporation.

Open Windows is a registered trademark of Sun Microsystems Computer Corporation.

VLSI Technology for Smaller, Cheaper, Faster Return Link Systems

Kathy Nanzetta, Parminder Guman, Toby Bennett
Data System Technology Division, Code 520

Jeff Solomon
RMS Technologies, Code 520.9

Jason Dowling, John Welling
Loral Aerosys, Code 521

Mission Operations and Data Systems Directorate
NASA, Goddard Space Flight Center
Greenbelt, Maryland 20771

ABSTRACT

Very Large Scale Integration (VLSI) Application-specific Integrated Circuit (ASIC) technology has enabled substantially smaller, cheaper, and more capable telemetry data systems. However, the rapid growth in available ASIC fabrication densities has far outpaced the application of this technology to telemetry systems. Available densities have grown by well over an order magnitude since NASA's Goddard Space Flight Center (GSFC) first began developing ASICs for ground telemetry systems in 1985. To take advantage of these higher integration levels, a new generation of ASICs for return link telemetry processing is under development. These new submicron devices are designed to further reduce the cost and size of NASA return link processing systems while improving performance. This paper describes these highly integrated processing components.

1. INTRODUCTION

The rapid growth of chip fabrication densities has had a tremendous positive impact on NASA telemetry data systems. Each year, new data system implementations are getting smaller, cheaper, and more powerful due to the availability of higher integration components developed through improved VLSI fabrication processes. For ground telemetry systems, many of these components are the latest standard commercial microprocessors and solid-state memories developed for general purpose computing. Although general purpose components have improved telemetry data system implementations, even greater improvements have been gained with the addition of components developed specifically for telemetry processing.

The Data Systems Technology Division (DSTD) at NASA GSFC first began developing VLSI ASIC components for ground telemetry processing in 1985 [1]. This effort led to a series of more than a dozen different telemetry processing components implemented in silicon and Gallium Arsenide (GaAs) technologies. The high integration levels offered by these components enabled the development of VLSI-based systems that offered an order of magnitude improvement in performance, cost, and size over previous telemetry processing implementations. The inherent advantages of these systems has led to their wide spread use across a number of NASA programs. Over 100 VLSI-based telemetry systems have been deployed in support of such programs as the Small Explorer missions [2,3], Deep Space Network (DSN), Hubble Space Telescope (HST), and the Earth Observing System (EOS).

Integrated circuit technology has progressed very rapidly since the DSTD developed its first series of VLSI telemetry processing components. Many of the original ASIC components still in use were implemented in 2 micron semi-custom Complementary Metal Oxide Semiconductor (CMOS) gate array technology. This technology capable of fabricating parts with usable densities of up to 10,000

logic gates. While such levels of integration were impressive at the time, they are modest by today's standards. With available densities approaching 1,000,000 logic gates, current submicron semi-custom technologies offer an opportunity to once again improve the performance and shrink the cost and size of telemetry data systems.

To make full use of today's available VLSI densities, the DSTD is developing a new series of very high integration VLSI ASIC components for return link telemetry processing. The new ASICs are intended to integrate into a single device much of the functionality contained in current printed circuit card subsystems. This is expected to reduce system reproduction costs to well less than 20 % of the cost of previous VLSI implementations. These next generation VLSI components are targeted towards space science missions using the widely adopted packet telemetry protocols recommended by the Consultative Committee for Space Data Systems (CCSDS). However, they also offer a level of programmability and generic capability that make them useful for other missions with unique protocol formats. Efforts to date have focused on the development of three individual ASICs. One ASIC for Reed-Solomon error correction has been implemented and tested. Two more ASICs for frame synchronization and CCSDS service processing are currently in the design stage. In this paper, we first describe the functions required for CCSDS return-link processing. We then describe the architecture and key features of each of the three next generation VLSI components that implement these functions.

2. CCSDS RETURN LINK PROCESSING

In the past, telemetry formats tended to be unique for each new spacecraft. This led to the successive development of telemetry data systems for each new mission. This mission-unique development cycle led to very high costs for the acquisition and maintenance of data systems on a NASA-wide basis. To reduce these costs and to promote interoperability between ground processing elements, NASA adopted space data protocol standards outlined by the CCSDS, an international collaborative body composed of many of the world's space agencies. Most future NASA missions are now planning to use the CCSDS protocols. This yields great potential for significant cost savings across all future NASA flight and ground data systems.

Return link processing is an area of particular interest for cost savings. Systems implementing return link functions are used throughout NASA in ground stations, control centers, science data processing facilities, spacecraft verification equipment, compatibility testing, and launch support facilities. Demand for CCSDS return link processing systems is expected to increase dramatically beyond current uses with the advent of the EOS program. Starting in 1998, the EOS will fly a series of remote sensing spacecraft to monitor the earth's environment. Many of these spacecraft will be capable of broadcasting CCSDS formatted science data directly to the user. Because of the scope of the EOS program, it is expected that there will be a large user base for direct broadcast data and, therefore, an even greater demand for cost-effective CCSDS return link processing systems.

Return link processing takes place after the acquisition, demodulation, and digitization of signals transmitted from the spacecraft. Return-link processing systems generally extract framed data from incoming serial bit streams, correct framed data, validate the protocol structures within the frame, and extract user data. Figure 1 depicts an example return link processing chain for packetized CCSDS telemetry.

Frame synchronization is the process of delineating framed data structures from the incoming serial bit stream. CCSDS telemetry uses a specific pattern to mark frame boundaries. Because space to ground transmission induces numerous types of data disturbances, NASA frame synchronizers employ sophisticated measures in searching for these markers to ensure correct synchronization of data.

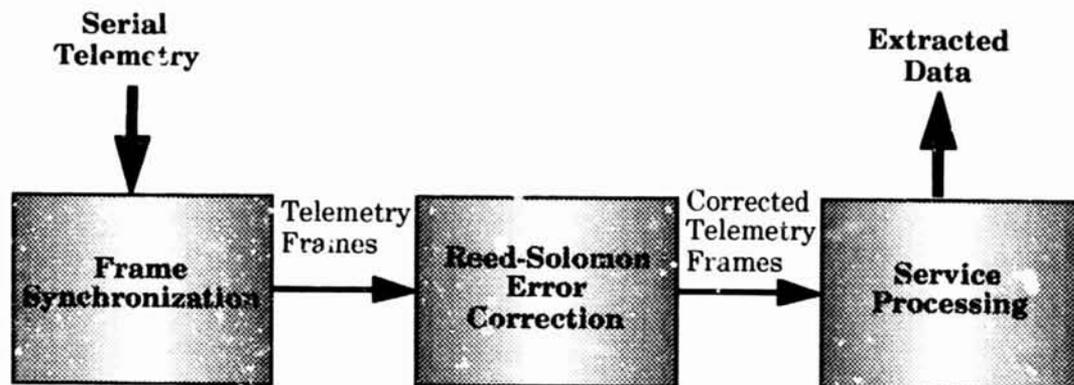


Figure 1. Example CCSDS Return Link Processing Chain

Reed-Solomon error correction removes errors introduced during the transmission process. The CCSDS recommendations specify a very powerful block error correction code to protect internal data and protocol structures of the frame. This Reed-Solomon code is applied prior to transmission in the form of appended check symbols. The data and check symbols together represent code words that are decoded on the ground to correct transmission errors. To increase the burst error correction capability of the code, a technique known as interleaving is used. Interleaving systematically alternates the symbols of multiple code words within a frame so that when a burst error occurs, it is distributed across more than one code word.

CCSDS Service Processing demultiplexes, extracts, and validates user data from the composite stream of telemetry frames. The CCSDS protocols use data driven protocol constructs and the concept of virtual channels to assign portions of the composite stream to different spacecraft instruments. Virtual channel assignments also identify the kind of processing to be performed on the data from each instrument. To extract user data, Service Processing uses protocol constructs contained in the headers of telemetry frames to identify virtual channels and the type of processing required. Packet data types are generally the most complex data type to process because individual virtual channels can carry multiplexed streams of variable length packets from different sources. Packets are also allowed to span the frames of a given virtual channel. Therefore, packet processing requires not only locating, extracting, and validating packets in frames but also piecing together packets that cross frame boundaries.

Current VLSI-based CCSDS return link telemetry systems developed by the DSTD implement these functions on several 9U form factor VMEbus printed circuit cards. The cards are densely populated with commercial VLSI components and the DSTD's first generation ASIC components using dual sided surface mount technology and plug-in daughter card assemblies. The three next generation of ASIC components currently under development will allow integration of all these functions onto a single card. Each of these new 0.6 micron CMOS ASICs is designed to minimize the number of required supporting components. Remaining supporting components mainly consist of commercial high density memories which are expensive to implement in ASIC technology. The three next generation components, described in the following paragraphs, are known as the Parallel Integrated Frame Synchronizer chip, the Reed-Solomon Error Correction chip, and the CCSDS Service Processor chip.

3. PARALLEL INTEGRATED FRAME SYNCHRONIZER CHIP

The Parallel Integrated Frame Synchronizer (PIFS) chip is currently under development and is scheduled to be completed in summer of 1995. It is being designed using 0.6 micron CMOS gate array technology. As its name implies, the PIFS chip uses a parallel algorithm to perform telemetry frame synchronization. This is different from previous generation VLSI frame synchronizer chips which used serial processing approaches. While parallel approaches are not new, they require significantly more

logic than serial approaches. Because new VLSI processes have so greatly lowered the cost and size of logic functions, greater complexity is no longer a disadvantage.

Figure 2 depicts the expected cost, size, performance, and power advantages of the next generation frame synchronizer based on the new PIFS chip as compared to present VLSI serial implementations. Currently, the DSTD uses two different frame synchronizers to meet the full range of NASA spacecraft data rates at reasonable costs. A frame synchronizer based on lower cost CMOS technology is used for telemetry data rates up to 20 Megabits per second (Mbps) [4]. For higher data rates, a more expensive frame synchronizer based on GaAs and Emitter Coupled Logic (ECL) technology is used [5]. With operating rates up to 500 Mbps, the PIFS-based next generation frame synchronizer will replace both implementations at a lower cost. This will allow the additional logistical advantage of having one frame synchronizer that meets the full range of NASA spacecraft data rates.

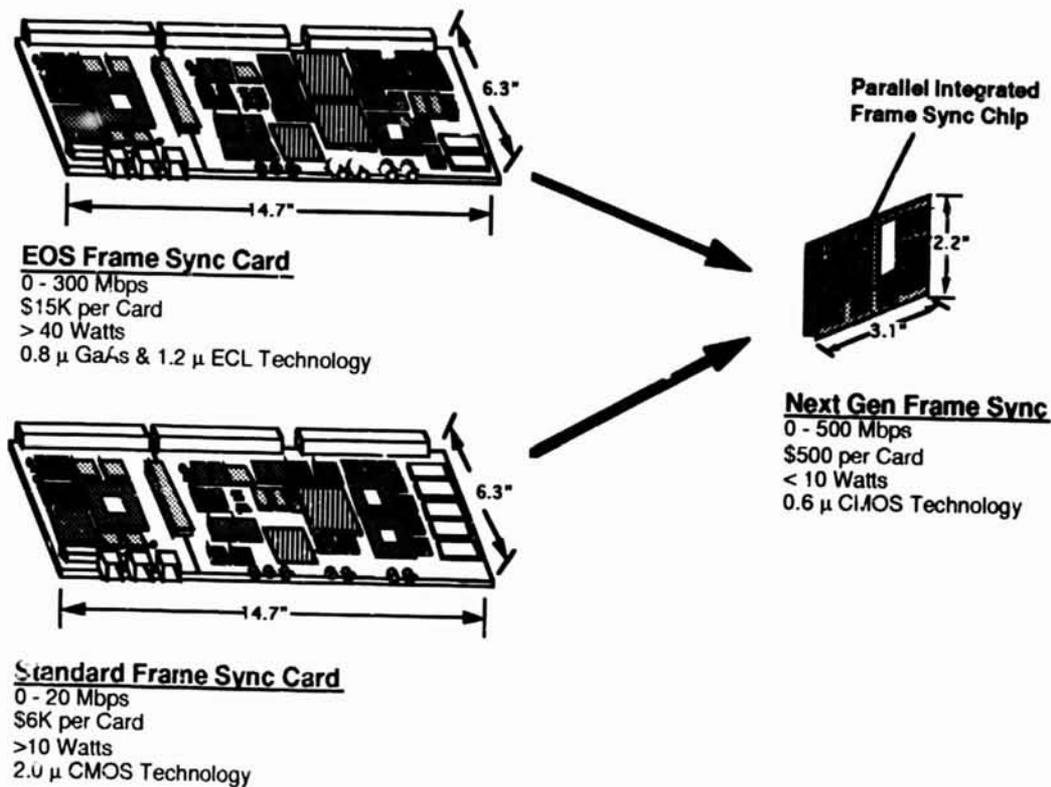


Figure 2. Next Generation Frame Synchronizer Targeted Level of Integration

A functional block diagram of the PIFS chip is shown in Figure 3. The PIFS chip is controlled by a set of internal registers that are configured through a standard microprocessor interface prior to operation. The registers allow programmability to meet the needs of many different space missions. Marker patterns, compare masks, correlator tolerances, acquisition strategy, and slip tolerances are just a few of the programmable parameters within the registers. During operation, data enters the chip in one of two ways. For very high data rates, serial data is first externally converted to byte-wide parallel data and then fed into an internal First-In, First-Out (FIFO) memory. For rates below 50 Mbps, serial data can be input directly where it is converted to parallel on-chip. The use of the FIFO memory allows the PIFS internal logic to run off a separate master clock. This feature coupled with a data-flow architecture has several advantages over previous implementations including lower latency, easier processing of nested or asynchronously blocked data, and automatic pipeline flushing. As data passes

through the chip, correlations are performed, synchronizer locations are calculated, and data is aligned to frame boundaries before being output. The PIFS chip generally contains a superset of the functions contained in previous VLSI card-level implementations including cumulative quality accounting, time stamping, and real-time quality trailer generation. One exception is reverse data handling which becomes unnecessary with the ubiquitous future use of onboard Solid State Recorders. If compatibility with older spacecraft using tape recorders is required, this function can be accomplished by adding external Programmable Logic Devices (PLD) and random access memories. The PIFS also adds the capability to synchronize all current weather satellite formats.

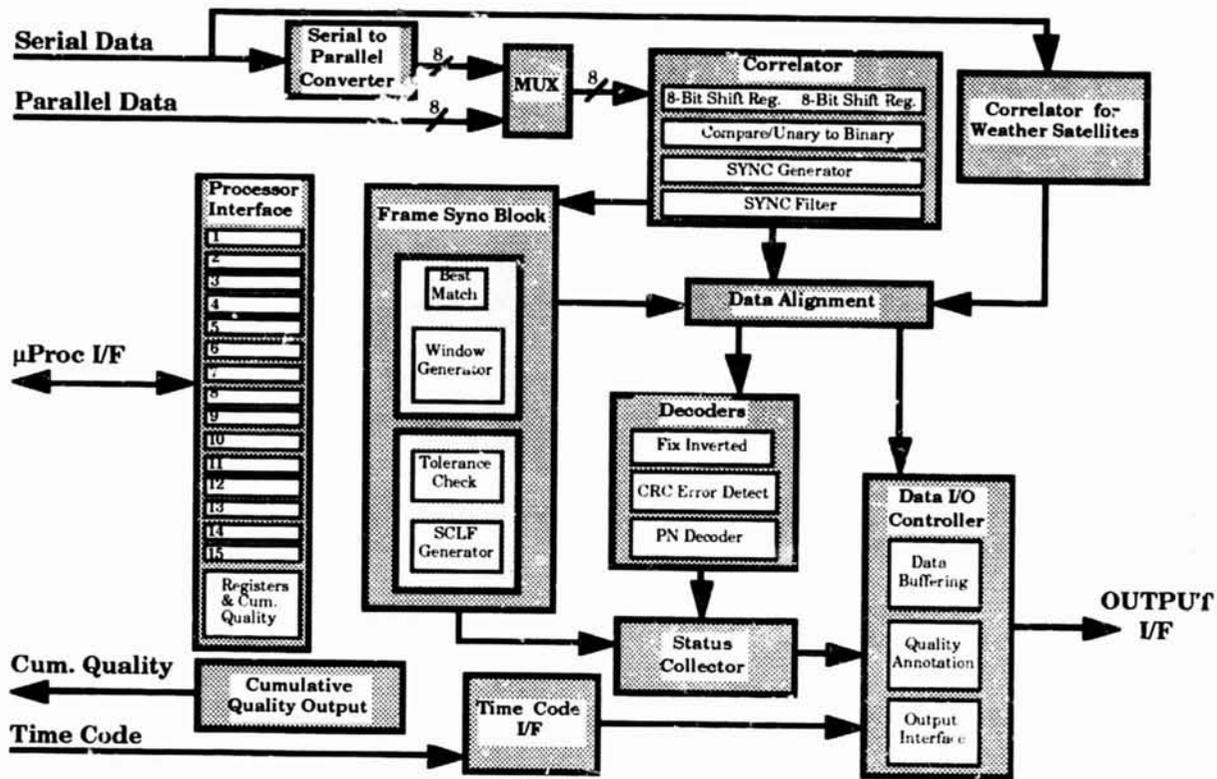


Figure 3. PIFS Chip Functional Block Diagram

4. Reed-Solomon Error Correction Chip

With over 125,000 logic gates, the Reed-Solomon Error Correction (RSEC) chip is the largest ASIC implemented to date by the DSTD. This chip, completed in February 1994, is implemented in 0.6 micron CMOS hybrid standard cell / gate array technology. The RSEC chip integrates much of the functionality currently contained on two VMEbus card subsystems. Elements of the RSEC chip include CCSDS Reed-Solomon block and header decoders, 16 KBytes of synchronous random access memory, and a pipeline of four memory controllers that perform deinterleaving, error correction, real-time quality annotation, and frame filtering and routing. The chip has been recently tested at sustained operating rates of well over 500 Mbps. This level of integration and performance coupled with the complexity of the CCSDS Reed-Solomon code make it quite possibly the most powerful error correction device in the world!

A functional block diagram of the RSEC chip is shown in Figure 4.

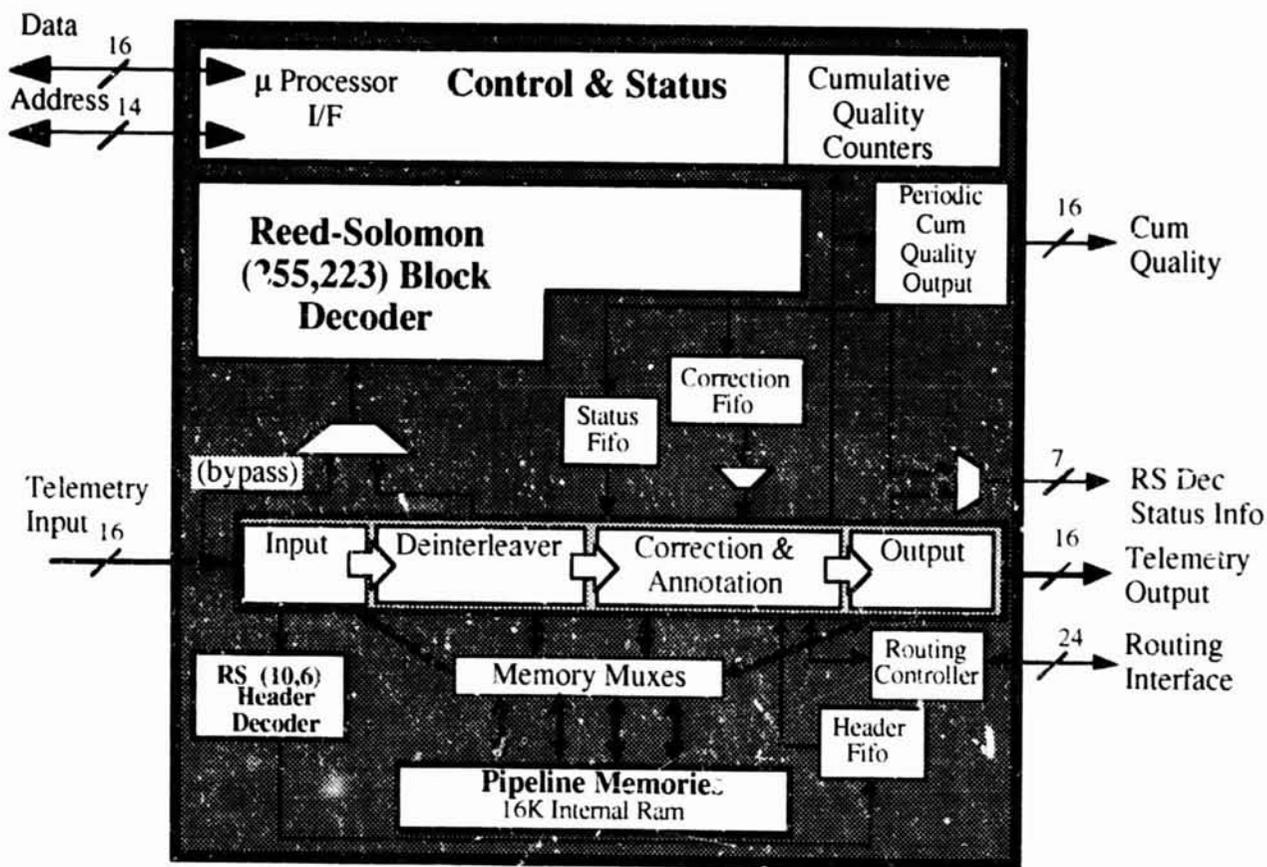


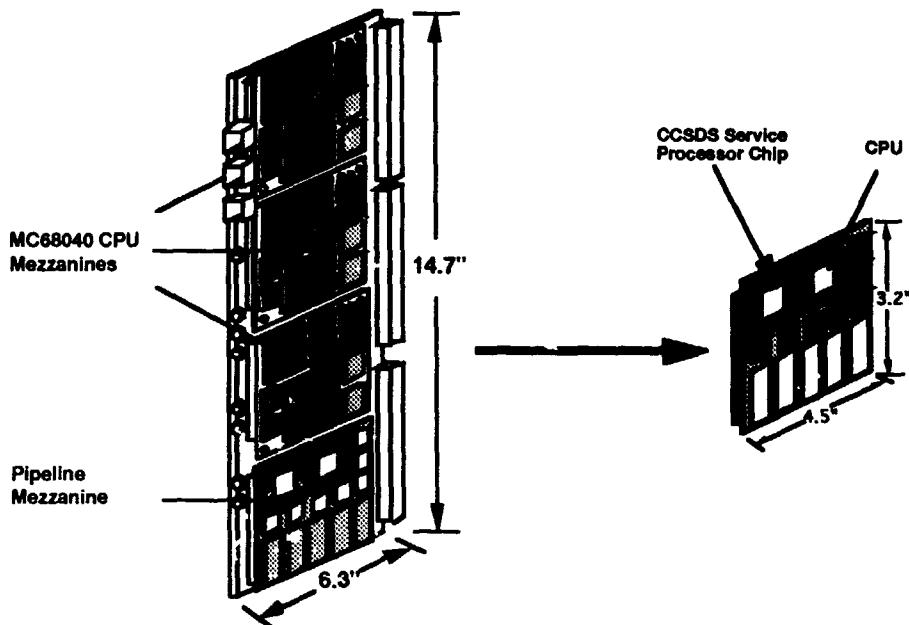
Figure 4. RSEC Chip Functional Block Diagram

5. CCSDS Service Processor Chip

The CCSDS Service Processor (CSP) chip is currently under development and is expected to be completed in the summer of 1995. Implementing the CSP chip is the most ambitious of the three chip development efforts. It involves a radical change from the current Service Processor card architecture. The current Service Processor has the highest complexity of any card implemented by the DSTD. It employs three Motorola MC68040 processors, five VLSI ASIC components, over 6 MBytes of memory, and a host of commercial VLSI components. When the architecture for this card was first developed in the late 1980's [6], the CCSDS protocols were still under development. To maintain the flexibility necessary to accommodate changes in the protocols, most of the workload was implemented in software running on the three processors. Only the most generic data movement and extraction functions were placed in the VLSI ASICs.

The CCSDS recommendations have since stabilized allowing many of the current software functions to be integrated into the CSP chip. A major goal of the CSP chip is to reduce implementation cost by integrating enough functions to eliminate two of the current card's processors. Another goal of the CSP chip is to significantly increase packet processing throughput to alleviate a shortcoming in the CCSDS recommendations which allows the creation of very small packets. At even modest data rates, small packets can lead to very high packet rates. Because each packet requires a similar amount of protocol processing, bursts of small packets can quickly overload ground processing systems. Even the current Service Processor, by far the highest performance implementation to date, is easily overwhelmed by bursts of very small packets at relatively low data rates. Moving many of these functions into the CSP

chip is expected to substantially increase packet throughput. The targeted level of performance and integration of the next generation Service Processor based on the CSP chip is depicted in Figure 5.



Current Service Processor Card

50 Megabit per second
 15,000 CCSDS packets per second
 \$12K per card
 3 On-board High Performance CPUs

Next Generation Service Processor

300 Megabit per second
 100,000 CCSDS packets per second
 \$1.5K per card
 Single medium performance CPU

Figure 5. Next Generation Service Processor Targeted Level of Integration

6. CONCLUSION

The development of three new VLSI ASIC components for return link telemetry processing has been discussed. These components are planned for use in the next generation of VLSI systems now under development and will find application in a number of NASA ground data systems. The RSEC chip is already planned for delivery in high performance systems that will be used in the integration and test of the EOS-AM spacecraft and the EOSDIS Test System. The full complement of components will be first demonstrated in a prototype very low-cost ground capture and processing station for EOS direct broadcast data.

This new generation of return link processing components will help lower the cost and increase the performance of NASA's future data systems. However, return link processing is just one of many areas where high integration ASIC technology can be used to create cost effective system solutions. In the future, the DSTD will target high integration ASIC solutions to lower the cost of digital signal processing, telemetry stream simulation, and science data processing.

7. REFERENCES

1. Chesney, J., Speciale, N., Horner, W. and Sabia, S., "High Performance VLSI Telemetry Data Systems", Proceedings of the American Institute of Aeronautics and Astronautics, Second International Symposium on Space Information Systems, 1990.
2. K. Brentzel and N. Speciale, "Small Explorer Project (SMEX) Telemetry Processing Systems," Proceedings of International Telemetry Conference, 1993.
3. J. Shi, T. Mao, J. Chesney, and N. Speciale, "Fast Auroral Snapshot Explorer (FAST) packet Processing System (PPS)," Proceedings of International Telemetry Conference, 1993.
4. N. Speciale and K. Wunderlich, "High Speed Frame Synchronizer Utilizing VLSI Technology" Proceeding of the International Telemetry Conference, November 1988.
5. K. Wunderlich, J. Chesney, and T. Bennett, "Frame Synchronization at 300 Mbps and Beyond..." Proceeding of the International Telemetry Conference, November 1990.
6. G. Grebowsky and C. Dominy, "VLSI High Speed Packet Processor" Proceeding of the International Telemetry Conference, November 1988.

8. NOMENCLATURE

ASIC	Application-specific Integrated Circuit
CCSDS	Consultative Committee for Space Data Systems
CMOS	Complementary Metal Oxide Semiconductor
CSP	CCSDS Service Processor
DSN	Deep Space Network
DSTD	Data Systems Technology Division
FIFO	First-In, First-Out
GSFC	Goddard Space Flight Center
HST	Hubble Space Telescope
ECL	Emitter Coupled Logic
EOS	Earth Observing System
EOSDIS	Earth Observing System Data Information System
FIFO	First-In, First-Out
GaAs	Gallium Arsenide
Mbps	Megabits per second
MO&DSD	Mission Operations and Data Systems Directorate
NASA	National Aeronautics and Space Administration
PIFS	Parallel Integrated Frame Synchronizer
PLD	Programmable Logic Devices
RSEC	Reed-Solomon Error Correction
VLSI	Very Large Scale Integration

Telemetry Distribution and Processing For The Second German Spacelab Mission D-2

E. Rabenau Satellite Operational Services GmbH
W. Kruse Deutsche Forschungsanstalt für Luft- und Raumfahrt (DLR)

Abstract

For the second German Spacelab Mission D-2 all activities related to operating, monitoring and controlling the experiments on board the Spacelab were conducted from the German Space Operations Control Center (GSOC) operated by the Deutsche Forschungsanstalt für Luft- und Raumfahrt (DLR) in Oberpfaffenhofen, Germany. The operational requirements imposed new concepts on the transfer of data between Germany and the NASA centers and the processing of data at the GSOC itself. Highlights were the upgrade of the Spacelab Data Processing Facility (SLDPF) to real time data processing, the introduction of packet telemetry and the development of the high-rate data handling front end, data processing and display systems at GSOC. For the first time, a robot on board the Spacelab was to be controlled from the ground in a closed loop environment. A dedicated forward channel was implemented to transfer the robot manipulation commands originating from the robotics experiment ground station to the Spacelab via the Orbiter's text and graphics system interface. The capability to perform telepresence from an external user center was implemented. All interfaces proved successful during the course of the D-2 mission and are described in detail in this paper.

INTRODUCTION

D-2 was launched on April 26, 1993. It successfully accomplished the mission goal

Acronyms

CCSDS	Consultive Committee for Space Data Systems
CAP	Command Acceptance Patterns
CDF	Command Data File
CMD	Command
DLR	Deutsche Forschungsanstalt für Luft- und Raumfahrt
DPS	Data Processing System
DTS	Data Transfer System
ECIO	Experiment Computer Input/Output
EGSE	Experiment Ground Support Equipment
GSFC	Goddard Space Flight Center
GSOC	German Space Operations Control Center
HDRR	High Data Rate Recorder
HRFL	High Rate Forward Link
HRM	High Rate Multiplexer
ISC	Johnson Space Center
KUSP	Ku-Band Signal Processor
LAN	Local Area Network
OD	Operational Downlink
PCM	Pulse Code Modulation
PPF	Payload Parameter Frame
SCIO	Subsystem Computer Input/Output
SIPS	Spacelab Input Processing System
SLDPF	Spacelab Data Processing Facility
SPARCS	Spacelab Realtime Packet Converter System
STL	Subtimeline
TAGS	Text and Graphics System
TDM	Time Division Multiplexer
WAN	Wide Area Network

after 10 days in space. To meet the mission objectives the D-2 mission management required the control center to handle all experiment related data, voice, video and other communication services which resulted in the following requirements:

- (1) Transferring the complete Spacelab high rate telemetry to GSOC. Capturing, storing and distributing the data.
- (2) Processing the Spacelab Experiment Computer and Subsystem Computer Input/Output (ECIO/SCIO) as part of the high rate telemetry and making it available for "quick-look" display to allow the mission operations support team to monitor the progress of the mission, status and health of the experiment computer and facilities.
- (3) Transferring, capturing, processing and displaying the Payload Parameter Frame (PPF) telemetry. The PPF consisted of 151 data words of the Spacelab operational downlink processed by the Johnson Space Center (JSC).
- (4) Forwarding command data blocks originating from the GSOC Command System in order to remotely manipulate the Spacelab and experiment computers.
- (5) Receiving and distributing command acceptance and trajectory data blocks.
- (6) Supporting robot experiment operations by forwarding manipulation commands with minimum delay.
- (7) Supporting telescience operations from the Microgravity User Center in Cologne, Germany.
- (8) Supporting the retrieval of data in chronological order from the data archive in order to facilitate troubleshooting and detailed evaluation of the progress of the experiment.
- (9) Maintaining the original data rates in the order of one update/repetition per

second for most data channels in regard to distribution, processing and display.

- (10) Delivering the experiment-related high rate Spacelab telemetry to processing equipment provided by the experimenters. The computers installed and operated by the experimenters are hereafter referred to as Electrical Ground Support Equipment (EGSE)
 - (i1) Offering services prior to the actual mission during the Spacelab integration phase and the "integrated" simulations. Interfaces were implemented to support data flow from the Spacelab integration facility and the Spacelab Training Assembly, both located in Northern Germany. These interfaces, along with the voice, video, fax and other communication services necessary for the overall operations, are not discussed in this paper.

TRANSFER OF TELEMETRY DATA

The concept of the transfer of data as implemented from the requirements is pictured in Figure 1. It distinguishes between the high-rate data generated on board Spacelab and auxiliary data transferred in NASCOM block data format.

Spacelab High Rate Telemetry

The Spacelab provided a high-rate data system for the transfer of the ECIO, SCIO and PCM-experiment-science data. The ECIO and SCIO contained housekeeping, monitoring and low-rate science data. 5 experiment and 2 computer output channels with an overall bitrate of 607.39 kbps were processed by the High Rate Multiplexer (HRM) and nominally down linked at 1 Mbps via the Ku-band signal

processor (KUSP). During Ku-band loss of signal the data was recorded onto the High Data Rate Recorder (HDRR) from where it was regularly dumped at a rate of 24:1. Table 1 provides a listing of the HRM channels and data rates.

Table 1 HRM Telemetry Channels

HRM Channel	Data Rate kbps
Experiment 1	15.39
Experiment 2	248.0
Experiment 3	81.92
Experiment 4	164.8
Experiment 5	20.48
ECIO	51.2
SCIO	25.6
Total	607.39

The composite HRM signal was captured, processed and transferred to the GSOC by the Data Transfer System (DTS) implemented at the Spacelab Data Processing Facility (SLDPF), Goddard Space Flight Center (GSFC). The DTS incorporated the enhanced Spacelab Input Processing System (SIPS) and the newly developed Spacelab Realtime Packet Converter System (SPARCS).

The DTS consisted of a triple redundant system that provided real time processing and performed the following tasks:

- (1) Captured the composite HRM onto tapes and monitored its data quality
- (2) Extracted 7 HRM data, the HDRR dump, voice and time channels and monitored their quality
- (3) Captured the extracted HDRR dump on tapes and played back the data at the nominal data rate upon request in

parallel to the real-time data transmissions

- (4) Processed the HRM data by the SIPS which output data records containing a number of HRM minor frames for each data channel
- (5) Packetized the data records into Transfer Frames by the SPARCS and output the frames at a rate of 1024 kbps.
- (6) Replayed data from the tapes upon request in parallel to the real-time data transmissions.

All minor frames of the HRM data, output by the SLDPF, were frame synchronized and tagged with Spacelab time and downlink quality.

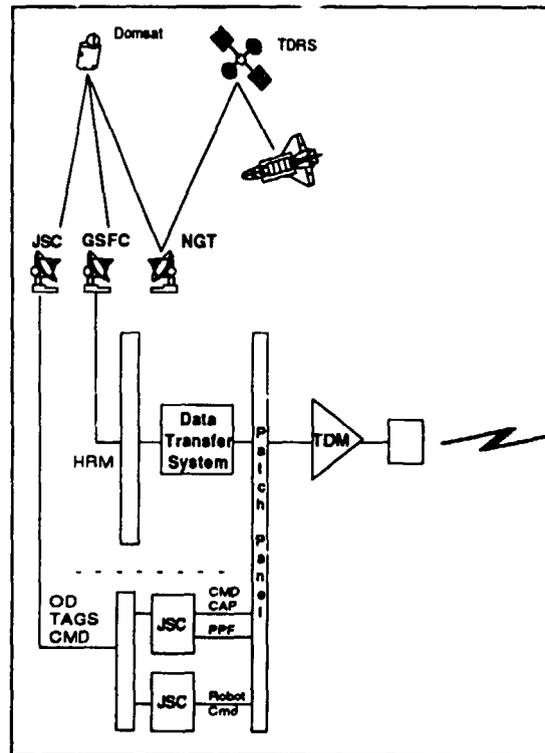


Figure 1 Telemetry Data Flow Schematic

Auxiliary Data

Auxiliary data to support the mission was exchanged between the Johnson Space Center (JSC) and the GSOC. This included the PPF data set which was downlinked as part of the Orbiter Downlink data stream using either channel 1 of the KUSP or S-band propagation, processed and output by JSC at the rate of 1 block per second.

Command data blocks generated by the GSOC command system were forwarded to JSC at the maximum rate of 1 command data block per 3 seconds.

The robot experiment remote manipulation commands which were generated by the robot ground station were forwarded by GSOC to JSC using a dedicated link (HRFL). A new interface was implemented at JSC to validate and merge the commands into the Text and Graphic System channel (TAGS) for high speed uplink.

JSC periodically output trajectory messages (TRAJ) and generated acceptance data blocks (CAP) in response to the GSOC commands.

Time Division Multiplexers and Satellite Links

A triple redundant Time Division Multiplexer (TDM) system was set up both at GSFC and GSOC and provided the gateway to the communication links across the Atlantic. The TDM operated at an aggregate rate of 2048 kbps and incorporated 3 data, 22 voice, 2 FAX and 1 WAN channel. Two diverse satellite paths via two different Intelsat satellites transported the TDM aggregate. Ground stations were installed at GSFC and GSOC. Nominally one satellite link was dedicated for the transfer of real-

time and the other for the transfer of playback and replay data.

TELEMETRY DATA PROCESSING

The GSOC handled the D-2 telemetry data by its front end, data processing and display systems and communicated with EGSEs provided by the experimenters. Figure 2 shows the GSOC systems in a high-level functional block diagram focusing on the interconnections between the systems. Figure 3 depicts the functional flow of data and the user interfaces.

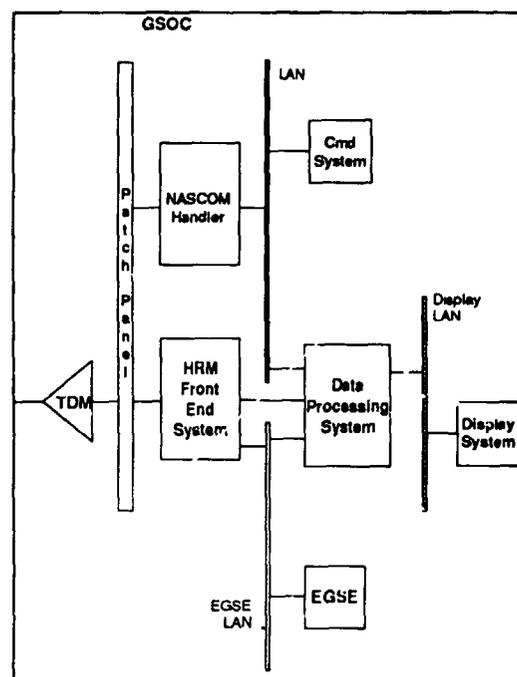


Figure 2 Functional Block Diagram

Front End Processing System

All incoming and outgoing telemetry data were transported to and from the GSOC Front End System via the Time Division Multiplexer system (refer to Figures 1/ 2).

- (1) Redundant APTEC® high speed input/output processors were

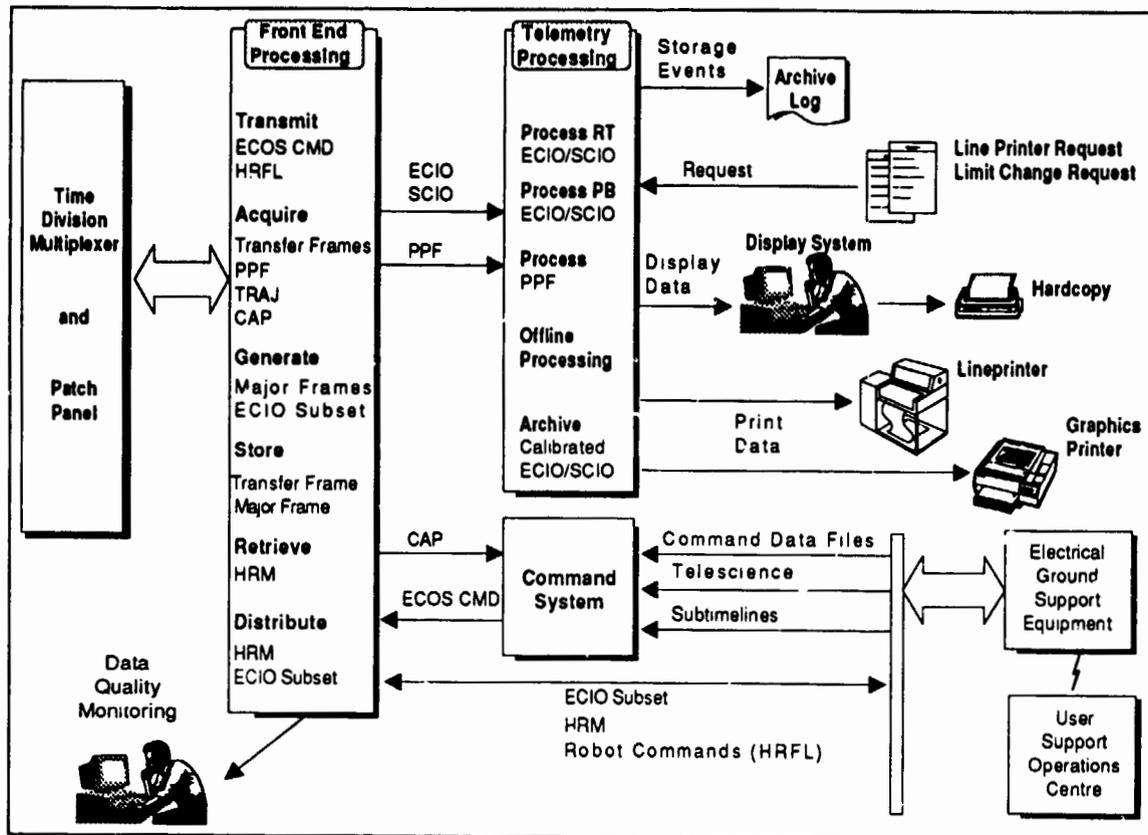


Figure 3 D-2 GSOC Data Processing User Interfaces

implemented to process the Transfer Frames and embedded data. Nominally the APTEC® real time and hot backup systems were operated in parallel. In detail the high-rate front end system performed the following tasks.

- (a) Two frame synchronizers received the Transfer Frames from two different satellite links in parallel at 1024 kbps each and forwarded both data streams to the processor which performed all operations in parallel.
- (b) The 5 HRM data channels were restored from the Transfer Frames and synchronized on major-frame level (optional).
- (c) Quality monitor information of the Transfer Frames and the embedded Spacelab minor frames was generated and displayed for on-line evaluation.
- (d) Subsets from the ECIO and SCIO were generated and transferred to the Data Processing System via a special backup interface at the rate of 80 kbp
- (e) HRM data from the ECIO subsets were converted into standard GSOC network record blocks providing time-tags and data quality information and transferred to the EGSEs via a dedicated LAN at the rate of 550 kbps.

- (1) All Transfer Frames containing payload data and the restored HRM data frames were stored in the high-rate data archive which was based on a video tape storage device. Major-frame data could be retrieved from the archive and transferred to the EGSE. After the mission the high-rate data archive was reduced by building data records arranged in chronological order and made available on different media for post-mission evaluation by the experimenter.
- (2) A set of redundant processors was implemented to handle all incoming and outgoing auxiliary data in NASCOM block-format via two 19.2 kbps lines. The NASCOM line handlers
- (a) demultiplexed the NASCOM data blocks and routed the PPF, command acceptance and trajectory data blocks to the DPS, the Command System and the Orbit/Big Screen System respectively
 - (b) monitored the data quality using the established GSOC NASCOM block monitoring
 - (c) received commands from the Command System (ECOS CMD) and the robot EGSE (HRFL) and routed the data to the respective TDM channel.

Data Processing System (DPS)

The Data Processing System was designed to process the Spacelab computer input/output channel data and the Payload Parameter Frame data sets as received from

the front end system. For that purpose VAX 6320 processors were operated in a cluster environment. High-speed line printers and laser printers were available to generate computer printouts.

About 3200 parameters were computed as standard Spacelab data types, e.g. engineering unit, character, string and time conversion and up to 4th order polynomial calculations. About 600 parameters were derived from the standard data types and required special processing. These included complex shuttle attitude calculations, ring buffer rearrangements and value sensitive calculations. The software further flagged the out-of-limit conditions of selected parameters and allowed for the change of the out-of-limit values on-line. The computed telemetry was distributed to the display processing system over a dedicated local area network (LAN) using a "broadcast" data transfer method. The total amount of telemetry parameters to be transferred was in the order of 4000 including system parameters.

Telemetry data was stored within the data processing system computers. The storage contained the converted and calibrated telemetry parameters for both real time and playback transmissions in chronological order. The contents of the storage were made available as computer printouts to the experimenters and mission operations support personnel on request.

Display Processing System

At the tail end of the processing chain the telemetry output by the DPS could be observed on "quick-look" displays incorporated in a display processing system. The system consisted of 50 independent computer workstations picking up the

"broadcast" parameters from the dedicated LAN. In this configuration each workstation was independent and did not influence the network or other workstations in case of failure.

Each workstation comprised a 19" color monitor, an ink-jet printer, a keyboard and a mouse. Provisions were made for attaching a second monitor. Data was presented as display pages in windows on the display workstation screen. Pages were selected from the display page directory. Graphics and alphanumeric representations and limit violation color coding were supported. The contents of each display page could be printed on the attached printer. The printer supported color printouts. The mouse provided all user interface actions, such as format selection, window manipulation and print initiation. Additionally an auxiliary timing unit allowed the setting of alarms in GMT and MET.

EGSE Services

The control center only distributed the experiment high rate data. Provisions were made for the experimenters to bring in their own computer equipment and plug it into the network. The requirement for experimenters to operate external to the control center was supported for the telescience experiment.

The EGSE's ranged from single PC-type computers to whole compute networks. A total of 9 EGSE's were operated during the D-2 mission. The EGSE's were physically connected to a dedicated LAN, a thick-wire ETHERNET with transceivers as the interface and had to comply to the DECnet protocol standard or the DEC Pathworks standard for DOS/Macintosh. User data connections did not differentiate between

internal and external users. Figure 4 provides an overview of typical EGSE connections to the LAN. Data transfer mechanisms were established for

- (a) continuous data, i.e. high-rate telemetry, robot manipulation commands and telescience data, by establishing logical links using the DECnet task-to-task protocol services or non-handshake transfer mechanisms for higher data rates which would otherwise pose network throughput problems
- (b) non-continuous data, i.e. transfer of experimenter generated Command Data Files (CDF) and ECOS Subtimeline Files (STL), by using the file transfer services provided by the VMS/DECnet/DEC Pathworks operating systems.

High-rate data was distributed to the EGSE in real-time providing the respective data channel was active. Provisions were made to transfer HDRR dump playback data in parallel. The control center also retrieved and replayed data from the high-rate data archive upon request. ECIO subsets were distributed to the EGSE in real-time. There was no option to retrieve and distribute ECIO subsets from the HRM data archive.

Experiment operational support data, such as command data strings (Command Data Files-CDF) and ECOS Subtimeline files (STL) were generated locally by the experimenters and forwarded to the control center's Command System for validation, packaging and transfer. The need for almost instantaneous experimenter interaction in the flow of the experiment further called for the implementation of a telescience command interface from an external user center to the Command System. The robot manipulation

31/105

LESSONS LEARNED SUPPORTING ONBOARD SOLID-STATE RECORDERS

Jeff Shi, Tony Mao, Tim Clotworthy

RMS Technologies, Inc.

Code 520.9

Gerald Grebowsky

Data Systems Technology Division

Code 521.1

NASA, Goddard Space Flight Center

Greenbelt, MD 20771

ABSTRACT

With the advance of semiconductor technology, Solid-State Recorders (SSR) have matured and been accepted as primary onboard data storage devices. Their high reliability, simpler interface and control, and high flexibility have made the SSRs a superb choice in today's spacecraft design. While there are many benefits, the use of SSRs may also add significant complexity to ground data systems. For instance, real-time and playback data may be interleaved into the same data stream, making data sequencing and time ordering difficult. Stored data may be played back out of time order, increasing processing load significantly. Data may also be played back after being sorted by Virtual Channels in the SSR, potentially creating bursts in packet rates that exceed the real-time processing capabilities of the ground systems.

This paper presents a summary of lessons learned through the efforts in supporting a number of NASA's missions that employ SSRs. It describes various problems encountered through the design process, and their potential impact on ground system performance, resources, and cost. Recommended approaches to minimizing the impact are demonstrated by examples. The discussion leads to the conclusion that the use of SSRs demands an even higher level of cooperation between spacecraft and ground system designers in order to build the most cost-effective end-to-end system.

1. INTRODUCTION

With the advance of semiconductor technology, Solid-State Recorders (SSR) have matured and been accepted as primary mass data storage devices onboard spacecraft. Its high reliability, simpler interface and control, and high flexibility have made the SSR a superb choice for today's spacecraft designs. The use of SSRs also brings benefits to ground data processing systems. One of the most obvious advantages is the elimination of reversed data playback associated with the use of tape recorders. On the other hand, the flexibility of SSRs may also add complexity to ground data systems. For instance, real-time and playback data may be interleaved into the same data stream, making data sequencing and time ordering difficult. Stored data may be played back out of time order, increasing processing load significantly. Data may also be played back after being sorted in the SSR, potentially creating bursts in packet rates that exceed the capabilities of the ground systems.

The Data Systems Technology Division (DSTD) at Goddard Space Flight Center (GSFC) has provided support for a number of missions flying SSRs, including the Solar Anomalous and

Magnetospheric Particle Explorer (SAMPEX), Fast Auroral Snapshot Explorer (FAST), Submillimeter Wave Astronomy Satellite (SWAS), and Advanced Composition Explorer (ACE). In addition, planned application of SSRs in other missions such as the Earth Observing System (EOS-AM) have been reviewed.

The DSTD has developed a new-generation Packet Processing System (PPS) to perform science data processing for the FAST mission [1]. Scheduled for launch in August 1994, FAST is the second mission of the GSFC Small Explorer (SMEX) program, and one of the first missions based on Consultative Committee for Space Data Systems (CCSDS) recommended packet telemetry data formats [2]. Through the PPS development, extensive efforts have been made to support unique data scenarios generated with the flexibility of the SSR.

This paper presents a summary of lessons learned directly from FAST Packet Processing System development, and indirectly from reviewing other missions. Section 2 describes general characteristics of the onboard SSRs. Section 3 presents a detailed discussion of impacts, and recommendations. Section 4 summarizes the discussion with conclusions.

2. SSR CHARACTERISTICS

Until recently, tape recorders have been the primary means to store data onboard a spacecraft. These recorders were used to store data when not in contact with ground stations, or communication satellites. Tape recorders were stream-oriented devices implying that data can only be accessed sequentially. To preserve power and increase the lifetime of tape recorders, data was typically played back without rewinding, generating a bit-reversed data stream during a playback pass. In addition, the recording and playback scenarios did not assure erasure of old data from the tapes. These scenarios often resulted in a number of old data fragments at either end of a data stream.

In contrast, the SSR is built upon Random Access Memory (RAM) technology and provides the capability for accurate and fast search, which is crucial for downloading data. Any set of stored data can usually be randomly accessed by selectable addressing in the memory. Accordingly, it offers the capability for more flexibility in data recording and playback. The obvious benefit to ground processing from the introduction of the SSR is the elimination of the tape recorder bit-reversed playback.

However, this same flexibility can lead to new challenges for ground data processing. This is especially true if the use of the SSR flexibility effects the sequential order, grouping, or block structure of recommended data formats. These problems are discussed in detail in Section 3.

3. IMPACT DEFINITION AND RECOMMENDATIONS

One of the primary function required by ground data processing is the removal of artifacts introduced by the telemetry path to reproduce data that has the same form as data coming directly from the sensor. This implies that the data products should present data in its original time order with minimum errors and distortions. In support of this functionality, ground data processing incorporates the capability to detect and account for discrepancies in the expected order or structure of the received data. The potential impacts of the SSR application on ground data processing considered here fall into three categories:

1. Time order. The manipulation of time order, as described in the first three subsections, can increase system resource requirements and cost.

2. Data rate. When supporting the CCSDS packetized data format, the ground system's data processing capability or capacity is not simply defined by the input clock rate alone, but also by a packet rate. An example in Section 3.4 shows how under a constant clock rate, packet rates can push capabilities of ground processing systems to their limits.
3. Errors and gaps. The last two subsections are dedicated to discussions of errors and gaps induced during SSR operations, and the impact on the ground processing quality and accounting tasks.

3.1 MANIPULATION OF TIME ORDER

One of the main features of the SSR is that data can be played back in almost any order. This is in sharp contrast with tape recorders that can only be accessed sequentially.

The FAST mission offers an excellent example of how this capability may be implemented. The capacity of the FAST SSR is 128 Mbytes (1 Gigabits). At 8 Megabits per second (Mbps), high-resolution instruments can fill up the SSR in merely 2 minutes! To maximize the use of this limited storage, the FAST Principal Investigator (PI) has developed a complex algorithm to screen sensor data and to take samples, each of which may contain several thousand bytes of sensor data. The algorithm divides the SSR into hundreds of partitions, as shown in Figure 1. Each partition is further divided into a small buffer and a large buffer so that a small portion of data prior to a sensor data sample can be preserved.

When an observation begins, sensor data is taken and compared against a preset threshold. If the data value is less than the threshold, the data is stored sequentially into the small buffer, which operates as a ring buffer. If the data value exceeds the threshold, a sample is taken by jumping to the beginning of the associated large buffer and filling to the end with subsequent sensor data. This process repeats until all available partitions are filled. Then, if samples are still being taken, a comparison is made to all stored samples according to predefined criteria. If the new sample is better, it will overwrite an old sample, which may be anywhere from the first partition to the Nth partition. As the observation continues and the sensor value fluctuates, this scheme will result in fragmented data in the SSR. When the SSR is played back sequentially from a low to high address, the ground system will receive these data fragments that are completely out of time order.

Although ground data systems are designed to handle data fragmentation, the large number of fragments in a data stream will have adverse impact on system performance and resources. As the number of fragments increases, it will take longer for the PPS to merge them together into data sets. More fragments require more system processing and storage resources, and increasing system cost. In extreme cases where millions of data fragments may be generated, ground data systems can be brought nearly to a halt.

There are two remedies to this problem. One is to inform the PI about the impact of data fragmentation and agree to a design limit. In the FAST PPS development, the PI projects 200 data fragments per sensor and agrees to stay within that limit. As a result, there will be about 4000 (200 x 20 science sensors) fragments coming down to the PPS every session. This approach will minimize the impact on flight software development, but limit system's ability to adapt to different data scenarios.

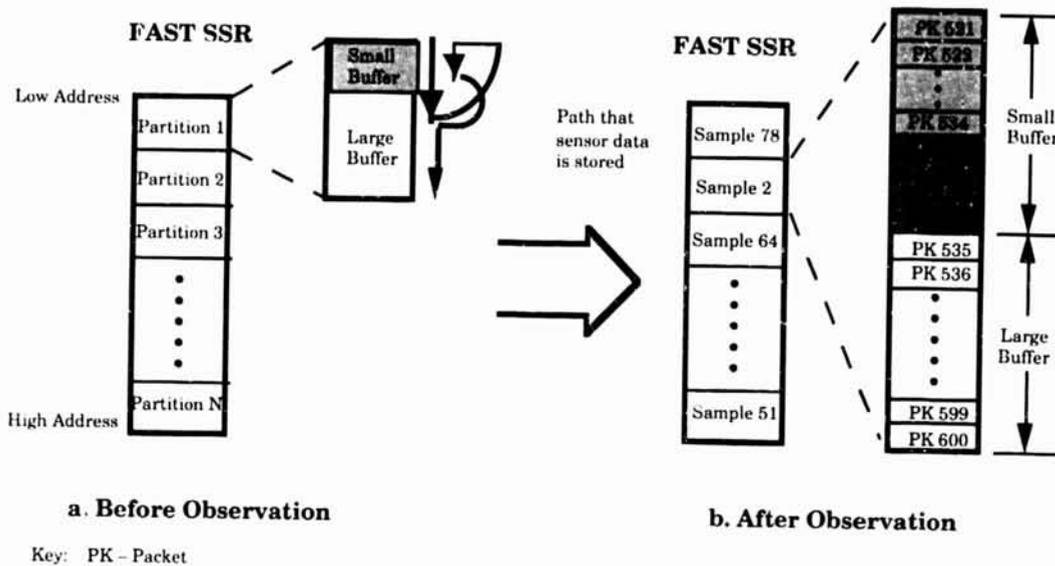


Figure 1. Simplified FAST SSR Memory Management Example

An alternative is for the spacecraft data system to maintain a memory map. This map will keep track of time order and the boundaries of the fragments. During a downlink, the spacecraft data system will use information in the memory map to dump recorded data fragments so that their original time order can be preserved. As a result, the ground data processing can be greatly simplified because there will be only one data fragment per sensor instead of thousands. The drawback will be increased complexity in the flight software design.

3.2 INTERLEAVE OF REAL-TIME AND PLAYBACK DATA

During ground data processing, separation of real-time and playback data is important for sequence check and time ordering. For missions based on tape recorders, there is a clear distinction between real-time and playback data because real-time data is forward and playback data is reversed. They will never be mixed on a frame-by-frame basis since frame synchronization requires a consistent data direction.

Because playback data from the SSR is also in forward order, it is possible to interleave real-time and playback data into the same data stream. This is desirable by flight operations teams because it allows them to continuously monitor critical spacecraft parameters in real-time while receiving recorded payload data. The problem is that packet sorting can not be performed based only on Path Identifier (ID), a combination of Spacecraft ID (SCID) and Application Process ID (APID), because both real-time and playback data will have the same Path ID. To address this problem, the FAST spacecraft assigns different Virtual Channels (VC) to real-time and playback data. In this manner, the PPS is able to separate real-time and playback data by using the VCID in addition to Path ID, as illustrated in an example in Figure 2. An alternative scheme is to use the REPLAY flag defined by CCSDS in a frame primary header so that the sorting can be accomplished by using the REPLAY flag in addition to Path ID.

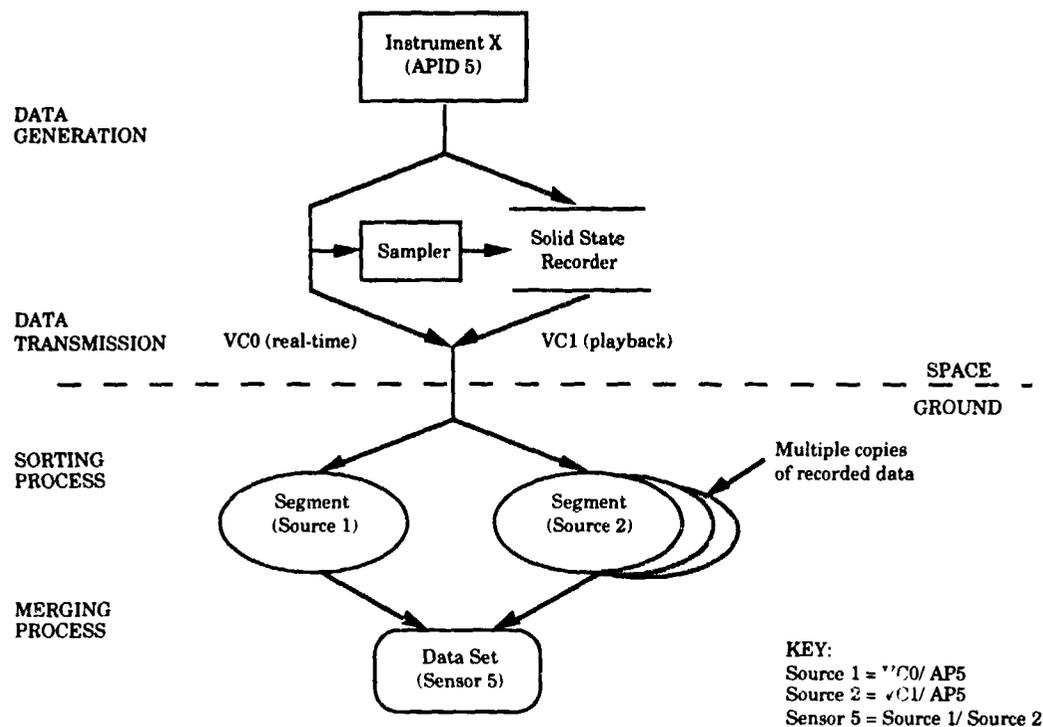


Figure 2. FAST SSR Data Scenario Example

3.3 MULTIPLE PLAYBACK OF DATA

As a contingency operation, many missions have planned to redump a certain section of recorded data when the data quality of the first dump is unsatisfactory. Because of the SSRs fast response time, it is feasible for ground controllers to send commands up to require another dump of the same data, or a subset of it based on real-time processing status. As illustrated in Figure 2, multiple copies of the same data will be received by ground data systems during one pass, which increases processing load since all redundant data has to be identified and deleted based on data quality. However, this normally represents a minor impact as only limited attempts can be made for retransmission.

3.4 RECORDING SORTED DATA

Another common use of the SSRs flexibility is to sort sensor data as it is stored in the SSR. A typical implementation is to partition the SSR into a number of buffers, one for each sensor or each VC. Consequently, data will be stored in appropriate buffers according to APID or VCID. During a downlink pass, data will be dumped one buffer at a time, i.e., one sensor or VC at a time. There are several advantages to this scheme. First, it may simplify the onboard memory management plan by dividing a large buffer into smaller, but dedicated, buffers. Second, it allows prioritization of data playback. Third, it reduces the processing load of ground systems by performing sensor or VC demultiplexing and sorting functions onboard the spacecraft.

However, the sorting and buffering can change packet rates significantly, and its potential impact on ground systems must not be overlooked. The CCSDS telemetry is a packetized data stream in which many packets, large or small, are multiplexed together. The packet processing load is

proportional to the number of packets per second. Each ground system has a limited ability to process CCSDS source packets as measured in packets per second. Typically, high-rate sensors (e.g. science instruments) generate large packets, and low-rate sensors (e.g. housekeeping sensors) generate small packets. For a fixed data rate, as average packet size decreases, the number of packets per second increases.

For example, assume a fictitious spacecraft has just two sensors, an engineering sensor and an instrument sensor. Every second, the engineering sensor generates a packet of 20 bytes, and the instrument sensor 1000 packets of 1000 bytes. For an 8-Mbps downlink without buffering, this scenario represents a packet rate of 1001 packets per second. On the other hand, if all packets are sorted and buffered, the same 8-Mbps downlink will produce 1000 packets per second for the instrument buffer, and 50,000 packets per second for the engineering buffer! In this scenario, the burst packet rate is increased 50 times. Suppose a ground system is designed to process data in real time at 5000 packets per second – it will still overflow by the later scenario even though it can handle the average packet rate by a wide margin.

This problem was first experienced in the SAMPEX mission. As a result, a recommendation was made to and adopted by the FAST spacecraft development team to interleave recorded engineering data with science data during high-rate downlink with a ratio of 1:5. This represents a small change in flight software, but saves significantly in ground system design and implementation. This is especially true when real-time processing of engineering data is desired.

A second approach calls for packet rate buffering to be performed by the ground data systems. This is very difficult to implement for a number of reasons. It requires expensive resources to buffer frames before packet processing. Packet delivery latency will be much longer and unpredictable. Scheduling of buffered data playback will be very tricky, if not impossible, if an attempt is made to match data packet rates to system capability.

3.5 CREATION OF GAPS

During a ground contact (pass), FAST spacecraft engineering packets are generated at a higher rate and transmitted to ground in real time. As a backup, these packets are also stored in an engineering buffer, in case retransmission is required. However, due to buffer limitation, the packets are sampled and only a subset of them, e.g., one out of four, is stored, causing discontinuity in packet sequence count (e.g., 1, 5, 9, etc.). There are many complications when these packets with large numbers of gaps are received by ground data systems. First: how does the system know this discontinuity is intentionally created, and not caused by transmission errors? Second: how are these packets merged into the ones with sequence counts of different discontinuities? Worse yet: what if packets are sampled at different rates from time to time?

The FAST PPS implemented limited capabilities to handle these cases. Nevertheless, it requires a lot of effort and resources, and still only covers a fraction of the possible scenarios. In general, it is dangerous to tinker with the packet sequence counter, and every effort should be made to avoid this practice if level zero processing is required.

3.6 CREATION OF FORMAT ERRORS

The above discussions treated data scenarios that use the flexibility inherent in the SSR. Ground processing complications will also be encountered if the implementation of the SSR does not maintain any continuity inherent in the format of the data generated. In the case of CCSDS recommended formats, storage and addressing of data in the SSR should support playback while

maintaining frame and byte boundaries. Violation of this condition will result in loss of data, as demonstrated the following example.

In this SSR design, the onboard telemetry generation is running at the very low rate of 6944 bits per second. But even at this low rate, the SSR still can not record and play back at the same time. Like tape recorders, two SSR units of 80 Mbytes each have to be configured with a utilization factor under 50 percent.

Like tape recorders, this SSR is addressable only in 1-Kbyte blocks, rather than on any byte boundaries. Worse, this 1-Kbyte block is asynchronous to byte boundaries, i.e., it may start and stop in the middle of a byte. As a result, each time a write operation begins, a transfer frame is lost and a portion of stored data is corrupted. In addition, this adds significant complexity to ground data processing because many frames may be corrupted with gaps and errors even though communication link quality is good.

4. SUMMARY

The application of onboard Solid-State Recorders has been discussed from a ground data processing perspective based on experience gained through the SAMPEX, FAST, SWAS, and other space missions. Characteristics of the SSR have been described in contrast with conventional tape recorders. Their advantages and potential negative impacts are detailed using several examples. Many of the data scenarios described have been simulated and tested using an advanced Spacecraft Data Simulator at the DSTD during the development of the FAST PPS [3].

In general, the SSR, with its speed, reliability, and flexibility, offers a technically superior choice for space data systems storage and data processing. With sophisticated management algorithms, the SSR can achieve many objectives that are simply impossible to achieve with conventional tape recorders. However, since the SSR may impose significant performance, resource, and cost impacts on ground data processing systems, the SSR management algorithms should be developed in a coordinated effort with ground system developers. Important parameters such as average and burst packet rates, and maximum number of data fragments per pass should be defined in system specifications to guide both space and ground data system designs. With such an integrated effort, adverse impacts can be avoided or minimized so that the most cost-effective space-ground data systems can be achieved.

5. REFERENCES

1. Shi, J., Horner, W., Grebowsky, G., Chesney, J., "Fast Auroral Snapshot Explorer (FAST) Packet Processing System (PPS)," Proceedings of International Telemetry Conference, 1993, pp. 445-459.
2. "Packet Telemetry," CCSDS 102.0-B-2, Blue Book, Consultative Committee for Space Data Systems, November, 1992.
3. Shi, J., Gordon, J., Mirchandani, C., Nguyen, D., "Spacecraft Data Simulator for the Test of Level Zero Processing Systems," Space Ops 1994.

6. NOMENCLATURE

ACE	Advanced Composition Explorer
APID	Application Process Identifier
CCSDS	Consultative Committee for Space Data Systems
DSTD	Data Systems Technology Division
EOS-AM	Earth Observing System AM
FAST	Fast Auroral Snapshot Explorer
GSFC	Goddard Space Flight Center
LZP	Level Zero Processing
PI	Principal Investigator
RAM	Random Access Memory
SAMPEX	Solar Anomalous and Magnetospheric Particle Explorer
SCID	Spacecraft Identifier
SMEX	Small Explorer
SWAS	Submillimeter Wave Astronomy Satellite
SSR	Solid-State Recorder
VC	Virtual Channel
VCID	Virtual Channel Identifier

**Experience with the EURECA Packet Telemetry
and
Packet Telecommand System**

By

Erik Mose Sørensen and Paolo Ferri

European Space Agency / European Space Operations Centre
Darmstadt, Germany

ABSTRACT

The European Retrieval Carrier (EURECA) was launched on its first flight on the 31st July 1992 and retrieved on the 29th of June 1993. EURECA is characterised by several new on-board features, most notably Packet telemetry, and a partial implementation of packet telecommanding, the first ESA packetised spacecraft. Today more than one year after the retrieval the data from the EURECA mission has to a large extent been analysed and we can present some of the interesting results.

This paper concentrates on the implementation and operational experience with the EURECA Packet Telemetry and Packet Telecommanding. We already discovered during the design of the ground system that the use of packet telemetry has major impact on the overall design and that processing of packet telemetry may have significant effect on the computer loading and sizing. During the mission a number of problems were identified with the on-board implementation resulting in very strange anomalous behaviours. Many of these problems directly violated basic assumptions for the design of the ground segment adding to the strange behaviour. The paper shows that the design of a telemetry packet system should be flexible enough to allow a rapid configuration of the telemetry processing in order to adapt it to the new situation in case of an on-board failure. The experience gained with the EURECA mission control should be used to improve ground systems for future missions.

Key Words: Packet Telemetry, Packet Telecommanding.

1. INTRODUCTION

The European Retrieval Carrier (Eureca) is a reusable platform supplying power, cooling, ground communications and data processing services to a variety of independently operated payloads (ref 1). Fifteen experimental facilities are carried to support more than fifty individual experiments. The operational altitude was 500 Km. The Operations Control Centre (OCC) was at ESA's European Space Operations Centre (ESOC) in Darmstadt, West Germany.

The primary groundstations were at Maspalomas in the Canary Islands and Kourou at French Guinea. During the deployment and retrieval phases contact was maintained via the NASA Communications Network and the STS.

At ESOC, operational data processing was carried out on the Eureca Dedicated Computer System (EDCS) that hosts the mission-configured Spacecraft Control and Operations System (SCOS) (ref 2) and the Eureca-Specific Software (ESS) applications.

The Eureca-A1 mission has characteristics differing quite considerably from those of missions hitherto supported at ESOC. One of these is the use of Packet Telemetry and Packet Commanding. EURECA was the first ESA application of Packet Telemetry and Commanding.

2. WHY PACKET TELEMETRY AND COMMANDING FOR EURECA

Spacecraft previously and currently controlled from ESOC all use a time-division multiplexed (TDM) telemetry, in which fixed-size subframes are generated and downlinked at constant rate. In the simplest case a given parameter appears at a fixed address in the subframe and this parameter reports the value of some on-board physical quantity, sampled in principle at the subframe rate. Many spacecraft make rather more sophisticated use of TDM telemetry, essentially because their operations and on-board applications cannot live with the restrictions of fixed sampling/fixed telemetry address. Thus innovations have appeared such as switchable formats, programmable formats and floating formats (this last named being an ad hoc packetisation). These sophistications illustrate a weakness of TDM telemetry, namely its inflexibility of handling a variety of on-board data sources, generating data at temporally varying rates, possibly as determined by an elaborate plan of instrument operations. The traditional TDM approach of allocating fixed

proportions of the available bandwidth to each source then becomes both restrictive and wasteful.

Eureca is a re-usable spacecraft supporting a different payload complement on each flight (15 instruments on the first flight). It also has to be assumed that most instruments are controlled with "unknown design" and that each instrument would require on-board flexibility to cover different mission phases and instrument modes. Packet Telemetry provides powerful capabilities to satisfy variable data rates and configurations, also providing abilities for late definition and changes. With Packet Telemetry the source can generate observational data when needed, hence the occurrence pattern or rate may be selected according to the phenomenon being observed. Packet telemetry provides variable partitioning of downlink avoiding unnecessary loading of resources. Another important consideration was that the packet telemetry is a standard where other options would have required special development with no or little reuse leading inevitably to higher cost in the long term.

The Packet telemetry recommendation (ref 3) uses two principal data structures, the source packet and the Transfer Frame, source packets being multiplexed within transfer frames. Each on-board source must label its data packets using CCSDS defined headers, although no requirements on the contained data are imposed. The transfer frames are of fixed length, optimised for high-performance transfer to the ground. The concept of Virtual Channels (VCs) also exists, to allow separation between data of different priorities, for example real-time data needed for operations and non time-critical dump of science data stored on board. VCs are identified at the transfer frame level. In the case of Eureca there are two VCs, VC0 and VC1, to handle real time and playback data respectively. Playback data is downlinked from on-board bubble memory and will contain bulky payload data as well as housekeeping data from the out-contact periods.

The Eureca telecommanding system is a hybrid between the older command standards (Ref 4) and the new Packet command standard (Ref 5). The reason for this lies in the way it has been implemented on board. Command decoders using the old standard have been used as a basis, but the extra services of the packet commanding have been built into the on-board computer. Thus when the on-board computer is nominally activated, the commanding system acts like a packet command system, using a subset of COP 1 of the standard (ref 5). If the OBC is off, the old standard has to be used. This paper will only concentrate on the experience in using the COP-1 Protocol.

NOTE: In this section, although the word COP-1 is used, EURECA has only implemented a subset of the COP-1. The

EURECA terminology and services are not completely compatible with the latest issue of the CCSDS recommendation.

COP-1 is a closed-loop Telecommand Protocol that utilises sequential ("go-back-n") retransmission techniques to correct Telecommand Blocks that were rejected by the spacecraft because of error. COP-1 allows Telecommand Blocks to be accepted by the spacecraft only if they are received in strict sequential order. This is controlled by the necessary presence of a standard return data report in the telemetry downlink, the Command Link Control Word (CLCW). A timer is used to cause retransmission of a Telecommand Block if the expected response is not received, with a limit on the number of automatic retransmissions allowed before the higher layer is notified that there are problems in sending Telecommand Blocks. The retransmission mechanism ensures that:

- No Telecommand Block is lost
- No Telecommand Block is duplicated
- No Telecommand Block is delivered out of sequence

The COP-1 protocol has also expedited service. This service is used for exceptional spacecraft communications. Typically, this service is required for recovery in absence of telemetry downlink (i.e no CLCW), or during unexpected situations requiring unimpaired access to the spacecraft data management system.

3. THE GROUND CONTROL SYSTEM

The introduction of Packet Telemetry makes it possible to define Packet Types, and for each of these packet types to define a standard for the format and presentation of data in the Packet Data Field. The following packet types are defined for Eureca: Housekeeping 1, Housekeeping 2, Time, Acknowledge, Exception, Report, Acknowledge and Private Packets. Housekeeping 1 (HK1) packets are similar to the subframes of TDM systems, containing snapshots of on-board parameters which can be subjected to limit and other checks and displayed on alphanumeric and graphic displays. The other packet types are different and require specific processing, thus making the processing system more complex.

One of the major changes going from a TDM to Packet Telemetry system is the change to an event driven system (packets arrive asynchronously, rather

than at fixed format rates). This impacts both design and computer loading.

The Architectural Design of the Eureka Dedicated Computer System (EDCS) is based on a Telemetry Processing Chain and a Telecommanding Chain. The Telemetry Processing Chain consists of a Telemetry Receiver, Telemetry Processing Task, Command Verifier, Filing and Display Tasks (alphanumeric, graphical displays, report/exception displays). The Telecommand Chain consists of the Manual Commanding Stacks, Automatic Commanding Queues, Command Verifier, Command Uplinker, Command Filing Task, Display and Configuration Tasks. The Communication between these individual tasks is based on the Buffer Manager, a task responsible for passing Telemetry and Command buffers around the system. Telemetry and Command buffers are given to the Buffer Manager and asked tasks are informed that a data buffer is available for processing. The Buffer Manager does not pass around the actual data buffers, only small mailbox messages are sent to the relevant tasks with a reference to the data buffer. This architecture is very convenient for Packet Telemetry, the Packets are distributed according to the packet type. If for a mission other packet types are required, such architecture makes it possible to setup a new task to process these new packet types without disturbing the functionality of already existing tasks.

As for a TDM spacecraft, the computer load on a packet TM system is dominated by telemetry processing and display support (neglecting any project specific peculiarities). Commanding tasks account for only a small fraction (3-5%) of the load. Two main considerations distinguish the load characteristics of the ground computer system supporting a packet telemetry. Firstly, there will not be one format, but a set of packets, of different lengths each having different processing needs. Secondly, the packets are generated asynchronously, not at a constant rate, so it is essential to have a traffic model, which gives a fairly realistic representation of average and peak packet rates. In the case of Eureka, such models are needed for pass and post pass activities, which are quite different. During real-time processing (pass operations), the packet rate is (worst case) 12/s. This generates a much higher load than the rather low daily average data rate (2kbits/s) might lead one to suppose. By contrast, to give a TDM example Hipparcos (a geostationary spacecraft) has a continuous data rate of 23 kbits/s but produces one subframe each c. 10s. The loading of the Hipparcos Dedicated Computer System (HDCS) is comparable to that of the EDCS (possibly a little lower) despite the Hipparcos's much higher bit rate. Similar as for the ground system the on-board system must be carefully analysed and a software system budget should establish a clear reference case for on-board

and space ground traffic scenario, which can be used for system testing. Critical on-board areas are computer load, timing of cooperation or dependant applications, packet buffer sizes and number of packet buffers.

Data delivery to users is greatly facilitated by use of packet telemetry, which already results in demultiplexing according to application ID. This also simplifies the provision of security, i.e. protection of privacy of datasets. Eureka users require rapid access to their data, which rules out the traditional method science data delivery, dispatch on magnetic tapes.

The COP-1 protocol increases the complication of the command uplinker software, which has to handle for every telecommand with a number of messages coming from different units at the station in addition to the telemetry messages from the spacecraft. Testing this software in a realistic environment became absolutely necessary due to the importance of the timing aspects of the problem and this forced extension of precious testing time with the spacecraft flight model connected to a ground station interface.

4. THE ON-BOARD SYSTEM

The large number of independent processors on-board EURECA increases the likelihood of unexpected behaviours which result in corruption of the format or contents of the Telemetry Packet produced. During the mission a number of problems were identified with the on-board implementation resulting in very strange anomalous behaviours. Many of these problems directly violated basic assumptions for the design of the ground segment adding to the strange behaviour. Below is a table listing the problems experienced during the mission:

TRANSFER FRAMES

Problem	Consequence	On-Ground Detection	Prevention
Received a corrupted Transfer Frame (already corrupted before the FECW was calculated) The problem was with a spillover with Idle Frames between.	Ground system reported protocol error because the expected spillover data were not available. Packet Discarded	Always use the First Header Pointer in the Transfer Frame Header as the Master to locate the first Packet in a Frame. If inconsistent with the Packet Length from the last Packet in the previous Frame discard this Packet and report a protocol error.	Ground Testing
When the on-board Data Handling System changes mode from High Speed Link to Low Speed Link it cannot maintain the Transfer Frames proper (spillover etc.)	As above	As above	Spacecraft Design.
Received a Transfer Frame with two Idle Packets and with a non-idle packet between.	Allowed according to the standard.	Do not assume that an Idle-Packet always is at the end of a Transfer Frame.	

PRIMARY HEADERS

Problem	Consequence	On-Ground Detection	Prevention
Received Packets where the Secondary Header Flag was set to 1 but the Packet Length Field had the value 0 (SH is fixed to 6 octets for EURECA)	Ground system detected a corrupted packet reported a protocol error. Time calibration not possible.	Maintain a list of allowed length of each Application ID and check every received Packet. Check consistency between the information in the Primary Header and the Packet Length. Check the value of the P field in the Secondary Header if the P field is used.	Ground testing.
In one experiment the Source Sequence Counter is implemented as a 16 Bit Counter instead of the 14 Bit defined in the standard.	Ground system reported segmentation protocol errors because the SSC has been extended into the Segmentation Flags in the Primary Header. Packet discarded.	Normally build into the packet decommutation algorithm.	Ground testing shall check that all instruments handles correctly the wraparound of the SSC. This require normally a long test run. workaround: Restart experiment at regular interval.
In one experiment the Source Sequence Counter is shared between four different Application IDs.	Ground system reports jumps in Source Sequence Counter. Accounting for these Application IDs not possible.	Normally build into the packet decommutation algorithm.	Ground testing.

Problem	Consequence	On-Ground Detection	Prevention
General: Due to onboard power/cooling constraints it is necessary to activate/deactivate instruments frequently. In such cases the experiments resets the Source Sequence Counter to 0.	Allowed according to the standard	The Source Sequence Counter is not very useful in these cases. The ground design must take into account such type of operations	-

SECONDARY HEADERS AND TIME CALIBRATION

Problem	Consequence	On-Ground Detection	Prevention
Received Secondary Headers where the Time Field was shifted one octet.	Proper time calibration cannot be performed.	May be difficult. For real-time received telemetry it is possible to make a plausibility check against current time. However this does not work for playback of on-board stored Telemetry. In the playback case another plausibility checks must be implemented.	Ground testing
Many experiments have problems with the stability of their local clocks resulting in: Unacceptable drift Large jumps in time when they synchronize with the Master Clock. This can even cause the time to jump backwards.	Proper time calibration cannot always be performed. In case the time jumps backwards this may cause problems for the filing system. However this depends on the design of the filing system.	As above.	Design of the overall time concept including requirements on drifts of master and local clocks. During ground testing verify that the implementation is according to specification.

5. OPERATIONAL EXPERIENCE WITH PACKET TM/TC

One of the main advantages of packet Tm is that the TM source can in principle decide what data to send when to the ground. This concept was extensively applied on EURECA, and the ground segment and operations concept used it as a basic assumption. While this proved to work well in the nominal cases, it became a problem in cases of on-board failures. In some cases the on-board unit which experienced that failure took the wrong decision on what TM to send, limiting the visibility to the ground of the causes of the failure. Some failures affected the functionality of the unit to the extent that the unit stopped generating TM or even started an endless loop in which event TM packets were generated continuously, overflowing the on-board data storage. Interaction between the subsystem TM generation and the system level decisions taken by DHS in case of specific failures were also very difficult to handle. In the case of AOCS special application software had to be written within DHS to

guarantee extended TM generation in case of subsystem anomalies. This did not succeed in several cases during the flight, and the corr TM coverage of critical failures was lost as a consequence.

The implementation of the packet TM concept had a major positive effect on the on-board communications between "intelligent" instruments and subsystems and the central DHS over the DHS data bus. Low level protocol problems in the bus interface units were often cured at higher level by the packet transfer protocol. Those units which were not able to generate packets suffered from the low level problems, causing significant complications to the operations. One negative aspect of the EURECA implementation of packet TM was that the DHS application software was not able to read the contents of the TM packets generated by the other subsystems or instruments. This artificially limited enormously the system level fault management capabilities of the DHS. In particular the information contained in the Housekeeping packets

and the Event packets (Report and Exception) was essential to detect and isolate problems with the subsystem or instrument which could be easily recovered at system level. This limitation of the DHS shifted the system level fault management to the ground control, which was in most of the cases only able to intervene after several hours, due to the limited visibility of the spacecraft from the ground stations (about 5% of the mission time).

The use of the different packet types by the different packet TM sources on-board (12 instruments and 2 subsystems) was not always correctly reflecting the definitions imposed by the design specifications. In particular an improper use of Report and Exception packets was causing some problems in flight operations. The ground segment was designed under the assumption that Exception packets would only report anomalous behaviours, and Report packets would indicate progress or completion of nominal activities; in several cases it was found out during final system testing or even during flight that this clear distinction was not always observed.

Another clear directive for the design of TM packets was that all TM parameters for which direct ground monitoring was required should have been included in Housekeeping packets. The ground segment was designed on this assumption and therefore was not supposed to open and process science packets. This rule was also in several cases not properly followed and the ground had to work around the problem by including some specific science packets in the list of TM packets to be processed. This was not trivial also due to the fact that no formal documentation was available to describe science packets, and the relevant information had to be extracted from various sources like meetings, private conversations and informal documents.

The packet TM implementation had a significant impact on the operational database preparation. Most of the TM parameters were contained in several different Housekeeping packets; this had an impact on the size of database tables and complicated the handling of derived parameters, which had to be defined and inserted in all TM packets containing a contributing parameters. A large amount of manpower had to be invested in the generation of the Event packets database. This was mainly caused by the large number of possible event packets (of the order of 2500 at the end) to be defined, but also to the lack of description of these packets in the AIT database. The contents description and meaning of each event packet had to be extracted in most of the cases directly from the on-board software code which was generating it. This manual work had to be repeated every time a new version of the application software was generated and copied to ESOC, even after launch.

Event packets were the most powerful tool the flight controllers had to monitor the spacecraft and payload activities and to identify and diagnose anomalies. The lack of AIT database in this area reduced significantly the quality of the overall ground testing.

A final consideration should be made on the opportunity to involve flight operations personnel in the definition of the contents of Housekeeping packets. These packets were originally designed following engineering considerations and disregarding completely the utilisation during operations. This forced a complete redesign of the packets at a later stage in the development of the spacecraft, with impact on both the AIT/AIV programme and the operations preparation.

For commanding no real problems was encountered during flight with this concept. Its flexibility was properly exploited by the database editors specially designed for this mission in the mission control system. The block protocol and the related retry mechanism in case of failed uplink verification of a TC block worked very well, but were very difficult to test and tune before flight.

6. LESSONS LEARNED

The following lessons have been learned about packet telemetry and telecommand systems from development of the Eureka spacecraft control system and during the mission:

1. Sizing of ground and on-board computer systems needs to be carried out carefully, using a good traffic model for the generation of the various packets.
2. Very careful consideration has to be given to matching the design of the spacecraft and packet control system to the characteristics of packet telemetry. "Fudging" a TDM system work with packet telemetry is not advised and at the best is likely to be highly inefficient
3. On-Ground Testing must take into account the use of Packet Telemetry. This must include functional tests to verify 1) all implemented features of the Packet Telemetry (segmentation etc.), 2) proper wraparound of counters, 3) stability of on-board clocks (master and slave), 4) performance tests to verify on-board loading of the system in

typical operational scenarios.

4. Ground system must be able to handle errors in the implementation of packet telemetry 1) check the consistency of all static fields in transfer frames and packets, 2) design the system to be robust against implementation errors, 3) design the system to minimise the impact on other users in case of implementation errors, 4) include knowledge of the on-board implementation (expected application id's, expected packet lengths etc.), 5) provide proper reporting for detected errors, 6) give operational staff proper visibility of detected errors, 6) provide tools to disable error reporting of "known errors"

5. The COP-1 protocol has proven to be very reliable and is able to recover transmission error with a minimal operational impact. There have been a number of occasions where the COP-1 protocol has successfully recovered an error. These cases all concern link degradation, and involved the following circumstances:

During the deployment phase with a bad RF link between the Shuttle and EURECA

During the deployment phase where the EDCS did not receive a Command Acceptance Pattern (CAP) from NASA. During ESA ground station passes where the spacecraft was configured with the wrong antennae.

During ESA ground passes where commanding was executed down to 0° elevation, resulting in degradation of the telecommand and telemetry links).

During on-board antenna switch over.

When the OBC failed to allocate a telecommand buffer (due to an OBC overload condition).

Although not all of the above cases were foreseen in the design of the COP-1 protocol (in particular case 2 and 6) the COP-1 protocol has always successfully recovered the error with a maximum of two retries. It is also important that during EURECA routine operations with a normal link budget the COP-1 protocol has never been in retry (i.e. no transmission errors).

6. The design of the commanding system in the control centre must consider end-to-end protocols (in particular needed for uplinking on-board

software patches and master schedules) and provide elements that makes it possible to recover in case of ground failures.

7. Introduction of Packet Telemetry and Commands is a major step towards standardisation of on-board and ground systems. In order to fully archive this goal it will be necessary to define standards covering more of the format than that specified in present standards. At the very least local standards are needed for each packet type to avoid proliferation. Ref 7 describes some current ESA work on this topic.

7. CONCLUSION

The packet TM/TC concept proved very powerful in supporting complex operations of an autonomous low-Earth orbiter like EURECA. The system supported a heavy downlink and uplink traffic corresponding to a total of 10 million transfer frames containing 35 million packets and 240000 commands were sent. Most of the above described problems do not relate to the overall concept but to the implementation, which suffered in the EURECA mission from the lack of previous experience. We have found a number of problems with the actual implementation of the Packet Telemetry Standard but we have not found any problems with the standard itself.

The lessons learned from this mission could be easily taken into consideration in the design of future missions applying the same approach to the space-ground interface.

8. REFERENCES

1. Andresen, R.D. and Nellessen, W. "Eureca in the Columbus Scenario." 1988, Space Technology 8, 45
2. Kaufeler J-F and Mazza M., "A New Generation of Spacecraft Control System - SCOS," 1988, ESA Bulletin 56.
3. Consultative Committee for Space Data Systems, 1984, 'Recommendation for Space Data System Standards: Packet Telemetry', "Blue Book".

4. PCM Telecommand Standard ESA PSS-45 Issue 1 April 1978
5. Consultative Committee for Space Data Systems, 1986, 'Recommendation for Space Data System Standards: Telecommand, Part-2: Data Routing Service, Architectural Specification', "Blue Book".
6. M. Jones et al, 1990, Ground handling of Packet Telemetry and Commanding: The Eureka Dedicated Control System, submitted to the ESA Journal.
7. Kaufeler, J-F, 1990, Packet management standard, Proceedings of First International Symposium on Data Systems for Spacecraft Control.

**A MODULAR, SOFTWARE REPROGRAMMABLE, TELEMETRY
PREPROCESSOR FOR OPEN SYSTEMS BACKPLANE
ARCHITECTURES**

Steve Talabac
CTA

Paper Not Available

Mission Management

Page 275

1. Mission Planning

MM.1.a	JPL Operations Cost Model for Flight Projects <i>John Carraway</i>	277
MM.1.b	Galileo Mission Planning for Low Gain Antenna Based Operations <i>R. Gershman, K. L. Buxbaum, J. M. Ludwinski, B. G. Paczkowski</i>	279-286
MM.1.c	A Distributed Planning Concept for Space Station Payload Operations <i>Jeff Hagopian, Theresa Maxwell, Tracey Reed</i>	287-294
MM.1.d	Mission Planning for Space-Based Satellite Surveillance Experiments With the MSX <i>R. Sridharan, T. Fishman, E. Robinson, H. Viggh, A. Wiseman</i>	295-303
MM.1.e	Engineering <i>Ulysses</i> Extended Mission <i>Shaun Standley</i>	305-309
MM.1.f	Accuracy Analysis of TDRSS Demand Forecasts <i>Daniel C. Stern, Allen J. Levine, Karl J. Pitt</i>	311-318
MM.1.g	Joint Operations Planning for Space Surveillance Missions on the MSX Satellite <i>Grant Stokes, Andrew Good</i>	319-326
MM.1.h	Multi-Mission Operations From the Headquarters Perspective <i>Guenter Strobel</i>	327
MM.1.i	International Mission Planning for Space Very Long Baseline Interferometry <i>James S. Ulvestad</i>	329-336

* Presented in Poster Session

JPL OPERATIONS COST MODEL FOR FLIGHT PROJECTS

**John Carraway
Jet Propulsion Laboratory**

Paper Not Available

GALILEO MISSION PLANNING FOR LOW GAIN ANTENNA BASED OPERATIONS

R. Gershman, K. L. Buxbaum, J. M. Ludwinski, and B. G. Paczkowski
 Jet Propulsion Laboratory
 California Institute of Technology
 Pasadena, CA

ABSTRACT

The Galileo mission operations concept is undergoing substantial redesign, necessitated by the deployment failure of the High Gain Antenna, while the spacecraft is on its way to Jupiter. The new design applies state-of-the-art technology and processes to increase the telemetry rate available through the Low Gain Antenna and to increase the information density of the telemetry. This paper describes the mission planning process being developed as part of this redesign. Principal topics include a brief description of the new mission concept and anticipated science return (these have been covered more extensively in earlier papers), identification of key drivers on the mission planning process, a description of the process and its implementation schedule, a discussion of the application of automated mission planning tools to the process, and a status report on mission planning work to date.

Galileo enhancements include extensive reprogramming of on-board computers and substantial hardware and software upgrades for the Deep Space Network (DSN). The principal mode of operation will be onboard recording of science data followed by extended playback periods. A variety of techniques will be used to compress and edit the data both before recording and during playback. A highly-compressed real-time science data stream will also be important. The telemetry rate will be increased using advanced coding techniques and advanced receivers.

Galileo mission planning for orbital operations now involves partitioning of several scarce resources. Particularly difficult are division of the telemetry among the many users (eleven instruments, radio science, engineering monitoring, and navigation) and allocation of space on the tape recorder at each of the ten satellite encounters. The planning process is complicated by uncertainty in forecast performance of the DSN modifications and the non-deterministic nature of the new data compression schemes. Key mission planning steps include quantifying resources

or capabilities to be allocated, prioritizing science observations and estimating resource needs for each, working inter-and intra-orbit trades of these resources among the Project elements, and planning real-time science activity. The first major mission planning activity, a high level, orbit-by-orbit allocation of resources among science objectives, has already been completed; and results are illustrated in the paper.

To make efficient use of limited resources, Galileo mission planning will rely on automated mission planning tools capable of dealing with interactions among time-varying downlink capability, real-time science and engineering data transmission, and playback of recorded data. A new generic mission planning tool is being adapted for this purpose.

1. MISSION OVERVIEW

Galileo is on its way to Jupiter to study the giant planet's atmosphere, satellites and magnetosphere with the most capable suite of instruments ever placed on a planetary spacecraft. Galileo is actually two spacecraft currently traveling attached. The Probe will separate in July 1995 and enter the Jupiter atmosphere on December 7, 1995. For about 75 minutes during Probe descent, data from its seven instruments will be relayed to the Orbiter for subsequent transmission to Earth. The Orbiter will then conduct a 23-month-long tour of the Jupiter system including ten close encounters (200-2700 km altitude) with the Galilean satellites while returning data from its eleven instruments. Details of Galileo's science objectives and the instruments sent to accomplish them are provided in Reference 1.

A high level timeline of the mission is shown in Figure 1. Galileo was launched on a Venus-Earth-Earth-Gravity Assist (VEEGA) trajectory in October 1989. This trajectory has provided opportunities to return science data from the first two asteroid encounters (asteroids Gaspra and Ida) as well as data from close flybys of Venus and Earth (twice). Galileo's images of Ida provided an unexpected

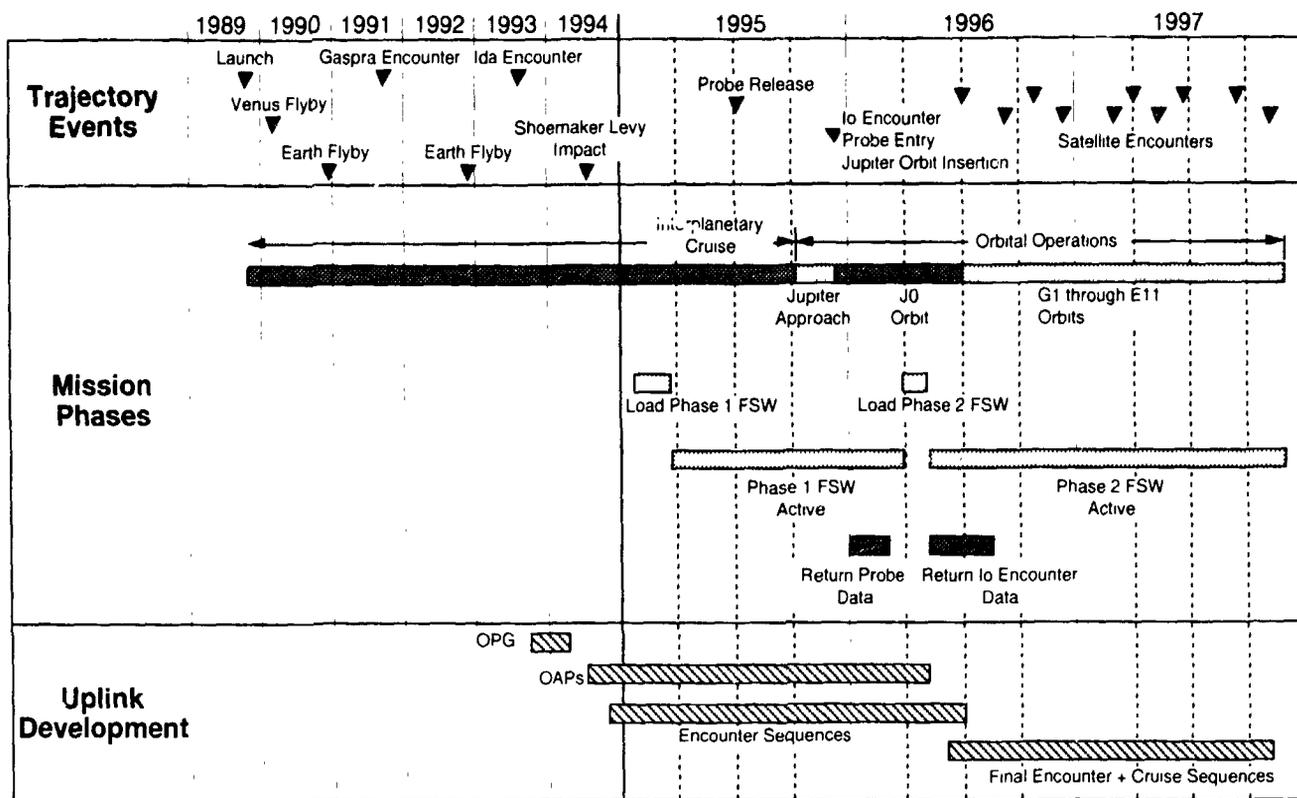


Figure 1. Mission Overview

bonus, discovery of a small moon orbiting the asteroid. Shortly after submission of this paper, Galileo will observe a remarkable target-of-opportunity, the impact of comet Shoemaker-Levy 9 into Jupiter.

The Galileo design incorporated a High Gain Antenna (HGA) capable of downlinking an 800x800 pixel image in one minute. At launch, the HGA was folded umbrella-fashion to fit in the Space Shuttle bay; and, for thermal reasons, deployment was not scheduled to occur until about 1.5 years after launch. The deployment sequence resulted in a partially open antenna, and a wide range of corrective actions has been unsuccessful (see Reference 2). In late 1991 a new mission concept using the Low Gain Antenna (LGA) was devised to capture most of the original science objectives if the HGA could not be opened. The new concept is summarized here, details can be found in Reference 2.

In cooperation with the Deep Space Network (DSN), systems are being developed that will provide two orders of magnitude improvement in the downlink of science information from Galileo to Earth. Half of this improvement will be in actual data rate improve-

ments resulting from application of advanced error-correcting coding techniques and advanced technology receivers that enable shifting all of the power of the radio signal into the telemetry side-bands and also facilitate arraying of multiple tracking stations. The other order of magnitude improvement will be achieved by increasing the information density of the downlink via reprogramming of onboard computers to apply state-of-the-art data compression techniques (References 3 and 4) as well as extensive onboard editing of data from the science instruments.

The Galileo science community estimates that 70% of the original science objectives can be achieved by the new mission concept. This includes all of the objectives associated with the Probe, since the data quantity is small and the full data set can be recorded on the Orbiter and returned using the LGA even without the spacecraft software and DSN enhancements.

Figure 2 illustrates the new operational concept for a typical orbit. Since most of the key opportunities for imaging and other remote sensing occur in a 7-day "encounter" period centered (roughly) at perijove,

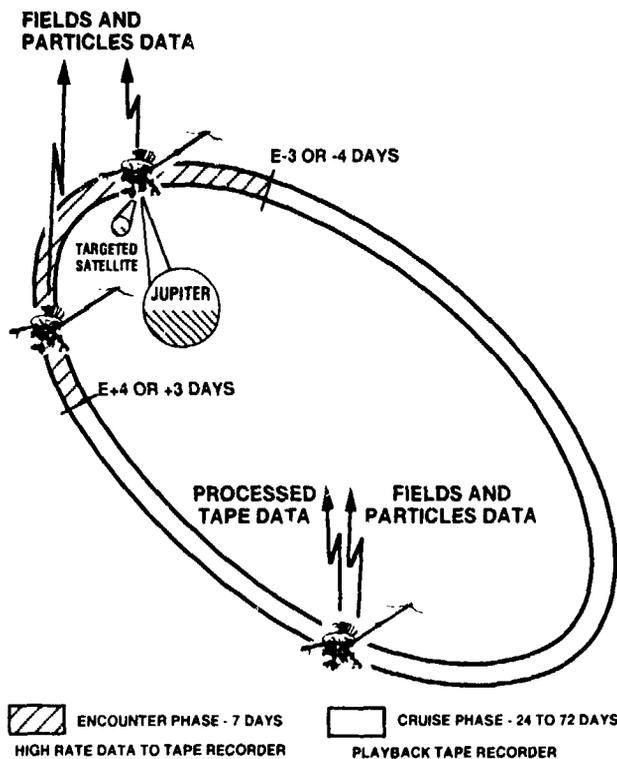


Figure 2. Typical Orbit

these observations can be recorded and subsequently played back in compressed or edited form during the "cruise" period between encounters (24-72 days). In addition to the return of recorded encounter data, a continuous stream of highly-edited real-time data (predominantly from the fields-and-particles instruments) can be downlinked throughout both the encounter and cruise periods.

The flight software (FSW) modifications that provide these new capabilities (designated "Phase 2" in Figure 1) are currently being developed and will be uplinked in the spring of 1996. The Phase 1 FSW modifications will be uplinked early in 1995 and will provide for protection of the Probe data against tape recorder problems by storing key data in the on board computer.

2. DRIVERS ON MISSION PLANNING

With a mission design that includes six years of interplanetary cruise and two years of orbital operations, the subject of what mission planning to do when has long been debated within the Project. The need for early development and testing of the highly critical sequences for Probe data relay through the

Orbiter and the Jupiter Orbit Insertion maneuver was never at issue, but there has been less certainty about the level of detail of planning for orbital operations. For the original mission concept there was concern about the difficulty of building and implementing eleven complex satellite flyby sequences (an Io flyby on the day of Jupiter orbit insertion plus ten orbital encounters), with substantial contention among the eleven orbiter instruments for observation time and sequence memory (particularly the four instruments on the scan platform). So the pre-launch Project Plan called for early development of detailed plans that would precisely allocate these resources.

The modifications for LGA-based operations added to the list of critical resources while making precise early allocation of these resources a lot more difficult. The most significant resource for LGA-based operations is the downlink capability (usually referred to on the Project as "BTG" or "bits-to-ground", although commonly measured in megabits). Space on the tape recorder ("bits-to-tape") is also a crucial commodity, since the recorder can only be filled once for each satellite flyby and for the "best" orbits (long periods between flybys coupled with small Earth-Jupiter range) there is enough BTG capability to empty the tape recorder at acceptable compression ratios. The criticality of the tape recorder to the LGA-operations concept has also added the cycle-life of the recorder to the list of resources that must be closely managed.

The interplanetary cruise phase encounters have provided experience in dealing with these scarce resources and have generally confirmed the need for detailed early planning. The Venus and Gaspra flybys were constrained largely by space on the tape, since there was ample playback capacity at subsequent Earth flybys; the Earth flybys themselves were useful exercises in dividing up observing time; and the Ida flyby was the first experience with severe BTG limitations. These experiences left no one doubting the wisdom of having detailed plans in place well in advance of the high activity periods.

The Galileo mission planners must, however, now deal with a high degree of uncertainty in allocating BTG (their most critical resource). The DSN enhancements discussed in Section 1 include the first application of new technology in several areas, and, while confidence is high, no comprehensive end-to-

end performance test will be possible until shortly after the Phase 2 FSW modifications are loaded in the Spring of 1996. Uncertainty in performance of data compression algorithms is also a major hindrance to precise planning. Compressibility of some imaging data (and the corresponding BTG allocation) will be known to within only a factor of two a priori.

Another driver on the planning process is the continuing pressure on operations budgets of NASA missions. The mission plan must be structured so that it can be implemented with a staffing level substantially reduced from the original Project Plan.

3. THE MISSION PLANNING PROCESS

Galileo mission planning and sequence development have always used a top-down design process. The products are as follows: (1) the Orbit Planning Guide (OPG) providing a high level orbit-by-orbit allocation of resources across the tour, (2) Orbit Activity Plans (OAPs), one for each orbit, which suballocate resources among individual activities in a time ordered listing, (3) a set of Orbit Profiles for each orbit, in which the OAP activities are expanded in terms of sequence components which can be automatically converted to (4) an uplinkable command file of 1000-3000 commands. Steps (1) and (2) are viewed as mission planning and are the focus of this paper.

In 23 months the Galileo Orbiter will navigate through an eleven-orbit tour. Experience during interplanetary cruise has shown that the complete sequence planning process for each orbit will take considerably more than two months. Hence the sequence planning process must begin before the Jupiter tour begins. This has led to a schedule (see Figure 1) under which the OPG was completed in February 1994 and orbit-by-orbit sequence development began in July 1994. In the pre-arrival planning, the encounter sequence for each targeted fly-by of a Galilean satellite will be developed in full detail immediately following the OAP development. All OAPs and encounter sequences are scheduled for completion prior to the first satellite encounter of the tour (July 1996).

The Galileo mission planning process is intertwined with the structure of the Galileo science community. The Galileo flight team at JPL is organized to interface with and support the science investigator teams which are organized by instrument. Each of the

instrument and radio science experiments on the Probe and on the Orbiter, is lead by a Principal Investigator (or Team Leader for SSI and Radio Science) with a group of Co-Investigators (or team members). Most of the Galileo investigators are located at other institutions than JPL. The Principal Investigators, Team Leaders, and a number of Interdisciplinary Scientists comprise the Project's senior science planning agency, the Project Science Group (PSG). The PSG has subcommittees - called working groups - which cross-cut the instrument teams to deal with top level priorities and plans in the three major discipline areas called out in the Project Plan: Atmospheres, Satellites and Magnetosphere. All of the Orbiter investigator teams are represented at JPL by an operations support team lead by a Science Coordinator. Through periodic meetings and on-going dialogue of the PSG and the working groups, the mission goals are turned into operations plans at JPL.

As part of the planning process, resources are allocated as early as possible during development. Tape usage (bits-to-tape), telemetry usage (bits-to-ground), and propellant usage (kilograms) were allocated to the discipline working groups as part of the OPG. Within the discipline working groups and as part of the Orbit Activity Plans, those resources get sub-allocated to the eleven instruments and radio science. Tape recorder cycles and sequence memory usage cannot be allocated until a high level sequence is available; they are first allocated in the OAP. As part of sequence adaptation during orbital operations, all of these resources are subject to some re-allocation.

In addition to distributing the key spacecraft resources among the three science disciplines, the OPG also describes the high-level plan for how each science discipline will accomplish its science objectives consistent with the distribution of resources. The process of developing the resource allocations was influenced by a number of factors: experience with the previous (pre-launch) OPG, experience with Galileo planetary encounters on the way to Jupiter, scoping exercises and of course, schedule. Allocations of resources across science discipline areas, based on scientific consideration, are always difficult to get agreement on; the investigators, science elements of the JPL team and the Project Scientist worked together to arrive at the current position. An initial allocation of resources to the working groups over the whole tour was developed by the Project

Scientist. This initial allocation provided the basis for further negotiation and trading of resources between the working groups with the outcome being orbit-by-orbit allocations, driven by and consistent with the characteristics of the orbital tour.

The first two weeks of the 8-week OAP development cycle involve two parallel tasks: building an engineering and navigation "skeleton" plan and initiating work on satellite encounter remote sensing designs for the critical period around closest approach. The skeleton schedules and allocates resources for spacecraft systems maintenance and calibration, attitude updates, optical navigation imaging, radiometric navigation, and orbit trim maneuvers. The remote sensing design uses sophisticated 3-D cartographic tools to account for target ephemeris, spacecraft trajectory, and scan platform dynamics in laying out mosaic patterns and target-to-target scan platform slews. This must be done at a fine level of detail at the beginning of the OAP to get a handle on the resource needs of the observations near closest approach.

Next, OAP development enters a 4-week iterative period in which the remainder of the science observations are assigned, resource needs are estimated, the activity timeline is built, deviations from operating constraints are identified, and all of this is iterated where conflicts are found. During this period the working groups divide BTG and other resources among the participating instruments and the instruments divide them among individual observations. This includes separate BTG allocations for tape re-

order playback and real-time science. Conflicts with the "skeleton" are also subject to iteration.

The final two weeks of the OAP cycle are devoted to a last round of constraint checking, review of the integrated product by all participants, and approval by project management.

4. ORBIT PLANNING GUIDE RESULTS

This section summarizes the results of the OPG development completed in February 1994 (Reference 5). In particular, Table 1 summarizes the results of the OPG negotiations among the working groups for allocating BTG and tape recorder space for the orbital tour. The table gives the total BTG available to science during the cruise phase for each orbit (in megabits), the percentage of the BTG allocated to each working group, and the percent allocation of the encounter tape load. The working group allocations for the J0 encounter (J0) and the G1 orbit were combined because the expectation is that all of the J0 data cannot be returned prior to the G1 encounter. Some J0 data will be carried over and played back during the G1 cruise period. For the C9 orbit, the total telemetry capability has not been fully allocated to the working groups at the OPG level since it is more than enough to play back the tape. Some additional recording and play back during the cruise period of the orbit is planned.

A number of science trades were necessary to develop the allocations in Table 1. The long-range,

Table 1. OPG Resource Allocation

Orbit	Capability (MBTG)	Satellites		Magnetosphere		Atmosphere	
		% of BTG	% of Tape Load	% of BTG	% of Tape Load	% of BTG	% of Tape Load
J0	170		50%		13%		13%
G1	225	35%	35%	48%	5%	17%	38%
G2	155	22%	58%	70%	18%	8%	25%
C3	110	49%	53%	20%	3%	31%	45%
E4	100	50%	50%	17%	3%	33%	48%
E6	110	40%	53%	40%	8%	20%	40%
G7	90	40%	55%	40%	8%	20%	38%
G8	95	35%	45%	40%	10%	25%	45%
C9	460	24%	53%	40%	16%	14%	55%
C10	200	28%	40%	40%	10%	32%	50%
E11	115	40%	40%	30%	10%	30%	50%
Totals	1930	33%	46%	41%	8%	21%	41%

short-duration orbits of C3, E4, E6, and G7 posed particular difficulty. For the satellite working group (SWG), these orbits contain the high priority target Europa. In the case of the magnetospheric working group (MWG) continuous real-time monitoring of Jupiter's dynamic magnetosphere is their highest science objective. The atmospheric working group (AWG) is more flexible with respect to acquiring specific science objectives during these orbits, but they still require that their primary science objectives be met by the end of the mission. The compromises made for these orbits consisted of the MWG reducing their requests on the downlink telemetry during C3 and E4 in order to accommodate the SWG's requests for telemetry during these scientifically important orbits, and SWG and AWG reducing their telemetry requirements for G2, which permitted MWG to utilize most of the capability for this orbit. As a result, the MWG developed the concept of two magnetospheric sub-tours, one at the beginning of the orbital tour and the second during the last orbits. The sub-tour concept is illustrated in Figure 3.

As a result of the science trades made to generate the resource allocation table, each of the working groups will address the most important of their key scientific questions about the Jovian system. For AWG, the focus of the science instruments will be an integrated study of small areas of Jupiter ("features") and those observations that are unique in terms of instrumental capability or geometric opportunity.

The MWG's primary science objective is the magnetospheric survey. In order to investigate the large-scale topology and temporal behavior of the magnetosphere, the concept of two sub-tours was introduced. In addition to the above sub-tours, it is important that the region inside $50 R_j$ be continuously sampled for each orbit. A major objective in the second sub-tour is the journey into the unexplored regions of Jupiter's magnetotail. MWG's second primary objective has also been retained: high resolution coverage of the close flybys of the Galilean satellites.

The SWG satellite priorities are Io (single flyby), Europa, Ganymede, and Callisto. For the imaging experiment a high priority objective is to achieve global coverage complementary to that of Voyager as well as limited coverage 100-1000 times higher resolution than Voyager. For Near-Infrared Mapping Spectrometer (NIMS), the global coverage objective is to achieve coverage of a high percentage of the surface at modest spatial and spectral resolution, since all coverage of the satellites in the NIMS wavelength regime is new. The Photopolarimeter Radiometer observation set includes thermal and polarization observations. The ultraviolet experiment set includes limb scans as high priority. Most of the remaining observations for SWG consist of focused studies of very limited spatial extent for specific features or regions on the satellites.

5. MISSION PLANNING TOOLS

The flight software changes associated with operating the Galileo spacecraft using the LGA provide significant challenges and added complexity in the development of the science sequences. There are now complex interactions among collection and transmission of real-time science, transmission of engineering data, collection of recorded science, and playback of recorded data. For example, changes to the real-time science collection rate during the cruise portion of the orbit affect the amount of recorded science that can be played back during the same period. In a sample orbit planning exercise (SOPE) conducted in 1993 in order to understand the process of how science sequences are developed using the new Phase 2 flight software, it became clear that a mission planning tool would be needed to efficiently and successfully develop the flight sequences. The SOPE illustrated the need to modify an activity plan

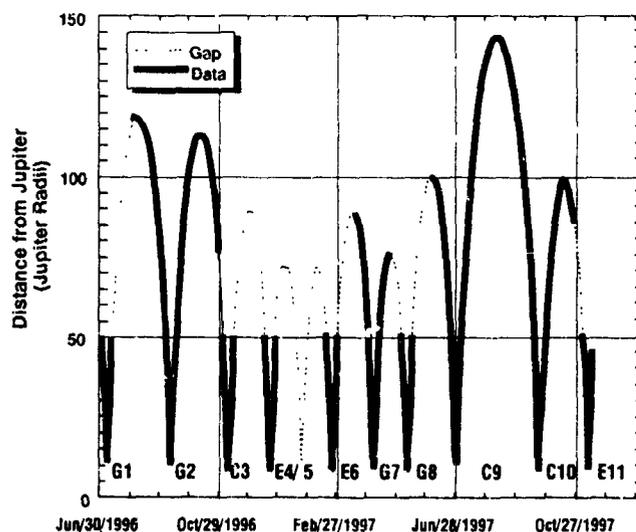


Figure 3. Magnetospheric Survey Subtours

in development often and provide for fast turn-around estimates of the effects on spacecraft resources. In addition, in light of the current economic environment on Galileo, reductions in the mission operations workforce also require that automation tools be developed.

The key mission planning tool that is being developed as a result of these needs is called **MIRAGE**, for **Mission Integration, Real-time Analysis, and Graphical Timeline Editor**. The **MIRAGE** software will expedite integration and conflict resolution, and provide modeling of spacecraft resources for science and engineering activities. It utilizes a graphical user interface with a timeline representation of the sequence in development. The **MIRAGE** software allows the user to quickly and easily manipulate science and engineering activities and provides for immediate feedback on the expected spacecraft resource usage resulting from these changes. The resources modeled within **MIRAGE** include onboard computer buffer usage, real-time science BTG, recorded science tape usage, play back BTG, tape recorder start/stop cycles, sequence memory usage, and resource claim violations with respect to the scan platform, the spacecraft attitude, and the real-time and record telemetry formats.

MIRAGE is the Galileo adaptation of the multi-mission **PLAN-IT-2** (for **Plan Integrated Timelines, version 2**) science planning software developed at JPL (see Reference 6). **PLAN-IT-2** is an activity scheduling program that provides for sequence visualization to aid in the resolution of conflicts between spacecraft activities. It is written in LISP and runs on a UNIX workstation. **PLAN-IT-2** presents the sequence to the user in the form of a timeline display showing the activities, conflicts, and any constraints that need to be considered in the sequence. The decision to use **PLAN-IT-2** in the development of the **MIRAGE** software was driven by several factors, including the limited amount of software development time for **MIRAGE**, the immediate availability of a graphical user interface for timeline displays, and the capability to incorporate Galileo-specific constraint checking and spacecraft models. Adaptation of **PLAN-IT-2** for Galileo involved reconfiguration of the display; incorporation of Galileo-specific resource constraint checks; definition of the format, content, and representation of the science and engineering activities; incorporation of resource model-

ing; and configuration of the internal time system and time representations. An example screen from the Galileo adaptation of **PLAN-IT-2** is shown in Figure 4.

The primary use for **MIRAGE** is in the development of the OAPs. **MIRAGE** will compile the desired engineering, real-time science, and recorded science activities, model and track the resources listed above, and summarize resource usage by science instrument, science working group, or activity.

For the OAP integration activities, **MIRAGE** will be used in a sequence integration workroom environment. Here, all flight team members responsible for producing a conflict-free integrated plan will use **MIRAGE**'s interactive and real-time capabilities to negotiate activity timings, move, delete, and/or update the activities, and display the effects of those changes in spacecraft resources. Workroom tools will include a large screen for display of **MIRAGE** outputs like Figure 4.

Two other tools being developed by Galileo to further increase the amount of automation involved in the sequence development process are **SCAN-IT**, which is a sequence review tool to provide automated checking of spacecraft and instrument flight rules, and **OAPLINK**, which is a tool used to expand high-level activities into sequence components. The **SCAN-IT** software is a Galileo adaptation of an existing multi-mission sequence review tool, which is a Unix based program and written in LISP. The adaptation process involves the incorporation of the relevant flight rules via a set of **SCAN-IT** scripts. The **OAPLINK** software has been in use on the Galileo flight team for the past couple of years.

6. IMPLICATIONS FOR FUTURE MISSIONS

While some of the work described here is peculiar to Galileo's anomaly response situation, a number of the mission planning factors discussed in this paper have far-reaching implications. First, data compression is likely to be an important element of future space missions and the mission planning implications of data compression described here, particularly the need to deal with the resulting uncertainty in effective downlink capability, will be widely applicable. Another conclusion is that software tools are now available to support activity planning and re-

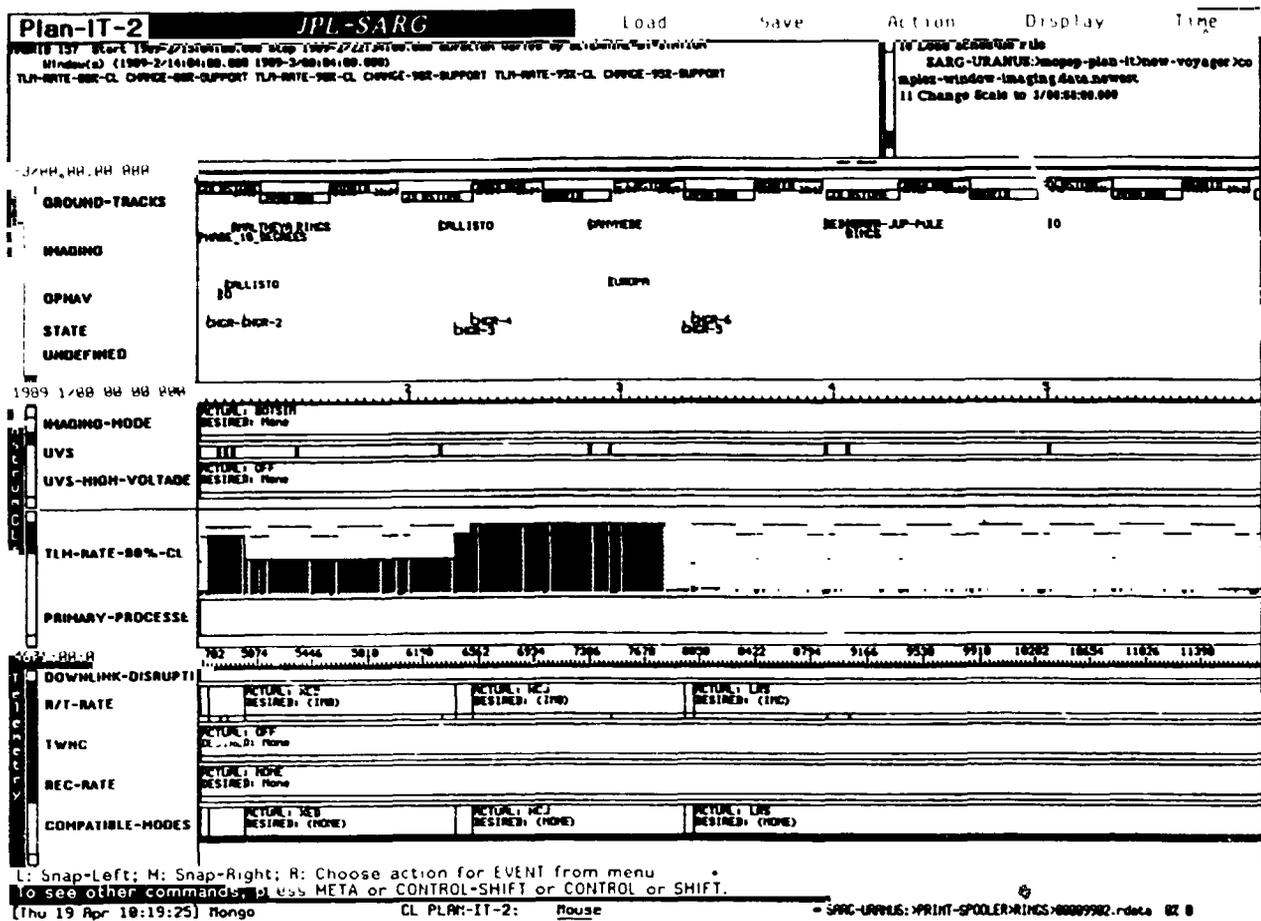


Figure 4. Sample Plan-It-2 Display

source allocation. These have great value and should be considered in the earliest stages of designing mission operations systems.

7. ACKNOWLEDGMENTS

The research described in this paper was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

REFERENCES

1. Russell, C.T., (Ed.) (1992). *The Galileo Mission, Space Science Reviews*, 60 (1-4), Kluwer Academic Publishers.
2. O'Neil, W. J., et. al. (1993, October). *Performing the Galileo Jupiter Mission with the Low-Gain Antenna (LGA) and an Enroute Progress Report*. (IAF-93.Q.5.411), presented at the 44th Congress of the IAF, Graz, Austria.
3. Rice, R.F., (1993, November 15). *Some Practical Universal Noiseless Coding Techniques., Part III, Module PSI 14, K+* (JPL Publication 91-3). Pasadena, CA: Jet Propulsion Laboratory.
4. Cheung, K. & Tong, K., (1993, April 2). *Proposed Data Compression Schemes for the Galileo S-Band Contingency Mission. Proceedings of the 1993 Space & Earth Science Data Compression Workshop*, Snowbird, Utah.
5. Paczkowski, B., et. al. (1994, February). *Galileo Orbit Planning Guide* (Internal report D-11524). Pasadena, CA: Jet Propulsion Laboratory.
6. Eggemeyer, W. & Cruz, J. (1990). PLAN-IT-2: The Next Generation Planning and Scheduling Tool. *Telematics and Informatics*, 7 (3/4), 189-207.

31763

A DISTRIBUTED PLANNING CONCEPT FOR SPACE STATION PAYLOAD OPERATIONS

Jeff Hagopian

Theresa Maxwell

Tracey Reed

Advanced Planning Branch (EO45)

Marshall Space Flight Center, AL 35812

ABSTRACT

The complex and diverse nature of the payload operations to be performed on the Space Station requires a robust and flexible planning approach. The planning approach for Space Station payload operations must support the phased development of the Space Station, as well as the geographically distributed users of the Space Station. To date, the planning approach for manned operations in space has been one of centralized planning to the n-th degree of detail. This approach, while valid for short duration flights, incurs high operations costs and is not conducive to long duration Space Station operations. The Space Station payload operations planning concept must reduce operations costs, accommodate phased station development, support distributed users, and provide flexibility. One way to meet these objectives is to distribute the planning functions across a hierarchy of payload planning organizations based on their particular needs and expertise. This paper presents a planning concept which satisfies all phases of the development of the Space Station (manned Shuttle flights, unmanned Station operations, and permanent manned operations), and the migration from centralized to distributed planning functions. Identified in

this paper are the payload planning functions which can be distributed and the process by which these functions are performed.

1. INTRODUCTION

The key to any successful project, be it a complex space mission or a simple family picnic, is proper planning and preparation. The planning approach used must be tailored to meet the specific needs of the problem at hand. The Space Station payload operations planning problem is considerably different from the payload operations planning problem associated with current Shuttle missions. The characteristics of this problem which make it so very different are: large numbers of geographically distributed payload users (e.g., users in the United States, Japan, Canada, Europe, etc.), multiple operations control centers, continuous operations, diverse and dynamic payload complements, and a desire for operational flexibility. With these characteristics in mind, it is crucial that a payload operations planning concept be developed which meets the needs of the payload user community and the Space Station program.

2. PLANNING CONCEPT

Because of the diverse and dynamic payload complement, no one organization will have

the knowledge and expertise required to perform all of the detailed planning. Since the knowledge and expertise is spread across the various organizations and users, it makes sense to distribute the planning as well. While there are many possible ways of supporting distributed planning, the hierarchical distribution of resources appears to be the approach which is best suited for Space Station payload operations planning. An overview of this concept is provided in the following sections which describe the architecture, resource envelopes, and planning process. The architecture and resource envelopes are discussed first to provide the reader with a basis for understanding the planning process. Rather than expressing this concept using Space Station specific terminology, the concept is described in general terms which can be applied to other planning problems.

2.1 Architecture

Figure 1 provides an overview of the architecture which supports this approach. This architecture consists of various levels of planning, where the functions of a particular level are performed by one or more organizations.

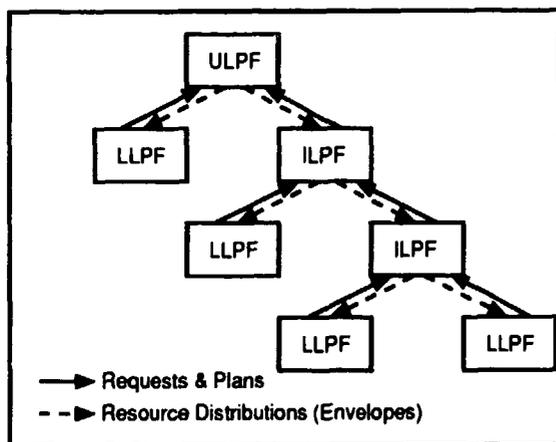


Figure 1. Architecture

In general, there are three basic levels of planning: 1) Upper Level Planning Function

(ULPF), 2) Lower Level Planning Function (LLPF), and 3) Intermediate Level Planning Function (ILPF).

The ULPF represents the controlling authority and is ultimately responsible for the integrated plan of payload operations. There is only one ULPF, although there may be many organizations which support its functions. The LLPF represents the individual users of the Space Station. These individuals have specific payload operations which need to be scheduled, and are in competition with one another for the limited resources available to support those operations. The ILPF represents the organization or organizations which serve as the interface between the ULPF and the LLPF. In most cases, the ILPF represents the sponsoring organization or country of the users. In cases where there is no ILPF organization, the LLPF interfaces directly with the ULPF. There may be multiple ILPF levels, where one ILPF organization exists to serve the ILPF organizations which fall under its authority. Refer to Figure 1 for a pictorial representation of this architecture and the relationships between the ULPF, ILPF, and LLPF organizations.

The basic premise of this concept is that resources are distributed in a manner which allows for concurrent planning at each level in the architecture. Requests for resources are passed from the LLPF upwards through the ILPF level(s) to the ULPF. The ULPF, taking into account all of the requests for resources, distributes the available resources to the ILPF. Each ILPF then distributes its resources to the level below it, either another ILPF level or the LLPF. At the LLPF level, the users develop plans within their resource distributions and pass those plans back up through the path to the ULPF. Each level, by having a view into all of the requests for resources at its level, can ensure an equitable

distribution of resources to best satisfy the needs of its users. The flow of this information from one level to the next is depicted in Figure 1.

2.2 Resource Envelopes

Resources are distributed to the planning levels in the form of resource envelopes. A resource request is the *time-independent* distribution of the magnitude of a resource over time. In contrast, a resource envelope is the *time-dependent* distribution of the magnitude of a resource over time. The development of envelopes involves assigning a resource request to a specific time period. Figure 2 shows an example of a resource envelope. Resource envelopes are created for each resource that constrains planning. For example, there are envelopes for power, data, crew, etc. The resource requirement shown in Figure 2 represents a power profile required to perform an operation or group of operations. The ILPF or LLPF organizations may request a resource in excess of the actual requirement to allow for the desired operational flexibility. The resource requests are submitted to the appropriate planning level for resource envelope development.

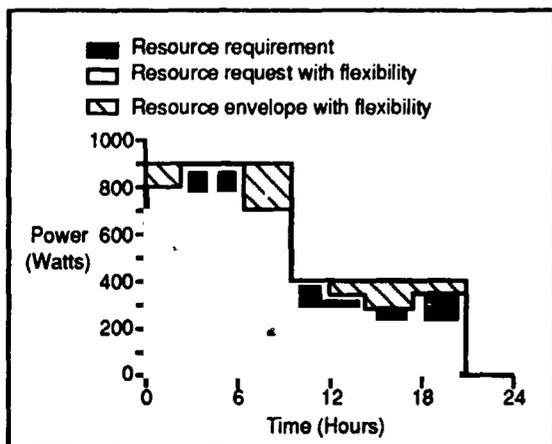


Figure 2. Resource Envelope

Resource envelopes are developed to satisfy resource requests within the resource avail-

abilities and other constraints. The resource envelope defines a profile that is greater than or equal to the resource request. Additional flexibility may be added to the resource request to simplify the resulting profile. Once a resource envelope is developed and distributed to the appropriate level, additional envelopes can be created at that planning level based on the resource availability profile provided in its resource envelope. These envelopes are created in a manner which ensures that no overbooking of the resource occurs. Figure 3 illustrates the distribution of resource envelopes.

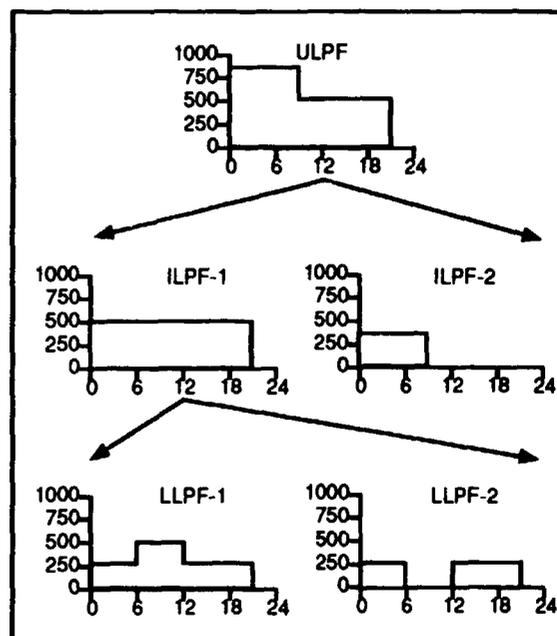


Figure 3. Envelope Distribution

2.3 Planning Process

The process for developing payload operations schedules is usually tailored to the environment in which the planning is performed. Problem characteristics, planning cycles, unique product requirements, functional interfaces, and planning software capabilities factor into the definition of the planning process. The distribution of planning responsibilities will also significantly

affect the design of the planning process. The Space Station payload planning process will therefore differ somewhat from the processes used for Space Shuttle/Spacelab payloads or for unmanned free-flyer payloads. However, there are also similarities. The Space Station planning process must support manned operations, like Shuttle/Spacelab, as well as continuous operations and unmanned periods, like the free-flyers.

The key to developing a distributed planning process is that all planning processes are built upon the same fundamental set of planning functions:

- **Constraint Definition**
Defines all constraints on scheduling, including the scheduling horizon, groundrules, definition of resources and system configurations, resource availability profiles, etc. Resources may represent physical objects, such as equipment; systems services, such as power; or environmental conditions, such as microgravity or orbital daylight.
- **Requirements Definition**
Defines the requirements of each operation to be scheduled. These requirements may include resource usage profiles, temporal relationships to other operations, and performance requirements (number of performances of the operation and their required distribution over time).
- **Scheduling**
Produces conflict-free schedules which satisfy the scheduling requirements within the defined constraints.
- **Product Generation**
Produces integrated payload plans and data which can be used to analyze and/or execute the schedule.

The major difference between a centralized

planning process and a distributed one is *who performs* each of the functions.

Typically, the requirements definition function is performed by those organizations or individuals who have in-depth knowledge of the operations to be scheduled, such as the users who sponsor the payloads on the Space Station. In the planning architecture discussed earlier, these organizations and/or individuals would belong to the LLPF. In a centralized planning environment, the other planning functions are performed by a single centralized authority, represented in the planning architecture by the ULPF. Figure 4 represents a typical centralized planning process.

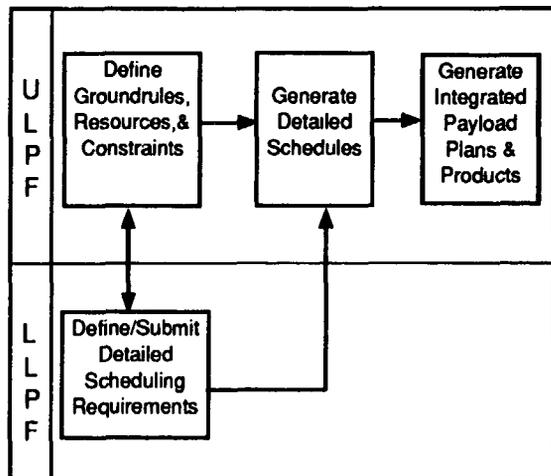


Figure 4. Centralized Planning Process

In a distributed planning environment, the responsibility for performing each of the planning functions may be distributed across the entire hierarchy of payload planning organizations (ULPF, ILPF, LLPF), as discussed in the architecture section. The degree to which the planning functions can be distributed depends on many factors, including the abilities and desires of the various organizations to actively participate in the planning process.

Figure 5 depicts a distributed planning process with each of the planning functions fully distributed across the various planning levels (ULPF, ILPF, LLPF). A discussion of this process follows. To simplify the discussion, Figure 5 is shown with exactly one ILPF level between the ULPF and LLPF. The process can easily be modified to accommodate an architecture with multiple ILPF levels or no ILPF level at all. It will also support centralized planning if the ULPF organization performs all of the planning functions except requirements definition, which must be done by the LLPF.

The *Constraint Definition* function may be distributed if there are particular resources or groundrules which are unique to a single payload (LLPF) or group of related payloads under a common ILPF organization. For example, a group of life science payloads under a common ILPF might share the use of a life science glovebox. Such constraints may be defined at the appropriate ILPF or LLPF

level. Space Station systems services, crew, and all other constraints which apply across multiple ILPF organizations must be defined and controlled by the ULPF. Although constraints may be defined at any level, it is extremely important that all organizations are planning against a common and consistent set of constraints. Visibility into all levels is required to ensure that conflicts in constraint definition do not occur. For example, the creation of three distinct resources with the name "Glovebox" by different organizations would complicate the schedule integration function later in the process.

As in the centralized process, the *Requirements Definition* function is primarily performed by the LLPF. In the centralized process, the LLPF submits detailed scheduling requirements which the ULPF can utilize in scheduling and product development. In a distributed process, however, the LLPF submits requests for resources within which it can perform its own detailed scheduling. As

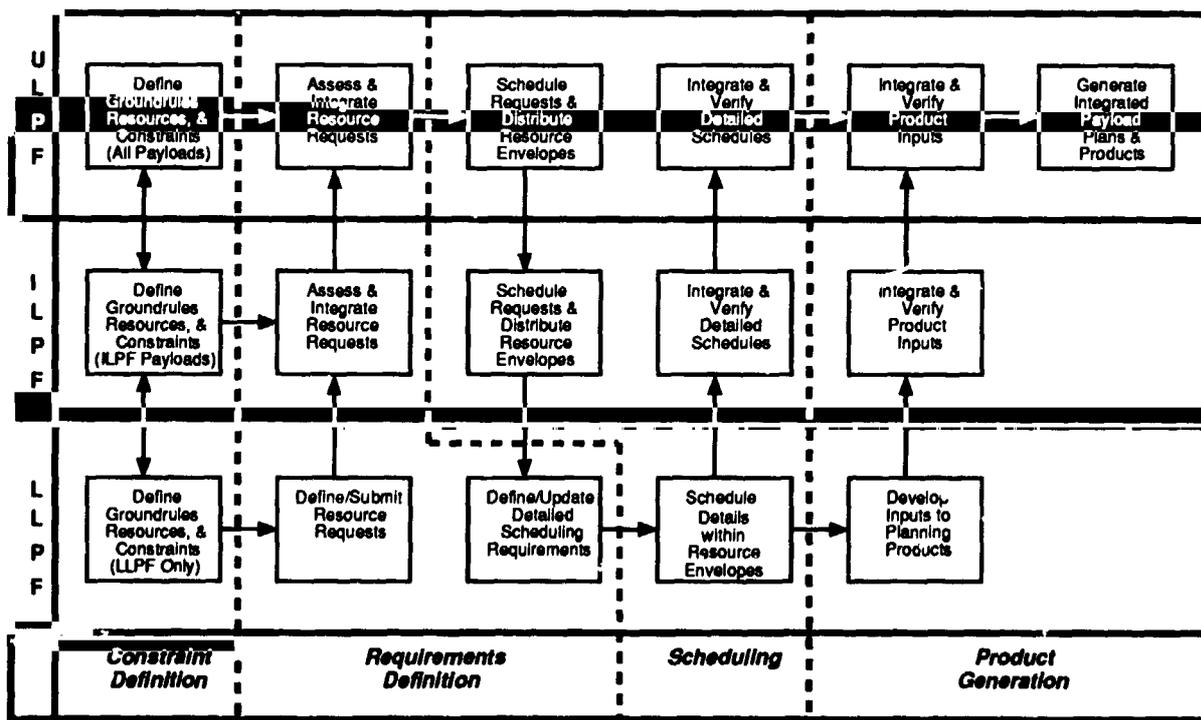


Figure 5. Distributed Planning Process

was discussed in the section on resource envelopes, a resource request may represent the exact requirements of a specific operation, or it may grossly define a set of resources which accommodates the requirements of one or more operations. A gross resource request will provide the LLPF with any desired flexibility in the detailed scheduling step of the process.

Each ILPF collects and assesses the resource requests submitted by its associated LLPF organizations. Conflicts between LLPF requests are resolved at this point. Based on its objectives and priorities, the ILPF may choose to forward any or all of the individual LLPF resource requests to the ULPF. The ILPF may also choose to merge multiple LLPF resource requests into larger ILPF resource requests. This may provide the ILPF with some desired flexibility in the scheduling step of the process.

When all of the ILPF/LLPF resource requests are submitted, the ULPF is ready to begin the *Scheduling* process. By having visibility into all users' needs (via the resource requests), the scheduling process can ensure an equitable distribution of resources across the entire payload complement. First, the ULPF schedules the integrated set of resource requests against the defined constraints. From this integrated schedule, resource envelopes are then constructed for each ILPF. These envelopes may contain resources in excess of what was requested by the ILPF. A key aspect of this concept is that the sum of the distributed resource envelopes created at any level cannot exceed the resource availabilities (no overbooking of resources allowed). This ensures that the detailed schedules created at lower levels will not produce constraint violations when integrated together. Note that the ULPF may only distribute resource envelopes for those

resources which are under its control.

Next, each ILPF follows a similar process to divide its resource envelopes into individual LLPF resource envelopes. Any resources under the control of the ILPF may be distributed at this time.

Detailed scheduling of specific operations is then performed by the LLPF within the resource envelopes assigned by the ILPF. Prior to scheduling, the LLPF completes the *Requirements Definition* process for its operations by defining/updating the detailed scheduling requirements.

The last step in the *Scheduling* process is the integration of the independently developed detailed schedules. Integration is performed in an upwards fashion through the ILPF to the ULPF. Each planning level verifies that the detailed schedules it integrates are compatible with the appropriate resource envelopes. As part of the integration function, the ULPF may perform any additional planning tasks required to finalize the integrated schedule of payload operations.

The *Product Generation* function may also be distributed to a certain extent. Some additional information, not required for scheduling, must be associated with the payload schedule in order to generate the products which are used by the onboard crew, onboard software, and ground controllers to execute the schedule. Examples of these product inputs include identification of the detailed procedures to be executed for each scheduled operation, and associated notes. Since the LLPF has the most intimate knowledge of the payload operations and procedures, it builds the product inputs, which are then integrated by the ILPF and ULPF for inclusion in the final products.

3. ANALYSIS AND EVALUATION

As with any complex concept or process, there are a number of strengths and weaknesses associated with the distributed planning concept described in this paper. A discussion of the known advantages and disadvantages follows.

3.1 Advantages

The distributed planning concept provides a number of advantages which make it particularly attractive as a solution to the Space Station payload operations planning problem. Following is a brief summary of these advantages:

- Reduces the operations costs of the ULPF organization through the increased participation of the ILPF and LLPF organizations.
- Provides operational flexibility at the appropriate level of fidelity through the use of resource requests and resource envelopes. This flexibility results in a plan which is better able to accommodate changes during plan execution.
- Places responsibility for planning at the level where the knowledge and expertise exists. The end users (LLPF) are active participants in the process and are not simply viewed as data providers.
- Results in the production of conflict-free plans through the use of resource envelopes which do not allow for the overbooking of resources.
- Supports the transition from centralized to distributed planning, as well as a mixture of both centralized and distributed concepts. The planning process remains fairly stable regardless of the number of organizations performing the various planning functions.
- Ensures equitable distribution of resources among the payloads through

visibility into the integrated set of resource requests.

3.2 Disadvantages

The distributed planning concept also has a number of disadvantages associated with it. Many of these disadvantages are a direct result of the distribution of planning functions and would probably manifest themselves in other distributed planning concepts. Following is a brief summary of these disadvantages:

- Increases operations costs to the ILPF and LLPF organizations due to their more active role in the planning process.
- Results in less efficiency in the planned utilization of resources. The flexibility built into the resource requests and resource envelopes results in the scheduling of resources which may not actually be utilized.
- Results in longer planning cycles due to the active involvement of all levels in the planning process. Sufficient time must be provided to allow each level to perform its required functions, as well as to account for the transfer of information from one level to the next.
- Requires a significant amount of coordination to define the planning constraints. The success of this concept depends on all of the various organizations using a well defined and consistent set of planning constraints.
- Results in numerous and complex interfaces to support the distribution of the planning functions. Organizations involved in the process will be geographically distributed and will be working in facilities which may or may not be similarly equipped.
- Requires a rigorous configuration management process to ensure that all organizations are using the most current

data and that changes to the data are only made by authorized organizations.

4. CONCLUSIONS

The complex and diverse nature of the payload operations to be performed on the Space Station will require a change in the current payload operations planning philosophy. The unique characteristics of the Space Station payload operations planning problem drive the need for a distributed payload operations planning concept.

The key to a successful payload operations planning concept is to develop an approach which will meet the needs of the payload user community and the Space Station program. The authors believe the distributed planning concept presented in this paper provides a robust and flexible planning approach which will support the phased development of the Space Station, accommodate a large number of geographically distributed users, accommodate diverse and dynamic payload complements, as well as provide for operational flexibility. There are significant

benefits to be gained with this concept if the Space Station program is willing to accept the disadvantages. The authors feel this is a viable concept which is being actively pursued for implementation. This concept will need to be revisited to accommodate changes as the Space Station program evolves. Also, it is acknowledged that certain functions associated with this concept will require further study and development.

5. REFERENCES

1. NASA, International Space Station Alpha Program. Concept of Operation and Utilization, SSP 50011-01, Vol. 1, Principles, March 1994.
2. NASA, International Space Station Alpha Program. Concept of Operation and Utilization, SSP 50011-03, Vol. 3, Processes, April 1994.
3. NASA, Johnson Space Center. Operations Phase Assessment Team II Basic Report, April 1993.

MISSION PLANNING FOR SPACE-BASED SATELLITE SURVEILLANCE EXPERIMENTS WITH THE MSX

R. Sridharan, T. Fishman, E. Robinson, H. Viggh and A. Wiseman
Lincoln Laboratory, Massachusetts Institute of Technology

Abstract - The Midcourse Space Experiment is a BMDO-sponsored scientific satellite set for launch within the year. The satellite will collect phenomenology data on missile targets, plumes, earth limb backgrounds and deep space backgrounds in the LWIR, visible and ultra-violet spectral bands. It will also conduct functional demonstrations for space-based space surveillance. The Space-Based Visible sensor, built by Lincoln Laboratory, Massachusetts Institute of Technology, is the primary sensor on board the MSX for demonstration of space surveillance. The SBV Processing, Operations and Control Center (SPOCC) is the mission planning and commanding center for all space surveillance experiments using the SBV and other MSX instruments. The guiding principle in the SPOCC Mission Planning System was that all routine functions be automated. Manual analyst input should be minimal. Major concepts are: (1) A high level language, called SLED, for user interface to the system; (2) A group of independent software processes which would generally be run in a pipe-line mode for experiment commanding but can be run independently for analyst assessment; (3) An integrated experiment cost computation function that permits assessment of the feasibility of the experiment. This paper will report on the design, implementation and testing of the Mission Planning System.

1.0. INTRODUCTION

The Mid-Course Space Experiment consists of a set of payloads on a satellite being designed and built under the sponsorship of Ballistic Missile Defense Organization (formerly, Strategic Defense Initiative Organization) of the Department of Defense. The major instruments are a set of long-wave infra red sensors being built by Utah State University, a set of sensors operating in the visible wavelength and ultraviolet wavelengths, being built by Johns Hopkins University's Applied Physics Laboratory, and a visible wavelength sensor designed and built by Lincoln Laboratory, Massachusetts Institute of Technology. The satellite bus is being built by JHU/APL who is also acting as the integrator for all the payloads and associated systems. The MSX satellite, shown in Figure 1, is due for launch in late 94 from the Vandenberg launch complex into a near-sun-synchronous orbit.

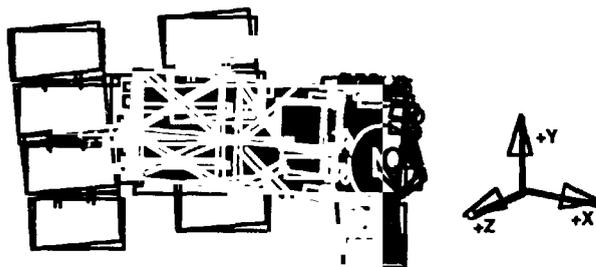


Fig. 1. MSX spacecraft

1.1. MSX Missions and Operations

The MSX satellite is being launched to conduct a series of measurements on phenomenology of backgrounds, missile targets, plumes and resident space objects (RSOs); and to engage in functional demonstrations of detection, acquisition and tracking for ballistic missile defense and space-based space (satellite) surveillance missions.

Eight Principal Investigators are associated with the MSX project. The PIs develop experiment plans that are then prioritized by the BMDO's Mission Planning Team. JHU/APL's Mission Operations Center commands the MSX to carry out the experiments and collect science data. The data are returned to the PIs for analysis and for refining the experiments.

1.2. SBV Processing, Operations and Control Center (SPOCC)

The SBV Processing, Operations and Control Center, located at Lincoln Laboratory, MIT is a component of the APL's Mission Operations Center. In this role, SPOCC (Ref. 1) generates the necessary commanding for the MSX and its sensors for all space-based space surveillance experiments defined by the PI for Surveillance; and converts and calibrates the returned science data before turning them over the SPI's Surveillance Data

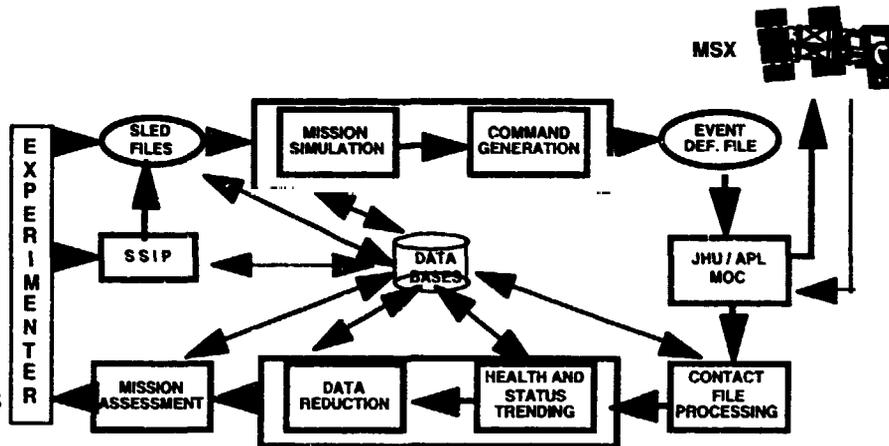


Fig. 2 : SPOCC Software System Architecture

Analysis Center. Further, SPOCC maintains the health and status of the Lincoln Laboratory's SBV sensor on board the MSX. The software architecture of SPOCC and its interaction with the MSX are shown in Figure 2.

SPOCC has four major functions:

1. SPOCC provides the facility for translating the Surveillance PI's experiments into feasible data collection events on the MSX. The Mission Planning System was designed for this purpose;
2. SPOCC monitors the health and status of the SBV using returned telemetry from the spacecraft;
3. SPOCC provides the capability to decommutate and reduce, to engineering units, the science data collected by the SBV in support of experiments; and
4. SPOCC, in association with the SBV brassboard, provides the facilities to alter, and test, software on-board the SBV in response to changing requirements.

This paper will describe the first part of SPOCC - the Mission Planning System.

The Mission Planning System in SPOCC was originally conceived and designed to support the detailed commanding of the SBV sensor on the MSX. However, because of changing requirements, it has expanded to encompass the commanding of the SPIRIT 3 and UVISI sensors also in support of surveillance experiments. This paper will describe the original design; and use the other sensors as an example of its (limited) adaptability.

1.3. SBV Hardware and Software

A working knowledge of the SBV hardware and software (Ref. 2-4) is essential to understand the design choices made in the Mission Planning System.

The SBV contains of an off-axis imaging telescope with an aperture of 15 cm and a CCD camera at the focal plane. The design improves the off-axis light rejection capability of the telescope over conventional on-axis designs and thus enables the SBV to point within 100 Km of the earth limb without saturation of the focal plane. The camera consists of four CCD arrays each

420x420 pixels, laid out along the Z-axis of the spacecraft. The instantaneous field-of-view at each focal plane is 1.4 deg x 1.4 deg. Distortion due to the off-axis design causes the total instantaneous FOV to be ~6.6 deg. x 1.4 deg.

The SBV carries a redundant pair of Signal Processors whose function is to detect moving targets in a stationary background. The Signal Processor (Ref. 3) collects a set of raw camera frame data (4 - 16 frames) and applies a space-time filtering algorithm on these data. If the telescope is pointed in an inertially invariant direction, the stars would be stationary and the Signal Processor will detect streaks corresponding to any resident space object in the field-of-view of a CCD array. If, on the other hand, a RSO is being tracked, its image would appear stationary and the stars would generate streaks. Only one focal plane array can be processed at a time. Typically, the SP takes a total time of 50 seconds from the initiation of the frame integration on the camera focal plane to writing out the results of star and streak detection. The algorithm can be controlled to produce a small number of stars for positional reference and a limited number of RSO streaks. A data compression of 10^5 - 10^6 is achieved by the SP.

The entire operation of the SBV is internally controlled by an Experiment Controller. Timed commands are stored in the EC and sent to the various components. Another major function of the EC is to store the results from the Signal Processor in its memory until such time as a downlink event can be initiated.

The SBV has been designed for space-based surveillance of RSOs. The large field-of-view enables rapid search. The off-axis design enables low and high altitude RSOs to be detected and tracked near the earth limb, near the moon and within 25 deg of the sun without saturation of the focal planes. The Signal Processor design optimizes the detection of RSO streaks against a stationary background. The data compression, and the collection of positional data on stars and streaks, permits positional accuracies of the order of a third of a pixel (4 arcsec) which is adequate to support the current requirements of space surveillance. Use of internal memory to store the results and downlinking of the data on demand to a ground station enables the SBV to avoid using the on-board power-hungry tape recorder for storage of data. Further, as in many low altitude experimental satellites, real-time communication is not available and the on-board storage of processed results enables the effective use of limited downlink opportunities.

1.4. MSX Spacecraft

A working knowledge of the MSX spacecraft (Ref. 5) and its capabilities and limitations is necessary to understand the design of and the design choices made in the SPOCC mission planning system. The instruments of concern to the Surveillance PI are the SPIRIT 3 radiometer and the UVISI imagers and spectrometers, apart from the SBV.

The MSX (Figure 1) is a large satellite with all major sensors coaligned rigidly along the X-axis. Thus re-pointing any sensor is equivalent to reorienting the entire spacecraft.

The MSX is severely resource limited (Ref. 6). Power is generated by two solar panels. If all the instruments are on and the MSX is tracking a target, the power demand is greater than what can be generated by the solar panels even at full illumination. The excess demand is serviced from rechargeable Nickel-Hydride batteries. Further, the MSX is in a near-sun-synchronous orbit, and as a result, there are extended shadow periods (up to 20 minutes long in an orbital period of 103 minutes).

The data storage capability of the MSX is limited. Only one tape recorder can be used at a time, and the total data that can be stored is ~36 minutes of data at 25 Mb/s; and 180 minutes of

data at 5 Mb/s. These data can be relayed down to only the APL ground station. It takes 2-3 passes over the APL ground station to read out all the data on a tape recorder of data.

The MSX has severe geometrical constraints (Ref. 6). The most significant of these is levied by SPIRIT 3 sensor which is cryogenically cooled by solid hydrogen. Thermal input into the sensor from the earth and the sun must be minimized to conserve the depletion of the cryogen and prolong the life of the sensor. This necessitates pointing constraints on the +X-axis and the -Y-axis of the spacecraft. The other sensors have pointing restrictions along the +X-axis also.

2.0 SPOCC MISSION PLANNING SYSTEM

The Mission Planning System has the following requirements:

- 1) Command the MSX spacecraft for all surveillance experiments correctly;
- 2) Command the SBV in all its operational modes correctly;
- 3) Command SPIRIT 3 and UVISI in a restricted set of operational modes correctly in support of Surveillance experiments;
- 4) Monitor constraints and resource usage;
- 5) Provide a high level language interface to the experimenter;
- 6) Ensure that modes of operation that are incompatible with the health, safety or operational philosophy of the instruments or the spacecraft are precluded; and
- 7) Provide a pipelined operational capability in support of rapid and automated generation of commanding for experiments.

The components of the Mission Planning System are shown in Figure 3.

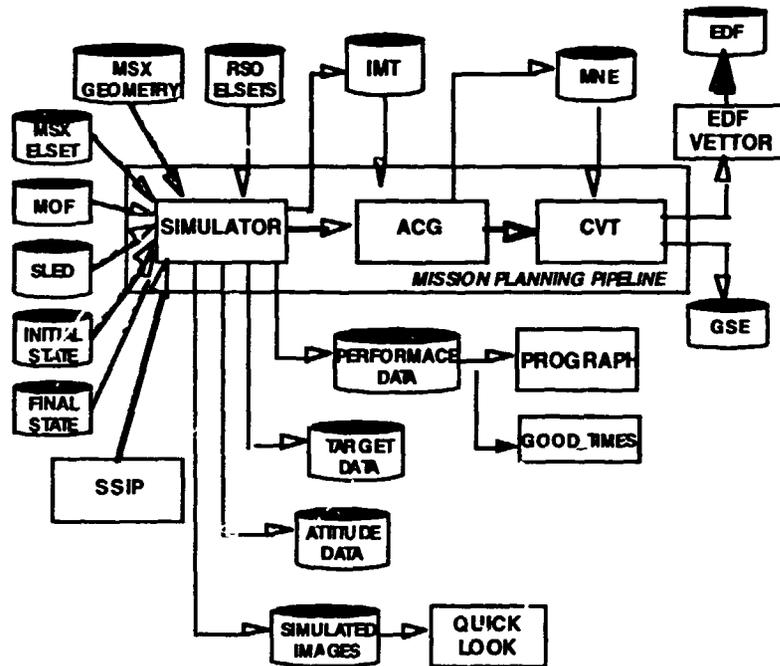


Fig. 3 : Mission Planning System

3.0. THE SIMULATOR

The Simulator is the heart of the mission planning pipeline. It simulates the functioning of the SBV and the MSX and produces the data necessary to both command the spacecraft and to analyze the experiments.

3.1. Architecture

The Simulator is driven by SLED files, either manually created or automatically generated by a component called SSIP(see below). The Parser interprets the SLED code and produces a time ordered, parameterized queue of events to be simulated along with a set of associated data tables. The Simulator takes each SLED generated event and decomposes it into a series of simulation events. Each simulation event corresponds to a state change in the simulation, a change in the attitude control system or a new set of spacecraft or sensor commands. These events are in turn used to drive a standard discrete time simulation. Several graphical, textual and data base/file outputs are produced for analysts to examine and also for further processing by the rest of the Mission Planning System.

The primary sensor for the Simulator is the SBV. Hence, there is a detailed model of all the permitted operating modes and timelines of that sensor. The distributed nature of commanding of the MSX has necessitated an agreement with APL that all surveillance experiments will command the SBV regardless of any other sensor used. Thus, the timeline of the commanding for any experiment is primarily driven by the SBV. The SPIRIT 3 and UVISI sensors, when invoked, are used in a restricted set of modes tailored to fit within the constraints of the SBV timing.

3.2. SLED and the Parser

SLED is a high level language used to define an experiment for the mission planning pipeline (Ref. 7). The major concepts in the SLED language are:

1. It is a high level language which permits a description of the experiment.
2. It frees the user from the details of the commanding of the sensors.
3. It frees the user from worrying about detailed timing of the sensor commanding or the experiment operation.

SLED allows a user to describe an entire experiment and simulation in a compact format. The logical structure of the language is shown in Figure 4. The SLED parser, which is the front end of the simulator, takes a SLED input file and produces a set of data structures used to drive the simulation. More importantly, the parser has extensive error checking functions which prevent inappropriate sensor and infeasible spacecraft events from being generated.

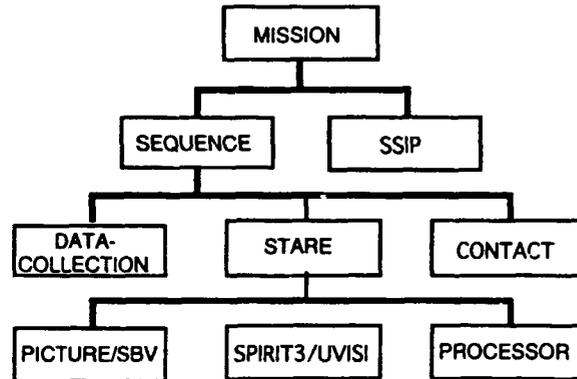


FIG. 4 : Logical Structure of SLED

3.3. Models

3.3.1. Orbital Mechanics

ORBLIB, a set of routines developed at Lincoln Laboratory, is used to determine the position of the MSX, resident space objects, the moon and the sun. The simulator is also able to accept ephemeris files for the MSX produced by JHU/APL.

3.3.2. Attitude Control System

The attitude control system is modeled using software developed at APL. It is essentially the same as the system on the spacecraft with mechanical inputs and outputs modeled in software. It takes as input a set of files corresponding to spacecraft commands and uploadable parameters

and produces an attitude history. Optionally, the operator can select a very simple model, which ignores spacecraft dynamics, for quick look and opportunity analysis.

3.3.3. Power/Thermal Systems

There is a detailed model of the power system which was also developed at APL. It includes modeling of the solar panels, batteries and power electronics. Again, there is a simpler model available for quicklook analyses.

APL has also developed a detailed nodal analysis of the spacecraft's critical temperatures. It models the effects of solar radiation and internal power consumption. In particular, it calculates the temperature of the battery and solar cells which are used as input to the power model. At present it is not implemented and much simpler assumptions are being used (i. e. constant battery temperature). Both the power and the thermal models ignore transients.

GRC, under contract to APL, has developed a model for the thermal behavior of the SPIRIT 3 sensor cryogen. The model takes as inputs the relative position of the sun and earth, the operating mode of the SPIRIT 3 and the temperatures of the baffle, shell and sunshade. It tracks aperture heat load, baffle temperature and cryogen usage.

3.3.4 Contact Scheduling

The MSX is a low altitude satellite that must download data stored onboard during short contacts with fixed ground stations (on the order of 10 minutes). While the downloading of tape recorded data is scheduled by APL, the downloading data stored in SBV RAM is scheduled by the Simulator. During a typical surveillance experiment, or data collection event (DCE), the onboard SBV RAM may be filled and downloaded several times, requiring several contacts.

The mission planning process is an iterative one. Well in advance of running a DCE on the MSX, the Simulator is run to determine what opportunities exist for a particular experiment. APL selects the ones that fit in with the other DCEs being scheduled for other PI teams and sends SPOCC schedule files that reflect when DCEs will run. For each scheduled DCE, the simulator is run to determine how much data is collected in the SBV RAM during the course of the experiment and thereby how many contacts are required. A request for contacts is then made through APL. APL responds with contact scheduling information. The list of contacts, combined with the DCE schedule, is used by the simulator to plan the final DCEs which is run on the spacecraft. The Simulator contains logic to pause data collection during contacts, and maneuver the spacecraft as necessary for contacts over the APL ground station which require a specific attitude.

3.4. SSIP

An operational space surveillance sensor must be able to respond to tasking from the controlling agency by automatically scheduling the tasks in a sensible, prioritized order taking into account visibility, detectability, sensitivity, dynamics, etc. The Space Surveillance Interface Processor provides an automated capability to generate such a schedule - an ordered list of searches for or tracks of RSOs - internally to the Simulator. No on-board capability is being implemented. Instead a ground-based Interface Processor will demonstrate the operational capability.

At present, there are two schedulers, one for geosynchronous searches and one for tasking experiments. The structure of the software allows for the addition of more scheduling algorithms.

SSIP takes a tasking file as input and produces SLED files which are in turn used by the simulator. The tasking file allows the user to specify a complicated scheduling scenario in a very compact format.

There are two concepts of interest in SSIP, pseudo-objects and the figure of merit (FOM). Pseudo-objects are used to produce search spaces. For instance, to search along the geosynchronous belt a set of pseudo-objects would be generated. Each object would have a mean anomaly approximately one field of view apart. As SSIP generates a search for each object, the search space is covered. The FOM is a computed scalar used to determine which object should be tracked next. It is calculated by multiplying a series of weighting factors. These factors pertain to the geometry and dynamics of the orbits of the RSOs, the reflectivity-area product of the RSOs and the characteristics of the background against which the RSOs are detected.

3.5. Outputs

3.5.1. Instantiated Mission Timeline

The instantiated mission timeline or IMT file is a time ordered, time-tagged list of the events that occurred during the simulation. The IMT file is passed on to the ACG/CVT where it is translated into spacecraft commands. It is an ASCII text file which can also be examined by the operator to see the results of a simulation.

3.5.2. The PLOT and Attitude Files

A data file containing the details of the simulation is also produced. The data includes constraint angles, power and thermal data, target information and ground station information. This in turn can be used by the PROGRAPH and Good_times processors of the Opportunity/Feasibility Analysis System (Ref. 8).

The attitude data file contains a detailed listing of the position and attitude of the spacecraft, the status of the SBV (i. e. CCD number, gain etc.) and target data. It is also a operator readable ASCII text file. In addition the attitude file can be used to produce simulated SBV imagery.

3.5.3. Cost Reports

The simulator also produces a compact listing of resource usage data of most interest to an analyst. These data includes power/thermal values, avoidance angles and timing information. These costs are validated by comparing with the more detailed models used by APL.

4.0 COMMAND GENERATION

The Simulator, as described above, generates an Integrate Mission Timeline (IMT) file. The IMT file contains a sequence of spacecraft and sensor commanding events. The Automatic Command Generator (ACG) and Command Vector & Translator (CVT) complete the mission planning process by converting the high level event description in the IMT file to the SBV and MSX commands that will accomplish the specified events.

4.1 Automatic Command Generator (ACG)

The ACG expands each event in the IMT into a sequence of mnemonic commands. The ACG parses the IMT file, building an event queue from the sequence of spacecraft and sensor commanding events. Each event is processed sequentially, and is converted into one or more mnemonic commands. Each mnemonic represents either a 32 bit SBV serial command, or an APL command packet for an MSX subsystem or another sensor such as the SPIRIT III. The mnemonic commands are written out to an MNE file, short for mnemonics. The mnemonic commands are an intermediate level of commanding designed to be easier to read than 32 bit hex commands and is used primarily for debugging purposes. The mnemonic commands can also be used for writing SBV contingency scripts that do not require simulation of the spacecraft and other sensors.

4.2 Command Vettor & Translator (CVT)

The CVT translates the MNE file mnemonics into the commands that will be transmitted to APL for commanding the MSX and its sensors. The CVT vets and translates each SBV mnemonic into its corresponding 32 bit serial command value. The CVT translates each APL command packet mnemonic into the sequence of APL command domain identifiers corresponding to the command packet for that spacecraft subsystem or sensor. The 32 bit commands and domain identifiers are output to the Event Definition Format (EDF) file which is then transmitted to APL. The Mission Operations Center at APL processes the EDF file, converts the domain identifiers into 32 bit serial commands and builds a command upload for the MSX spacecraft.

The CVT also generates a second output file named the REQ file. The REQ file is used for testing the experiment's SBV commands on the SBV brassboard. The REQ file contains the same 32 bit SBV commands as the EDF file, but the MSX command domain identifiers are replaced by commands to the brassboard's ground support equipment (GSE). As the SBV commands execute on the brassboard, the GSE simulates the MSX spacecraft subsystems that interface with the SBV.

5.0. DATABASES

During a Simulator run, several types of data are retrieved from various databases. A Master Object File (MOF) database provides information on resident space objects. A second database provides schedule information regarding which data collection events are to run when. Information regarding the contact schedule for data downloads is also present in a third database. Both of these databases are used for planning DCEs to take advantage of the ground station contacts available for downloading collected data.

6.0 PIPELINE AND AUTOMATION

The Simulator, ACG, and CVT constitute the core of the SPOCC Mission Planning System pipeline that is run end to end for each data collection event to be run on the MSX. The SPOCC Mission Planning System incorporates several types of automation to minimize human operator workload during mission planning (Figure 5).

The event schedule and contact schedule information arrives from APL as various files which are automatically processed upon receipt, archived, and the appropriate data entered into the databases. Several other files needed for mission planning, such as orbital geometry files, also arrive from APL and are automatically processed and archived.

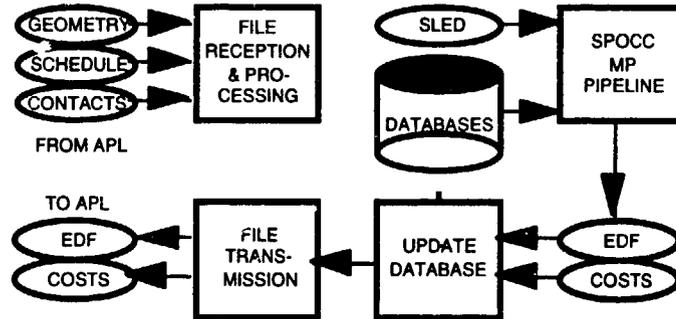


Fig. 5: Automation of SPOCC Mission Planning Pipeline

The pieces of the pipeline can be run either individually, or the whole pipeline can be run with one call to a script. A script is also available to assist the mission planners in properly naming SLED files, as well as one which automatically archives the pipeline output files, transmits the EDF and cost report files to APL, and updates a database as to what files were sent.

7.0 TESTING

Several levels of testing are used to ensure to correct operation of the pipeline during its development. As new features and capabilities are added, developmental unit testing on the individual pieces of the pipeline are carried out. With each major release of the whole pipeline, a series of standard regression tests are run to verify the new release.

As each new type of data collection event is developed, its REQ file is first run through the SBV brassboard to verify that the SBV portion of the commanding is correct. The EDF is then sent to APL for feasibility testing with their MSX spacecraft simulation. Several types of surveillance data collection events have also been run on the MSX spacecraft hardware itself during ground testing as part of the MSX integration and testing effort.

8.0 SUMMARY

A capable mission planning system has been designed and built for space surveillance experiments with the MSX satellite. While primarily designed for experiments with the SBV sensor built by MIT/LL, the system has been expanded to accommodate other sensors on board and also the commanding of the MSX itself. The entire system is designed to be driven by an experiment description in a high level language and a set of data bases. The system can be operated in a pipelined fashion. Comprehensive unit, subsystem and system testing is accomplished with specially designed regression tests. This is followed by validation through a brassboard of the SBV and the spacecraft simulator. The system will be operational at launch of the spacecraft, expected at the end of 1994.

9.0 REFERENCES

1. The SPOCC team : "SPOCC Design Document, revision 6.0", Doc. no. 91PM-SPC-0024, MIT Lincoln laboratory, 28 Feb. 1992.
2. D.C.Harrison : "Space-Based visible (SBV) Program : System definition document", Project Report SBV-10, MIT/Lincoln Laboratory, 6 Feb. 1991.
3. P. Chu : "Efficient Detection of Small Moving Objects", Technical report TR-846, MIT/Lincoln Laboratory, 21 July 1989.
4. T.P.Opar : "Space-Based Visible Band Sensor Detection performance Analysis", Project Report SBS-9, MIT Lincoln Laboratory, 16 Aug. 1988.
5. Mission Research Corporation : "MidCourse Space Experiment (MSX) Preliminary Data Management Plan, DoD SDIO Report, 15 July 1991.
6. Anon : "Mission Operations, Constraints and Requirements Handbook (MOCARH)", MSX report, MSX Mission Operations Center, Johns Hopkins University / Applied Physics Laboratory, 1994.
7. SPOCC Mission Planning Team : "The Simulator Reference manual, Version 3.0", MIT Lincoln Laboratory, 1994.
8. R.Sridhara *et al* : "Opportunity Analysis for Space - Based Surveillance Experiments" MIT Lincoln Laboratory Technical Report No. 1011, 1994.

Engineering *Ulysses* Extended Mission

Shaun Standley*
European Space Operations Centre,
Darmstadt, Germany.**

Abstract

The *Ulysses* Mission is a collaboration between the European Space Agency (ESA) and the National Aeronautics and Space Administration (NASA). The mission is unique, enabling exploration of the heliosphere within a few astronomical units of the Sun over a full range of heliographic latitudes - adding a third dimension to our understanding of the Solar System.

The advanced scientific instrumentation on *Ulysses* continually measures the properties of the heliospheric magnetic field, the solar wind, solar radio bursts and plasma waves, galactic cosmic rays, energetic particles, solar X-rays, and interstellar neutral gas. By the end of 1995, the spacecraft will have completed measurements at heliographic latitudes up to 80 degrees over a single orbit of the Sun. The properties of the heliosphere are solar cycle dependent, and *Ulysses* first orbit of the Sun will have taken place around a solar minimum. In order to characterize the heliosphere over a full (11 year) solar cycle, it is desirable to continue measurements over a second orbit of the Sun, a new *Odyssey* that will extend through 2001. Since the spacecraft was only designed for a five-year mission, a number of technical challenges have been surmounted in order to demonstrate the engineering feasibility of this unparalleled scientific opportunity.

This paper describes the changes that were necessary to the *Ulysses* mission engineering and mission operations in order to ensure continual, effective payload operation throughout 1996-2001.

The Mission

Ulysses was launched in October 1990 on the space shuttle *Discovery*. Following deployment, the spacecraft was accelerated by an IUS and PAM-S

into an in-ecliptic transfer trajectory to Jupiter. A gravity assist flyby was necessary at Jupiter in order to produce *Ulysses*' inclined, heliospheric trajectory as depicted in figure 1.

The primary objective of the *Ulysses* mission is to characterize the heliosphere over the full range of solar latitudes. The spacecraft carries instrumentation¹ to perform measurements of the interplanetary magnetic field, solar wind plasma, radio and plasma waves, energetic particles, cosmic ray isotopes, interstellar neutral gas, and interplanetary dust. Full descriptions of the instruments and scientific objectives of the mission² are not included here, but are thoroughly treated in the above referenced publications.

The spacecraft³ is spin-stabilized, with a High Gain Antenna (HGA) mounted with its boresight along the spin axis. Because the spacecraft is not subject to large perturbing forces, it maintains a stable inertial attitude for long periods of time. Attitude maneuvers are necessary to compensate for apparent Earth drift, keeping the HGA pointed within about 1° of the Earth. The precise magnitude of the allowable offpointing depends on the link budget for a given mission phase. Attitude control is provided by catalytic decomposition thrusters fueled by hydrazine.

Ulysses has both X and S-band transmission capability, but X-band only is used to maintain the downlink via the HGA. The spacecraft also has front and rear S-band quadrofilar helix type low gain antennas (LGAs); S-band transmission was provided for use during the launch and early mission, and is only used now during limited periods of Radio Science investigation.

The spacecraft is powered by a Radioisotope Thermoelectric Generator (RTG)^{4,5}, providing about

* Systems Engineer, *Ulysses* Flight Control Team.

** European Space Operations Centre personnel are situated at the Jet Propulsion Laboratory, Pasadena, California, as part of the joint ESA/NASA Mission Operations Team.

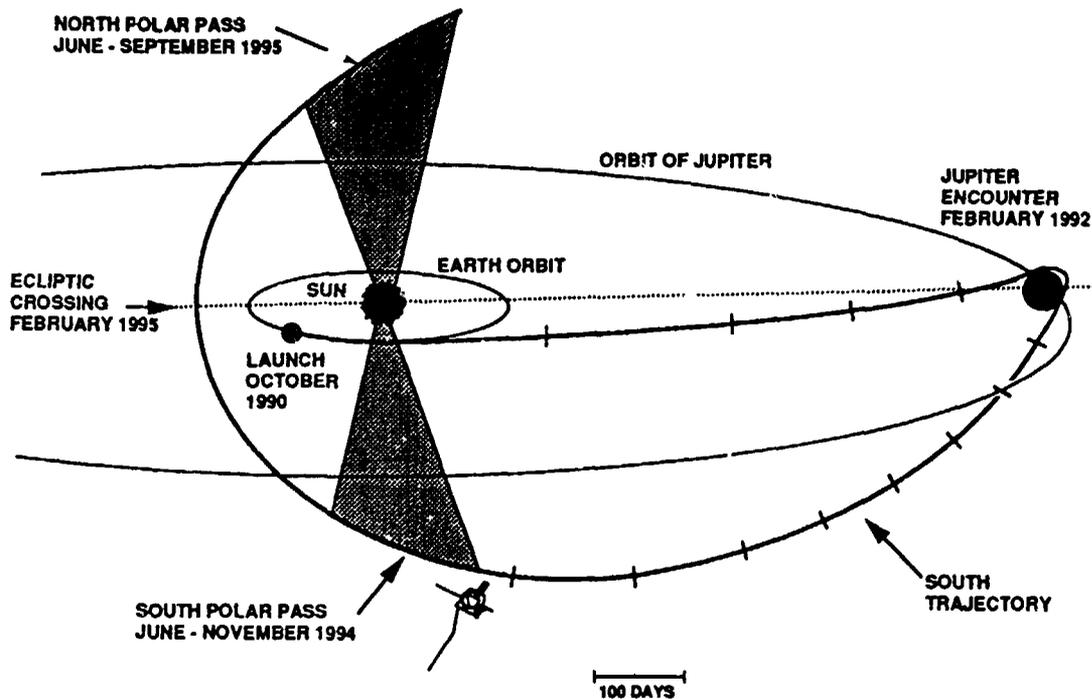
287W at beginning of mission. Solar heating varies from 1500W/m^2 at beginning of mission to about 45W/m^2 at apnelion; such large variations in environmental conditions make thermal control and power management of particular importance. Power not consumed by operational units and heaters is dumped to resistances inside the spacecraft to contribute to general heating. This power can be diverted to resistances outside the spacecraft when the interior becomes too warm. Dissipation is shifted between dedicated heaters, internal power dumpers (IPDs) and external power dumpers (EPDs) to achieve the optimal thermal state for the spacecraft.

reconfiguration from anomalous modes, and carry out on-board data processing.

The spacecraft is operated by a joint ESA/NASA team situated at Jet Propulsion Laboratory in Pasadena, California.

Ulysses at Solar Maximum

The basic properties of the heliosphere are solar cycle dependent. *Ulysses* southern polar pass and upcoming northern polar pass will take place at solar minimum, the equivalent passes in 2000 and 2001



The spacecraft Data Handling Subsystem (DHS) performs the usual functions of acquiring, decoding and accepting incoming commands and distributing these commands to the instruments and platform subsystems. Only 40 commands can be stored as a 'sequence' on-board, so a significant amount of commanding in near-real-time. All telemetry acquisition and processing is performed by the DHS, with data storage on two redundant 45Mbit tape recorders.

The DHS incorporates a software package tailored to *Ulysses*, with applications that monitor spacecraft health and safety, initiate recovery and

will take place at solar maximum. The extended mission will enable characterization of the heliosphere not only over all latitudes, but over the full 11 year solar cycle, greatly enhancing the scientific return of the mission as a whole.

Beyond 1995 *Ulysses* will also form an important complement to ESA's Solar Heliospheric Observatory (SOHO), and NASA's WIND mission. These spacecraft, from positions close to the Earth, will study the Sun, monitor the solar wind, the heliospheric magnetic field, and energetic particles.

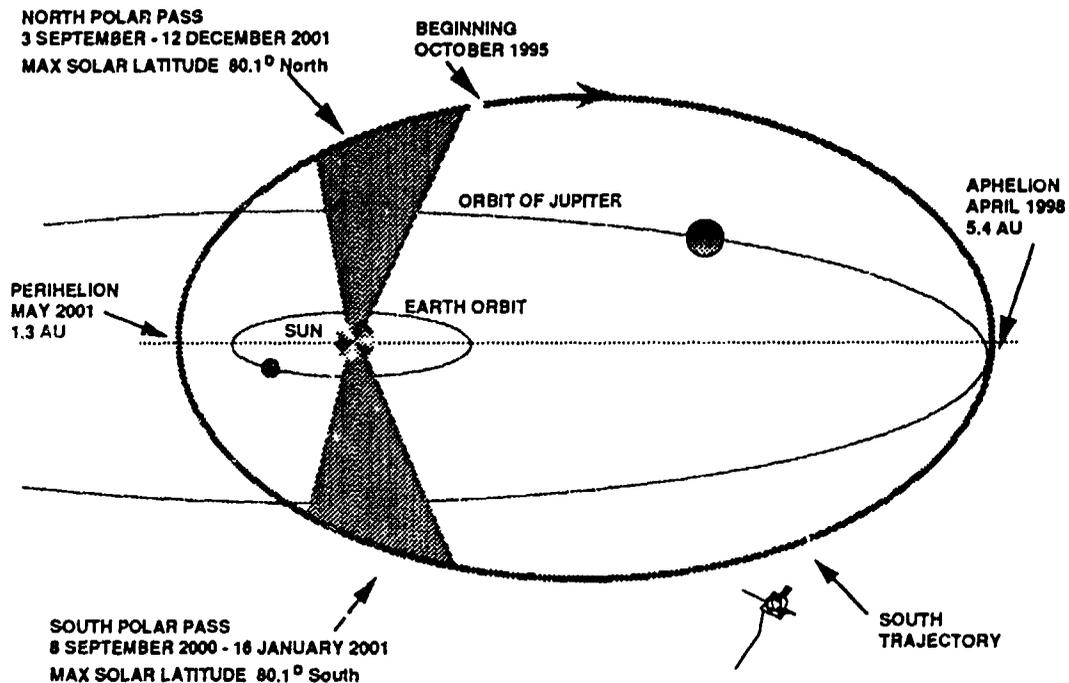
Current Status

Both scientific^{6,7,8} and engineering^{9,10} reports on the status of the *Ulysses* mission have been given periodically in the past. The first four years of the mission have not been trouble free, but there has been no malfunction or degradation that seriously threatens the viability of a second solar orbit. Mission operations have proceeded well, as evidenced by the near continuous flow of science data since launch, and with a few exceptions the spacecraft has performed nominally.

In November 1990, shortly after launch, the

spacecraft body are subjected to the same motion of their fields of view.

The reason for the nutation has been established¹¹ as thermal bending of the axial boom causing torque reactions on the central body, complicated by a severe underperformance of the spacecraft's passive nutation dampers. Fortunately, thermally-induced nutation can only occur under certain conditions of solar distance and Solar Aspect angle, which are not met for long phases of the mission. More importantly, the spacecraft's on-board conical scanning control mode (CONSCAN) has been used successfully to control nutation instability on several



spacecraft started to nutate immediately after the deployment of the spacecraft's axial boom. Nutation causes the spacecraft spin axis to describe a rosette-like pattern, rather than staying fixed in one direction. Over a period of weeks, this nutation built up to 6.5°, and eventually disappeared on 18 December 1990. Nutation represented a danger to the spacecraft as the flexing motion caused at the root of the axial boom could cause the boom to collapse and damage the spacecraft. The motion also causes the High Gain Antenna (HGA) to depoint, eventually resulting in loss of telemetry as the link margin decreases. Instrument pointing is also effected, as instruments mounted rigidly to the

occasions, and although the threat of nutation must be taken seriously, it no longer compromises the viability of the mission. In March 1992, following *Ulysses* flyby of Jupiter, the spacecraft's redundant systems were checked out. The second Central Terminal Unit (CTU2), a major component of the on-board computer, was found to be malfunctioning. Two bits in a register used for telemetry formatting became linked by a short-circuit, resulting in widespread corruption of data in the telemetry format. Fortunately, CTU1 is still in perfect condition so telemetry formatting during the mission has been and will continue to be uncorrupted. CTU2 will only be used in the event of a malfunction of the

prime unit, and is still perfectly viable as an emergency backup. The corruption produced by the bit linkage is predictable, so the anomaly is by no means catastrophic in terms of data recovery, even if CTU2 had to be used for long periods. Fifty percent of data words are corrupted, and result mainly in increased ambiguity in the science data.

Ulysses' principal emergency mode is termed Disconnection of Non-Essential Loads (DNEL), and consists of the entire payload being switched off followed by a general reconfiguration of the spacecraft platform to redundant systems. At the time of writing, five DNELs have occurred during the mission. These have been attributed to short duration current surges in the Main Switch (which connects the science instruments to the main bus) that occur at the same time as a Reaction Control Subsystem latching valve transition during routine maneuvers. Recovery from DNEL takes 12-48 hours, and the occurrence of DNEL at the current rate is not considered a threat to science continuity. The spacecraft latching valves isolate the propellant tank in the middle of the spacecraft from the thruster clusters. These valves are routinely closed when a maneuver is not taking place, and are closed automatically by on-board logic if significant spin rate or attitude perturbations are detected. In early 1994, new information on the valves' manufacturing history indicated an increased likelihood of failure under certain operating conditions. Maneuver operations were changed so that the number of times the latching valves were cycled was minimized.

Concerns

Despite the demonstrably excellent health of *Ulysses* science instruments and engineering subsystems, continuation of the mission is still dependent on adequate consumables to sustain the spacecraft through another six year orbit. Consumables of concern are power, as supplied by the RTG, and attitude control hydrazine.

Because of launch delays, *Ulysses* beginning-of-mission power was about 287W, and is expected to meet its end-of-northern-polar-pass requirement of 245W. After this, in order to ensure all instruments can be operated throughout the second solar orbit, several modifications will be necessary to the spacecraft operational configuration:

- Hot-redundant units such as the redundant receiver will be powered down when necessary.
- Operations causing power peaks in daily activities, such as attitude manoeuvres, tape recorder operations, commanding, and some instrument reconfigurations will be separated.
- The most power-efficient unit of a redundant pair will always be used.
- At later stages of the mission, thermal safety margins established by the original mission design will be reduced in the light of operational experience of *Ulysses* thermal behavior.

By modifying routine operations as above, it is possible from a power point of view to operate all the science instruments through the second northern polar pass in December 2001.

Apart from power, hydrazine fuel mass remaining is also a potential concern. Fuel is necessary for routine Earth pointing maneuvers, to keep the HGA correctly aligned for telemetry transmission. Fuel may also be required for nutation damping maneuvers, should nutation reoccur in late 1999.

Fuel consumption due to routine attitude maneuvers can be predicted with confidence¹² based on historical performance and knowledge of future mission geometry. Because of the excellent orbit injection accuracy in the early mission, large amounts of the fuel budgeted for trajectory correction maneuvers has not been used. The fuel remaining is ample for routine attitude control and nutation damping, should this become necessary.

Conclusion

The continuation of *Ulysses* observations over a full solar activity cycle are an unparalleled scientific opportunity. The excellent health of the spacecraft instruments and engineering subsystems, coupled with stringent management of consumables, makes a second solar orbit achievable.

Acknowledgments

This paper includes relevant information in the *Ulysses* project contained in project presentations and review material from mission continuation studies. The Author would like to thank colleagues in the *Ulysses* Flight Control Team who are an integral part of mission continuation planning, and for the encouragement and direction received from the Project Science personnel in both Project Offices.

¹ *Ulysses Instruments. Astronomy and Astrophysics - Supplement Series*, 92, 1-440.

² *Ulysses: A Journey above the Sun's Poles. Trans. AGU*, 72, 241-248.

³ Hawkyard, A. & Buia, P. *The Ulysses Spacecraft. ESA Bulletin 63*, August 1990.

⁴ Bennett, G.L. and Campbell, R.W. *Performance of the RTGs on the Galileo and Ulysses Missions. Proceedings of the European Space Power Conference, Graz, Austria August 1993.*

⁵ Mastal, E.F., Campbell, R.W., *RTGs - The Powering of Ulysses. ESA Bulletin 63*, August 1990.

⁶ *The Ulysses Mission: In-Ecliptic Phase. Geophysical Research Letters*, 19, 1235-1314.

⁷ *Ulysses at Jupiter. Science*, 257, 1449-1596.

⁸ *Ulysses Encounter with Jupiter. J. Geophys. Res.*, 98, 21, 111-21, 252.

⁹ García-Pérez, R. *Ulysses Log: 1992*, Proceedings of the Second International Symposium on Spacecraft Operations, Pasadena, California, November 1992.

¹⁰ Angold et al. *Ulysses Operations at Jupiter. ESA Bulletin 63*, August 1990.

¹¹ Crellin, E.B. *Thermo-Elastic Dynamic Instability for Ulysses. European Space Technology Centre, Working Paper 1637*, October 1991, Noordwijk, The Netherlands.

¹² *Extension of the Ulysses Mission. ESA Science Programme Committee, ESA/SPC(93)25. Paris, 12 May 1993.*

ACCURACY ANALYSIS OF TDRSS DEMAND FORECASTS

Daniel C. Stern
Office of Space Communications
National Aeronautics and Space
Administration

Allen J. Levine
Networks Division
NASA Goddard Space
Flight Center

Karl J. Pitt
Space Systems Division
The MITRE Corporation

Abstract

This paper reviews Space Network (SN) demand forecasting experience over the past 16 years and describes methods used in the forecasts. The paper focuses on the Single Access (SA) service, the most sought-after resource in the Space Network. Of the ten years of actual demand data available, only the last five years (1989 to 1993) were considered predictive due to the extensive impact of the Challenger accident of 1986.

NASA's Space Network provides tracking and communications services to user spacecraft such as the Shuttle and the Hubble Space Telescope. Forecasting the customer requirements is essential to planning network resources and to establishing service commitments to future customers. The lead time to procure Tracking and Data Relay Satellites (TDRSs) requires demand forecasts ten years in the future – a planning horizon beyond the funding commitments for missions to be supported.

The long range forecasts are shown to have had a bias toward underestimation in the 1991 - 1992 period. The trend of underestimation can be expected to be replaced by overestimation for a number of years starting with 1998. At that time demand from new missions slated for launch will be larger than the demand from ongoing missions, making the potential for delay the dominant factor. If the new missions appear as scheduled, the forecasts are likely to be moderately underestimated.

The SN commitment to meet the negotiated customer's requirements calls for conservatism in the forecasting. Modification of the forecasting procedure to account for a delay bias is, therefore, not advised. Fine tuning the mission model to more accurately reflect the current actual demand is recommended as it may marginally improve the first year forecasting.

BACKGROUND

NASA's Space Network (SN) provides tracking and communications services to user spacecraft such as the Shuttle and the Hubble Space Telescope. The Space Network space segment consists of operational geostationary Tracking and Data Relay Satellites (TDRSs) at longitudes of 41W, 174W, partially operational satellites at 171W and 275W, and a spare at 46W. TDRSs are controlled from the White Sands Ground Terminal (*Cacique*) and the Second TDRSS Ground Terminal (*Danzante*) in New Mexico. Each TDRS communicates with user spacecraft by using one of two Single Access (SA) antennas or by using a multiple-access phased array antenna.

The SN began to support customers in late 1983 with the launch of TDRS-1. Implementation of the complete system of three relay spacecraft was delayed by loss of Challenger along with its TDRS-2 payload in January 1986. The accident also brought about a re-evaluation and a slowdown in the pace and number of Shuttle-launched missions, many of which were slated for SN support.

Shuttle operations resumed in September 1988 with the successful launch of TDRS-3. Six months later, TDRS-4 was launched. The completion of checkout of the third operational TDRS in June 1989 marked the initiation of a fully operational SN. Mission load grew from early support of Shuttle, Landsat 4 and 5, ERBS, SME, and SMM to include COBE, HST, UARS, EUVE, TOPEX, and classified missions. Replenishment of aging relay spacecraft and the addition of spare capacity was accomplished with launches of TDRS-5 and -6 in 1992 and 1993.

MISSION MODEL HISTORY AND PURPOSE

The generation of a "mission model" for the prediction of the users and their communications requirements has been a key activity since early in the formation of the tracking networks. Major studies for the Spaceflight Tracking and Data Network (STDN) -- as the ground-based network was known -- were conducted yearly, with additional updates in between as demanded. When plans for the creation of the TDRSS (or the Space Network, as it later became to be known) began to emerge, the studies began to include prospective TDRSS users. Starting in 1978, the first study devoted to TDRSS was produced.

The primary purpose of the Network Support Capability Studies (also referred to as loading studies or forecasts) was to ensure that the projected TDRSS would have adequate resources to handle the upcoming customers so that a commitment to potential new customers could be made. This purpose is still valid, but in more recent years the activity has grown in importance as support for the procurement of replenishment TDRS and, consequently, has been the subject of close scrutiny within and outside the agency.

Unfortunately, mission modeling is not an exact science. Political and economic environment, unforeseen technical problems, and technology developments tend to determine actual events and diminish the validity of the forecast. On the other hand, some stability is lent to the model by the tendency for operational missions to be extended beyond their original planned life as expected new missions fail to happen.

MISSION MODEL AND DEMAND MODEL DEVELOPMENT

The first step in developing the mission model is to survey the mission planning offices at the NASA centers and NASA Headquarters for Space Network user requirements documents, written or in process. Any missions not yet approved require confirmation of the appropriate program office at NASA Headquarters. Additional offices at the centers or Headquarters are surveyed for information on non-NASA programs, such as those of other government agencies or commercial or foreign entities, that are planning on Space Network support.

After all the missions using the Space Network have been identified, an examination of the overall telecommunications requirements is performed, and the missions are prioritized to facilitate schedule conflict arbitration. Although requirements documentation states the needs of the respective missions, a meeting or conversation with mission project representatives generally provides additional useful information, such as operational constraints, relationships with other missions, further explanations of the mission goals, and relative importance of the specific support requirements. This information, along with the experience of the analyst, is sometimes used to extend the mission duration from that formally stated in the documented requirements. The list of prioritized missions and requirements thus developed constitutes the *mission model*.

The mission model is then further developed into a *demand model*. This is the aggregate of all the mission requirements on the Space Network. Set up activity, such as Single Access (SA) antenna repositioning (slew) time, is also included. This aggregate is then compared to the availability of the Space Network resources by using the Network Planning and Analysis System (NPAS). The results are provided as the percentage of the customer's requirements that can be met.

For the purposes of this analysis a simplified version of the demand model is used and referred to as *demand forecast* (or simply *forecast*). In the demand forecast, detailed mission requirements,

such as orbit and number of contacts per day, are reduced to the average total SA hours per day required in each year or quarter-year of the forecast period. The total SA hours include the effect of two minutes of slow time per communications contact.

The *actual demand* data are taken from monthly network support reports. A valid comparison with the demand projections requires that the Shuttle be in flight. Because the monthly report data include intermittent Shuttle flights, the actual Shuttle data were subtracted from the monthly totals and the effect of a Shuttle in flight was adjusted by adding the assumed full-period shuttle support. This permits the use of all the monthly data points. Actual demand data also include the effect of slow time.

MISSION MODEL CHRONOLOGY

The earliest mission model data for the Space Network (SN) are found in a Network Support Capability Study of July 1976. Five missions were listed to have TDRSS support as soon as it became operational in early 1981. Because no specific TDRSS service requirements were provided, a demand forecast is not available for analysis.

Beginning in December 1978, more detailed studies were conducted at least yearly. Eight studies spanning the period from 1978 to 1993 were analyzed for this paper. A summary of the model characteristics is provided in Table 1, followed by further description of their contents.

Table 1. Characteristics of Mission Models Used in Study

Model Name	Pub Date	Years	No.*	STS-V, STS-K**	Other Significant SA Missions (4 to 24 hr/day)	No. Approved or in orbit
1978	12/78	81-90	22	Yes	HST, LANDSAT, ADV. GEOL, STEREO SAT, OER	27%
1980	10/80	84-88	21	Yes	HST, LANDSAT-D, NOS	29%
1982	6/82	84-89	13	Yes	HST, LANDSAT-D	59%
1985	2/85	85-91	11	Yes	HST	64%
1989	5/88	89-97	19	No	SSF, HR SO, HST, EOS, TRMM	47%
1990	10/89	90-97	18	No	SSF, HST, EOS, TRMM, HR SO	79%
1992	3/92	92-99	17	No	SSF, HST, TRMM, EOS, LSAT-7, AXAF	100% (11 in orbit)
1993-1	—	93-99	15	No	SSF, HST, TRMM, EOS, LSAT-7	100% (9 in orbit)

* Only missions with a single-access requirement are included.

** A Vandenberg Air Force Base-launched Shuttle (STS-V) supported simultaneously with a Kennedy Space Center-launched Shuttle (STS-K). An STS-K is included in all studies.

The 1978 and 1980 models are characterized by optimistic projection for a large number of users for the yet-to-be built network. Most missions were in their study phase, and were included in the models because the budget environment appeared to support them. Large requirements for STS-K (Kennedy) and STS-V (Vandenberg) dominate these early models, as well as those produced through 1985. Requirements for simultaneous support of the second Shuttle begin between calendar years 1984 and 1985. The models assume that both Shuttles are required simultaneously but with varying contact time needs. The total contribution of the Shuttles therefore varies from one to two full links.

The 1989 and the later models are characterized by faded optimism and greater scrutiny of Shuttle-launched missions, resulting from the 1986 Challenger accident. A steadily diminishing set of study missions is included. The final two models include no study missions – all the missions are either in orbit or approved.

The 1989 model was produced in May 1988, several months before the resumption of Shuttle flights. The 1989 and later models include only a Kennedy-launched Shuttle; Vandenberg had, by then, been dropped as a Shuttle launch site.

The 1989 and 1990 models also included HRSO. The requirement was for a full link in the first model and was reduced to 10 hours per day in the 1990 model. The mission was dropped out in the later models. HRSO is the most significant consideration in comparing the 1989 model with those of 1990 and 1992.

All models, starting with that of 1989, also include a continuous coverage requirement for Space Station, resulting in a step increase in demand. Slips in Space Station start dates moved the requirement start from late 1995 into 1996. The program further slipped to 1998 start date causing a noticeable change to the 1993-1 model.

FORECASTS VERSUS ACTUAL DEMAND

All eight forecasts as well as the actual demand are plotted for comparison in Figure 1. The plot suggests a division of the studies into two sets: 1978 to 1985 and 1989 to 1993. The first set covers the pre-Challenger accident as well as the pre-operational SN period. The second set covers the period where the SN is near or at full operation.

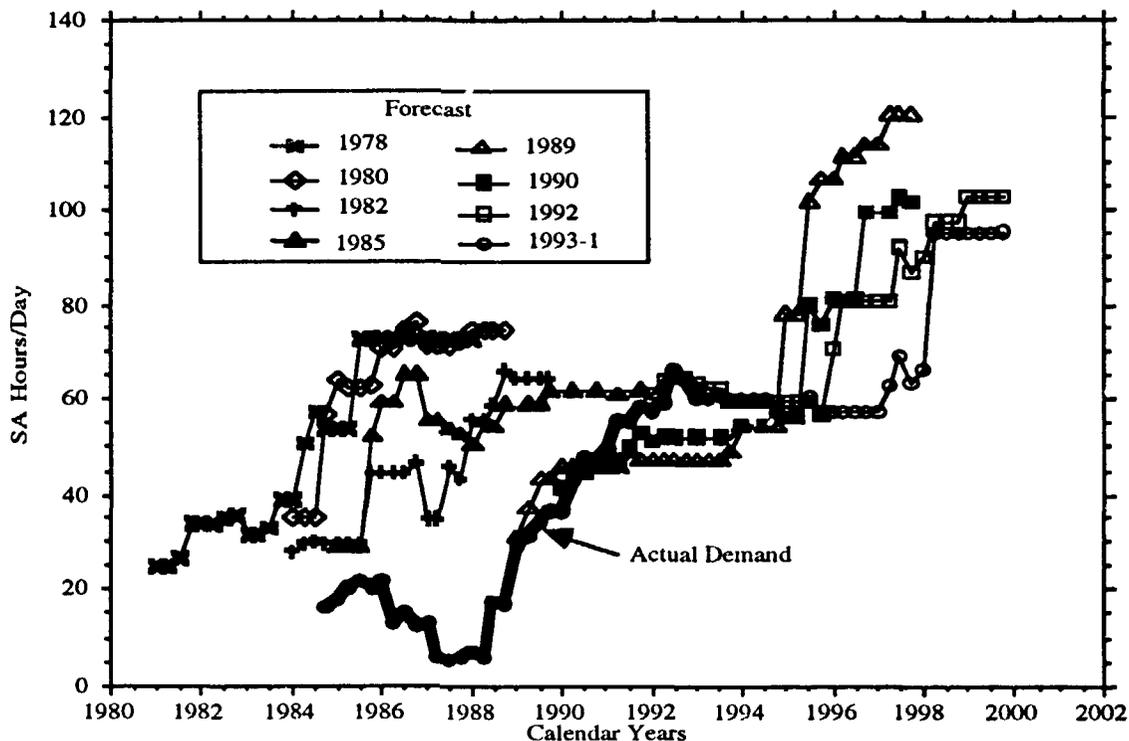


Figure 1. Comparisons of Forecasts and Actual Demand.

As stated previously, the Challenger accident suspended all shuttle launches for 32 months. An attempt was made to account for the Shuttle suspension period by adding the early forecasts out 32 months. The resulting altered plot improved the predictions but substantial differences

remained between forecasts and actual. It appears that the change in launch rate and suspension of some missions curtailed the originally predicted user build-up.

The maturity of the Space Network may have been another factor in the accuracy of the forecasts. The percentage of approved and ongoing missions for each Model (see Table 1) appears to be correlated with the accuracy of the forecasts. Unfortunately, the Challenger accident masks an accurate analysis of this effect for the forecasts made 1978 through 1985.

It is useful to examine the uncertainty in the forecasts beginning with 1989 under an assumption that the forecasts are updated to reflect more accurate information. If the mission model from the forecast of 1989 is compared to the forecast of 1993, a net loss of one mission is expected due to the change in forecast span, yet a loss of 4 missions occurred. Six missions were lost (four never occurred and two were removed due to forecast span) whereas two were gained (one due to the forecast span and the other an unexpected user). In Table 1 the 1993 forecast is shown to be fully approved with a majority ongoing, while only 47% of the missions in the 1989 model were approved. This suggests that at least two or three of the four missions lost can be attributed to the lack of approval.

STATISTICAL ANALYSIS OF 1989-1993 DEMAND PROJECTIONS

Due to the extensive impact of the Challenger accident of 1986, only the post-Challenger forecasts are considered predictive. The statistical analysis is therefore limited to the forecasts beginning with 1989. Visual analysis of the 1989-1993 forecasts plotted in Figure 1 suggest that in the short term the forecasts are relatively accurate, experiencing small errors due to fluctuations in actual demand. Large changes in forecast are due to user program changes in the out-years.

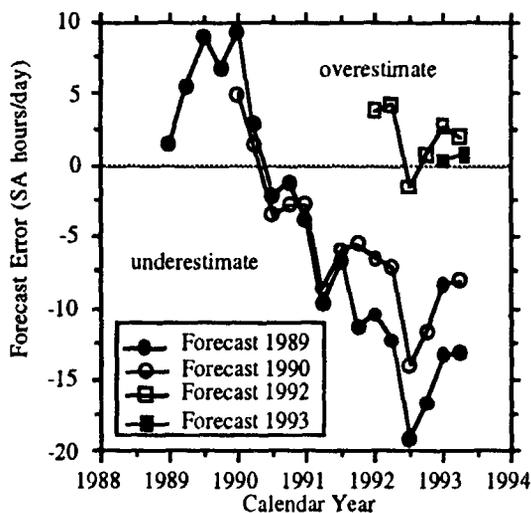


Figure 2. Error in Forecasts Versus Date.

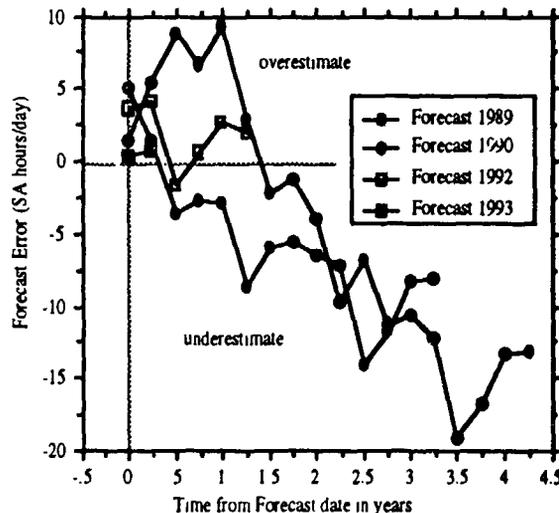


Figure 3. Error in Forecasts Versus Time from Forecast

Short-Range Forecast Errors. Figures 2 and 3 show the difference between forecasted SA hours and the actual demand. Positive differences indicate overestimation. In Figure 2 the errors are plotted as a function of the year for which the forecast was made, whereas in Figure 3 the errors are plotted as a function of time from when the forecast was made. The figures show that there has been a tendency to overestimate in the first year or so into a forecast and increasingly

underestimate for several years after that. Whether this experience indicates a true trend, i.e., can be expected to repeat in future forecasts, is discussed below.

Events, such as a slip of a mission start date, can cause errors in more than one forecast. This is particularly true for two successive forecasts because both forecasts cover substantially the same future period. The similarity in the 1989 and 1990 curves is evidence of this error-data correlation. The effect of the correlation is to reduce the amount of independent data available for statistical inference. It is therefore useful to see if uncorrelated data is available for analysis.

The first year of each forecast is likely to provide such uncorrelated data. Errors occurring within the first year of a given forecast are not likely to be correlated with errors in the first year of a subsequent forecast because the forecasts are spaced at least one year apart. Table 2 and 3 are the error statistics for just the first year of the forecasts.

Table 2. Error in first year after forecast

	Mean	Std. Dev.	Count	Minimum	Maximum
Error (hrs/day)	2.23	3.64	14	-3.54	8.92

Table 3. Hypothesis Testing of First Year Error for Mean = 0

	P-Value	95% Lower	95% Upper
Error (hrs/day)	.039	.13	4.33

Table 2 shows that first year errors ranged from a negative 3.5 years (underestimate) to a positive 8.9 (overestimate) with a mean of 2.2 years and a fairly large standard deviation of 3.6. To further measure the strength of the mean error, a hypothesis was tested to see how likely the sample mean could result from a random variation about a true mean of zero. The result, given in Table 3, shows that the mean equal to zero has a likelihood given by P of about 4%. The hypothesis therefore falls outside the 95% confidence range. This implies that there is reasonable evidence that the first year of a forecast tends to be somewhat overestimated.

There are three mechanisms that can account for the observed tendency to overestimate early in the forecast period. One is delays in mission start due to launch slip or other schedule slips. Another contribution of the overestimation is an artifact of the analysis. For simplicity the analysis assumes that the user's actual demand matches his requirements. Phenomena, such as antenna blocking, restrict contact time opportunities and reduce the actual demand. These would tend to overstate the requirement used in the analysis (but are taken into account in the detailed NPAS model). A third source of the error may be a true overestimation in the mission model caused by documenting worst case requirements. (E.g., the user's documented requirement is for 20 minute contacts but the project normally schedules just 18 minutes.)

Long-Range Forecast Uncertainty. As was stated earlier, Figures 3 and 4 also indicated that there has been a pattern of underestimation of the longer term forecasts, yet the statistical evidence, being correlated and small, in itself is not predictive. The mission models from 1989 and 1993 were therefore examined for statistically significant trends. Along with the appearance and disappearance of entire missions, there were parameter changes that are analyzed in Tables 4 and 5.

Table 4. Comparison of User Values for Forecasts in 1989 and 1993

	Mean	Std. Dev.	Count	Minimum	Maximum
Extension (yrs)	1.75	.39	6	1.25	2.00
Service Growth (min/day)	23.25	74.68	12	-87.00	147.00
Delay (yrs)	2.06	1.26	4	.50	3.50

Table 5. Hypothesis Testing of User Values for Mean = 0

	P-Value	95% Lower	95% Upper
Extension (yrs)	.00	1.34	2.16
Service Growth (min/day)	.30	-24.20	70.70
Delay (yrs)	.05	.05	4.08

In Table 4 the row "Extension (yrs)" contains the slippage of the end-of-support date of six missions that appeared in both forecasts. The mean slip was 1.75; no missions were terminated early. The hypothesis test of the mean shown in Table 5 demonstrates that the mean is strong evidence of a positive life extension, with 1.3 years a conservative estimate. The "Service growth" statistics, derived from 12 missions, have large variations with a positive mean of 23 min/day. The hypothesis test of the mean does not provide evidence of a non-zero mean; i.e., the mean is not statistically significant. The "Delay" statistics, based on 4 missions that slipped from 0.5 to 3.5 years, indicates moderate support of a positive mean delay of 2 years. The small sample size, however, results in a lower 95% confidence bound for the mean of .05 – barely distinguishable from 0.

To summarize the analysis performed above, the data indicate a tendency to overestimate forecasts in their first year, extend ongoing missions, and possibly delay new missions. Using a conservative bias (based on the lower 95% confidence bound), one can conclude that there is a total short-term overestimate of 0.13 hrs/day, 1.3 years of mission life extension, no mission delay, and no growth in service level.

CONCLUSIONS

The long range forecasts, show a bias toward underestimation when the impact of mission extensions is larger than the impact of user delay. This effect is seen in the 1988 - 1990 period which underestimated the demand realized in the years 1991 to 1993, and will likely underestimate 1994 and 1995. The trend of underestimation can be expected to be replaced by overestimation for a number of years starting with 1998. Then demand from new missions slated for launch – particularly the space station – will be larger than the demand from ongoing missions with a potential for extension, and the potential for delay will dominate. If the new missions appear as scheduled, the forecasts are likely to be moderately underestimated.

Using statistics to predict the accuracy of future forecasts is not simply a matter of extending the past trends. Like the Challenger accident, chance events, changes in the economic and political environment can render past trends obsolete. One difference already noted is that the most recent forecasts consist of only approved missions. This lends a greater degree of conservatism to the forecasts. On the other hand, a slip or cancellation in the space station alone will have a substantial impact on the demand.

The SN commitment to meet the negotiated customer's requirements calls for conservatism in the forecasting. This requires that only statistically convincing (likely to be true) evidence be used to modify the mission model. Modification of the forecasting procedure to account for a delay bias is not advised. Fine tuning the mission model to more accurately reflect the current actual demand is recommended as it may marginally improve the first year forecasting...

References

- NASA, GSFC, Networks Division. *Keeping Track; A History of the GSFC Tracking and Data Acquisition Networks: 1957 to 1991.*
- NASA, GSFC. (1976, July). *Network Support Capability Study, STDN No. 114.*
- NASA, GSFC. (Internal Presentation, December, 1978). *TDRSS Mission Model and Loading Expectations 1981-1990.*
- NASA, GSFC. (Internal Presentation, August, 1980). *STDN Mission Model and Network Loading.*
- NASA, GSFC. (Internal Presentation, June, 1982). *STDN Mission Model and Network Loading.*
- NASA, GSFC. (1985, February). *Network Support Capability Study, Rev. 3, STDN No. 114*
- NASA, GSFC. (Internal Presentation, May, 1988). *SN Mission Model and Network Loading.*
- NASA, GSFC. (Internal Presentation, October, 1989). *SN Mission Model and Network Loading.*
- NASA, GSFC. (Internal Presentation, March, 1992). *SN Mission Model and Network Loading.*
- NASA, GSFC. (Monthly editions, 1978 to 1993). *Mission Requirements and Data Systems Support Forecast.*
- NASA, GSFC. (Monthly editions, 1978 to 1993). *Summary of STDN Operations.*

JOINT OPERATIONS PLANNING FOR SPACE SURVEILLANCE MISSIONS ON THE MSX SATELLITE

Grant Stokes, Lincoln Laboratory, Massachusetts Institute of Technology
 Andrew Good, Johns Hopkins University/Applied Physics Laboratory

Abstract - The Midcourse Space Experiment (MSX) satellite, sponsored by BMDO, is intended to gather broad-band phenomenology data on missiles, plumes, naturally occurring earthlimb backgrounds and deep space backgrounds. In addition the MSX will be used to conduct functional demonstrations of space-based space surveillance. The JHU/Applied Physics Laboratory (APL), located in Laurel, MD is the integrator and operator of the MSX satellite. APL will conduct all operations related to the MSX and is charged with the detailed operations planning required to implement all of the experiments run on the MSX except the space surveillance experiments. The non-surveillance operations are generally amenable to being defined months ahead of time and being scheduled on a monthly basis. Lincoln Laboratory, Massachusetts Institute of Technology (LL), located in Lexington, MA, is the provider of one of the principle MSX instruments, the Space-Based Visible (SBV) sensor, and the agency charged with implementing the space surveillance demonstrations on the MSX. The planning timelines for the space surveillance demonstrations are fundamentally different from those for the other experiments. They are generally amenable to being scheduled on a monthly basis, but the specific experiment sequence and pointing must be refined shortly before execution. This allocation of responsibilities to different organizations implies the need for a joint mission planning system for conducting space surveillance demonstrations. This paper details the iterative, joint planning system, based on passing responsibility for generating MSX commands for surveillance operations from APL to LL for specific scheduled operations. The joint planning system, including the generation of a budget for spacecraft resources to be used for surveillance events, has been successfully demonstrated during ground testing of the MSX and is being validated for MSX launch within the year. The planning system developed for the MSX forms a model possibly applicable to developing distributed mission planning systems for other multi-use satellites.

INTRODUCTION

The Midcourse Space Experiment (MSX) is a satellite-based experiment sponsored by the Ballistic Missile Defense Organization (BMDO) to be flown in a low-earth orbit beginning in late 1994. MSX was initially conceived as the first extended duration, long wave infrared (LWIR) phenomenology measurement program sponsored by BMDO; however, these early objectives have evolved into a more comprehensive experiment. MSX is now a multi-year experiment designed to collect broad-band phenomenology data on missiles, plumes, naturally occurring earthlimb backgrounds and deep space backgrounds. In addition, MSX will be used to collect spacecraft contamination data, to integrate, validate, and transfer advanced technologies to current and future BMDO systems, and to conduct functional demonstrations of space-based space surveillance

The Johns Hopkins University Applied Physics Laboratory (APL) is the integrator and operator of the MSX satellite. MSX will be launched from Vandenberg Air Force Base into a near-polar, low-earth, near sun-synchronous orbit. The MSX, shown in Figure 1, consists of the satellite superstructure, three primary optical sensors, contamination instrumentation and the spacecraft support subsystems. The optical axes of the three primary sensors (Space Infrared Imaging Telescope (SPIRIT III), Space-Based Visible (SBV) sensor, and Ultraviolet/Visible Imagers and Spectrographic Imagers (UVISI)) are parallel to one another and point in the +X direction. The support subsystems consist of the power subsystem, the thermal control

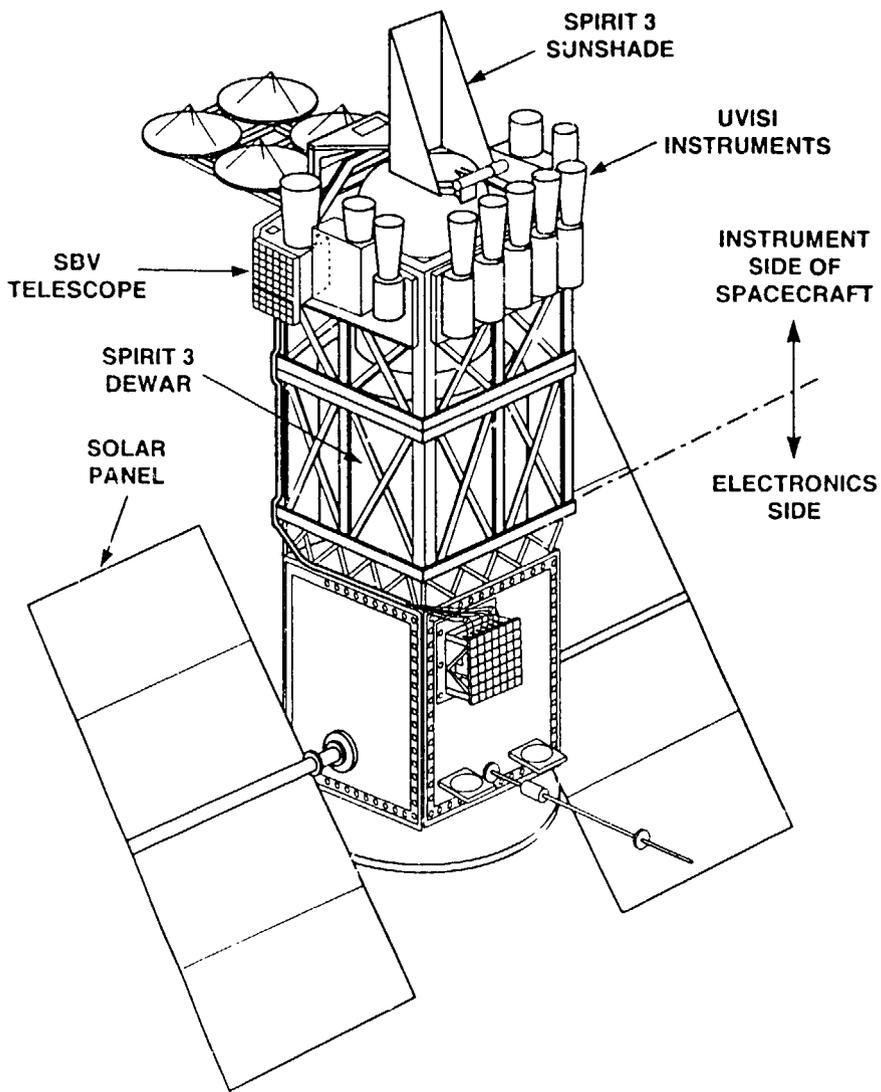


Figure 1 MSX satellite

subsystem, the command and data handling subsystem and the attitude determination and control subsystem. In addition, MSX houses a Beacon Receiver and On-board Signal and Data Processor (OSDP).

The SPIRIT III sensor has been developed by the Utah State University Space Dynamics Laboratory (USU/SDL). It is a passive mid to very long wavelength infrared (M/VLWIR) sensor and is the primary instrument aboard MSX for collecting target and background phenomenological data. SPIRIT III consists of a telescope with a 35.5 cm diameter aperture, a six-channel interferometer, a six-band radiometer and a cryogenic dewar/heat exchanger. The lifetime for SPIRIT III operations, which will be limited by the cryogen supply, is currently projected to be 18-24 months.

The UVISI sensor has been developed by APL with a primary mission to collect data on celestial and atmospheric backgrounds. Other UVISI missions include target characterization in the UV regime and observation of contamination particulates in conjunction with the contamination instruments. The UVISI sensor consists of four imagers and five spectrographic imagers (SPIMs) covering a spectral range from far UV to near infrared. The imagers include wide and narrow field-of-view sensors in both the visible and UV ranges and also include filter wheels to select various passbands. UVISI also includes an image processing system which will be used for closed-loop tracking of targets and aurora.

The SBV sensor, developed by the Lincoln Laboratory, Massachusetts Institute of Technology (LL), is the primary visible wavelength sensor aboard MSX. It will be used to collect data on target signatures and background phenomenologies, but the primary mission of SBV will be to conduct functional demonstrations of space-based space surveillance. SBV incorporates a 15 cm, off-axis, all-reflective, reimaging telescope with a thermoelectrically-cooled CCD focal plane array. SBV also includes an image processing system, experiment control system, telemetry formatter, and a data buffer for temporary data storage.

The collective suite of MSX instruments and supporting subsystems provide a broad range of data collection potential; however, a significant number of operational constraints have been imposed by spacecraft and instrument designers in order to achieve safe operations and to maintain the desired mission life (five years overall including two years for SPIRIT III). These constraints include limitations on boresight pointing relative to the sun, moon, and earth, restrictions on warming of the SPIRIT III dewar and baffle, bounds on battery depth-of-discharge and temperature, and thermal and duty cycle limits for the on-board tape recorders. The combination of these operational constraints with the BMDO goal of 14 data collection events per day represent a significant challenge to the MSX flight operations system.

The MSX flight operations system consists of facilities at APL (Operations Planning Center (OPC), Mission Control Center (MCC), Mission Processing Center (MPC), Performance Assessment Center (PAC), and Attitude Processing Center (APC)), at LL (SBV Processing, Operations and Control Center (SPOCC), and at the USAF Test Support Complex (TSC) at Onizuka Air Force Base. This collection of facilities is referred to as the "extended" MSX Mission Operations Center (MOC). A BMDO-led Mission Planning Team (MPT) instructs the MOC on a monthly basis on the type, number, and priority of experiments to be conducted. The OPC/SPOCC then develop operations planning products (e.g., schedules, contact support plans, command loads) which are provided to the MCC and TSC for execution. Spacecraft science and housekeeping data are collected by the MCC and TSC and then processed by the MPC, APC, and PAC as well as disseminated to the MSX data community.

SPACE SURVEILLANCE

Currently the United States maintains a world wide network of ground based sensors tasked with the acquisition of tracking data on all man-made objects in orbit around the earth. These sensors include a network of passive optical systems which are limited to a short duty cycle by poor weather and by daylight. Since foreign based sites are progressively more expensive and inconvenient to support, it is natural to ask whether ground based sensors could be supplemented or replaced by satellite based sensing systems. Satellite based sensors are not limited by daylight operation or poor weather and a single satellite borne sensor can sample the entire geosynchronous belt satellite population several times per day.

One of the missions of the MSX satellite is to demonstrate the feasibility of space-based space surveillance operations. One of the three principle MSX sensors, the SBV sensor has been specifically designed to provide visible-band satellite tracking data. The SBV consists of a six inch optical telescope with high off-axis rejection optics designed to acquire good quality satellite track data quite near the bright earth limb. In addition to the visible data from the SBV, track and optical signature data from the other MSX sensors is of interest to the space surveillance community. This is especially true for data from the S₁ IRIT III long-wave infrared sensor which promises the ability to detect satellites in the shadow of the earth.

The mission planning required to execute space surveillance activities is fundamentally different from that required to execute the other MSX missions. Normally space surveillance sensors are tasked on a day at a time basis by Space Command. In addition, Space Command provides special updates to the sensor tasking for special events, such as new launches, which require reactions on short time lines (minutes to hours). This operational tempo is significantly shorter than the normal MSX mission planning process which requires the operation to be well defined at the monthly planning level, which occurs as much as 10 weeks before the execution of the event on the spacecraft. If the routine MSX planning timeline were followed and space surveillance experiments were pre-planned, the ephemeris of many low altitude satellites targeted for observation will have changed enough to put them out of the sensor field of view by the experiment execution time. In addition, the normal MSX planning procedure contains no provision for generating observations in response to quick reaction experiments such as the launch of a new satellite.

The mission planning for the Space Surveillance experiments on the MSX satellite requires the ability to leave considerable flexibility in the experiment timing and attitude profile to be followed by the MSX in the experiment execution until late in the experiment planning process. Under "normal" circumstances the details of the operation, consisting of the list of satellites to be observed, the attitude profile for the MSX and the data acquisition times can be defined one to two days before the execution on the MSX. Special "quick reaction events", such as acquiring track data on a newly launched satellite in its transfer orbit to the geosynchronous belt, require reaction times on the order of hours.

JOINT PLANNING PROCESS

The mission planning required to operate a satellite as complex as the MSX is a large task under any condition; however, it is complicated further by the breadth of the experimental missions to be conducted by the satellite. Most of the MSX experiments are amenable to a long-term planning process either because their targets are slowly changing (eg., naturally occurring earth limb and deep space backgrounds) or because they are under the control of the experimenter (eg., dedicated missile shots). This long-term planning process allows time for the mission planners to communicate with the Principle Investigators to clarify the details of a specific experiment in the planning process. On the other hand the space surveillance experiments designed at Lincoln Laboratory, Massachusetts Institute of Technology require fundamental modifications late

in the planning process on timelines that admit little manual intervention. Thus, the MSX program was faced with a fundamental decision to either implement a highly automated and expensive general purpose planning system which would accommodate the complete set of diverse MSX experiments or to build a long-term planning system for the majority of the experiments and allow a link into the planning process from a more automated system dedicated to planning the space surveillance experiments. For reasons of economy and to minimize the complexity of the entire implementation, the second option was chosen. Since the expertise needed to fulfill the space surveillance mission planning function resides at Lincoln Laboratory, the center for surveillance experiment planning was located there in the SBV Processing, Operations and Control Center (SPOCC).

In order to simplify the planning procedures and to allow the parallel planning of experiments at APL and LL centers, the following three principles were adopted by the organizations involved:

I. The planning team at LL is responsible for complete operation of the MSX spacecraft and all its sensors during the time period scheduled for a surveillance experiment. Thus, the LL team will receive the MSX in a given standard configuration, known as parked mode, will generate all the command information for both the satellite and sensor sub-systems required to implement the data collection and will return the spacecraft to the standard parked mode upon completion of the event. The LL planning team is responsible for abiding by all spacecraft constraints and operating rules during the conduct of surveillance events

II. The long-term planning for the space surveillance events will consist of allocating time intervals and resource budgets to the space surveillance events. Thus, it has been agreed that the specific modes of satellite operation for surveillance experiments will be left to be filled in the day prior to conduct of the event. However, during the long-term planning process, the experiment will be scheduled during a specified time interval and the integrated effect on the MSX resources, such as battery depth-of-discharge (DOD) and changes to the spacecraft thermal state will be agreed on a "not to exceed" basis.

III. The final responsibility for safe spacecraft operations will belong to APL which will check all command information generated by LL. The check will be automated and will be conducted shortly before upload of the commands to the MSX.

These three principles enable the parallel planning of operations at the two centers by clearly separating the responsibilities of each planning center during each of the planning intervals necessary to operate the MSX. However, they also require an overlap of capability between the two planning sites because both must be able to generate command information for the entire satellite. This duplication was accepted as a cost of having a distributed planning system.

The planning system for the MSX goes through four phases of activity as shown in Figures 2 and 3 in order to generate a data collection event for the satellite. The phases and the interaction between the planning centers for surveillance events are described below:

Opportunity Analysis - The planning centers are given experiment priorities on a monthly basis by the BMDO run Mission Planning Team. The priorities are provided six weeks before the start of the month being planned. Once the priorities are received each planning center, the OPC at APL and the SPOCC at LL, analyzes the experiments for which they are responsible to determine feasible times for which data may be collected. For surveillance experiments, items such as target visibility, sun angle and proximity to the earth limb or earth shadow are considered and a list of feasible times is compiled. The opportunity list includes the start and duration of each feasible event start time, the event duration, the relative desirability of that particular feasible time compared with others on the list, an indication of the accuracy of the estimated event start time (eg., if the

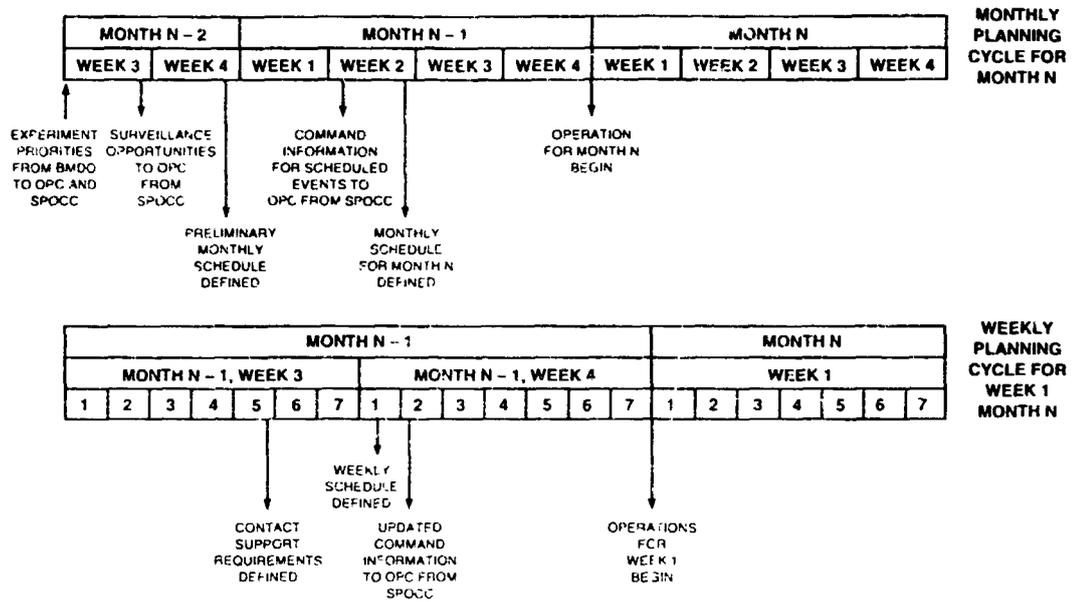


Figure 2 Monthly and weekly planning cycles for MSX experiment

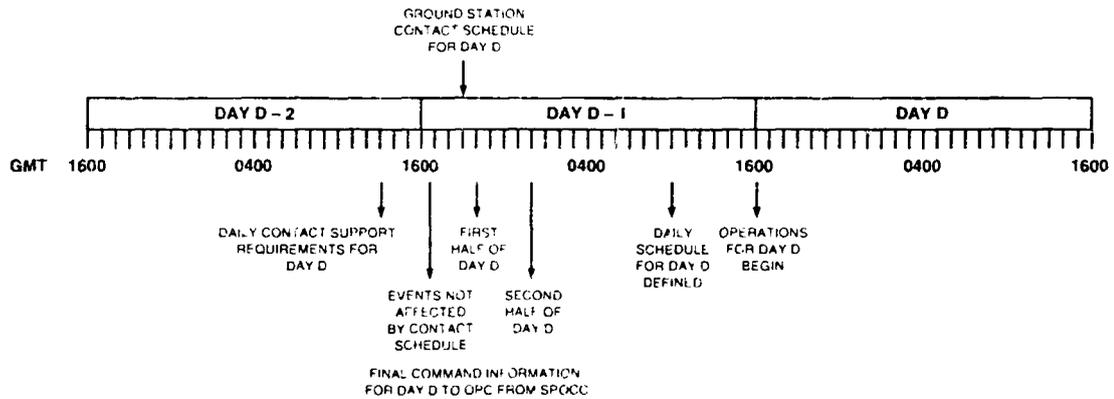


Figure 3 Daily planning cycle for MSX experiments

243594-1

satellite to be observed has a low altitude, the time it becomes visible will not be precisely known (10 weeks in advance) and a pointer to an example set of command information for that type of event. The space surveillance opportunity list and the example command information sets are provided to the OPC for integration with the other experiments in the Monthly Planning Process.

Monthly Planning - The OPC combines the opportunity lists for each of the different types of experiments and constructs a schedule of data collection events to be conducted during the month. Since the MSX spacecraft is not designed for 100% duty cycle, the scheduling process must pay close attention to the use of spacecraft resources. In addition, the cryogenic SPIRIT III sensor is very sensitive to the thermal state and history of the MSX. In order to estimate the resources which will be used by the space surveillance events, the OPC analyzes the sample command information provided by the SPOCC for each event type and estimates the change in battery DOD and the thermal deltas for critical elements. These estimated resource expenditures now become a "not to exceed" budget for the conduct of the surveillance data collection event. The actual pointing and targets may be considerably different, but the integrated effect on the spacecraft resources may not be any larger than that defined during the monthly scheduling process. The OPC generates a monthly schedule for the MSX operations during the month and, after suitable iteration with BMDO and the SPOCC, the schedule is published and the SPOCC provides the OPC with preliminary command information for all of the space surveillance events as scheduled. The Weekly Planning process is then started for the first week of the planning month as shown in Figure 2.

Weekly Planning - Weekly planning is largely used by the OPC to update non-space surveillance experiments to reduce the amount of work needed at the daily planning level. In addition, the uplink and downlink requirements for the earth stations in the SGLS network are compiled and input into the scheduling process at the TSC. For surveillance experiments, the automated SPOCC planning system is re-run taking into account the updated ephemeridies for the intended targets (if known at the time) and the MSX, and an update of the event start times is provided to the OPC along with revised command information for each event to be executed during the planning week.

Daily Planning - The final mission planning occurs at the daily planning level, which occurs the day before the events are to be executed on the MSX, as shown in Figure 3. At that time the final uplink/downlink schedules are known, the orbital geometry of the MSX and the targets are available with sufficient accuracy and tasking lists are available from Space Command for tasked experiments. At that time the SPOCC generates final sets of command information for each event during the day and provides them to the OPC for analysis and inclusion in one of the three command upload creation cycles run during each day for the MSX. The SPOCC is responsible for generating command information that is compliant with all MSX constraints, operation rules and resource budgets determined during the scheduling process. The OPC conducts a final, automated analysis of the events as provided by the SPOCC and, if they are compliant with the agreed rules, incorporates them into the command load.

Quick Reaction Events - A number of space surveillance events require shorter timelines than provided by the daily planning process described above. These include events such as the launch of a new satellite, which is scheduled well in advance, but the specific launch time is not known with sufficient accuracy until after the launch. A series of special procedures have been developed to plan events requiring a very quick response from the planning system. The procedures require that an interrupt window be defined at the monthly planning level. The window defines a range of times during which normal MSX operations can be disrupted in order to collect data on a specific event if it happens. The ability to capture the event depends on the availability of suitable pre-scheduled ground station uplinks which may be used to uplink new commands to the MSX. Once a quick reaction event has been declared, the SPOCC will generate commands to observe the satellite based on tipoff information from Space Command (such as the time of launch in the case of a new launch) and will forward the new commands to APL for inclusion in an uplink which will

cancel the existing commands and replace them with those required to execute the quick reaction event observations. Preliminary timing tests run on the planning process indicate that the SPOCC can have the required command information ready for transmission to APL within 30 minutes of the launch and that APL can process the results in time to track a satellite in a transfer orbit to geosynchronous altitude. Final timing tests and procedure verification will take place after a period of operational experience with the MSX under the normal planning process.

CONCLUSIONS

In order to accommodate the mission planning for a broad range of diverse experiments to be run on the MSX satellite, a distributed mission planning system has been defined and implemented. Under this model, the MSX mission planning is accomplished for all non-surveillance experiments using a long-term planning process at the APL OPC. Space surveillance experiments are planned by LL and carried in the APL planning schedule as event durations and resource utilization budgets without the details of the operation which are provided to the OPC during the final Daily Planning process in command ready form.

This system of distributed mission planning has been developed for a complex, multi-function/multi-mission spacecraft where the expertise needed to conduct mission planning for various mission types is distributed between two locations. The advantage of the process as defined is that the two planning centers can conduct the mission planning functions in parallel, each adding the details of the operation as they are available or according to the capabilities of each planning system. The event is held in the master schedule by budget allocations and schedule place holders until the final details are available. Having each planning center responsible for generating command information for the entire spacecraft for the events for which they are responsible simplifies the interaction between planning centers considerably since each can consider the other's events as "black boxes" until the final details are provided in a complete package. The disadvantage of this approach is that each planning center needs to understand and be capable of commanding every satellite function that will be needed to satisfy their events.

Given that many of the satellites launched currently are large multi-function payloads containing a broad range of instruments, collecting data for a diverse user set, the MSX planning system experience may yield broadly applicable lessons learned. The main requirement to implementing such a cooperative planning system has been a mutual understanding of each participant's mission requirements and a willingness on the part of all parties to consider all the alternatives and to negotiate a sensible approach to solving the mission planning puzzle.

**MULTI-MISSION OPERATIONS FROM THE HEADQUARTERS
PERSPECTIVE**

Guenter Strobel
NASA-HQ

Paper Not Available

p. 8

**INTERNATIONAL MISSION PLANNING FOR SPACE
VERY LONG BASELINE INTERFEROMETRY**

James S. Ulvestad*
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, CA 91109

ABSTRACT

Two spacecraft dedicated to Very Long Baseline Interferometry (VLBI) will be launched in 1996 and 1997 to make observations using baselines between the space telescopes and many of the world's ground radio telescopes. The Japanese Institute of Space and Astronautical Science (ISAS) will launch VSOP (VLBI Space Observatory Programme) in September 1996, while the Russian Astro Space Center (ASC) is scheduled to launch RadioAstron in 1997. Both spacecraft will observe radio sources at frequencies near 1.7, 4.8, and 22 GHz; RadioAstron will also observe at 0.33 GHz. The baselines between space and ground telescopes will provide 3-10 times the resolution available for ground VLBI at the same observing frequencies. Ground tracking stations on four continents will supply the required precise frequency reference to each spacecraft, measure the two-way residual phase and Doppler on the ground-space link, and record 128 Megabit/s of VLBI data downlinked from the spacecraft. The spacecraft data are meaningless without cross-correlation against the data from Earth-bound telescopes, which must take place at special-purpose VLBI correlation facilities. Therefore, participation by most of the world's radio observatories is needed to achieve substantial science return from VSOP and RadioAstron. The collaboration of several major space agencies and the ground observatories, which generally follow very different models for allocation of observing time and for routine operations, leads to great complexity in mission planning and in day-to-day operations. This paper describes some of those complications and the strategies being developed to assure productive scientific missions.

*For the International Space VLBI Team

INTRODUCTION TO SPACE VLBI

The Very Long Baseline Interferometry (VLBI) technique (e.g., Thompson, Moran, & Swenson 1986) has been used for over 25 years to maximize the angular resolution of radio-emitting astronomical objects. Widely separated radio telescopes simultaneously observe the same radio source at the same frequency. The data are digitized and recorded at a rate of over 100 Megabit/s on wideband videotapes or cassettes. A highly accurate clock at each telescope is used to time-tag the data. Following an observation, the recorded data are physically transported to a special-purpose correlation facility; information about the observing conditions, recording, and calibration at each telescope also is transmitted to the VLBI correlator. Cross-correlation of data from each pair of radio telescopes is performed to derive the source "visibility" as a function of baseline length and orientation. The collection of source visibilities then is used by the radio astronomer to model or map the radio source and derive various astrophysical parameters.

At a given observing frequency, the resolution of ground-based VLBI is limited by the physical dimensions of the Earth. At the common VLBI observing frequency of 5 GHz, a 10,000-km baseline corresponds to an interferometer fringe spacing of about 1.2 milliarcseconds. Higher resolution can be obtained either by using a higher observing frequency or by placing one telescope of a VLBI system in space, first suggested seriously in the late 1970s (e.g., Preston, Hagar, & Finley 1976; Burke & Roberts 1979). Since different source components dominate at different frequencies, and brightness-temperature measurements depend on the physical baseline length rather than the angular resolution, the two approaches to higher resolution can be viewed as complementary.

Space VLBI (SVLBI) observations present challenges beyond those found in ground-based VLBI experiments. Cross-correlation requires an accurate model for the relative signal delay (and its derivatives) for each telescope pair. When one telescope is in space, this requirement translates to a need for highly accurate orbit determination. The observing frequencies and time of reception for each data sample must be accurately known, requiring a frequency reference on the spacecraft that is comparable in quality to a hydrogen maser. This reference can be generated by transferring the stability of an Earth-based frequency standard from each tracking station to the spacecraft; residuals from the two-way link are recorded for use at the correlator. Because VLBI data must be recorded at a rate of more than 100 Megabit/s for extended periods, a wideband downlink is necessary. Finally, the ancillary data required for correlation must be constructed from a combination of spacecraft telemetry and tracking-station logs.

The technology required for SVLBI was demonstrated in a series of observations carried out from 1986 through 1988 (Levy et al. 1986, 1989; Linfield et al. 1989, 1990). In those experiments, the Tracking and Data Relay Satellite System (TDRSS) was used together with large radio telescopes in Japan and Australia to observe a number of radio sources at frequencies of 2.3 GHz and 15 GHz. Interference fringes were found on baselines as long as 2.15 Earth diameters (close to the maximum baseline sampled), and crude models were made of the observed radio sources. The successful observations demonstrated the technical feasibility and scientific potential of SVLBI observations, and have led directly to the dedicated SVLBI satellites that are scheduled for launch in the next several years.

VSOP AND RADIOASTRON MISSIONS

The VLBI Space Observatory Programme (VSOP) satellite will be launched in September 1996 by the Japanese Institute of Space and Astronautical Science (ISAS). The RadioAstron spacecraft, part of the Spectrum series of missions under development in Russia, is being built by the Astro Space Center (ASC) and the Lavochkin Association and is scheduled for

launch in 1997. Each spacecraft will carry an 8-10 meter deployable radio telescope together with receivers capable of making observations at standard VLBI frequencies in the gigahertz range. The nominal mission lifetimes are approximately 3 years. VSOP will be in an elliptical orbit with an apogee height of about 22,000 km, while RadioAstron will be in an elliptical orbit with an apogee height of about 77,000 km. Table 1 summarizes a number of the features of the missions, while Figures 1 and 2 are sketches of the two spacecraft. The primary scientific goals of both spacecraft will be the imaging and modeling of the nuclei of active galaxies (quasars, BL Lacertae objects, and radio galaxies) as well as investigations of OH and H₂O maser emission within our own Galaxy. Although the operational lifetimes of the two spacecraft are expected to overlap, they will operate independently in the sense that they generally will not observe the same sources simultaneously.

Table 1. SVLBI Mission Characteristics

Mission	VSOP	RadioAstron
Telescope	8 m	10 m
Mass	800 kg	5000 kg
Data Rate	128 Mb/s	128 Mb/s
Frequency	22 GHz	22 GHz
	4.8 GHz	4.8 GHz
	1.7 GHz	1.7 GHz
		0.33 GHz
Perigee Ht.	1,000 km	4,000 km
Apogee Ht.	22,000 km	77,000 km
Period	6.6 hr	28 hr
Inclination	31°	51°

Both spacecraft will make use of an uplink tone near 8 GHz (RadioAstron) or 15 GHz (VSOP) to establish the on-board frequency reference. On board transponders will enable round-trip links with the ground tracking stations. The two-way phase on this link will be used to establish the error in the spacecraft frequency standard and to derive Doppler data needed for accurate orbit determination. Each spacecraft will downlink the wideband VLBI data at 15 GHz. For further descriptions of the RadioAstron and VSOP missions, see Kardashev and Slysh (1988) and Hirose (1991), respectively.

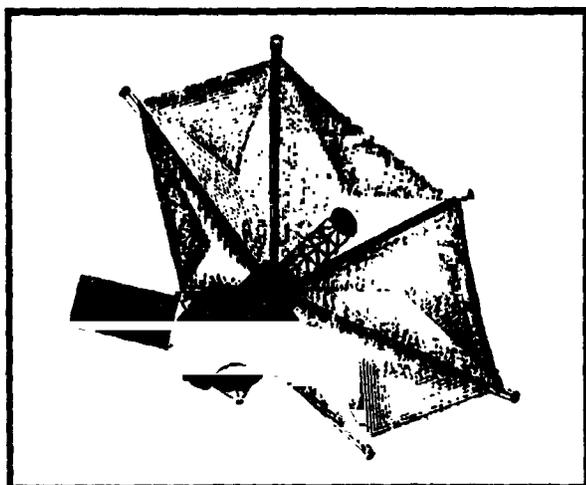


Figure 1. Sketch of VSOP spacecraft.

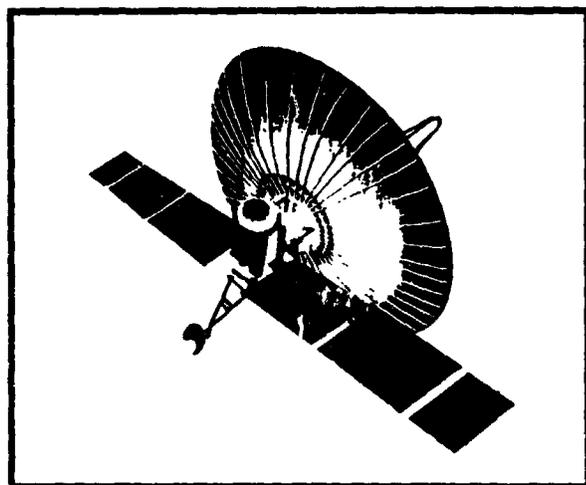


Figure 2. Sketch of RadioAstron spacecraft.

Because of the need to maintain a two-way phase link and a wideband data link during observations, scientific data can be gathered only when a spacecraft is in direct contact with a ground tracking station. Furthermore, the quality of the scientific results depends critically on the sampling of the aperture plane by the space-ground baselines, so a globally distributed tracking network is crucial to the success of VSOP and RadioAstron. Therefore, the U.S. National Aeronautics and Space Administration (NASA) is funding four ground stations that will be dedicated to tracking the two spacecraft. Three new 11-m antennas will be built, one each at the NASA tracking com-

plexes in California, Spain, and Australia. A fourth station will be at the National Radio Astronomy Observatory (NRAO) facility in Green Bank, West Virginia, and will make use of an existing 14-m antenna. In addition, VSOP will be tracked by a new 10-m antenna to be built at Usuda, Japan, while RadioAstron will be tracked by a 32-m antenna at Ussuriisk (near Vladivostok), Russia, and possibly by another antenna near Moscow.

INTERNATIONAL PARTICIPATION IN VSOP AND RADIOASTRON

The spacecraft tracking described above is only one aspect of the substantial international participation in the VSOP and RadioAstron missions. The flight receivers for RadioAstron are being built by Finland (22 GHz), the European Space Agency (4.8 GHz), Australia (1.7 GHz), and collaboratively by India and Russia (0.33 GHz). Highly accurate orbit determination will be provided by Japanese and Russian agencies and by NASA's Deep Space Network (DSN).

A required element unique to SVLBI is the participation of large networks of ground radio observatories, most of which are independent of the agencies building and tracking the spacecraft. Some of these ground observatories not affiliated with space agencies include the Very Long Baseline Array, the Very Large Array, and the 100-m Green Bank Telescope now under construction, all operated by NRAO; the members of the European VLBI Network, consisting of telescopes in England, Germany, the Netherlands, Sweden, Italy, and China, as well as associate members in Poland, Ukraine, Russia, Finland, Germany, and France; the Australia Telescope National Facility; Nobeyama Radio Observatory (Japan); the Communications Research Laboratory (Japan); Hobart Observatory (Australia); Hartebeesthoek Observatory (South Africa); and the Giant Metre Wave Radio Telescope (India). Other participating radio telescopes more closely related to the space agencies include the 70-m antennas of the NASA DSN, the 64-m ISAS antenna at Usuda, and the 70-m antennas located in Russia (Ussuriisk) and in the Ukraine (Evpatoria). Each observatory has its own method of allocating time among a variety of scientific requests, including VLBI and a host of other radio astronomical programs. Although the

ground telescopes are required for any science return from VSOP and RadioAstron, most are not under the control of the space missions. Therefore, a significant aspect of the planning for VSOP and RadioAstron has been the process of negotiation between the space missions and ground observatories, wherein the needs of the missions are balanced with the other scientific priorities of the observatories.

The primary bodies established for the scientific management of the missions are the RadioAstron International Scientific Council (RISC) and the VSOP International Scientific Council (VISC). Each is co-chaired by a representative of the Russian (RISC) or Japanese (VISC) project and a representative of the outside international community. The RISC and VISC contain representatives of the Russian and Japanese projects, foreign space agencies, and other participating organizations (including ground observatories). Because VSOP and RadioAstron face many of the same problems and must share resources such as tracking stations, ground telescopes, and correlation facilities, there is considerable overlap between the membership of the VISC and the RISC. Each organization meets formally twice per year, with additional informal discussions held during other international meetings.

SCIENTIFIC ACCESS AND GROUND OBSERVATORY PARTICIPATION

The policies for granting observing time on VSOP and RadioAstron are the subject of ongoing discussions that will be completed only when the announcements of opportunity are formally released. The standard practice for space astrophysical observatories has been to reserve some fraction (up to 100%) of the observing time for those individuals and organizations that have built the spacecraft or contributed scientific instruments. This reserved time often is used to carry out key science programs that are the primary goals of the missions. In contrast, the long-standing practice of most radio observatories is one of open access based solely on scientific peer review and independent of an individual's organizational or national affiliation; they typically have little or no reserved time. However, it is not possible to schedule SVLBI programs without some guarantee that particular time periods will be made

available to the space missions by the ground observatories, since the scientific return of any specific observation depends critically on the distribution of the participating ground telescopes.

For both VSOP and RadioAstron, the agreements that have been made to date specify an open peer-review process based on scientific merit and technical feasibility of each proposal. Scientific referees will be selected from among nominees provided by the participating organizations. A few key science programs (e.g., a survey for high brightness temperature, or monitoring of superluminal motions) will be listed in the announcements of opportunity. Many of the members of the key science teams may be selected based on their proposals. Representatives of organizations that have made substantial contributions to the spacecraft and mission development also may be added to the key science teams by the RISC and the VISC.

The Global VLBI Working Group, consisting of the directors of major radio observatories or their representatives, has negotiated ground-telescope participation with the space missions. Based on the general philosophy of access for the highest quality science, many ground observatories have now made commitments of some fraction of their observing time for at least the first year of the SVLBI missions. The expectation is that those commitments will be renewed if the quality of the science return during the first year is commensurate with that of the other science being done by these observatories. Typical commitments from the majority of the world's major radio observatories range from 10% to 30% of their total observing time in a year. In most cases, the commitments have been made to a general SVLBI pool of observing time that would cover both missions, with the understanding that the missions will divide that time as scientifically appropriate. Despite the substantial commitments of time from ground observatories, the need for significant numbers of telescopes to observe for a large part of a day in order to produce a single SVLBI image implies that the scientific return of the missions may be limited by the lack of ground telescopes, particularly if both spacecraft are in orbit simultaneously. Extensive observing simulations are in progress to determine the minimum numbers of ground tele-

scopes required to make observations of different types. Ultimately, it may be up to the investigators, the scientific reviewers, the international science councils, and the scheduler(s) to determine the scientific tradeoffs between a large number of observations employing a minimal number of telescopes and a smaller number of observations using more ground telescopes.

SCIENTIFIC SIMULATIONS AND SCHEDULING

The planning of the missions and analysis of the scientific return has benefited tremendously from the development of a variety of software packages that simulate different aspects of the missions. Simulation packages have been developed by D. Murphy at the Jet Propulsion Laboratory (JPL); R. Taylor and G. Young at the University of Calgary; H. Kobayashi and collaborators at ISAS; L. Gurvits, V. Yakimov, and collaborators at ASC; and I. Fejes and collaborators at the Institute of Geodesy, Cartography, and Remote Sensing in Hungary. (See Fejes et al. 1994, and Murphy et al. 1994.) One of the most important functions of the software is to simulate the aperture-plane coverage for different combinations of tracking stations and ground telescopes, given the known spacecraft constraints. The packages can produce plots of the aperture-plane coverage as a function of source position or time for an assumed set of participating ground telescopes, and some also analyze the detection thresholds and image quality for those coverages for an assumed source model. Two early successes of the JPL simulations were the realizations that the VSOP telemetry antenna mask and a RadioAstron radiator constraint significantly reduced the missions' scientific returns; subsequent redesigns reduced or removed those constraints.

The continuing development of the simulation software has two main goals. The first goal is to use simulations as an aid in scheduling the missions. The software would be used to analyze the technical feasibility of proposals and the possible tracking scenarios. Analyses of the aperture-plane coverage as a function of time (particularly important for the rapidly precessing orbit of VSOP) will be used to find the optimum time to schedule a particular scientific observation. As an example, Figure 3 shows

the synthesized aperture for a 5-GHz observation of 3C 345 using the combination of VSOP and the 10 VLBA telescopes at three different epochs separated by six months each (from the software written by D. Murphy). This diagram plots the east-west and north-south components of all sampled interferometer baselines. The top and bottom panels show changes in the synthesized aperture over time due to precession of the spacecraft orbit. The middle panel has no space-ground baselines because the radio source lies within 70° of the Sun and cannot be observed by the spacecraft. Two major differences between space-ground and ground-only VLBI are readily apparent: (1) the projected baselines for the ground telescopes alone (see middle panel) are much shorter than the space-ground baselines; and (2) the ground baselines (inner portion of all three panels) do not change from month to month.

The second use for the simulation software will be as an aid to the prospective user. The user software and associated user guides will be integral parts of the announcements of opportunity. It currently is thought that the main software packages to be used in proposal preparation will be those developed at JPL and at the University of Calgary. These packages will be used as a tool to familiarize the prospective user with the complexities of the SVLBI missions. Details of particular observations then can be simulated, enabling a stronger proposal to be written. The tools will also reduce the number of technically infeasible observations that are proposed, thus reducing the workload in the proposal evaluation process.

A strawman scheduling program has been developed by D. Meier of JPL (Meier 1994) to determine the need for ground radio telescopes in support of SVLBI observations. After making assumptions about the minimum number of telescopes needed for particular types of observations, the total requirements on the world's ground radio telescopes have been analyzed for the case when either VSOP or RadioAstron is flying alone, or when the two are in operation simultaneously. These requirements were of great use in the aforementioned negotiations for guaranteed ground radio telescope time.

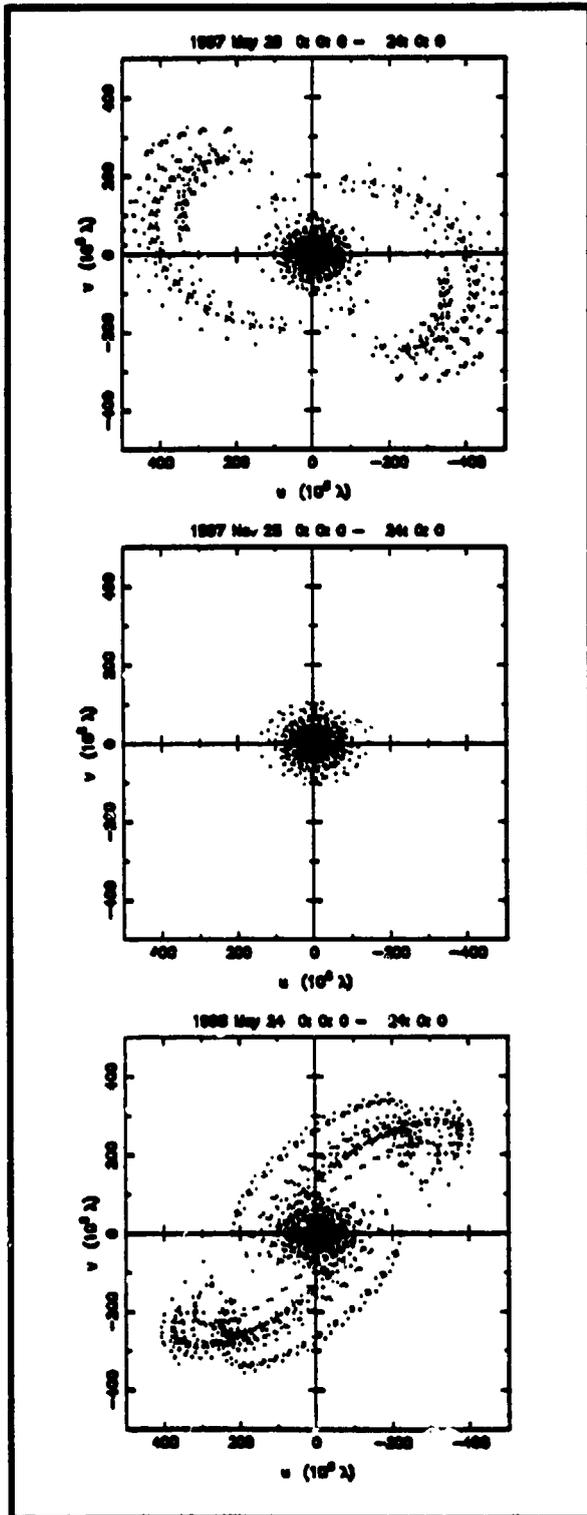


Figure 3. Aperture-plane coverage for 5-GHz observations of 3C 345 at 6-month intervals, using VSOP and the VLBA. Projected baselines are given in units of millions of wavelengths.

Additional software will be used to create the scientific observing schedule. This software would require inputs such as the source coordinates, the set of ground telescopes available at a particular frequency, and the quality of the aperture plane coverage as a function of time (based on the simulation software). The output would be an observing program that would achieve a high scientific return for a given set of constraints on ground and space resources. Because of the need to finalize the precise commitments of the ground telescopes, this schedule would be produced up to one year in advance of the appropriate observation period, but the scheduling procedures also must be flexible enough to accommodate contingencies aboard the spacecraft or at any of the supporting ground facilities.

MISSION OPERATIONS

The details of the operations of VSOP and RadioAstron have been entrusted to two parallel groups, the RadioAstron Science Operations Group (RSOG) and the VSOP Science Operations Group (VSOG). The groups' membership consists largely of representatives of the space agencies operating the spacecraft, but also includes affiliated international members such as the developers of the simulation software and (ultimately) representatives of the key science teams. The responsibilities of the RSOG and VSOG include preparation of the announcements of opportunity, development of simulation and scheduling software, production of both scientific and detailed schedules, allocation of ground resources, coordination of the daily operations of all international mission elements, calibration of the space radio telescope data, and overall mission performance assessment. Much of the work on simulations and scheduling has been, and will continue to be, performed under the auspices of the VSOG and RSOG. In the end, the scientific success or failure of VSOP and RadioAstron will depend on the effectiveness of the VSOG and RSOG in coordinating all the international participants.

The VSOG and RSOG have concentrated heavily on the duties involved with pre-launch science planning. Recently, a subgroup to both the VSOG and RSOG was formed in order to coordinate pre-launch planning of mission operations. This team includes representatives of

the different space agencies, tracking stations, and correlation facilities. Its key responsibility is the development of the detailed interfaces and procedures needed for exchange of data such as schedules, phase residuals, and correlator input logs. It also participates in development of plans for the in-orbit checkout phases and in generating agreements on the operational responsibilities of all mission elements.

A key aspect of the mission operations for SVLBI is the development of a reliable international system for data transfer. Schedule files and required updates must be made available in a timely fashion. A variety of tracking, telemetry, and VLBI data must take different, sometimes circuitous paths before arriving at the correlation facilities. The relative paucity of operations personnel implies that all data-transfer tasks must be automated as much as possible. Details of the international data transfer system for SVLBI, including the generation of correlator input files, are presented at this conference in a paper by Wiercigroch (1994).

ORBIT DETERMINATION

The primary means of orbit determination for VSOP and RadioAstron will be the two-way Doppler data derived from the 15-GHz and 8-GHz links between tracking stations and spacecraft. These data will be supplemented by range and range-rate data from the spacecraft command stations. Accurate predicted orbits are needed for the tracking stations to follow the spacecraft and to keep the two-way phase residuals at an acceptably low level. More accurate spacecraft trajectories, with position and velocity errors less than 100 meters and 1 cm/s, respectively, are required for the correlator models. In addition, acceleration errors much smaller than 10^{-7} m/s² are needed to enable long coherent integration times. Covariance analyses have revealed that the most difficult problem will be that of achieving the velocity and acceleration requirements near spacecraft perigee.

The two-way Doppler data used for orbit determination must be derived using a two way phase link that is a new feature for both VSOP and RadioAstron. The tracking stations under construction by different agencies have dif-

ferent implementations for that link and the derivation of the Doppler data. It remains to be seen whether they will yield data of comparable quality in order to produce the accurate orbit required for data correlation.

DATA ANALYSIS

VLBI data are recorded in real time, with the recordings brought together later for pairwise cross-correlation at a special purpose correlator. The VLBI correlators use models of delay and delay-rate to determine the window used for cross-correlation; fits to the correlator output are used to determine the location of the interference fringes and to derive visibility functions from the output data. The permitted values of delay and delay-rate must be considerably larger in SVLBI than for ground-only VLBI because of the longer baseline and higher relative speed between space and ground telescopes. Since one element is not fixed to the Earth, a new correlator interface must be built to include a spacecraft trajectory in the model. Measurements of the residual phase on the link between tracking station and spacecraft must be input at least 10 times per second in order to account for frequency variations caused by effects such as orbit errors and propagation of the uplink tone through the Earth's troposphere and ionosphere. Each VLBI correlator is a one-of-a-kind system of hardware, firmware, and software, and presents a unique technical challenge to the processing of SVLBI data.

The standard software used for analyzing much of the world's radio interferometry data is the Astronomical Image Processing System (AIPS), developed by NRAO; VLBI data are also processed using other software such as that developed at the California Institute of Technology. AIPS is being upgraded by NRAO in order to be capable of processing SVLBI data. New routines are being written to improve the detection of weak interference fringes and to follow those fringes forward or backward in time. Special-purpose software also is being written to enable improved modeling of the radio sources. Tests of some parts of this software have been performed using the experimental SVLBI data obtained with TDRSS, and more are anticipated in the future.

Problems associated with proposing SVLBI observations and analyzing the resulting data will be considerably more formidable than those associated with ground VLBI. Therefore, the international participants in the SVLBI missions need to provide as much assistance as possible to the scientists interested in using those missions. The simulation software described previously is an important part of the response to this challenge. On-line information, workshops, and articles in newsletters and the scientific literature also are being developed in order to assist prospective users. User support in analyzing SVLBI observations using the AIPS software will be made available by NRAO at their facility in New Mexico. Other mission participants will provide more limited support of data analysis.

ACKNOWLEDGMENTS

The International SVLBI Team includes, but is not limited to, the following individuals: E. Akim, V. Altunin, S. Ananthakrishnan, V. Andreyanov, M. Artyukhov, R. Bakitko, A. Berman, R. Booth, B. Burke, W. Cannon, L. D'Addario, P. Dewdney, J. Ellis, I. Fejes, D. Gabuzda, L. Gurvits, A. Gvamichava, H. Hirabayashi, H. Hirosawa, T. Ichikawa, M. Inoue, D. Jauncey, N. Kardashev, T. Kato, N. Kawaguchi, H. Kobayashi, S. Likhachev, D. Meier, K. Miyoshi, M. Morimoto, Y. Murata, D. Murphy, E. Nakagawa, T. Orii, V. Pavlov, V. Perminov, Y. Ponomarev, M. Popov, R. Preston, V. Rogalsky, E. Romanov, J. Romney, R. Schiuzzi, A. Shchikhet, V. Slysh, J. Smith, J. Springett, K. Sukhanov, R. Taylor, J. Ulvestad, S. Urpo, E. Valtaoja, K. Wellington, A. Wiercigroch, R. Wietfeldt, V. Yakimov, Z. Yamamoto, and A. Zensus. Many other individuals also are making important contributions to VSOP and RadioAstron, particularly in the design and implementation of the spacecraft and ground tracking stations.

The contents of this paper have been developed by J. Ulvestad based on current plans for SVLBI, and he accepts full responsibility for any errors. He thanks his colleagues D. Murphy, R. Preston, J. Smith, and A. Wiercigroch for comments on drafts of this paper. A portion of this work was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under contract with NASA.

REFERENCES

- Burke, B. F., & Roberts, D. H. (1979). VLBI in Space. *Bulletin of the American Astronomical Society*, 11, 467.
- Fejes, I., Murphy, D. W., Taylor, A. R., Yakimov, V., & Young, G. (1994). Space VLBI User Assistance Software. *VLBI Technology: Progress and Future Observational Possibilities*. in press.
- Hirosawa, H. (1991). VSOP Satellite System Overview. *Frontiers of VLBI*, 21-25, Tokyo: Universal Academic Press.
- Kardashev, N. S., & Slysh, V. I. (1988). The RadioAstron Project. *The Impact of VLBI on Astrophysics and Geophysics*. Proceedings of IAU Symposium 129, 433-440, Dordrecht: Kluwer Academic Publishers.
- Levy, G. S. et al. (1986). Very Long Baseline Interferometric Observations Made with an Orbiting Radio Telescope. *Science*, 234, 187-189.
- Levy, G. S. et al. (1989). VLBI Using a Telescope in Earth Orbit. I. The Observations. *Astrophysical Journal*, 336, 1098-1104.
- Linfield, R. P. et al. (1989). VLBI Using a Telescope in Earth Orbit. II. Brightness Temperatures Exceeding the Inverse Compton Limit. *Astrophysical Journal*, 336, 1105-1112.
- Linfield, R. P. et al. (1990). 15 GHz Space VLBI Observations Using an Antenna on a TDRSS Satellite. *Astrophysical Journal*, 358, 350-358.
- Meier, D. L. (1994). Space VLBI Scheduling Studies: Techniques and Results. *VLBI Technology: Progress and Future Observational Possibilities*. in press.
- Murphy, D. W., Yakimov, V., Kobayashi, H., Taylor, A. R., & Fejes, I. (1994). Space VLBI Simulations. *VLBI Technology: Progress and Future Observational Possibilities*. in press.
- Preston, R. A., Hagar, H., & Finley, S. G. (1976). VLBI with an Earth-Orbiting Antenna. *Bulletin of the American Astronomical Society*, 8, 497-498.
- Thompson, A. R., Moran, J. M., & Swenson, G. W. (1986). *Interferometry and Synthesis in Radio Astronomy*. New York: John Wiley & Sons.
- Wiercigroch, A. B. (1994). International Data Transfer for Space Very Long Baseline Interferometry (these proceedings).

Mission Management

2. Operations Planning

Page 337

MM.2.a	Geostationary Satellite Positioning by DLR/GSOC Operations and Management Methods <i>Peter Brittinger</i>	339-346
MM.2.b	Magellan Project: Evolving Enhanced Operations Efficiency to Maximize Science Value <i>Allan R. Chevront, James C. Neuman, J Franklin McKinney</i>	347-354
MM.2.c	Grand Mission Versus Small Ops Team: Can We Have Both? <i>Raúl García-Pérez</i>	355-360
MM.2.d	The Role of Mission Operations in Spacecraft Integration and Test <i>Raymond J. Harvey</i>	361-369
MM.2.e	Payload Operations Management of a Planned European SL-Mission Employing Establishments of ESA and National Agencies <i>Rolf Joensson, Karl L. Mueller</i>	371-376
MM.2.f	Evaluating Space Network (SN) Scheduling Operations Concepts Through Statistical Analysis <i>Carl Kwadrat, Nadine Happell</i>	377-388
MM.2.g	SPOT4 Management Centre <i>Yves Labrune, X. Labbe, A. Roussel, P. Vielcanet</i>	389-396
MM.2.h	Costs Optimization for Operations Concepts of Small Satellite Missions <i>Jean-Michel Oberto</i>	397
MM.2.i	Nickel Cadmium Battery Operations and Performance <i>Gopalakrishna Rao, Jill Prettyman-Lukoschek, Richard Calvin, Thomas Berry, Robert Bote, Mark Toft</i>	399-408
MM.2.j	Mission Operations Management <i>David A. Rocco</i>	409-416
MM.2.k	GRTS Operations Monitor/Control System <i>Richard A. Rohrer</i>	417-423
MM.2.l	International Space Station Alpha User Payload Operations Concept <i>Ronald A. Schlegheck, Elaine F. Duncan, William B. Crysel, James W. Rider</i>	425-433
MM.2.m	Shared Mission Operations Concept <i>Gary L. Spradlin, Richard P. Rudd, Susan H. Linick</i>	435-442

* Presented in Poster Session

MM.2.n Mars Pathfinder Mission Operations Concepts

443-450

*Francis M. Sturms, Jr., William C. Dias, Albert Y.
Nakata, Wallace S. Tai*

56

* Presented in Poster Session

GEOSTATIONARY SATELLITE POSITIONING BY DLR/GSOC OPERATIONS AND MANAGEMENT METHODS

Peter Brittinger
DLR / GSOC
Project Manager Geostationary Satellites
82230 Oberpfaffenhofen - Germany

Abstract - Starting with a short description of the GSOC (German Space Operations Center) and its role within the wider framework of the research institute DLR, this paper provides a review of the geostationary telecommunications satellites positioned by the GSOC. The paper then proceeds to describe the evolution of the operations and management structures and methods which have been effectively used to accomplish these missions.

1. INTRODUCTION

During the past 25 years the DLR-German Space Operations Center (GSOC) has operated an extensive variety of satellites. In particular, GSOC has specialized in the positioning and operation of Geostationary Communication Satellites and has successfully delivered and maneuvered eleven geostationary satellites to their on-station positions.

This paper describes the operational and management methods which have developed over the years to support the positioning of geostationary satellites. In particular this paper focuses on the following major topics:

- the role of GSOC within the DLR and its responsibilities in the preparation and execution of national and international spaceflight projects.
- the current track record in the field of communication satellites reaching from the German/French SYMPHONIE program through to the EUTELSAT II F4 mission. The paper discusses the specific features of the different programs and the special requirements that these missions put on the ground segment facilities and staff.

- the management structures of the different programs and its relationship to the GSOC project organization.
- the organization of LEOP (Launch and Early Operations Phase) Services as it is performed at GSOC. This is presented using the example of the EUTELSAT II program.

2. THE ROLE OF GSOC WITHIN DLR

The German Aerospace Research Establishment (Deutsche Forschungsanstalt fuer Luft- und Raumfahrt - DLR) is Germany's largest research institution for the engineering sciences and employs over 4.500 people at seven Research Centers.

Situated on the DLR site at Oberpfaffenhofen near Munich, the German Space Operations Center (GSOC) has over the past 25 years provided services for the operation and support of a wide variety of manned and unmanned space missions. Currently the GSOC is responsible for the German National space program and in addition supports both ESA and NASA activities.

The current generation of DLR spacecraft control systems and facilities have been developed and maintained over the previous 10 years with the specific requirements of multimission functionality. The implementation which has resulted, has proven the strategy to be both flexible and cost effective. This has subsequently enabled the DLR to use essentially the same software, systems and facilities to support a wide variety of spacecraft missions including manned missions, scientific missions and telecommunications spacecraft both in LEOP and routine mission phases.

The experience available through the GSOC is reflected in the wide variety of space missions which have been supported since its establishment and include:

- Geostationary Satellites (SYMPHONIE, TV-SAT, DFS, EUTELSAT II);
- Interplanetary Missions (HELIOS, GALILEO);
- Earth-Orbiting Scientific Missions (AZUR, AEROS, AMPTE, ROSAT);
- Manned Spaceflight Missions (FSLP, SPACELAB D1 and D2, MIR, COLUMBUS);
- Ground Station Support (e. g. GIOTTO, EUMETSAT);
- Sounding rocket programs (ARIES, TEXUS, MAXUS).

Currently the GSOC operates eight control rooms at the Oberpfaffenhofen site. This includes the original facilities and a new complex built to support manned space missions which has been equipped with highly modern facilities and systems. To date this new facility has been used successfully for the D2 mission in May 1993, and the MIR '92' E mission.

Since the start of 1994 the new complex has also been available and utilised for unmanned projects, i.e. conducting LEOPs, routine operations and for the support of scientific missions such as ROSAT.

At the DLR ground station in Weilheim the GSOC also operates two 15 meter S-Band Antennas and one 30 meter X-Band Antenna. In 1996 the facilities will be enhanced by the addition of a Ku-Band Antenna.

3. GEOSTATIONARY SATELLITES POSITIONED BY GSOC

GSOC has been active in the positioning of Geostationary satellites for over 20 years, starting with the first European efforts in this area - the German/French SYMPHONIE program. Since 1974 GSOC has successfully positioned eleven satellites in geostationary orbit, whereby a number of factors are particularly significant :

- In the time between mid 1989 and the end of 1992 a positioning was executed on average every 5 months.
- In mid 1990 three missions ROSAT, DFS-2 and EUTELSAT II-F1 were launched and supported during a three month period.
- GSOC is in the unique position to be able to support launches and transfer orbit injections from practically every type of rocket.

Since 1974 GSOC has been awarded various contracts to position satellites in geostationary orbit, to perform "In Orbit Tests", routine operations and also to support so called "Hot Standby"-operation phases (Figure 1).

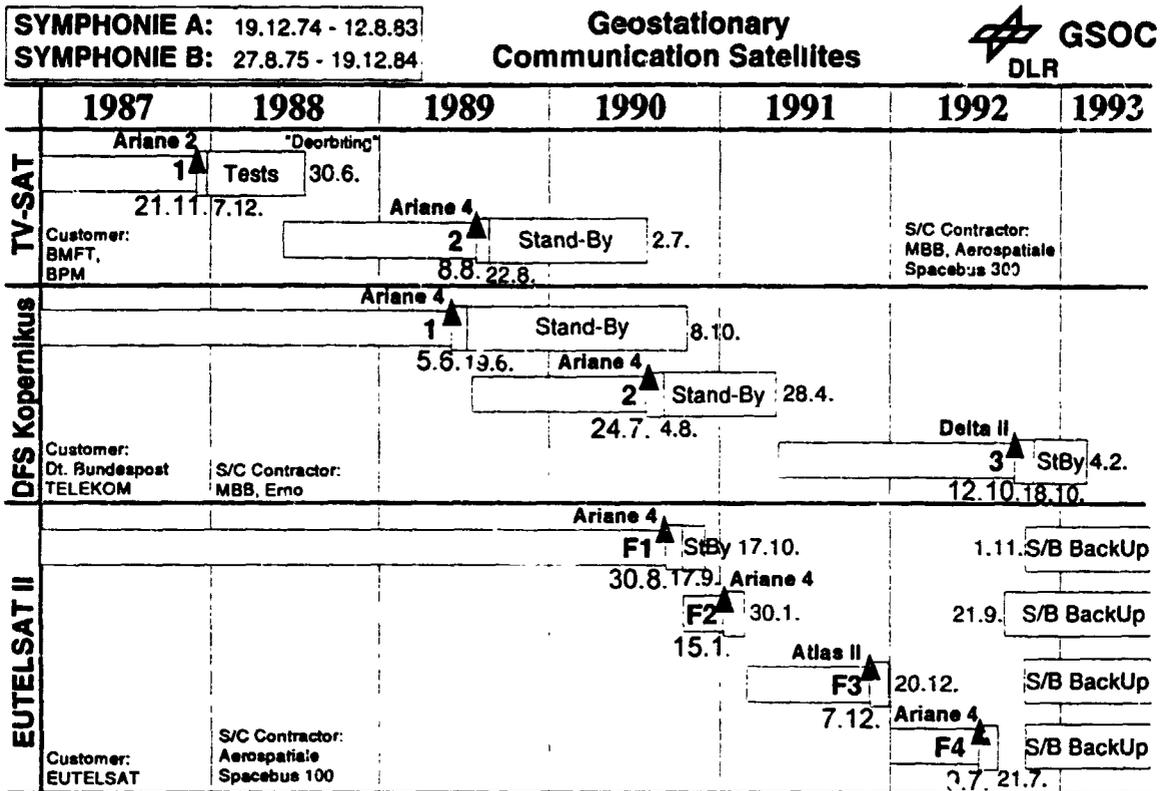


Figure 1: Communication Satellites positioned by GSOC

3.1 TECHNICAL & OPERATIONS ASPECTS

The following paragraphs outline the special technical and operational aspects of the various missions.

SYMPHONIE Program:

- SYMPHONIE A, Launch 19.12.74, DELTA
- SYMPHONIE B, Launch 27.8.75, DELTA

Following the launch, GSOC was responsible for the operations required to place the SYMPHONIE satellites at their dedicated positions in the geostationary orbit. For the first time in Europe, procedures for positioning a 3-axis stabilized geostationary satellite with optimized fuel consumption for routine operations and station-keeping were developed and successfully implemented. The on-station operation was executed by time and work sharing with CNES over a period of 10 years (the Satellite's designed lifetime was 5

years). Another significant factor for these missions was the fact that SYMPHONIE A/B were the first geostationary communication satellites to be brought into the so called "graveyard orbit" using the remaining fuel.

TV-SAT Program:

- TV-SAT 1, Launch 21.11.87, ARIANE 2
- TV-SAT 2, Launch 8.8.89, ARIANE 4 L

The TV-SAT 1 project made high technical and operational demands on GSOC. Due to a technical malfunction of the satellite it was not possible to deploy one of the two solar pannels of the spacecraft. Despite this problem the spacecraft was successfully positioned in its geostationary orbital position of 19° W. During the positioning it was necessary not only to define modified maneuvers, but also to define new test procedures to analyse and find a solution to the problem. The complexity and size of the actual program undertaken was possible

only because of GSOC's existing engineering know how, the flexibility of the equipment used, and the ability to react rapidly to software- and configuration changes. Despite this, it proved impossible to deploy the solar panel and as a direct result the unopened solar panel prevented the operating ability of the Ku-Band Tx-antenna and subsequently any routine operation in Ku-Band.

In the following months TV-SAT 1 was used for test purposes to gather experience for the follow-up missions and at the beginning of M. 1989, the on board thrusters were used to move TV-SAT 1 into a safety orbit 340 km above the geosynchronous orbit.

The TV-SAT 2 project in contrast was a perfect mission. Using routine operational planning and optimisation methods the satellite was put into its geostationary position in a record time of 11 days. Following "In-Orbit Tests" the responsibility for the routine operations was transferred step by step to the Deutsche Bundespost TELEKOM.

DFS Kopernikus Program:

- DFS-1 Kopernikus, Launch 5.6.89, ARIANE 44L
- DFS-2Kopernikus, Launch 27.7.90, ARIANE 44L
- DFS-3 Kopernikus, Launch 12.10.92, DELTA II

After injection into transfer orbit by an ARIANE 44L, the DFS-1 and DFS-2 satellites were positioned in the required geostationary orbit by the GSOC operations team using the classical 3-impulse method. Following the positioning, GSOC executed the In Orbit tests for the Deutsche Bundespost TELEKOM, and subsequently undertook routine operations for a period of one year. Following the step by step transfer of the routine operations to the TELEKOM control center at Usingen, the GSOC team remained in "Hot Standby" for a period of three months, ready at any time to resume routine operations if required.

The launch of DFS-3 with a DELTA II rocket meant a new mission profile when compared with an ARIANE or ATLAS launcher. With the continual development of the maneuver strategies, it was possible to create a maneuver sequence which

allowed positioning to take place in the absolute record time of six days.

EUTELSAT II Program:

- EUTELSATII-F1, Launch 30.8.90, ARIANE44LP
- EUTELSAT II-F2, Launch 15.1.91, ARIANE 44L
- EUTELSAT II-F3, Launch 7.12.91, ATLAS II
- EUTELSAT II-F4, Launch 9.7.92, ARIANE 44L
- (EUTELSAT II-F5, ARIANE failure 24.1.94)

With the positioning of EUTELSAT II-F1 a high standard system for LEOP Services was used. With this system the high level of EUTELSAT requirements were met in particular the redundancy concept. The mission operations experience gained from earlier positioning activities (SYMPHONIE, TV-SAT and DFS) were used effectively and the satellite was positioned within the shortest possible time. In addition specially developed optimizing programs allowed the fuel consumption to be minimized, thus extending the operational life time of the satellite. 17 days after launch the EUTELSAT II-F1 satellite was handed over to the customer for utilization. For a further 4 weeks GSOC was available for "Hot Standby" operations.

During the "Station Acquisition Phase" of the positioning of EUTELSAT II-F2, new strategies and manoeuvres were performed (using specially developed colocation software) in which the satellite flies around the operational control boxes of other geostationary satellites to avoid collisions.

The launch of EUTELSAT II-F3 using an ATLAS II rocket meant a new challenge for GSOC. The satellite was launched into an orbit outside the geostationary orbit (42.000 km). An additional perigee orbit maneuver was necessary, and was performed for the first time. The development of new operational procedures and the continuous development of the maneuver software allowed the GSOC operations team to meet the customer's request to position the satellite within two weeks.

EUTELSAT II-F4 was a normal routine positioning for GSOC. The satellite was handed over to the EUTELSAT Satellite Control Center in Paris after 11 days.

3.2 PROGRAM MANAGEMENT STRUCTURES

This section provides a short overview of the relationship between the program management structures of the overall program and the management structure of the GSOC. The section reviews the interfaces and how the two management structures worked together (Figure 2).

As part of the Ground System Project Group, GSOC was responsible for the German part of the Ground System and thus was responsible for the preparation and execution of the positioning of the satellites and subsequently for the routine operations. In a similar fashion, CNES / Toulouse was responsible for the corresponding French tasks.

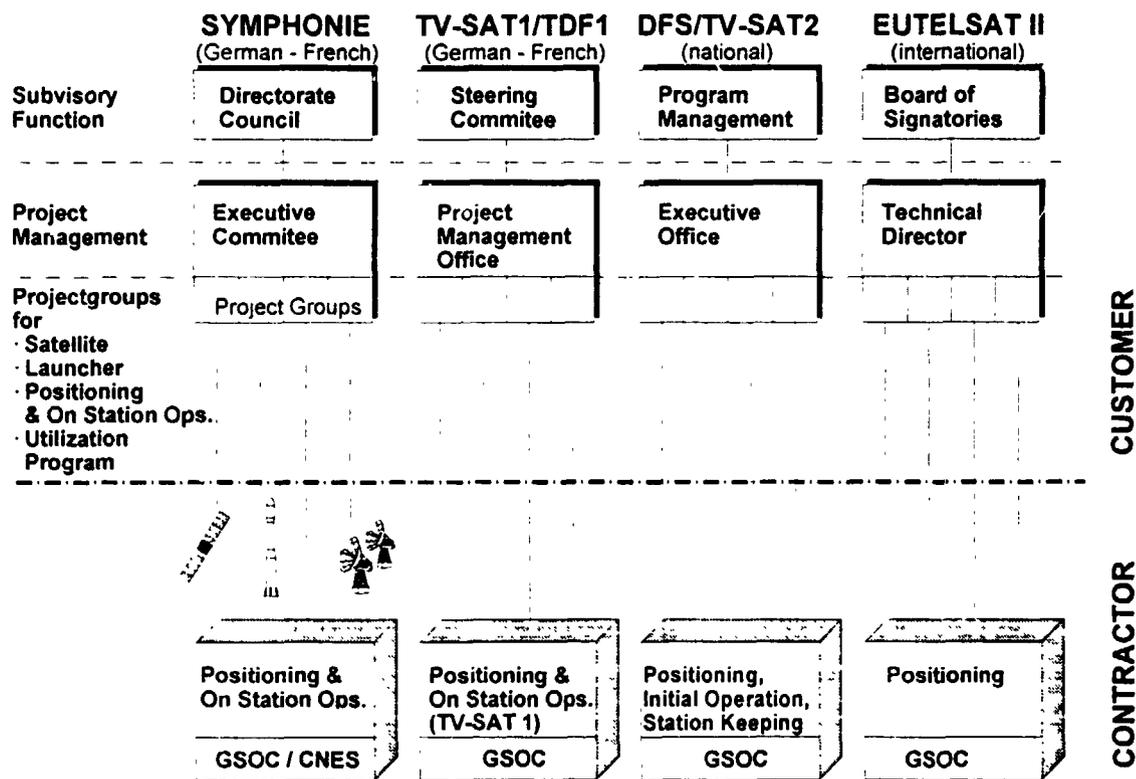


Figure 2: Program Management Structures

SYMPHONIE Program:

The German-French SYMPHONIE program was executed as a joint German French program, whereby all organizational units were manned with a mixture of French and German staff. The project organization was a bilateral management structure with three levels :

- The Directorate (as Subvisory Function)
- The Executive (as Project Management)
- The Project Groups (for specific work items)

TV-SAT 1 / TDF 1 Program:

Similar to SYMPHONIE, the Management Organization of the joint German-French TV-SAT 1 / TDF1 project contained three layers. The general guide lines for the execution of the project were defined by a steering committee. This steering committee was reported to by the bilateral PMO (Project Management Office) underneath which the Project Groups were organized for the actual execution of the project. This project office represented the interface to GSOC for all contractual and technical matters as far as its responsibility for the Ground System was concerned

DFS / TV-SAT 2 Program:

The TV-SAT 2 and DFS projects were national programs and the management lay solely in the hands of the Ministry for Post and Telecommunications (BPM). The program management was created as an Executive office for the management of national satellite systems under the FTZ (Fernmeldetechnisches Zentralamt). GSOC, being responsible for the positioning, had interfaces to the necessary project groups for technical matters, and directly to the Project Management at the FTZ for contractual matters.

EUTELSAT II Program:

The European Telecommunications Satellite Organization EUTELSAT is an international Organization with members in 46 countries. The highest control organization is the Board of Signatories. The project groups and project management report to the General Director who in turn reports to the Board of Signatories. The project management is responsible for the execution of the decisions made by the board, and also for the coordination of the complete program - i.e. also for the "LEOP-Services".

3.3 ORGANIZATION OF THE "LEOP - SERVICES" WITHIN GSOC

This part of the paper uses EUTELSAT II to describe not only the project organization, but also the Mission Operations themselves. The organization structure shown in figure 3 has been implemented since TV-SAT and has been successfully used for all follow-up programs. Within this project organization the DLR key persons are responsible for both the preparation and also for the execution of the mission.

The "LEOP-Services" Project Manager has the overall responsibility not only for the preparation phase (ground segment implementation), but also for the completion of the positioning where he performs the role of Mission Operations Director (MOD).

The Project Manager is supported by a Project Administrator primarily for project control but also for financial and contractual matters. In addition the project management is supported by an independent Quality Assurance Manager.

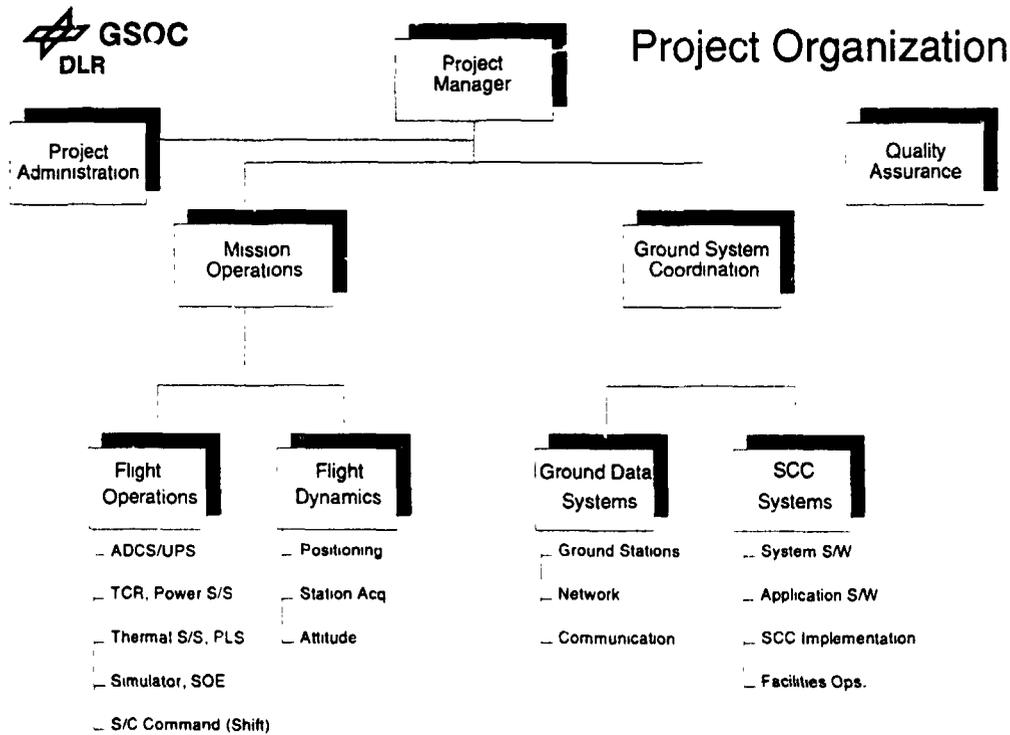


Figure 3: DLR / GSOC Project Organization for LEOP Services

Reporting directly to the Project Manager is the Mission Operations Team Lead (MOTL). Together with the Ground System Coordinator, the MOTL coordinates the work of the Team for the preparation and execution of a mission.

The DLR Mission Operations Team is created from Satellite and Ground System specialists who are allocated to the project from the various specialist departments of the DLR. This team of experienced Flight Dynamics, Flight Operations and Ground System Engineers is responsible not only for the preparation, but also for the execution of the mission.

As a direct result of this change to the structure, the project achieved a strongly subject orientated organization which provided better monitoring of the mission preparation and execution. The internal project monitoring with respect to the Mission Operations Team was much improved. The Mission Manager was hence released from this task and was able to focus his activities on the flight operations for the space segment. This becomes more important during the execution of the mission as the Mission Operations Team Leader is closely involved in the decision making process of the mission execution (see also Figure 4)

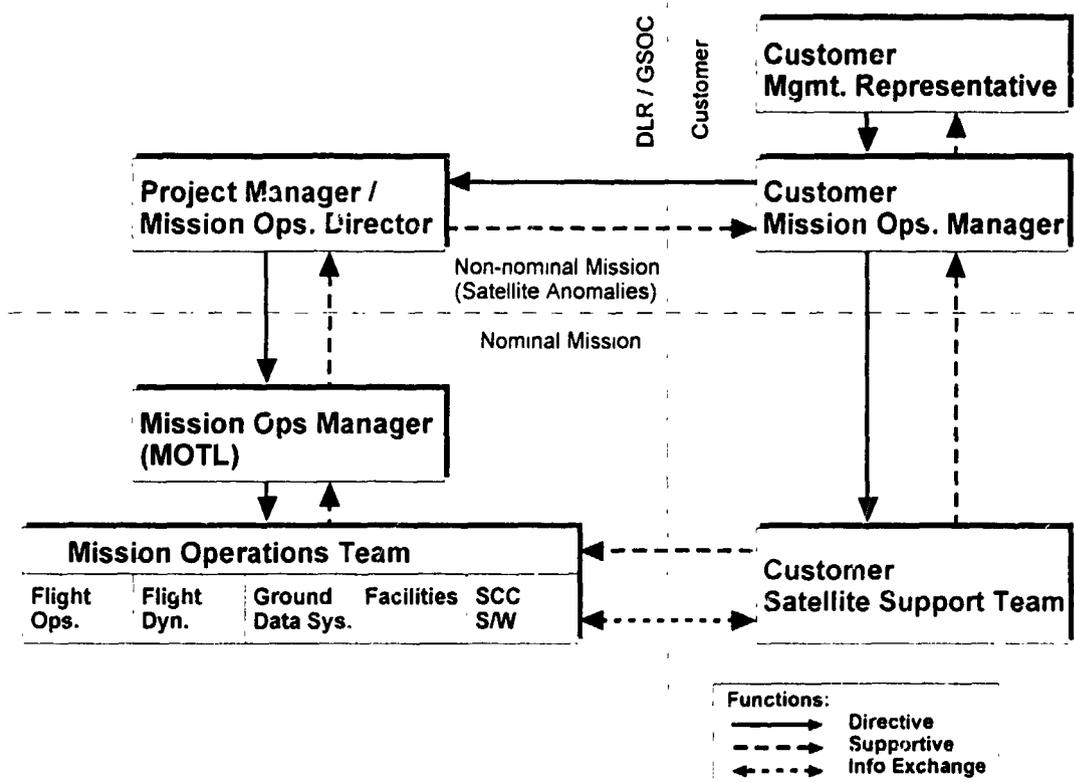


Figure 4: Organization of Mission Operations

Experience gained during the execution of the EUTELSAT II project allowed the refinement of the management structure, and more specifically determined the need for a Ground System Coordinator at the project level who provided direct assistance to the Mission Operations Team Leader.

The organigram (Figure 4) shows the principals of the organization of Mission Operations, whereby the interfaces for operational matters between the customer and contractor are also portrayed. Figure 4 shows how a positioning would be executed in close cooperation between GSOC and the customer.

As the Contractor, GSOC manages the mission under the control of the Mission Operations Team, lead by the Team Leader (MOTL). The mission is executed according to the Flight Plan which includes all nominal satellite operations together with a selection of predefined contingency procedures.

The customer is represented by the Customer Management Representative, Customer Mission Operations Manager and Customer Satellite Support Team.

The Customer Management Representative is the on site customer representative. He is responsible for the regulation of all mission related tasks including contractual matters as far as they relate to the responsibilities of the Customer Mission Operations Manager.

The Customer Mission Operations Manager follows the execution of Mission Operations, authorises the execution of emergency procedures and gives directives in the case of non nominal behaviour of the satellite. He is the only person who is authorised to give directives to the the DLR Mission Director or to the Customer Satellite Support Team.

The Customer Satellite Support Team, which is created from experts from the customer and the satellite manufacturer monitors the execution of the mission and compares the actual with the expected behaviour of the satellite. In case of non-nominal of the satellite, the Satellite Support Team provides the Mission Operations Team with inputs to correct the failure.

If the Mission Operations Team or the Satellite Support Team determine non nominal behaviour which is not covered by the Flight Plan, a special procedure has to be produced to cater for this behaviour. These special procedures are regarded as extensions, changes or adaptations to the existing Flight Plans and are produced in the form of "Recommendations". After release by the Customer Mission Operations Manager and the DLR Mission Operations Director, they are passed to the Mission Operations Team Leader (MOTL) for execution.

4. SUMMARY

Starting with SYMPHONIE, GSOC has been careful to systematically review and update the operational and management procedures and methods applied to positioning projects.

This approach has allowed the development of a set of standard geostationary positioning procedures and working methods which are optimised for modern communication satellites. These procedures and working methods have proved themselves during successful positioning activities.

From SYMPHONIE to the current series of EUTELSAT II spacecraft, the experience gained and retained over many years has been continuously used to both improve the ground operations facilities and also to enhance the operational capacity of GSOC specifically in the domain of geostationary satellite operations.

GSOC has proved its capability to adapt a variety of technical and management constraints as well as different contractual relationships.

The LEOP team at GSOC is able to react quickly and effectively to the most varied customer requests in a responsive and unbureaucratic fashion.

In this way the GSOC is in the position of being able to adapt its systems and operations to support practically any customer and any spacecraft manufacturer.

REFERENCE:

- [1] P. Brittinger, DLR/GSOC, *Experience in the Field of Geostationary Satellites, International Symposium on Spacecraft Ground Control and Flight Dynamics, Brazil February 1994*
- [2] P. Brittinger, DLR/GSOC, *Positionierung von geostationären Nachrichtensatelliten, DGLR/DLR-Symposium, Köln/Germany April 1994*

E 4.7
P. 8

**MAGELLAN PROJECT:
Evolving Enhanced Operations Efficiency
to Maximize Science Value**

Allan R. Chevront
James C. Neuman
J. Franklin McKinney
Martin Marietta Astronautics
Denver, CO

ABSTRACT - Magellan has been one of NASA most successful spacecraft, returning more science data than all planetary spacecraft combined. The Magellan Spacecraft Team (SCT) has maximized the science return with innovative operational techniques to overcome anomalies and to perform activities for which the spacecraft was not designed. Commanding the spacecraft was originally time consuming because the standard development process was envisioned as manual tasks. The Program understood that reducing mission operations costs were essential for an extended mission. Management created an environment which encouraged automation of routine tasks, allowing staff reduction while maximizing the science data returned. Data analysis and trending, command preparation, and command reviews are some of the tasks that were automated. The SCT has accommodated personnel reductions by improving operations efficiency while returning the maximum science data possible.

MISSION OBJECTIVES - The objectives of the Magellan program were to place a spacecraft with a radar sensor in orbit around Venus; obtain, reduce, and analyze the scientific data from the planet and make these results available to the public and the scientific community. The Magellan scientific mission objectives were:

1. To improve the knowledge of the tectonics and geologic history of Venus by analysis of the surface morphology and the processes which control it.
2. To improve the knowledge of the geophysics of Venus, principally its density distribution and dynamics.
3. To improve the knowledge of the small scale surface physics.

The objectives of the science experiments were:

1. Imaging: To produce contiguous images of at least 70 percent (with a goal of 90 percent) of the surface of Venus with no

systematic gaps except for one pole, and with a surface radar resolution of at least 260 meters (surface radar resolution is defined as the distance between the 3dB points of the main lobe of the radar system impulse function).

2. Altimetry: To produce maps of the topographic and radar scattering characteristics of the planet Venus with height resolution commensurate with the Synthetic Aperture Radar (SAR) range resolution and coverage commensurate with the SAR coverage.
3. Gravity: To refine the low degree and order gravity field of Venus and to produce high resolution (several hundred kilometer horizontal scale) gravity maps wherever possible.

HISTORY - A single Magellan (MGN) spacecraft was launched from Kennedy Space Center Launch Complex 39B on May 4, 1989, on board the Shuttle Atlantis. The launch vehicle was a Shuttle Orbiter/Inertial Upper Stage (IUS) combination. Once in the shuttle parking orbit, the IUS and MGN spacecraft combination was deployed from the cargo bay. After the orbit coast period, the IUS injected the MGN spacecraft into an Earth-Venus transfer trajectory.

The MGN spacecraft is powered by two single degree-of-freedom, sun-tracking solar panels and is three-axis stabilized by reaction wheels using gyros and a star scanner for attitude reference. When launched, the spacecraft carried a solid rocket motor for Venus orbit insertion. A small hydrazine system was provided for trajectory correction and certain attitude control functions. Earth communications with the Deep Space Network (DSN) is by means of S- and X-band channels, operating via low- and medium-gain antennas and a 3.7 meter high-gain antenna dish which is rigidly attached to the spacecraft. The high-gain antenna also functioned as the Synthetic Aperture Radar (SAR) mapping antenna during orbital operations.

Magellan followed a Type IV interplanetary trajectory to Venus, which represents a transfer angle around the Sun of slightly greater than 50 degrees. The use of the Type IV trajectory was required by the orbit geometry of Earth and Venus for the May 1989 opportunity. The Type I trajectory opportunity in October 1989 had been allocated to Galileo for its Venus-Earth-Earth Gravity Assist trajectory to Jupiter.

The interplanetary cruise phase lasted approximately 15 months. Cruise activities included calibrations of the gyros, the attitude reference unit and the antennas, daily star calibrations, three trajectory correction maneuvers to insure proper approach geometry, a functional test of the radar subsystem, and a three-day test of the mapping capability.

MC¹ arrived at Venus on August 10, 1990. By firing the solid rocket motor slightly before Venus closest approach, the desired periapsis latitude near 10 degrees North was attained. The spacecraft was placed in an elliptical orbit around Venus with a period of 3.26 hours. The planned in-orbit checkout (IOC) period was cut short because a timing idiosyncrasy in the on-board Attitude and Articulation Control flight software caused the spacecraft to enter safing during the star calibration of the second test mapping orbit. The imaging data processed from the 1.5 test orbits prior to safing was of such high quality that the project decided to terminate IOC following the safing recovery and enter directly into mapping operations.

The prime mission (Cycle 1) began on September 15, 1990, and lasted 243 days, the time required for Venus to make one rotation under the spacecraft orbit. Cycle 2 started on May 15, 1991, and Cycle 3 started on January 15, 1992, and continued to September 15, 1992. Typical activities during the radar mapping orbits are shown in Figure 1. Mapping operations were halted after the third cycle due to a transmitter failure. The science emphasis shifted to the third science experiment, gravity data. Cycle 4 was devoted to collecting gravity data and planning for aerobraking operations.

Following Cycle 4, Magellan's periapsis was lowered to place the spacecraft in the atmosphere each periapsis pass in order to slow the spacecraft and nearly circularize the orbit. Magellan was the first planetary spacecraft to use aerobraking to change its orbit. Since aerobraking had not been

accomplished before and the spacecraft was not designed to aerobrake, the planning tasks broke new ground. Aerobraking was successfully accomplished in seventy days (planned for eighty) and placed the spacecraft into a 500 km by 200 km orbit. Cycles 5 and 6 have collected high resolution gravity giving scientist their first view of the subsurface at the poles. In Cycle 5 and 6, Magellan has also performed several Radio Science Occultation, Bi-static and Quasi-specular Radar experiments.

Magellan's accomplishments include: mapping over 98% of the planet surface, obtaining high resolution gravity data over 95% of the surface, successfully accomplishing aerobraking to change the orbit, and performing several other experiments (Radio Science Occultation, Bi-static, Quasi-specular and Windmill).

OPERATIONS PROCESS - Prior to launch, the mission operations procedures and plans for the Spacecraft Team (SCT) were developed on the premise that the orbit would be repetitive and day-to-day tasks would thus be simple and low cost. The majority of the tasks (approximately 70%) would involve analyzing and trending the engineering data received from the spacecraft, while the remaining time would be for the command process. The data analysis and trending workload estimate was based on previous planetary spacecraft operations that relied on simple displays and manually analyzing engineering data. For Magellan, the spacecraft data analysis and trending would be performed using the new JPL multi-mission operations process and multi-tasking workstations, but trending was still perceived as a manual process.

The command process was developed using sequences based on repetitive mapping orbital operations, intending to minimize the effort required for commanding. The mission planning and sequence development processes relied on meetings, paper products and manual reviews similar to previous spacecraft's. Rather than the simple process envisioned, the command process resulted in complex operations that were continuously tweaked to improve the science data quality.

The mission progressed as planned and all flight sequence milestones were met. However, the allocation of the work force to accomplish the tasks was radically different. Immediately after

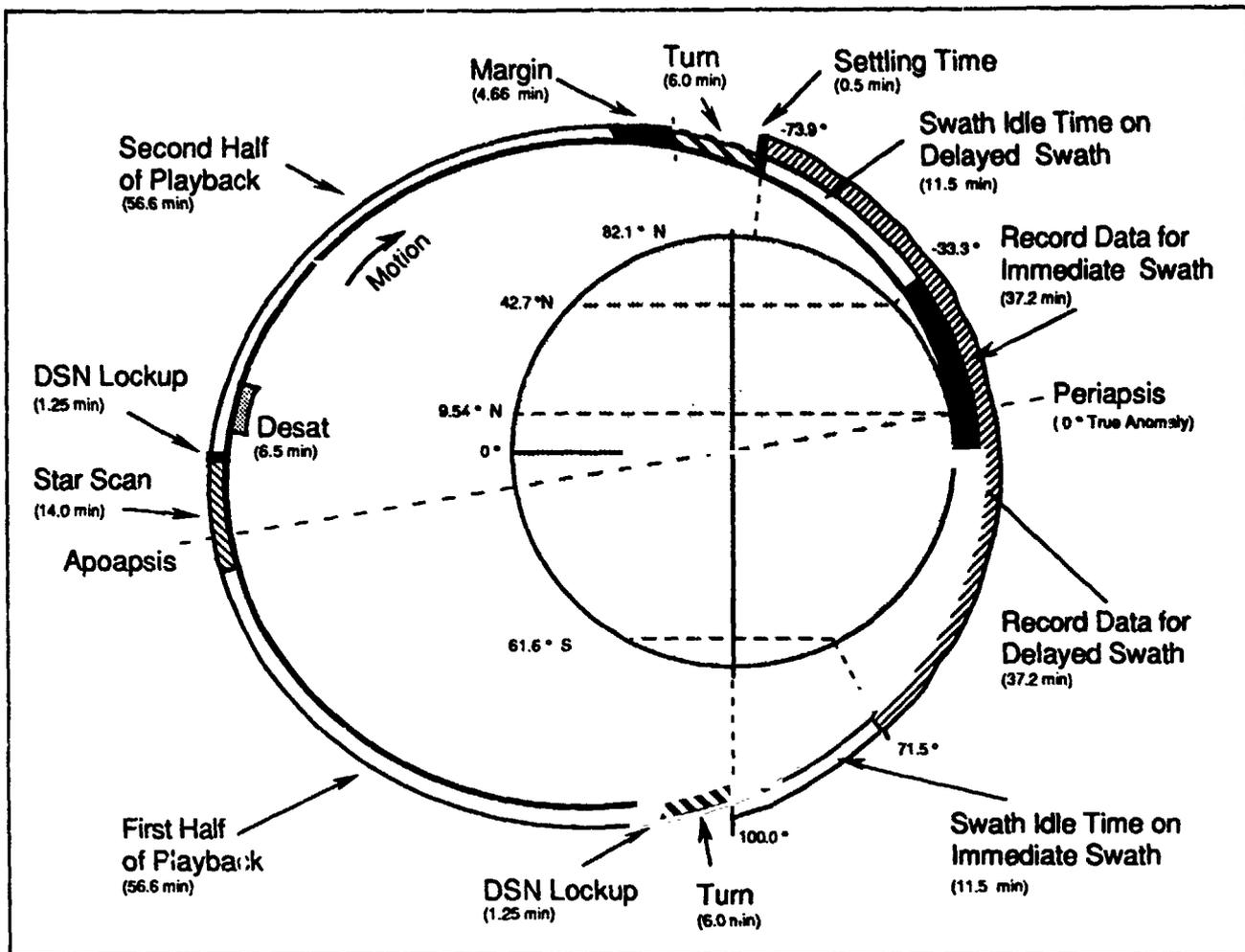


Figure 1 Typical Mapping Activities

launch, anomalies occurred on the spacecraft which required the subsystems engineers to spend an unexpected larger portion of their time performing sequence preparation and validation. The difficulty of operating and maintaining the health of the spacecraft early on (thermal control and star scan problems) combined with a labor intensive command process resulted in a very high percentage of the work effort being placed on commanding. This percentage decreased as the mission operations matured, but the final workload split was approximately seventy percent involving commanding and the remainder for data analysis and trending.

The SCT, faced with a labor intensive command process, a strong desire to obtain higher quantity and quality science data, and budget constraints realized the need to further reduce the time required for the spacecraft data analysis and trending. Using the framework of

the multi-mission operations tools and the processing capability of the workstations, the SCT developed batch scripts and other software routines that allowed the spacecraft data to be collected, analyzed, and displayed automatically. The data analysis and trending were performed during off hours and the results were available for the engineers to examine when they arrived in the morning. If the results were nominal, the engineers could then devote their time to the command process. If an anomaly was present, their time was used to determine the cause and corrective action, which generally involved more commanding.

The automation of the data analysis and trending was achieved because the tools allowed these tedious tasks to be performed by computers. In addition, the SCT was comprised of engineers with the necessary computer skills to automate tasks and encouraged by program management to

perform continuous process improvement. The automation process was further enhanced by the SCT conviction that the mission operations budget must be reduced if an extended mission was to be affordable. The planning for the successful Aerobraking operations accomplished during Cycle 4 was made possible by the mission operations savings during the earlier cycles. These savings were the direct results of the automation and improvement process.

MAGELLAN COMMAND PROCESS -

The Mission Operations Plan was for spacecraft sequences to be developed using a standard process (initially a twelve week duration) which relied on stored commands to perform the tasks necessary to obtain science data. In addition to mapping commands, health and maintenance commands were also to be included with the standard (stored) sequence. The standard command process utilized repetitive commands each orbit with minor periodic parameter updates to minimize operations costs.

However, this simplicity was not realized due to the program's desire to maximize the quality of the science data. To improve the science data, the frequency of parameter updates had to be increased. Additional complexity arose because of the need to manage thermal control and star scan issues. Prior to Venus Orbit Insertion, the SCT realized that the plan to send a new mapping sequence every week would be difficult to achieve at the current staffing level despite the time savings from automation of the data analysis and trending. The creativeness of the engineers was allowed to manifest itself, resulting in an extended sequence (two weeks) and manageable parameter updates the staff could accommodate while still obtaining the highest quality science data.

As spacecraft anomalies developed and operating idiosyncrasies became apparent, non-stored (called non-standard) commands were required to meet maintenance issues, investigate anomalies and conduct characterization tests. These non-standard commands were developed using the standard command process but could not be placed in the stored sequence due to their near real time nature.

Standard Command Process - As shown in the orbit profile (Figure 1), the mapping data collection and playback required the spacecraft to perform six maneuvers each orbit. In addition to developing the commands to maneuver the spacecraft, commands (Figure 2) were required to

control the radar mapping parameters, manage the tape recorders, perform desaturations of the reaction wheels, and manage the telecommunications system. A typical orbit had over a hundred separate commands to perform.

In addition, the SCT had over a thousand variables in flight software to track and maintain. In order to perform the mapping mission, mapping and flight software parameters were stored on-board. Mapping parameters included the orbit's periapsis time, radar parameters tailored for the upcoming terrain and a mapping quaternion polynomial coefficient file required to constantly change the mapping attitude. Flight software parameters were the star scan parameters and gyro bias and scale factors. Also updated each sequence load were safing parameters for possible use by the fault protection system. This complexity was underestimated prior to launch and combined with non-standard commanding contributed to the increase in effort required for the standard process.

The original plan, once on orbit, called for a new sequence of commands to be uplinked to the spacecraft every week. Before Venus Orbit Insertion (VOI), the plan was changed to uplinking a sequence every two weeks because it was realized the program could not support the workload involved to develop and review a sequence each week. Each standard two week sequence took approximately 12 weeks to develop which meant that the SCT was working on up to six sequences at a time. This workload was labor intensive due to the amount of time required to: generate parameters; develop and review three cycles (preliminary, intermediate and final) of Sequence Events Files (SEF); review other uplink products; and perform a test in the System Verification Laboratory. The standard sequence cycle was marked by meetings, reviews, and reams of paper. The amount of time spent in meetings was also underestimated. Meeting time included presentation preparation, future sequence planning, reviewing developed sequences, and presenting subsystem performance. In Cycle 1, it was estimated that a typical subsystem engineer could spend twenty to twenty-five hours per week in meetings. Adding to the complexity was tracking six sequences at once, ensuring the right parameters were developed and coordinating activities between sequences.

After the successful completion of Cycle 1, the number of people on the SCT slowly

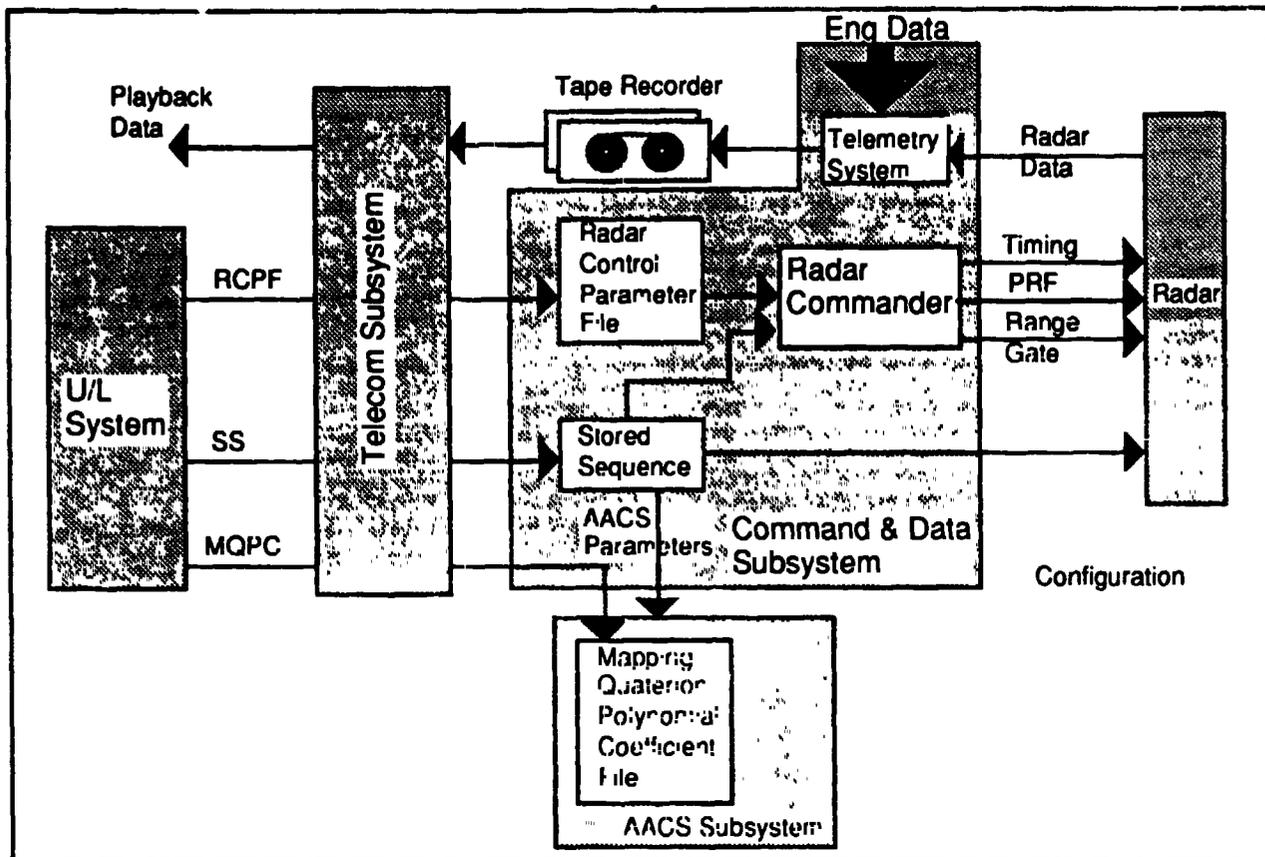


Figure 2 Typical Mapping Commands and Flow

decreased as personnel left the program. It was apparent that the existing standard process could not be supported with the smaller staff. A revised standard process was developed which eliminated the intermediate SEF products and took advantage of further automation to reduce time for command development, generation, and reviews. The new process took six weeks to produce the uplink commands thus reducing the number of sequences in work to three. The new process also reduced the amount of time required to prepare products by eliminating non-value added traditions such as management approval of technical parameters. To help achieve the reduced schedule, standard spacecraft maneuver times were developed which would reduce the analysis required for each sequence. The standardized maneuvers were never fully realized since spacecraft anomalies required each sequence to be as unique as in the first cycle. The new standard sequence process reduced the number of meetings, automated reviews and consumed less paper since electronic versions of uplink products were used. This six week process continued

through the third mapping cycle when mapping operations were halted due to failed transmitters.

The fourth cycle was a gravity only cycle and saw a reduction in the amount of work required because the mapping associated parameter development and reviews were not necessary. The program moved to three week sequences which meant only two sequences were in work at any one time since the standard process duration remained at six weeks. This allowed a smaller work force (thirty people) to continue to develop standard sequences and non-standard commands and prepare for aerobraking operations at the completion of the fourth cycle.

The program was presented with an opportunity to obtain high resolution global gravity data by nearly circularizing the orbit through aerobraking. Aerobraking the spacecraft was a high risk endeavor since it had never been attempted before with a planetary spacecraft. In addition, aerobraking was to be performed with a spacecraft that was not designed for it. The program engineers had to develop the aerobraking profile (attitude and duration) and commands for

performing aerobraking. The existing mapping block did not meet the requirements, therefore a new aerobraking block had to be developed and tested. This full time effort consumed approximately half of the SCT which meant the remaining half performed the nominal tasks to obtain the important gravity data. Aerobraking operations were developed to ensure that the necessary timing updates (to account for the shrinking orbit) were sent to the spacecraft in a timely manner. At the start of aerobraking, timing commands were sent to the spacecraft three times per week. As the orbit period shrunk, the timing commands were sent every day. At the end of aerobraking, timing commands were sent up to three times per day. Aerobraking was accomplished in seventy days, ten days ahead of schedule.

Following aerobraking, process improvements continued with a change in the way sequences were implemented to take advantage of the near circular orbit. The length of the sequences was increased to three weeks which meant the SCT was working on one sequence at a time. It was these types of improvements that allowed the SCT to collect high resolution gravity data and perform special experiments (Radio Science Occultation, Bi-static, Quasi-specular and Windmill) with a significantly smaller staff.

Non-Standard Command Process - Initially there was no set process to send a non-standard command to the spacecraft. Each subsystem would determine the need, develop the commands, and then present the results to the Mission Director for approval. This resulted in significant re-work as an alternative solution would surface during the presentation to the Mission Director. The alternative solutions led to confusion which resulted in a high number of command related incidents. The non-standard workload was a significant portion of the commanding process because solutions to problems were often re-worked several times.

Several months prior to Venus Orbit Insertion, the Proposed Engineering Change (PEC) process was developed. The PEC process brought discipline to the non-standard commanding effort and reduced the number of command incidents to near zero. The PEC process is started when a subsystem engineer completes a PEC form which describes the reason for the proposed change, the impacts if not implemented, the need date, and alternative

solutions. The PEC is then reviewed by the members of the SCT, updated and then presented to the Mission Director at the Engineering Review Board (ERB) for approval. If approved, the SCT is then authorized to implement the proposed solution and send the non-standard command(s) to the spacecraft. By holding a peer review, impacts to other subsystems are identified as well as better solutions not considered by the originator. This process forced a disciplined thought process which proved invaluable during anomaly recoveries. Over 270 PECs have been written. Of these, only sixteen have been disapproved; the majority were early in the mission as a result of conflicting requests. The small number of disapproval's indicates the PECs brought forth viable solutions in which all members of the SCT and program management concurred. Management developed confidence in the SCT and the solutions to anomalies due to the discipline created by the PEC process.

An example of process improvement is the Express Command. Express Commands are commands that have minimal impact to the spacecraft and were being presented as a PEC on a regular basis. Examples of express commands are memory read out, star scan parameter changes, and turning transmitter sub-carriers on or off. Express Command eliminated the repetitive workload of regularly presenting PECs to the ERB. One PEC was created that defined who could send a command, the conditions in which the command could be sent and the follow up action. Prior to Express Command every command had to be approved by the Mission Director. Now these commands required only approval by the appropriate subsystem, thus empowering the engineers.

REMOTE OPERATIONS - Magellan was the first JPL spacecraft to be flown from a remote location. This posed the problem of how to effectively communicate without face-to-face contact with the other person. The remote arrangement also required that JPL management give up much of its "routine control" over the SCT. By remotely locating the SCT, engineers with more Magellan spacecraft experience were enticed to support mission operations. All subsystem had team members who had been part of all phases of the program. If these engineers had been required to relocate to JPL, many of these experienced individuals would not have been part of the SCT.

The voice communication problem between the remotely located project teams was difficult to solve. Initially the teleconferencing capability was minimal (one speaker phone in a small conference room). Prior to VOI, a large conference room with multiple microphones was made available which improved the technical portion of the voice communication problem. Although both the Denver and the JPL sides worked very hard to effectively communicate, problems still existed. One of the main difficulties was understanding the other side's everyday workload problems. To maximize this understanding, representatives from both sides would travel to the other's facilities on a regular basis. These representatives were usually the Leads of various subsystems/teams. By spending one week every three months at the other person's location these leads developed an appreciation for each other's constraints and abilities. This rotating representative eliminated the belief that the other side "had it easier".

STAFFING - The success of the Magellan Venus Radar Mapping mission has been largely the result of the outstanding performance of the flight system, however, some credit must be given to the mission operations team and the staffing plan. The staffing plan included the

selection of the right people, the organizational structure at the beginning of the program, and systematic downsizing as the program matured. Much of the extended mission operations, including aerobraking, would not have been possible without significant reduction in the size of the SCT. The original plan provided for thirteen engineers monitoring the health of the spacecraft. As the program developed, it became apparent that the simple flight system developed as a low cost solution for the original VOIR program would involve a very complex mission operation if the Magellan science return was to be realized. In addition, the flight software and the fault protection system proved to be extremely complex and their verification and characterization continued during the Cruise Phase to ensure its readiness to support VOI and Mapping Operations.

At launch the SCT had sixty people organized as shown in Figure 3. This number grew to seventy as VOI approached. As mapping operation settled into a routine, the staff level was reduced to fifty by the end of the Cycle 1 and remained steady until the end of Cycle 2. At the end of Cycle 2, spacecraft telecommunication transmitter A's failure caused a major re-planning effort. The resulting potential funding cutoff

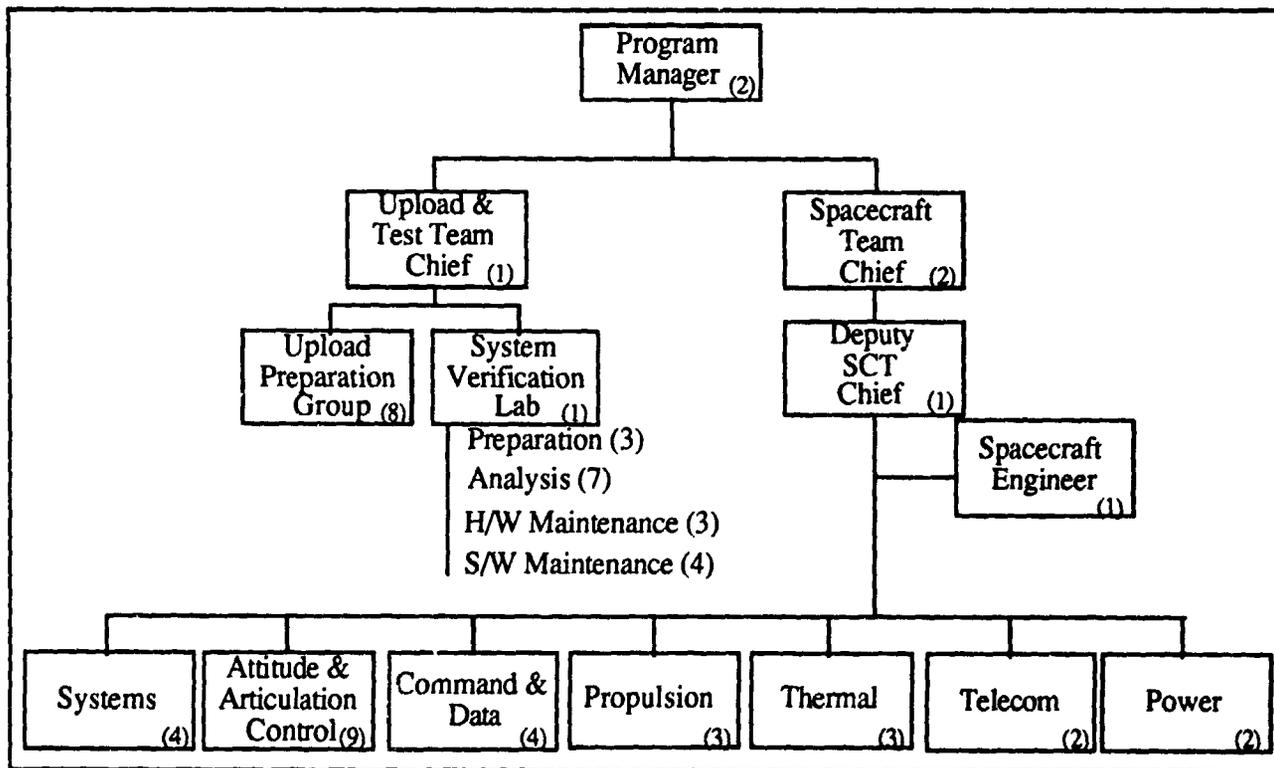


Figure 3 Spacecraft Team at Launch

encouraged the program to look for cost reductions to keep the mission alive. The process of staff reduction while maintaining team morale and productivity and continuing the science mission was a major challenge. This challenge was accepted by the team because of a fundamental belief that a continuation of the mapping mission would yield outstanding science results, but would only be possible if the size of the team and the resulting cost could be significantly reduced. The team size reduction was a product of recommendations and brainstorming sessions by the whole team. Any reduction in staffing was, therefore embraced by the team regardless of its impact on an individual.

The staff reduction effort continued until thirty people remained at the start of Cycle 4. The staffing leveled out at thirty people as the team continued mission operations and planned and conducted aerobraking. After the successful completion of aerobraking, the staff was reduced to fifteen people. At the end of the mission, the SCT was comprised of nine people some of whom were part time.

The staff reduction was accomplished in a positive manner since career growth opportunities on other contracts were available to nearly all of the team members with special skills gained from the Magellan experience. As these engineers left the program, the organization was restructured and/or the roles and responsibilities were distributed among the remaining team members. This process of "belt tightening" gave more responsibility and growth opportunities to the remaining staffing and had a positive impact on the overall team morale despite the ongoing staff reduction. The loss of senior staff did not significantly impact operations because remaining engineers were ready to assume their responsibilities. The automation process that was occurring simultaneously with the staff reduction enabled the available resources to return the maximum science data possible.

Management played an important role by identifying and keeping those individuals who could perform multiple tasks and/or encouraging individuals to become proficient in multiple tasks. This identification process was achieved by providing opportunities for engineers to excel. Throughout the program, management was not satisfied with the status quo. Instead, management encouraged the Leads to do more with less, so when funding faced reduction the

SCT was able to respond quickly with proposed staff reductions.

CONCLUSIONS - Software automation, process improvement, the management environment and a cooperative flight system are the main reasons Magellan has enjoyed such great success. The management philosophy created an environment of continuous process improvement that allowed the SCT to perform a wide variety of tasks with a steadily decreasing staff.

The areas that realized time saving due to automation were sequence preparation, sequence validation, and data analysis and trending. Sequence preparation saw significant savings through the electronic transfer of parameters and automating command generation procedures.

Sequence validation automation was achieved by the creation of software tools designed to replace the manual checks of the command products. Each subsystem had its own checklist that contained the steps necessary to manually validate the command products. As the engineers gained confidence in the checklists, software tools were written to perform the manual checks. This reduced the time to review a typical sequence from eight hours to two hours.

Data analysis and trending were also automated through the use of software and workstation tools. The primary source of automation was the generation of programs and scripts to carry out repetitive tasks that the engineers were required to perform each day. Many of these tasks were not completely characterized prior to launch, so the creation of software tools during spacecraft development was limited. After launch, when the spacecraft performance and ground data systems capabilities were better understood, each subsystem engineer produced a tool set that allowed them to perform their jobs more efficiently. It is important to note that the software coding and script writing was performed by the subsystem engineers and not a software development staff. This created the scenario where the end user was also the programmer, so the tools developed meet the needs without significant interaction.

ACKNOWLEDGMENTS - This work was performed by Martin Marietta Astronautics in Denver, Colorado while under contract to the NASA Jet Propulsion Laboratory for Magellan spacecraft operations.

GRAND MISSION VERSUS SMALL OPS TEAM CAN WE HAVE BOTH?

Raúl García-Pérez
Ulysses D. S/C OPS Manager
ESA/ ESOC
(ESA Operations Centre, in Darmstadt, Germany)
Ulysses office located at NASA/ JPL
4800 Oak Grove Blvd.
Pasadena, CA 91001, USA

ABSTRACT. - Space Missions are growing more ambitious, but resources are getting smaller. Is this a contradiction in terms, or is it a healthy challenge?

This paper offers the author's point of view as a member of a small Mission Operations Team that carries out an ambitious international mission (Ulysses ESA/NASA).

Table of contents:

- Introduction
- Today's facts
- How did we get here?
- Summary of our situation
- Objectives
- Alternatives
- Risks and problems
- Recommendations
- Conclusion

INTRODUCTION

So... Can we have both?

Right or wrong, the answer to this is being written by all of us, the people who work in the Space Business. We make the choices and in so doing we define the future.

In the other hand nobody is absolutely free to shape history. Forces like the economy and the development of the technology *invite* us to take certain decisions.

Actually it seems that we are at the same time invited and decided to have bigger but cheaper missions. Maybe the relevant question is no longer whether it is possible or not but:

How are we going to do it?

The following discussion will help us to answer this question. After all, we are the problem solvers in this game and this is a good place to talk about our solutions.

TODAY'S FACTS

Space and the economy

Space is nowadays a precious economic resource and the number of services it offers is increasing every day.

The missions that provide a commercial service grow as big as the market requires them to grow. They tend to use well-proven technologies and to build spacecraft in a production-line fashion.

The non-commercial missions tend to grow smaller under the economic pressure. Like the dinosaurs they disappear or evolve into birds under the pressure of the environment.

In the other hand we need the non-commercial missions to expand our knowledge. This tends to increase their mission cost, because it constitutes a bigger challenge than following the paved way.

Fast technology development

The new technologies may multiply the possibilities or lower the cost, but their novelty involves a greater risk.

Technology acts frequently as a hidden agenda. The conservative side of the project will defend the Mission Objectives above all, while some groups will be very motivated to develop a particular technology. This is not necessarily bad, because the new technologies are a desirable product of the space activities.

Man versus Machine

Here is another controversial issue, in which there is a case for either side.

Machines are more accurate, but they lack many human virtues. Robots are cheaper to fly, because they do not need life support. To

reduce the man power on the ground, we use artificial intelligence that is not cheap, but its development is an attractive hidden agenda.

In any case, we humans have an exploring heart and we cannot help to be part of the space exploration endeavor. This emotional imperative seems to be a key ingredient of progress because it generates motivation, which is essential for the future of any business.

HOW DID WE GET HERE

It is well known that the Space research started during the cold war. Now we are in the post cold war and the base of our economy is changing. Unfortunately fear to one another is a big incentive for the research and the economic growth; it is not surprising that we have lost some steam in this change process.

In the mean time we have become accustomed to re-direct part of our energy towards space exploration and to the business opportunities thereof. We should hope that this challenge would help us to substitute war as the prime incentive to advance science and technology.

SUMMARY OF OUR SITUATION

Like in the legend of Ulysses, we are caught between Scylla and Charibdes (a narrow strait between two opposite rocks). Nevertheless, there is nothing like a good challenge to make people think harder.

Before we talk about the way out, it would be a good idea to discuss what are our objectives.

OBJECTIVES

Do more and to do it for less

This is in simple terms the key question. It is possible but we must be aware that it requires significant cultural changes; we cannot expect to do it for less just following the traditional rules. We also must accept that changing the rules involves some risk.

Maintain people's motivation

People want to know, why should we pursue space research?

We need to conduct our missions in a way that satisfies people's needs and appeals to people's participation. We will not go very far if we do not take into account the fact that customers and professionals are just people

Insure operational flexibility

Space Operations is about: having options, knowing them well, and applying them as required. Usually a mission does not turn out as expected and we need to combine our options in a way that is different from the nominal plan.

Satisfy the customer

Who is the customer? The obvious customer is the user of the service that the mission provides. There are also indirect customers like the development of technology, the government, the tax payer, etc.

Be efficient

Avoid over-killing solutions in any part of the process. The key factor here is: how justifiable are our requirements?

We can probably agree that we want to accomplish most of the above requirements, but the question is still: how? Let us review some of our available alternatives.

ALTERNATIVES

Use small teams

There is a *critical mass* of people, beyond which a chain reaction occurs, that further increases their number and makes the organization less efficient. This critical number seems to be around 20 or 30 people

If you have a larger team you start to need more bureaucracy and more people to pull them together.

- *Hire the right people.* If we have a small team we do not have lots of redundancy. Therefore it is important to get the right combination of talent, personalities and experience.
- *Provide the right motivation.* The engine of human nature works mainly with two kinds of fuel: positive motivation and negative motivation. The best one is by far the positive motivation that we create by means of the following elements:
 - *An attractive vision and clear goals.* Examples: "We want to get *there* and achieve *that* great objective"; "We are here to deliver this product and to make this customer happy."
 - *A shared destiny.* "We are in the same ship, and we want to cooperate in order to safely arrive to our destination, so that we can share the success."
 - *Recognition and empowerment.* Let us show appreciation for the contribution of each member of the team. Let us allow each individual to feel his/her sharing of

the driver seat. We are more likely to volunteer our energy if we realize that it leads others towards the common goal.

- *The right pay-checks.* We can do wonders with a small team of great highly motivated people, but their motivation would not last long if we do not pay them well.

- *Co-locate people.* If everybody is in the same area, people will naturally talk to one another. Ideas will flow easier and they will need fewer memos and meetings. This has a magic effect on cost.
- *Lower cultural barriers.* People in a productive team may and should come from different backgrounds. They should be different in order to complement one another, but they should respect and welcome the differences. They should also be prepared for not being able to understand one another some times.

Use new technologies

A small team can implement a grand mission but they will require more powerful tools to be able to handle it.

The funding scheme could be a problem in the case of the operations technology. Normally the funding for operations is distributed over a number of years, and it is difficult to invest up front in powerful operations technology.

Keep the system simple

Do not incur in unnecessary complications. Both the spacecraft and the ground system should be as simple as possible. A mission becomes cheaper and also safer in this way. The following are different aspects that we can try in this area:

- *Small spacecraft.* A good way to keep it simple is to make it smaller. Sometimes a single small spacecraft cannot accomplish a grand mission, but for what we are saving we can afford to buy more than one.

The smaller the spacecraft the shorter the time and the cost to completion

- *More missions.* If they start to be cheaper we could have more; this means: "to distribute the eggs in different baskets." Also the learning curve of the new technologies gets faster if we launch more missions.
- *Provide feed-back to the requirements.* It is healthy to periodically check-out the relevance of the requirements that have a significant cost impact. Sometimes the customer does not really want what he asked for, particularly if he knows that it is going to cost him much more money or risk.

RISKS AND PROBLEMS

Now let us look for the obstacles that we have in our way.

Mission Failures.

Recently there have been some examples, but they seem to be distributed among missions with different sizes. We could expect higher risk from an ultra-low-cost mission, but the truth is that the big ones also fail, and when they do fail, they have a much bigger repercussion.

Size versus influence.

The smaller the project the smaller the weight it has within its organization, and the

reverse is also true. A project with a large budget represents a higher bet for the organization and will have a stronger voice when it comes to compete for resources.

This is an interesting management challenge. We have to protect the small-budget missions from being suffocated by the big ones, but we have to keep the big ones in good shape, because they damage badly the organization if they fail.

Lack of redundancy.

When reducing the cost, the redundancy is one of the things that tend to be suppressed. This applies both to the team and to the spacecraft and ground system design. This maybe OK if we lower the cost so much that we can do a second mission, but we must accept the fact that the mission becomes more likely to fail.

Nevertheless, the no-redundancy game could be very good for very small projects that we can afford to repeat several times by even trying different technologies.

Excess of automatism.

Auto-pilot is a great thing, but we normally do not fly on a plane without helm controls plus a pilot who knows how to use them. If the automatic function fails it is good to have a reliable "go to manual" key and a few well-trained people around.

Loss of interest

If we depend too much on the machines we have three negative effects:

1. People tend to forget how to operate in the manual mode that may be needed in an emergency. This requires continuous attention to training.
2. People lose interest for a system that does not give any opportunity to enjoy the

driver seat. That makes them to lose motivation to learn more about it

3. The public interest seems to react negatively to the machine winning the contest. Without this interest the space business would continuously decay.

If we are not careful, the human versus machine issue could severely damage the human motivation to pursue space research. Therefore, we should address and try to suggest a win-win solution to this problem among the recommendations below.

RECOMMENDATIONS

Small team with powerful technology

This is at least a possible solution to the problem: Grand Mission versus Small Operations.

In the term small team we should read all the good things that we have considered in the section ALTERNATIVES and not only the size of the team.

Special attention should be given to the funding peak required to buy up front the powerful hardware and software that will make it possible for a small team to handle the mission.

Harmonize human and machine

One would say that the artificial intelligence systems are no longer a simple tool but a knowledgeable colleague. If flying in auto-pilot is not very appealing to human nature, having to recognize that the machine is becoming an expert is quite hard on our pride.

The win-win solution to this conflict that we are going to suggest is to facilitate a good relationship between both sides

We humans could try to admit that the expert systems are becoming intelligent members of the team that have much to offer. Nevertheless, if a system has to be accepted as another member of the team it has to behave like one. It has to show the equivalent of *good manners*, and emulate the behavior of a reasonable human negotiator. It would not be a bad idea to program also some algorithms to respond to the human colleagues by showing recognition and nicely empowering the human initiatives.

For the programmers of the *expert systems* it is an interesting challenge to design such a *colleague-friendly* interface. Besides it will probably pay off to the developers as well, because a product as this is likely to capture the interest of many users.

Some people may think that this project is not worth the effort, but they should consider that Ignoring the human factor has always been very detrimental to any business.

CONCLUSION

Can we have, a grand mission but small operations team? Yes, we can.

How? We should try to combine a small affective team with *user-friendly* advanced technology.

It is indeed a challenge, but it is a very healthy and constructive one. The space business has probably grown a bit inefficient as part of its natural evolution.

We could compare our business to a mature apple-tree that has grown a bit too much. It is now the right time to prune it and to prepare it for a fruitful growth

CLARIFICATION

I have tried to share on this paper my personal ideas and opinions as an individual member of the Ulysses Operations Team. Nevertheless, my ideas do not represent the official opinion of the Ulysses Project.

The Role of Mission Operations in Spacecraft Integration and Test

Raymond J. Harvey
Johns Hopkins University The Applied Physics Laboratory
Johns Hopkins Road
Laurel, MD 20723

Abstract

The participation of mission operations personnel in the spacecraft integration and test process offers significant benefits to spacecraft programs in terms of test efficiency, staffing and training efficiency, test completeness, and subsequent cost containment. Operations personnel who have had real-time contact experience and have been responsible for the assessment of on-orbit spacecraft operations bring a unique view of spacecraft operations to pre-launch spacecraft test activities. Because of the unique view of the spacecraft/ground interface that experienced operations personnel have, they can propose optimum test approaches and optimum test data analysis techniques. Additionally, the testing that is typically required to validate operations methodologies can be integrated into spacecraft performance testing scenarios.

Introduction

Experienced mission operations personnel bring the unique view of operating a spacecraft to the integration and test (I&T) environment. Not only does the experienced mission operations person understand the functionality of the spacecraft at the system level, but also how the overall system will be operated after launch in the areas of mission planning, control, and assessment. Since these three functional areas of mission operations may be directly applied to integration and test, not only does this benefit the integration and test effort, but the mission operations effort as well, through validation of the operational concept, better training, and the practical experience of actually operating the spacecraft prior to launch. The participation of mission operations therefore benefits the entire program in both short and long terms in testing and operations efficiencies and the associated reduction in costs.

There are many different aspects where the mission operations personnel contribute to the integration and test effort from conceptual design of test procedures (planning), to the conduct of the test (control), and to the evaluation of the test (assessment). By involving these aspects of mission operations, the entire team may become involved in the various phases of testing. In addition to supporting the I&T effort, this process provides the opportunity to increase the coordination and communication within the mission operations team itself. In preparing for a spacecraft mission, the operations personnel are acquiring knowledge as to the capabilities which the spacecraft possesses, how these capabilities must be tested and validated during the early on-orbit checkout phase, and how to evaluate the performance of the spacecraft

Raymond J. Harvey is a member of the JHU/APL senior professional staff

throughout the mission. This knowledge and the experience from previous missions make operations personnel valuable assets to the integration and test effort.

Mission operations involvement also promotes spacecraft design optimization. The mission operations team knows how the spacecraft is "supposed" to operate. Their involvement in integration and test will demonstrate how the spacecraft will "actually" operate. In many fortunate cases, operations personnel can recommend subsystem modifications, if early enough in the process, such that on-orbit operations are more efficient and the mission operations development less complex. Obviously, for this to actually be effective, a certain amount of flexibility in the spacecraft design must exist, as well as a willingness on the part of program management to make modifications late in the spacecraft development process. From the program standpoint, however, it is often advantageous to implement the recommended changes, where time permits, to enhance the operational efficiency of the spacecraft over the course of the mission and potentially reduce lifecycle cost.

As summarized in Figure 1, there are many different aspects from which contributions may be made by the mission operations personnel in exchange for the invaluable experience gained by working with an operational spacecraft prior to launch. These aspects will be discussed below with practical examples provided where applicable.

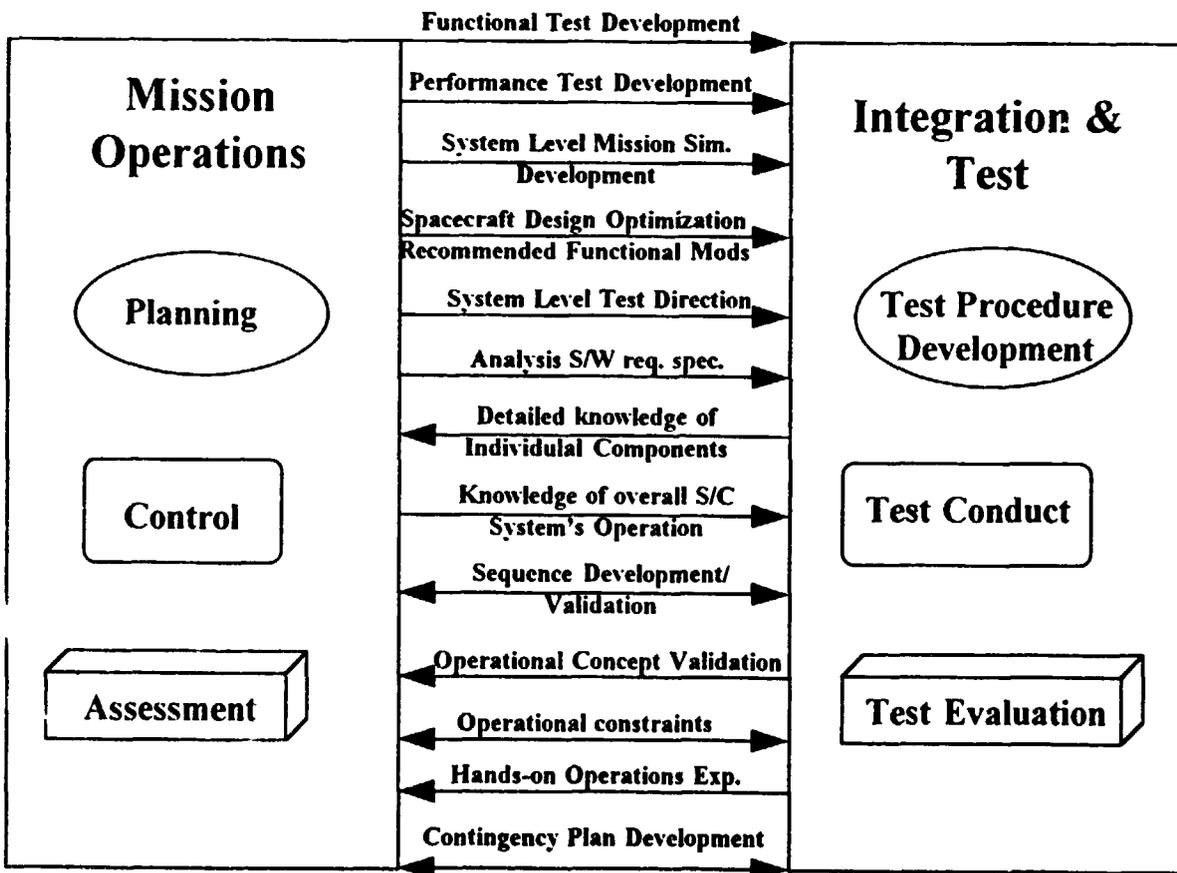


Figure 1: Mutual Benefits of Mission Operation's Role in Spacecraft Integration and Test

Mission Operations Involvement in I&T

Involvement of the mission operations team in I&T provides invaluable experience in many respects. It not only increases their knowledge of the spacecraft, validates their operational concept, capability, and command sequences, but it also benefits the I&T effort as well as the entire program. Many of the different roles within mission operations can be exercised by being involved with testing the spacecraft at the systems level. The different roles within mission operations are able to participate in the testing effort in different respects. These various roles come from the planning, control, and assessment teams. One of the planning team members is the "operations coordinator" (Marshall et al., 1992), who works with the program sponsor and PI teams to define how the spacecraft will be utilized on-orbit. Also from the planning team are "spacecraft specialists" who focus on the design and operation of the spacecraft and bring that knowledge to the mission operations team. The "spacecraft specialist" is the primary interface between the operations team and the spacecraft engineering design team. The mission operations control team provides the "mission controllers" who operate the consoles in performing the uplinks and downlinks to and from the spacecraft and monitor its state-of-health. During post-launch activities, the "spacecraft specialists" become the mission operations assessment team. The role of this team is to perform monitoring and trending of the spacecraft's health and status as well as to lead in the investigation and resolution of anomalies. This application to I&T is in test evaluation.

Participation of the "operations coordinators" provides the mechanism to use the mission operations planning system to design, develop, and generate command sequences that will subject the spacecraft to a post-launch type scenario. They also participate in the test evaluations to determine whether their instructions to the spacecraft produced the desired effect. This creates an optimum method of system level spacecraft testing as well as a chance to validate the planning team's capability of creating the scenarios and commanding the spacecraft to perform the scenario.

The involvement of the control team members include conducting the test, monitoring its progress in real-time, and identifying anomalies, just as they will be involved post-launch. Similarly, the assessment team members become involved in the real-time monitoring, post-test evaluation, and anomaly identification, investigation, and resolution. By the "mission controllers" and "spacecraft specialists" encountering actual spacecraft telemetry while the spacecraft is powered and operating within nominal ranges, they are able to experience how it responds to particular commands and sequences as well as how the spacecraft will operate within its design capabilities. This is especially advantageous when the involvement is during spacecraft environmental testing. This provides them with a baseline by which to evaluate the spacecraft's state-of-health, and to assess the performance of the spacecraft during post-launch operations. This information is highly critical to the mission operations team where they are a separate entity from the engineering design team, as is the case in many missions, including those of JHU/APL. In these cases where those who designed, built, and tested, do not operate the spacecraft, the "hands-on" experience is even more invaluable.

Another significant benefit from mission operations team participation in the testing efforts is in the area of contingency plan development. In many cases during spacecraft testing

when anomalies are encountered, contingency plan development occurs on the spot, in real-time. From there it is a matter of refinement.

Another important aspect involves acquiring knowledge and understanding of spacecraft operational constraints. The operations personnel also are in a better position, while participating the I&T, to recognize operational constraints and requirements at the system level. The "spacecraft specialists" typically maintain this type of information and have a better appreciation and understanding of what occurrences in the I&T process should be construed as operational constraints and work-arounds.

Mission Operation's Involvement in Functional, Performance, and System Level Test Design and Conduct

The acquired knowledge of the complex spacecraft design and operation by the mission operations personnel give them a distinct advantage over the integration and test team. By the time integration and testing begins, the I&T personnel typically have not concentrated on how to operate the spacecraft as a system, but as individual components. This insight on the part of the mission operations personnel provides them with the broader view of the spacecraft's capabilities and their intended use on-orbit. This is valuable when defining the functional and performance related tests which will be performed throughout the testing process. Since the capabilities of the spacecraft are understood by the operations personnel, they provide a unique perspective on which capabilities exist, the fact that they should be tested, and recommend how they should be tested. On a previous mission, numerous meetings were held to determine how to test the various components when delivered to the spacecraft. The lead subsystem design engineers were to present the testing requirements. In many cases it was difficult for the lead engineers to recall all of the subsystem capabilities without including the entire design team in the discussions - a very time consuming process that distracts the team from completing the final design. In these meetings, it proved beneficial to involve the mission operations personnel because they understood all of the capabilities required of the spacecraft after launch, thus requiring pre-launch testing.

For the definition of system level tests, mission simulations provide an optimal method of including the entire spacecraft. These mission simulations involve placing the spacecraft through the same sequence of events that would be required of it in collecting data post-launch. These mission simulations may be generated by the mission planning team's operations coordinators who have been working with the program sponsor and Principal Investigators (PI) in defining how the spacecraft will actually be used post-launch. They can bring this point-of-view to the testing effort and provide actual command sequences to perform the mission simulations. This not only tests out the planning system's ability to accurately generate these command sequences, but it also subjects the spacecraft, at the system level, to typical scenarios it will experience throughout the mission. This also demonstrates to the I&T team and design engineers supporting the effort, how the spacecraft will be used post-launch. In many cases on a previous mission, this type of exposure to actual operations, made the experts re-evaluate how their systems were actually commanded. This resulted in command sequence modifications in the I&T effort as well as the mission operations area.

An interesting aspect of mission operations participation in testing arose on a previous mission where the planning team's test scenarios were not constructed exactly as intended. In some cases the spacecraft was subjected to a more stressing case, thus actually improving on the test. The planned cases were eventually run; however, the slight deviation served as a more optimum test. This was initially thought of as a negative aspect, but when it was realized that no harm could be done to the spacecraft, it was viewed as a positive feature.

On a previous mission, certain more stressing test scenarios developed by the mission operations team were incorporated into the formal spacecraft baseline performance test that was conducted at various times throughout the I&T process, to prove that the spacecraft met the on-orbit mission requirements.

Mission Operations Involvement in Test Evaluation

Evaluation of the tests defined by the mission operation's "spacecraft specialist" team is analogous to post-launch performance assessment. Again, these tests are system level tests and require evaluation by individual support teams as well as mission operations. However, in order to evaluate the test, knowledge of the objectives is essential. The person developing the test case must convey to the supporting teams these objectives and coordinate accumulating the results and disseminating this information to the appropriate teams. In filling this role, mission operations personnel are not only directly involved with the performance assessment of the spacecraft itself, but also the evaluation of how close the planning process came to modelling how the spacecraft would perform that particular case. This provides the "operations coordinators" with feedback on their planning models and the "spacecraft specialists" with additional insight into what is involved in assessing spacecraft performance.

When the system level tests described in the previous section were to be executed, there was not one person on the integration and test team who completely understood the objectives of the test and therefore no one could realistically evaluate how the spacecraft system performed. The mission operations personnel understood the objectives since they defined the test, and therefore stood in a good position to direct the test development, execution, and evaluation. Particularly during the conduct of the test, when anomalies arose, someone understanding the test was required in order to be able to assess the situation and decide if particular anomalies could possibly be show-stoppers. In these cases, the mission operations person was relied upon for such evaluations. This not only increased the knowledge of the mission operations person, but provided an added benefit to the integration and test team, in that they were not required to dedicate a systems level person to learning and understanding the fine details of each test. In many cases, the mission operations representative kept the effort progressing when anomalies arose. By knowing the detailed objectives of the test from a systems perspective, the mission operations person was able to search for work-arounds to continue testing, as opposed to one team investigating the anomaly and the other teams watching and waiting. This allowed portions of the testing to proceed amongst the other teams while troubleshooting continued. The "hurry-up and wait" mode not only wastes valuable testing time of the spacecraft itself, but also that of the engineers and technicians providing support. If the mission operations person, with their unique perspective, can provide recommendations and suggestions on how to proceed, they again

benefit the entire testing effort.

Mission Operations Contributions to Ground Support System Development

A basic understanding and knowledge of how the spacecraft is operated, previous operations experience, and the ability to foresee how the spacecraft will actually be used on-orbit, enable the mission operations personnel to assist in the development of ground support system requirements, mainly in the areas of software. (The ground support system refers to the hardware and software used by the I&T team to test the spacecraft. It consists of the necessary elements to develop test scripts and command sequences, conduct tests, and to monitor and evaluate the performance of the spacecraft). Obviously, for every capability that a spacecraft possesses, there should be a way of testing that capability, and evaluating the performance of that test. In many cases, special ground support software must be developed to provide the capability to control the spacecraft a particular way.

Because of their viewpoint of how the spacecraft could and would be operated, mission operations personnel on a previous mission specified requirements for software tools to be used mostly in the areas of test evaluation and performance assessment. These types of tools were used in the integration and test effort not only to validate the spacecraft capability but also in troubleshooting anomalies. Some particular examples are described below:

Command Execution Verification - A post-launch requirement of verifying that every command in a stored sequence executed as expected, drove a software requirement for such a utility. This utility, in development for post-launch operations, was used extensively in evaluating system level performance tests. This software read a planned command sequence and for each command, accessed a look-up table for the appropriate telemetry parameters required for functional verification of each command. The software then accessed telemetry parameters from an archive file of raw telemetry and converted that telemetry from the appropriate time frame to engineering units for verification of proper command execution.

Command History Decoder - This tool was developed for mission operations such that command replicas downlinked in telemetry were reconstructed into a readable format. The mission operations personnel, recognizing the need for such a tool after launch, proceeded to have software developed to perform this task. This software was in turn made available on the GSS for the I&T team's use during testing to verify proper command system functionality.

History Buffer Decommutation - On a previous mission, there were several buffers which contained historical information concerning the health and status of the spacecraft. At the I&T level however, there was not an easy method of determining the contents of these buffers after they were downlinked (the only method consisted of manually decommutating raw data formatted in hex separated by minor frames). These tools, once in place, were used during the I&T process not only to verify the functionality of the buffers, but in reconstructing events on the spacecraft, particularly in the area of anomaly investigations. Particular buffers for which display capabilities were developed are briefly discussed below:

Autonomy - on this spacecraft, the command system had the capability of being programmed with rules which instructed it to monitor raw telemetry and perform comparison operations to determine if the telemetry showed to be in violation of a rule. If it was determined to be in violation, then the command system would execute a pre-defined command or command sequence. Rules were defined to safe-guard the spacecraft. The only method of determining the contents of these autonomy rules were to downlink a particular region of the command system's processor memory. There was no method of visualizing the contents of the rules without decommutating the raw data. Software was developed to read this raw data and convert it to a readable table. This tool was essential, in that the only way to determine which rule "fired" was to dump this region of memory and compare several counts. This software tool performed this comparison.

Orbit Memory - this buffer stored a subset of critical housekeeping telemetry parameters on a routine basis, to provide a continuous record of performance assessment trending. This buffer was downlinked through a particular telemetry format. The operation of this buffer was verified by the I&T team by using the manual method of decommutating raw data. This was acceptable for testing this capability, but not acceptable for using the data after launch to perform trending of these critical parameters. At the request of mission operations, this capability was designed into the GSS and was used in the I&T process to test the buffer's capability and used post-launch, for performance assessment trending.

Attitude History - this buffer routinely stored the spacecraft's attitude, such that while the on-board recorder was not in use and the spacecraft was outside of a station contact, the attitude would be known for that instant. This could be used to assess whether the spacecraft was maintaining the proper attitude orientation throughout the orbit. A capability was also designed into the GSS at the request of mission operations to access this data through a particular downlink telemetry format for use in verifying the spacecraft capability during I&T and for post-launch performance assessment.

Ephemeris Load Data Structure - although not particularly a buffer, this data structure contained the latest ephemeris that was uplinked to the attitude determination and control system. During system level testing, the spacecraft was loaded with an ephemeris such that the attitude system could be used in a mission simulation. These ephemeris loads were critical to the testing effort and therefore were routinely downlinked for verification of proper storage on-board. Again, the mission operations team defined the requirements and method of decommutation of this information such that it could easily be displayed and interpreted on the GSS.

Other software tools developed in support of the mission operations team were supplied to the I&T effort. These included a method to construct autonomy rules from a table which defined the input parameters. The format for these rules was complex such that a tool was required in order to create the rules with confidence of correctness. Another tool allowed for the autonomous comparison of a loaded rule with an image on the ground as opposed to inspection, to verify that it had been installed properly. A similar tool was developed to monitor the status of the rules. In providing these various tools to the I&T team, testing efficiency was increased.

Mission Operations Contribution to Spacecraft Design Enhancements

The involvement of mission operations personnel in the integration and test effort can result in an improved spacecraft. With their previous experience and their pre-launch involvement in the I&T process, the members of the mission operations team are in a good position to recommend modifications to components, where applicable and possible, to enhance the performance of the spacecraft post-launch.

On a previous mission, this proved to be particularly beneficial in several areas. Because of the mission operations "spacecraft specialists'" involvement in the system level operation of the spacecraft, they could anticipate the effect of one subsystem's capability on the entire system's performance. In a particular case, it was known that the solar arrays rotated at a particularly slow rate. The attitude system was designed to control the position of the arrays such that they could independently track the sun within a certain range. It became apparent during I&T simulations that solar array position control ceased while the spacecraft was in eclipse. This was because the attitude system positioned the arrays based on the measured sun direction derived from sun sensor inputs. This would require the arrays to be positioned upon exit from eclipse. This could require several minutes because of the rate at which they rotated. The mission operations personnel could envision the affect of this on power recovery and requested that the attitude software be modified such that the arrays could be positioned based on calculated sun direction as well as measured. In this way, the arrays would be at an optimal angle upon exit from eclipse, thus allowing maximum power recovery time.

A similar situation was identified during I&T where the arrays were not positioned when the spacecraft was maneuvering. For particular scenarios, positioning the arrays could require several minutes after the spacecraft position stabilized, but prior to actual data collection. Once again, it was requested that a modification be made such that positioning during maneuvers would be possible in order to maximize solar array rotation time.

Conclusion

The participation of experienced mission operations personnel in spacecraft integration and test has proven to be beneficial to spacecraft programs not only in the areas of mission operations system and team development, training efficiency, operational concept validation, and command sequence validation, but also in the areas of testing efficiency and completeness. The experienced mission operations personnel bring the unique point-of-view of having operated a spacecraft on-orbit to the testing effort. This results in more effective system level testing. It is advantageous for the entire mission operations team to become involved in that their planning, control, and assessment teams may assist in the areas of test development, conduct, and evaluation.

If the involvement of the mission operations team in the integration and test area can be coordinated at the onset of a program where particular responsibilities and authorities are given to the mission operations personnel, the benefits could be even more abundant. However, when these responsibilities are not established early, mission operations personnel may be constrained

by their commitments to their own system development and validation activities at the time I&T officially begins. Planned mission operations personnel participation can also assist the I&T team in that their responsibilities and team size may be reduced. An added advantage to the early involvement of mission operations is that an operation's perspective can influence the capabilities and efficiency of spacecraft pre-launch testability and post-launch operability. If this is taken into account at the beginning of a program from both teams' perspective, it will result in associated cost savings from the outset.

References

Marshall, M. H., & Landshof, J. A. (November 1992). Connecting Science and Operations: The Operations Coordinator. *Proceedings of the Second International Symposium on Ground Data Systems for Space Mission Operations (SPACEOPS 92)* (JPL Publication 93-5) (pp. 267 - 271). Pasadena, CA.

**PAYLOAD OPERATIONS MANAGEMENT OF A PLANNED
EUROPEAN SL-MISSION EMPLOYING ESTABLISHMENTS
OF ESA AND NATIONAL AGENCIES**

Rolf Joensson
European Space Technology Center (ESTEC)
Postbus 299, NL 2200 AG Noordwijk
Karl L. Mueller
Deutsche Forschungsanstalt fuer Luft- und
Raumfahrt (DLR), Linder Hoehe, D 51147 Cologne

ABSTRACT

Spacelab (SL)-missions with Payload Operations (P/L OPS) from Europe involve numerous space agencies, various ground infrastructure systems and national user organisations. An effective management structure must bring together different entities, facilities and people, but at the same time keep interfaces, costs and schedule under strict control.

This paper outlines the management concept for P/L OPS of a planned European SL-mission. The proposal draws on the relevant experience in Europe, which was acquired via the ESA/NASA mission SL-1, by the execution of two German SL-missions, and by the involvement in, or the support of, several NASA-missions.

INTRODUCTION

In the decade subsequent to SL-1, SL-utilization in Europe was performed mainly within the framework of the German SL-missions D1 and D-2. Building upon the contributions of DLR and German industry to SL-1, the D-missions were conceived such that the Mission Management was entrusted to DLR-management directorate in Cologne. The main project tasks to be managed were:

- integration & test of P/L & P/L-systems;
- P/L OPS (including P/L-crew training);
- control of development or interfaces to experiments, facilities or racks;
- NASA-interfaces (JSC as lead center).

Systems engineering and development of P/L-system H/W&S/W was performed by ERNO/Bremen (now DASA), as well as integration and test of the whole P/L complement prior to its delivery to KSC.

Experiment H/W&S/W (and respective user support) were built or provided mostly by German entities, but also by other parties.

P/L OPS was taken care of by DLR-technical directorate (with a SL-P/L simulator in Cologne, and the P/L OPS Control Center/ [POCC] in Oberpfaffenhofen).

During P/L OPS preparation, main activities took place at DLR-Cologne, subsequently moving to DLR-Oberpfaffenhofen, concentrated there during flight.

Accordingly, the POCC control team was composed of Rhinelanders and Bavarians (plus engineers from Northern Germany). For both D-missions, the lead position was manned by DLR personnel.

In addition to the D-missions, ESA and/or national agencies such as DARA were involved in other SL-missions via provision of either astronauts or experiments/facilities, thereby gaining further relevant experience.

Especially during IML-2, experimenters in user centers across Europe could control their experiments and/or transfer commands via MSFC POCC, using a precursor of the network planned by ESA for the Space Station era.

PLANNED EUROPEAN SL-MISSION

The contribution of ESA to the Space Station, the so-called Columbus Program, contains a Precursor Flights Program Element. The foremost goal of the Precursor Flights Program is "to prepare the European space user community, ESA and the participating states for the Space Station/Columbus era". The last programmatic document (Feb. 94) of ESA still maintains an SL-mission but, due to financial limitations, as a participation in a multilateral flight only. However, the program would be better served by an SL-mission led by ESA, as foreseen in earlier declarations of the Columbus Program, under the name "E1".

TECHNICAL SET-UP / SCHEDULE

In the last years, several investigations regarding an "E1" were carried out for ESA [Klein/Sobick, 1992; Mueller, 1992]. The last studies conducted for ESTEC could draw on recent NASA-experience with SL-missions of extended duration, and show the feasibility of the following configuration, though for some of the orbiters only [Joensson et al., 1994]:

Short tunnel, long SL-module, EDO-kit,
plus an exposed platform in cargo bay.

This would allow not only the accommodation of experiments and users from many disciplines other than micro-g, but also the operation of the payload in a manner more oriented towards the increment-type of operations planned for the Space Station era. In addition, the involvement of user centers could be further enhanced, and the ground infrastructure foreseen for the Space Station/Columbus era tested more extensively.

Since NASA plans to phase out SL during 1998, a launch prior to that date has to be aimed for. Taken together with the timespan of roughly 3.5 years, which is deemed necessary for preparing such an SL-flight as envisaged above, a launch in 1998 would only leave an absolute minimum on time before commencement of technical implementation.

ORGANISATIONAL SET-UP

The discussion of the mission implementation organisation foreseen will deal mainly with the pre-flight phase. The in-flight activities will be touched on only briefly, since those are too dependent on the actual requirements of the P/L.

For a rough overview of the pre-flight organisation see Fig.1; the outermost columns show only those tasks of DASA and USOCs which are considered relevant for the following discussion.

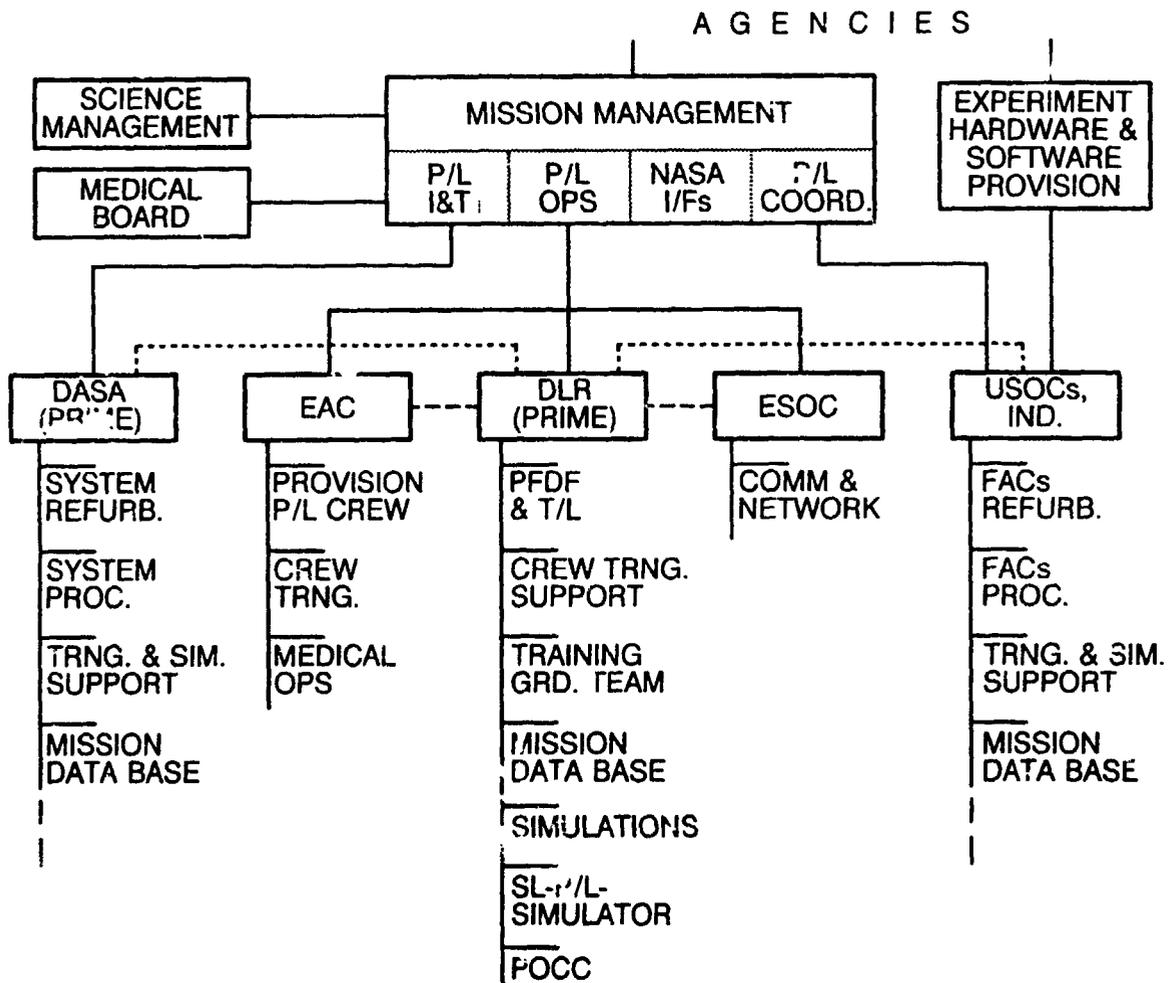


Figure 1 : Pre-flight Mission Implementation Organisation planned for E1 (adapted from Joensson et al., 1994).

One major difference as compared to D-missions is that mission management will be with ESA. The actual composition and location of that team will depend on negotiations with the agencies providing experiment H/W&S/W to E1.

Since for E1 every experiment H/W&S/W will be provided by third parties, mission management will control only the interfaces of the P/L system to the H/W & S/W in question (which may vary from simple experiments up to dedicated racks).

Integration and operations are foreseen to be contracted out again to DASA/ ERNO and DLR-technical directorate, but this time DASA and DLR will each have to lead a group of European firms, those consortia being structured and balanced according to the internal regulations of ESA.

Furthermore, the existing operations infrastructure has to be adapted to the existing ESA-organisation, which means

- that the tasks with respect to P/L-crew & P/L -Crew training plus medical operations will be under the responsibility of the European Astronaut Center (EAC) in Cologne,
- and that the European Space Operations Center (ESOC) in Darmstadt will be in charge of the network in Europe.

Moreover, whereas in D-2 two user centers were involved, for E1 at least three fully-fledged, national User Support Operations Centers (USOCs) in France, Germany and Italy will play a major role.

In addition to their standard services, it is likely the USOCs will be entrusted by their agencies with the development/refurbishment of experiment H/W&S/W for E1. This implies a transfer of tasks performed so far by industry to the USOCs.

From DLR, other tasks will be transferred to those USOCs, e.g. the development/adaptation of crew procedures for the above-mentioned experiment H/W&S/W. Similarly, the tasks concerning the crew procedures for P/L-system H/W&S/W (experiment-support and mission-specific equipment) will be shifted from P/L OPS to DASA.

The remainder of the tasks will, again, be the responsibility of DLR-technical directorate. However, whereas already for the D-missions subcontractors to DLR were employed, more of those firms, but from other ESA states, will have to be considered.

Regarding in-flight activities, the POCC control team might include members of EAC (crew I/F, medical operations) and of ESOC (network I/F), though it is still assumed the lead position will be manned again by DLR personnel.

Since many experiment operations will be performed as "telescience", this would require the capabilities to check and plan resources, to generate commands, to change and control procedures, and to archive data at the USOCs concerned.

Consequently, most experiment operations would be transferred from the DLR POCC to the USOCs, necessitating already in that area the use of a centralized P/L data base. However, such a common mission data base would, in addition, support the integration & test activities of DASA, and the performance of simulations by DLR, as well as the overall cooperation with NASA.

CONSEQUENCES FOR MANAGEMENT OF P/L OPS

Quite a number of tasks of P/L OPS, which for D-missions were under the sole responsibility of DLR-technical directorate, would in the case of an E1-mission be transferred from DLR to EAC and ESOC, and other activities be moved from DLR to USOCs and DASA.

Thus, the number of interfaces to be managed by mission management would increase significantly, and some of those will need some special attention.

EAC will be supported by DLR-technical directorate regarding P/L crew training in the frame of a special DLR-ESA agreement, and regarding medical operations by a consortium including a DLR research institute. As concerns the P/L-crew procedures tasks to be transferred, one of the USOCs to be considered will be the Micro-Gavity User Support Center (MUSC) of DLR.

However, the planned merger of the two DLR space operations departments, the Crew Training Center (housing the SL-P/L simulator) in Cologne and the German Space Operations Center (housing the POCC) in Oberpfaffenhofen into a single organisation will remove one interface.

Nevertheless, much of the P/L OPS relevant management which in the case of the D-missions was performed by P/L OPS itself, would for an E1 have to be performed from the level above, i.e. from mission management itself (as is foreseen for the Space Station/Columbus era).

Though all the parties concerned will use far more electronic tools, data bases and networks as compared with D-missions, the configuration control of those across DASA, DLR, EAC, ESOC and USOCs will require a significant effort not only by the parties just mentioned, but also by mission management.

However, one does not expect that inter-office communication will allow a paperless management of P/L OPS for an E1. Considering the multitude of parties involved, many papers will still have to be exchanged and evaluated, but made compatible only to the extent necessary, thus avoiding unnecessary efforts just for the sake of standardization.

Moreover, in the course of mission preparation, face-to-face contact of as many of the people likely to be involved in the actual flight (from working meetings to simulations) at the earliest possible stage will greatly enhance the probability of a successful E1 implementation.

CONCLUSION

The European SL-mission, E1, as described above is planned as a precursor to the Columbus era. The decentralization of activities foreseen for E1 will be a baseline for the Space Station/Columbus era. Therefore, many more parties will have to be involved in the project task P/L OPS as compared to the former D-missions, implying that far more interfaces would have to be controlled by mission management.

Due to the nature of tasks distributed among those parties, their interfaces would be rather complex, and use of modern tools for information dissemination will necessitate a considerable effort being put into configuration control.

REFERENCES

- Joensson, R. et al. (June 1994). Precursor Columbus Spacelab Flight, ESA Memorandum
- Klein, W.U. and Sobick, V. (March 1992). Columbus Precursor Spacelab Missions, Space Technol. 13, 83-85 (1993)
- Mueller, K.L. (March 1992). Payload Operations for Precursor Mission E1, Space Technol. 13, 155-160 (1993)

EVALUATING SPACE NETWORK (SN) SCHEDULING OPERATIONS CONCEPTS THROUGH STATISTICAL ANALYSIS

Carl Kwadrat and Nadine Happell
Stanford Telecom
1761 Business Center Drive
Reston, VA 22090 USA

ABSTRACT

The Network Control Center (NCC) currently uses the NCC Data System (NCCDS) to schedule customer spacecraft communication requests for the Space Network (SN). The NCC/Request Oriented Scheduling Engine (NCC/ROSE), which implements an operational concept called flexible scheduling, is being tested as a potential replacement for the NCCDS scheduler in an effort to increase the efficiency of the NCC scheduling operations. This paper describes the high fidelity benchmark tests being conducted on NCC/ROSE, the evaluation techniques used to compare schedules, and the results of the tests. This testing will verify the increases in efficiency and productivity that can help the NCC meet the anticipated scheduling loads well into the next century.

INTRODUCTION

The SN provides communication and tracking services to low earth orbiting spacecraft, such as the shuttle and Hubble Space Telescope (HST). These services are provided via two operational geosynchronous Tracking and Data Relay Satellites (TDRSs) and a ground terminal in White Sands, New Mexico. The NCC at the Goddard Space Flight Center (GSFC) is responsible for the management of SN resources, including the resource allocation function. Currently, customers submit relatively inflexible requests for communications and tracking support to the NCC (e.g., 20 minutes of S-band single access (SSA) support on the east relay satellite between 1200 and 1230) via Schedule Add Requests (SARs). However, customers generally have more flexibility than they are capable of expressing in the SAR messages. When scheduling conflicts occur, the NCC scheduler calls the customer(s), and using their true flexibilities, negotiates a resolution. Due to security restrictions, the NCC is prohibited from releasing information concerning the composite schedule to the general customer population, making conflict resolution even more difficult.

With projected increases in the network loading by the end of the century, extensive manual conflict resolution will not be viable. Therefore an operational concept called flexible scheduling is being evaluated (Moe, et al., Sept. 1993). Under this concept, customers are capable of expressing their full range of flexibilities in their request messages. Flexibilities to be included in the messages are: variable service and event durations, flexible service and event start times, open resource selection between equivalent resources, and backup or alternative event specification. In addition, flexible requests may express the recurring nature of requests (e.g., a 15 to 20 minute SSA support on any relay satellite once every orbit). With flexible requests, the scheduling system has more latitude in how to schedule a request

and avoid or resolve conflicts in an automated fashion. An added benefit is that conflicts are resolved as they are encountered, and not after other lower priority requests have been processed.

The Request-Oriented Scheduling Engine (ROSE) was designed as a general scheduling system capable of performing flexible scheduling (Weinstein, 1993). ROSE uses a scheduling language called the Flexible Envelope Request Notation (FERN) as an input format (Tong, 1993). Both FERN and ROSE are being adapted for use on the SN scheduling problem. ROSE is a candidate for replacing the current scheduling system and FERN is one of several candidate formats for replacing the current SAR messages (Meeks, 1994).

HIGH FIDELITY BENCHMARK

Part of the technology transfer process involves high fidelity benchmark tests to demonstrate the feasibility of using the NCC version of ROSE (NCC/ROSE) and the flexible scheduling concept under realistic SN scheduling scenarios (Moe et al, Nov. 1993). The benchmark tests are being conducted in two phases.

The purpose of Phase I tests is to verify that NCC/ROSE can perform SN scheduling. Specifically, NCC/ROSE must not schedule any requests in conflict based on SN scheduling constraints, and must not unnecessarily decline any request that could be legally scheduled. Phase I tests compare a schedule produced by the NCCDS to an NCC/ROSE generated schedule (neither schedule has undergone manual conflict resolution). A week of real requests submitted during a shuttle mission were used as inputs to both schedulers. The SARs were translated into FERN for input into NCC/ROSE. These requests reflect the current level of flexibility available in the electronic messages. Fig.1 illustrates the methodology used for the Phase I tests. Schedule run time, minutes of support scheduled, and number of events scheduled are the primary comparison metrics between the two schedules. The NCC/ROSE schedule also is converted back into inflexible requests and these requests are processed by the NCCDS. If the NCCDS does not reject any of these requests, then the NCC/ROSE schedule is a legal one.

The purpose of the Phase II tests is to evaluate the value added of flexible scheduling. For these tests, most of the customers capable of using flexible scheduling were interviewed in order to define their flexible requests. Flexible versions of the requests submitted for the test week were then generated. In order to support open resource selection and request recurrence, orbital data for the test week for these spacecraft were also collected as operational scheduling aids. In general, this data specified when the spacecraft could view which relay satellite, but also indicated other constraints that may be relevant to the requests.

The NCC/ROSE schedule generated with flexible requests is then compared to the NCCDS schedule after manual conflict resolution (Fig.2). The NCC/ROSE schedule again is converted into requests and submitted to the NCCDS for verification of a conflict free schedule. In addition, customers are interviewed to ensure that the conflict resolution options implemented by NCC/ROSE were acceptable. At the time of this writing, Phase II testing was ongoing.

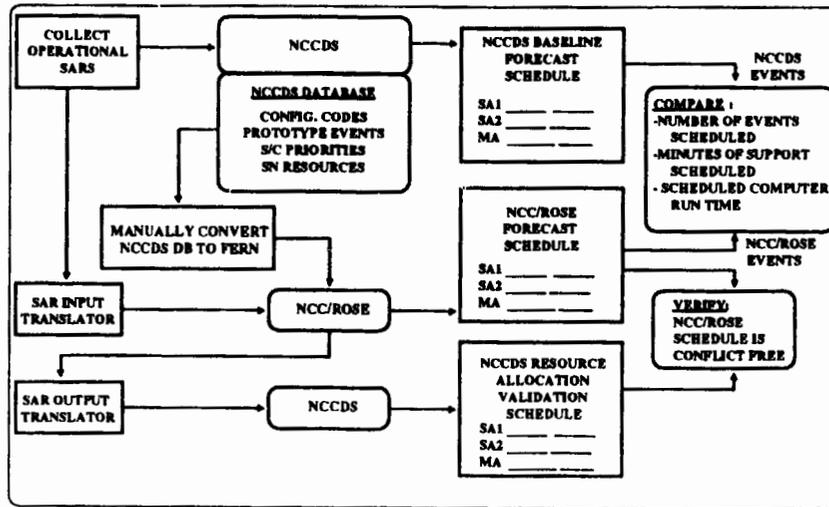


Fig. 1 - Phase I Methodology

NCC/ROSE can use two different algorithms to generate a schedule. Comparisons to the NCCDS are being made using both algorithms for Phase I and Phase II.

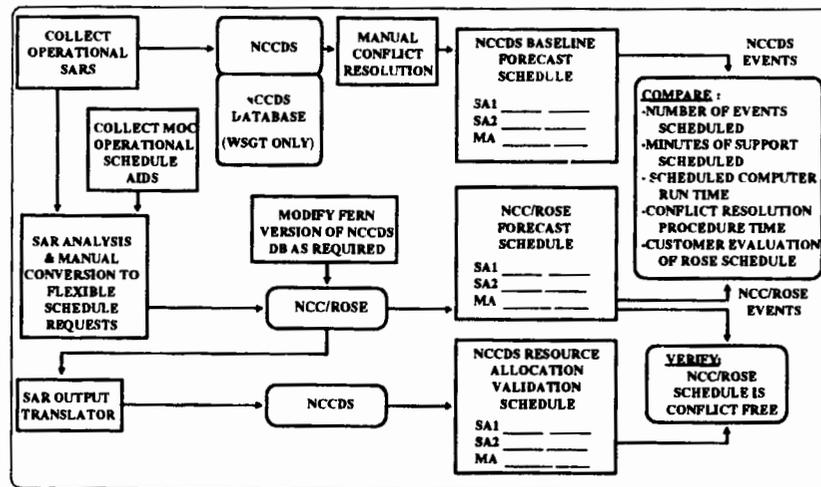


Fig. 2 - Phase II Methodology

EVALUATION TECHNIQUE

The evaluation method organizes the details of the comparisons between the NCCDS and NCC/ROSE schedules (Fig.1 and Fig.2). In addition, it characterizes the schedule differences via statistical evaluation metrics. When presented graphically, the metrics provide a composite view of schedule structure differences for all the SN customers and identifies anomalies for detailed analysis. Fig.3 shows an overview of the method used to make the comparisons.

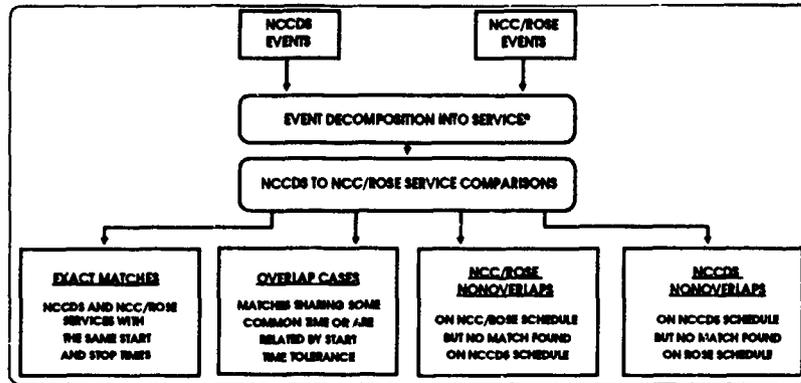


Fig. 3 - Schedule Comparison Method

The comparison method relies on the state transition diagram representation of the schedule shown in Fig.4 as a basis for generating the evaluation metrics. Each instance of a scheduled service is characterized by an ON transition state with an associated duration. The schedule period contains N instances.

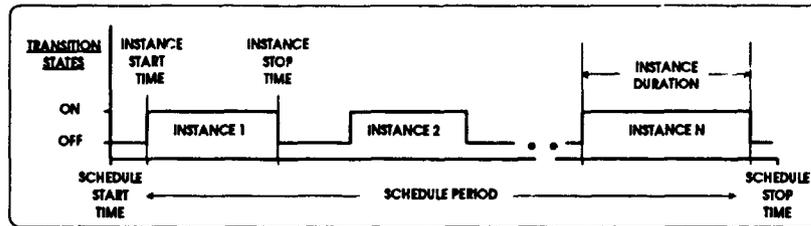


Fig. 4 - Representation of a Schedule

The NCCDS schedule consists of a series of events for all customers like the example HST events shown at the top of Fig.5. Each event contains one or more services. Event decomposition results in sets of customer service instances (bottom of Fig.5).

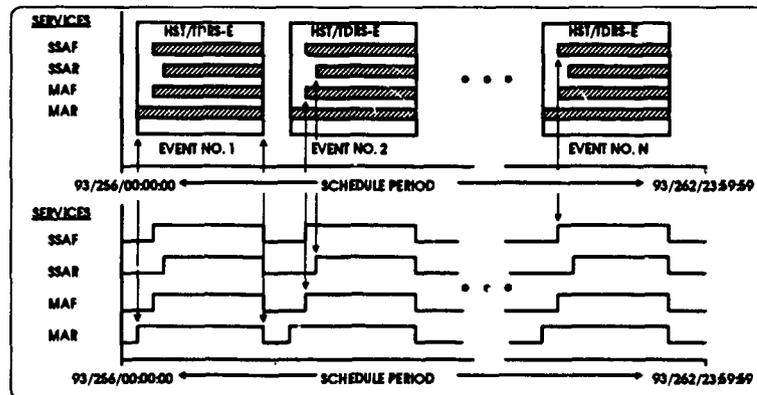


Fig. 5 - Event Decomposition into Services

Fig.6 shows the results of decomposing all of the NCCDS events into individual user resource schedules. The customer name, TDRS, and the TDRSS communication service requested identifies each schedule (e.g., STS, TDRS-E, SSAF).

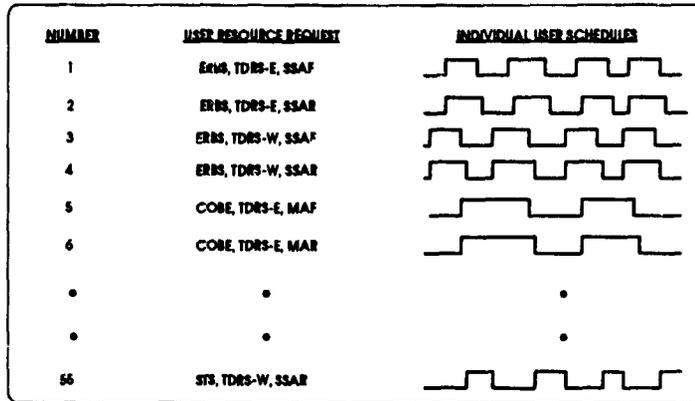


Fig. 6 - Decomposition by User Resource Requests

Fig.7 shows the criteria used in comparing the instances on the 55 NCCDS and NCC/ROSE user resource request schedules. Fig.7a through Fig.7c depict different match criteria while Fig.7d shows the no match criterion. Both overlap cases are the result

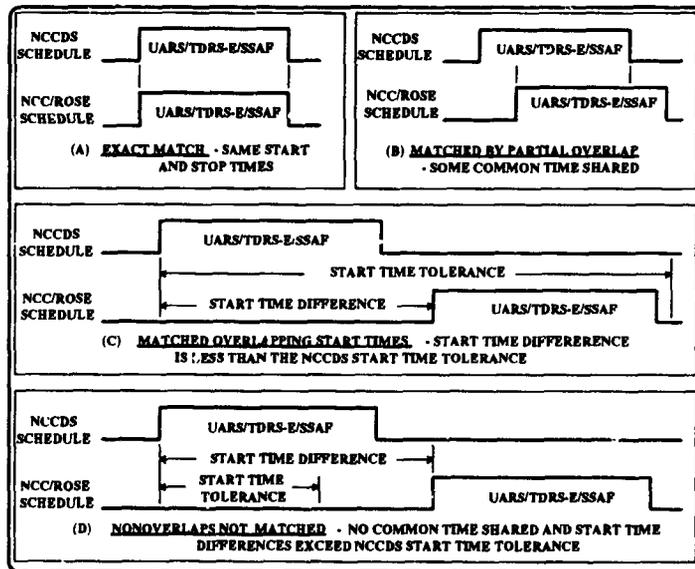


Fig. 7 - Service Instance Comparison Criteria

of exercising the NCCDS start time tolerances that allow an event to start anywhere in a specified time interval. Due to the open resource selection option, Phase II testing may produce overlap instances outside the SAR specified start time tolerance limits.

Instance counts and instance durations (Fig.4) form the basis of the evaluation metrics for each user resource request. Bar graphs provide a simultaneous view of all the customer

metrics. The 55 user resource requests (Fig.6), listed in order of increasing priority, form the abscissa of each graph. The percentage of the NCC/ROSE instances matching those on the NCCDS schedule forms the ordinate. Vertical lines separate the eight SN customer metric groups. The bar graphs presented below in Figs.8 through 12 illustrate Phase I results. Each bar graph contains the comparisons between the NCCDS results and NCC/ROSE earliest possible and lookahead algorithm results.

Fig.8 presents the results of the exact match comparisons (Fig.7a) indicating that lower priority users are less likely to have exactly matching instances than the high priority users.

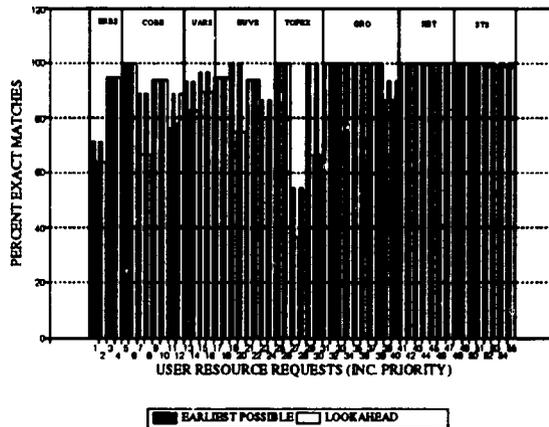


Fig. 8 - Exact Match Comparison Metrics

Fig.9 presents an assessment of instance start time differences for the overlap case results (Fig.7b and Fig.7c).

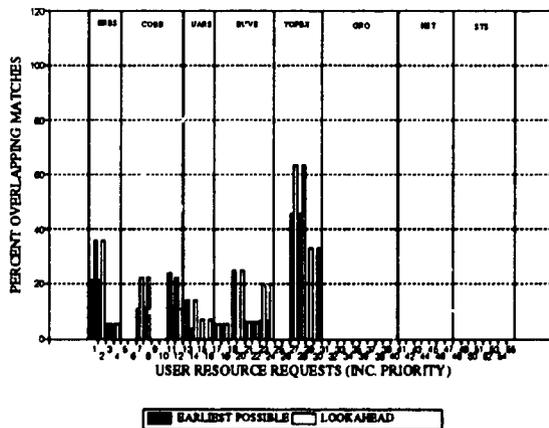


Fig. 9 - Overlap Comparison Metrics

Fig.10 shows the percent average start time difference metric for each user resource request. The ratio of average start time difference (Figs.7b and 7c) of all instances divided by the total of all the NCCDS instance durations (Fig.4) for a given user resource request forms the percent average start time difference metric. The average NCC/ROSE start time difference is

either positive, negative, or zero, corresponding to an average late, early, or equal start time with respect to the NCCDS schedule, respectively. Fig.10 shows that the NCC/ROSE earliest possible algorithm scheduled on average all of the overlap start times earlier than the NCCDS. The lookahead algorithm realized both leading and lagging average start time differences.

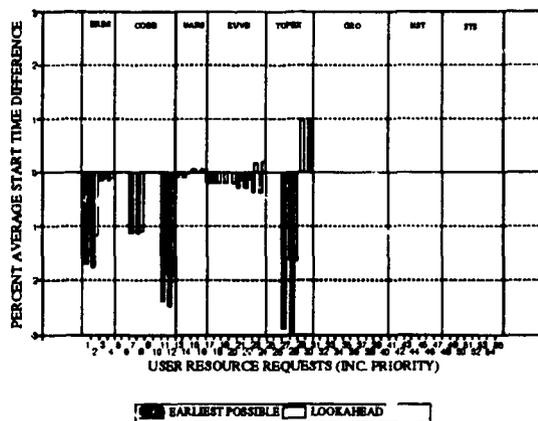


Fig. 10 - Average Start Time Difference Metrics

Fig.11 presents a composite of all the matching cases (Figs.7a, 7b, and 7c). This graph shows that the lower priority customers are more likely to have instances of a resource request dropped than high priority customers for both NCC/ROSE algorithms.

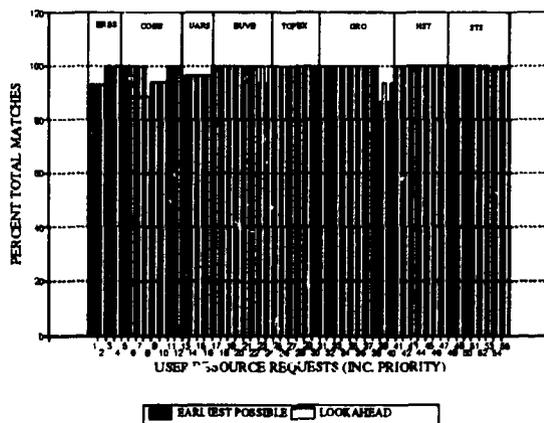


Fig. 11 - Exact and Overlapping Match Metrics

Fig.12 compares all of the NCC/ROSE scheduled instances (Figs.7a through 7d) to the NCCDS matching instances. COBE resource requests 11 and 12 for the lookahead algorithm exceed 100%, indicating that NCC/ROSE scheduled more instances than the NCCDS.

Replacing instance counts with instance durations (Fig.4) yields an analogous set of graphs corresponding to Figs. 8, 9, 11, and 12. The graphs compare the total time scheduled

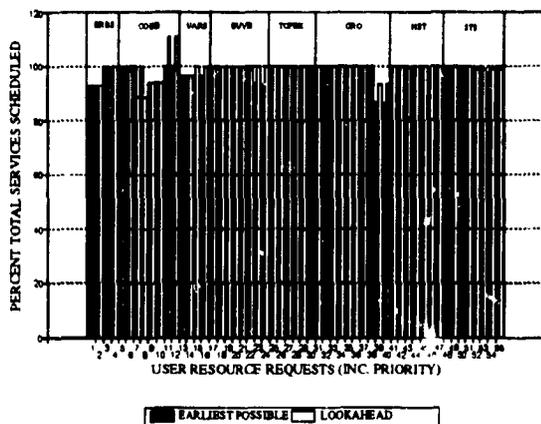


Fig. 12 - Total Instances Schedule Metrics

between the NCCDS and NCC/ROSE for each user resource request. Since the instance durations are not flexible for a given Phase I user resource request, the total service duration data is nominally proportional to the total instance data. This resulted in a set of percent time scheduled graphs that have virtually identical values in comparison to the instance scheduled graphs presented above. Flexible scheduling with variable instance durations will produce different results.

Phase II uses flexible requests for six of the eight SN customers. As such, the number of exact matches will decrease as the result of increased variability in instance start times and the added variabilities of instance duration, TDRS selection, and service selection. A shift from a highly populated exact match profile (Fig.8) to that dominated by large partial and nonoverlapping instance populations will accompany the transition from Phase I to Phase II testing.

PHASE I RESULTS

Table 1 shows a summary of the results of the NCCDS and NCC/ROSE scheduling operations for the earliest possible and lookahead algorithms (Kwadrat, 1994). NCC/ROSE scheduled the total number of events and total time within 1% of the NCCDS results for both algorithms.

Fig.13 shows two examples that illustrate the sources of the differences between the NCC/ROSE earliest possible and the NCCDS results presented in Table 1.

Fig.13a shows that an early EUVE start time selection by NCC/ROSE results in a conflict with a COBE instance. EUVE has a start time tolerance, COBE does not. The NCCDS uses the EUVE start time tolerance and the COBE instance is scheduled. Fig.13b shows the difference in antenna selection algorithms. NCC/ROSE places an HST instance on SSA antenna 1. The NCCDS placed the HST instance on SSA antenna 2. The NCC/ROSE

schedule omits the inflexible COBE request due to a conflict with the HST and shuttle events, while the NCCDS places it on the schedule.

Table 1 - Summary of Phase I Comparisons

NCCDS TO NCC/ROSE COMPARISONS	ALGORITHM	
	EARLIEST POSSIBLE	LOOKAHEAD
TOTAL NCCDS EVENTS SCHEDULED BY NCC/ROSE	99.4%	99.2%
TOTAL NCC/ROSE EVENTS SCHEDULED BY NCCDS	100.0%	99.8%
TOTAL NCCDS TIME SCHEDULED BY NCC/ROSE	99.6%	99.1%
NCCDS RUN TIME * (MINUTES)	45.2	45.2
NCC/ROSE RUN TIME ** (MINUTES)	5.3	9.0

* INCLUDES LOADING AND SAVING ALL CONFIGURATION CODE PARAMETERS
 ** INCLUDES LOADING BUT NOT SAVING ONLY THOSE CONFIGURATION CODE PARAMETERS REQUIRED FOR SCHEDULING (10%)

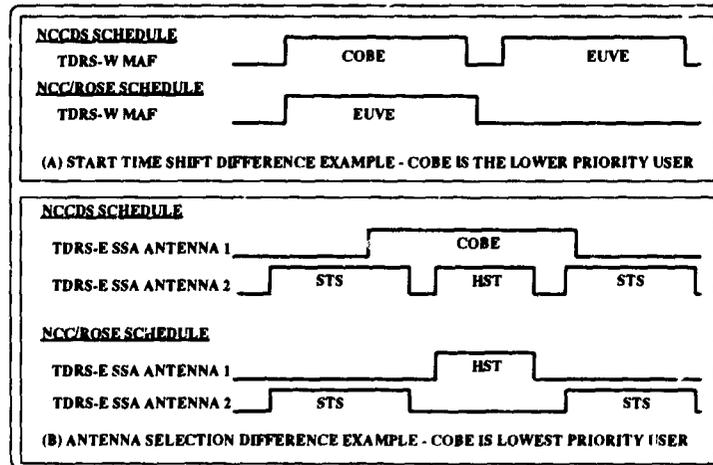


Fig. 13 - Earliest Possible Difference Examples

Heuristic algorithmic differences also account for differences in the lookahead algorithm results shown in Table 1. Fig. 14 shows two examples that demonstrate the impact of heuristic differences. Fig. 14a shows that NCC/ROSE used a UARS start time tolerance to permit the scheduling of ERBS.

The NCCDS elected not to shift the UARS instance, resulting in a rejection of the ERBS instance. Fig. 14b shows COBE being scheduled by the NCCDS but not by NCC/ROSE. EUVE is the only event with a start time tolerance. NCC/ROSE chose not to use the EUVE start time tolerance in order to schedule COBE. In addition, the NCC/ROSE lookahead uses a resource utilization algorithm to select antennas based on current load assessments. The NCCDS does not use this algorithm. This difference produced scheduling results similar to those shown in Fig. 13b.

A SUN Sparc 10 Workstation and a UNISYS mainframe are the host processors for NCC/ROSE and the NCCDS scheduling systems, respectively. The run times given in

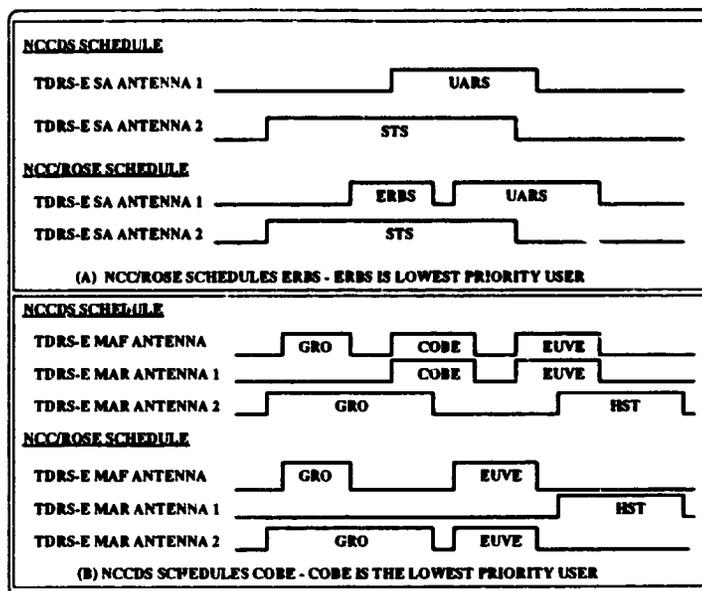


Fig. 14 - Lookahead Difference Examples

Table 1 are batch mode results. Configuration code processing differences between the NCCDS and NCC/ROSE are in part responsible for the run time differences.

PHASE II RESULTS

Phase II testing is in progress. The ERBS and COBE flexible schedule requests are operational. Due to a delay in the receipt of scheduling aids, the remaining six customers currently use the Phase I requests in the scheduling process. UARS, EUVE, GRO, and HST will also have flexible requests by the completion of Phase II testing.

Phase II schedules for the NCCDS included manual conflict resolution. There was an increase of 22% and 10% for ERBS and COBE instances, respectively, over the Phase I NCCDS schedule. Table 2 presents a summary of the preliminary Phase II ERBS and COBE results since they alone show the added effects of flexible requests on the NCC/ROSE schedule.

Table 2 - Phase II Preliminary Results

COMPARISONS	ERBS	COBE
NCC/ROSE TO NCCDS TOTAL INSTANCES SCHEDULED	80%	90%
NCC/ROSE TO NCCDS TOTAL TIME SCHEDULED	73%	90%

Fig. 15 shows the percent total NCCDS instances scheduled by NCC/ROSE for ERBS and COBE resource requests using the earliest possible algorithm. Fig. 15 is the Phase II counterpart of Fig. 12. Fig. 15 contains MA and SSA ERBS resource requests. All of the Phase I ERBS resource requests were SSA. The NCCDS manual conflict resolution

activities (Fig.2) and NCC/ROSE flexible service requests are responsible for the Phase II ERBS MA resource request metrics.

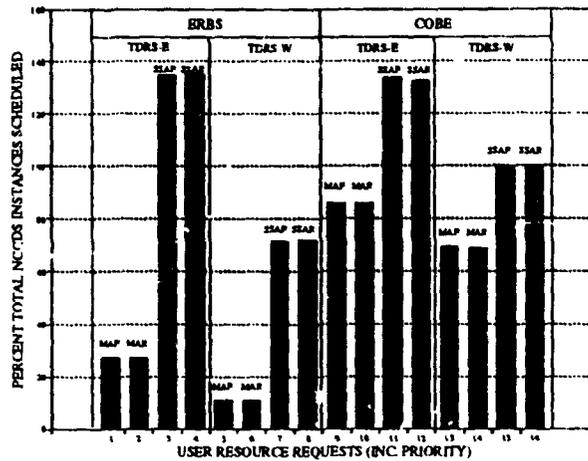


Fig. 15 - Total Instances Schedule Metrics

The number of ERBS and COBE SSA resource request instances have increased for both the NCCDS and NCC/ROSE results in making the transition from Phase I to Phase II. Fig.15 shows that for COBE at least NCC/ROSE appears to automatically choose, via flexible TDRS and service selections, more SSA scheduling on the alternate TDRS (the preferred alternative) in place of some of the MA selections made during NCCDS manual conflict resolution. The results for ERBS are less obvious and need further study.

Exactly matching instances form less than 1% of the Phase II ERBS and COBE comparisons. The Phase I exact matches (Fig.8) exceed 60% for both customers. This shows that the introduction of Phase II flexible requests significantly alters the NCC/ROSE schedule structure in comparison to the Phase I results.

As far as execution time is concerned, over 60 hours of NCCDS operator time were spent on manual conflict resolution. An NCC/ROSE run with flexible requests takes on the order of 5 minutes.

SUMMARY

The Phase I results verify that NCC/ROSE knows how to schedule SN services. All services that could be, were scheduled legally. However, the scheduling algorithms in NCC/ROSE are not quite as efficient as the algorithm in the NCCDS. Some improvements would probably be required prior to operational use.

The preliminary Phase II results are very promising. It was not expected that NCC/ROSE could perform conflict resolution as well as an NCCDS operator, but it might be able to resolve a significant portion of conflicts in an automated fashion. It appears that this is so. It

is hoped that these findings hold up after all appropriate customers are switched to the flexible requests.

The process of performing the tests has itself provided several valuable lessons. First, this effort required the cooperation of many different organizations, both government and contractor. With proper coordination, this collaboration has gone quite smoothly.

Still many technical stumbling blocks were encountered. The most cumbersome of which was dealing with the multitude of data formats and media for the operational scheduling aids for each customer. A single standardized interface is required prior to operational implementation of the flexible scheduling concept.

An important lesson learned was that it is more difficult than it appears to create a recurring flexible request. For flexible scheduling to work in an operational environment, it is critical that customer's have the proper tools to create and test their recurring flexible requests prior to submission to the NCC.

For flexible scheduling to be truly successful, the SN customer community must also change their mode of operations to take advantage of the enhancements. The more customers that submit flexible requests, the more benefit will be reaped by the entire SN community. And as the loading on the network increases, the more profitable the flexible scheduling strategy becomes.

ACKNOWLEDGMENTS

The benchmark tests have been a collaborative effort, requiring support from many different individuals and organizations. These have included: NASA GSFC Codes 522, 530, 534, 510 and 553, Loral Aerosys, Allied Signal Technical Services Corp., the SN customer scheduling operators and the NCC schedulers. A special thanks goes to Karen Moe of Code 522.

REFERENCES

- Kwadrat, C., (May 1994). *Phase I NCC/ROSE High Fidelity Benchmark Report*. Stanford Telecom, Reston, VA.
- Meeks, D., (May 1994). *Draft Survey of Potential SN Flexible Scheduling Request Formats*. Stanford Telecom Working Paper, Reston, VA.
- Moe, K., Wentz, L., & Maione, T. (Sept. 1993). *Space Network Flexible Scheduling Enhancements*. STel/TR93102, Reston, VA.
- Moe, K., Happell, N., Hayden, BJ, & Barclay, C. (Nov. 1993). Space Network Scheduling Benchmark: A Proof-of-Concept Process for Technology Transfer. *Proc. of Networks Technology Conference* (NASA CP-3241)(pp.103-120). Greenbelt, MD.
- Tong, G.M. (May 1993). *Flexible Envelope Request Notation (FERN) User's Guide*. GSFC/DSTL-93-012, Greenbelt, MD.
- Weinstein, S. (May 1993). *Request-Oriented Scheduling Engine (ROSE) User's Guide*. GSFC/DSTL-93-001, Greenbelt, MD.

SPOT4 MANAGEMENT CENTRE

Yves LABRUNE
CNES
18 avenue Edouard Belin
F - 31055 Toulouse Cedex

X. LABBE, A. ROUSSEL, P. VIELCANET
CISI
13 rue Villet
F - 31029 Toulouse Cedex

ABSTRACT

In the context of the CNES SPOT4 programme CISI is particularly responsible for the development of the SPOT4 Management Centre, part of the SPOT4 ground control system located at CNES Toulouse (France) designed to provide simultaneous control over two satellites.

The main operational activities are timed to synchronise with satellite visibilities (ten usable passes per day). The automatic capability of this system is achieved through agenda services (sequence of operations as defined and planned by operator). Therefore, the SPOT4 Management Centre offers limited, efficient and secure human interventions for supervision and decision making.

This paper emphasizes the main system characteristics as degree of automation, level of dependability and system parameterization.

1. PRESENTATION OF THE SPOT4 SYSTEM

1.1. Introduction

Since the 1977 decision made by the CNES (French Space Agency) to create the SPOT programme, there has been considerable success with the launches and operations of the SPOT1 (February 1986), SPOT2 (January 1990) and SPOT3 (September 1993) satellites.

The SPOT4 programme allows to ensure the continuity of this Earth Observation mission until the beginning of the next century (SPOT4 is planned to be launched in 1997).

1.2. SPOT4 technological aims

The payloads and passengers allow the SPOT4 mission to cover a wide commercial and technological field (eg. remote sensing, telecommunications, study of space environment).

The SPOT4 payloads are composed of :

- two identical HRVIR which are set up in such a way that it is possible both to get a repetitive coverage of the globe, and to form stereoscopic couples by the acquisition of oblique images,

- a new instrument, which is called VGT (vegetation), which mission consists of :
 - . studying and surveying vegetation and evaluating renewable resources, mainly in the agriculture field,
 - . studying and surveying the change in the continental biosphere at the global scale.

This instrument has been designed in order to observe most emerged land every day, the corresponding data being stored on-board in a mass-memory and transmitted back to the ground in visibility of the Kiruna and Toulouse stations.

The SPOT4 passengers are :

- DORIS (Doppler Orbitography and Radio-positioning Integrated by Satellite) which main purpose is to determine the orbit of the satellite with great accuracy (DORIS package uses an on-board orbit determination function),
- PASTEL which mission will be to transmit HRVIR images by laser optical link via a geostationary satellite (ARTEMIS) operated by the European Space Agency,

-ESBT which aims at experimenting the S band transmission of the housekeeping telemetry via the ARTEMIS satellite,

-PASTEC which mission is to increase the knowledge of the space environment and its influence on the behaviour of the satellite in orbit.

1.3. SPOT4 operational aims

According to the evolution of the Earth Remote

Sensing programmes, the realization of the SPOT4 ground segment must take into account more demanding operational requirements : reduced exploitation costs, better monitoring of the system operations (behaviour of satellite and payloads, status of image acquisition), reuse of SPOT4 developments for future programmes, harmonization and standardization of operations by the means of technological choices shared between different control centres, design of multimission software in order to ensure future reuse.

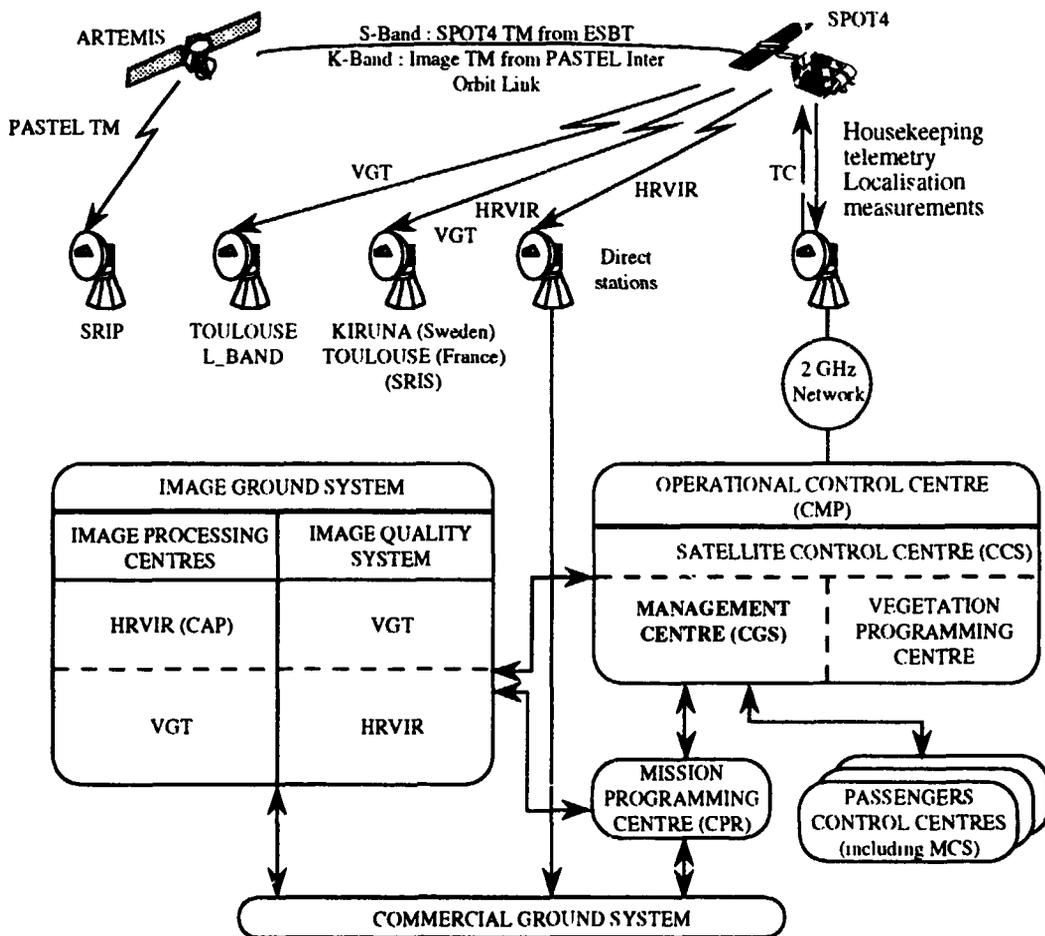


Figure 1 : SPOT4 System

2. PRESENTATION OF THE SPOT4 MANAGEMENT CENTRE (CGS)

The Management Centre is the entity of the Operational Control Centre (CMP) in charge of SPOT4 satellite programming and monitoring as shown in figure 1.

2.1. Main CGS functions

To perform the CGS mission, activities are divided into two types :

- critical tasks (eg. satellite commanding and monitoring, payload programming). These tasks follow a scenario which has been predefined; they execute on the Operations computer,
- preparation tasks (eg. instrument calibration). In order not to overload the critical tasks, these tasks are carried out on a specific computer (Preparation and Evaluation computer).

2.2. CGS hardware architecture

2.2.1. Operations computer

This HP 9000 serie computer supports the routine activities and manages the interfaces with the external centres (eg. CPR). Availability and security of the CGS data are granted by storage on mirror disks which can be accessed from both the Operations and the Back-Up computers as shown in figure 2.

2.2.2. Preparation and Evaluation computer

An HP 9000-755 computer is devoted to the specific tasks of operations engineers such as the preparation of non-routine activities on the satellite (eg. manual manoeuvre in degraded modes, spacecraft evaluation). For data retrieval or command execution, it gains access to the Operations computer via a local network.

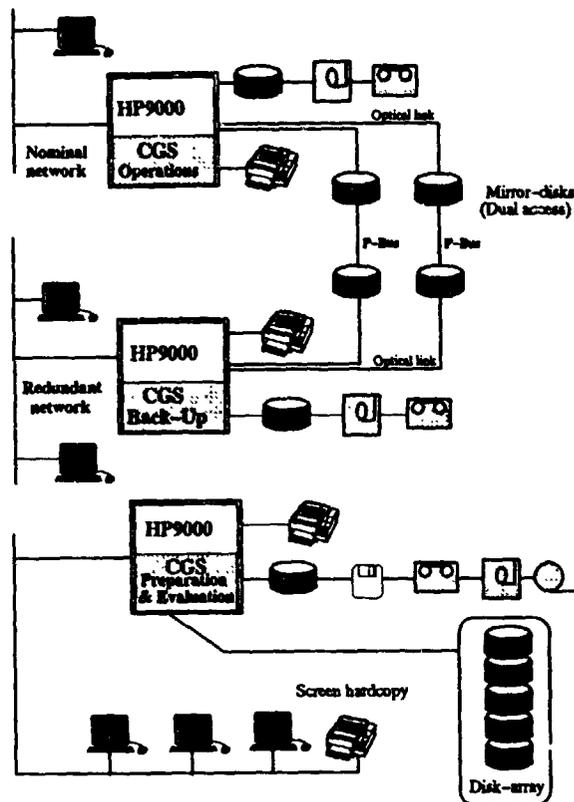


Figure 2 - Hardware Architecture

2.2.3. Back-Up computer

This computer provides redundancy for both the Operations computer and the Preparation and Evaluation computer. The use of mirror disks and operating check-points allows a fast restart with no loss of data.

2.2.4. CGS local network

In order to meet high performance requirements, the local network is divided in sub-networks carrying data specific to each segment. For high availability purposes, redundant schemes were implemented for both computers and sub-networks.

The network failure is avoided through the redundancy of the central router and each sub-network.

2.3. Software architecture

The CGS is composed of a set of sub-systems as shown in figure 3 :

- satellite monitoring sub-system which is in charge of the management of the satellite technological database and of the housekeeping telemetry off-line monitoring,
- passenger interface which sends the passenger commands to the satellite and plans the use of PASTEL and ESBT,
- flight dynamics sub-system which is in charge of SPOT4 orbit and attitude determination and control. It computes and prepares the related orbit and attitude manoeuvres,
- satellite and on-board software management sub-system which monitors the SPOT4 platform configuration,
- payload programming and monitoring sub-system which programs the HRVIR payload according to the commercial requests and the

technological evaluation needs (eg. instruments calibration) and monitors the images acquisition loop (on board command execution, ground acquisition and image archiving).

3. MAIN OPERATIONAL FEATURES

3.1. Introduction

The flexible design of the CGS answers variable needs among the various phases of the satellite life :

- launch and orbit positioning phase,
- flight acceptance phase,
- routine phase,
- anomaly mode.

After a brief overview of the CGS nominal operational environment this section presents the specific operational features of these phases and discusses the related implementation options.

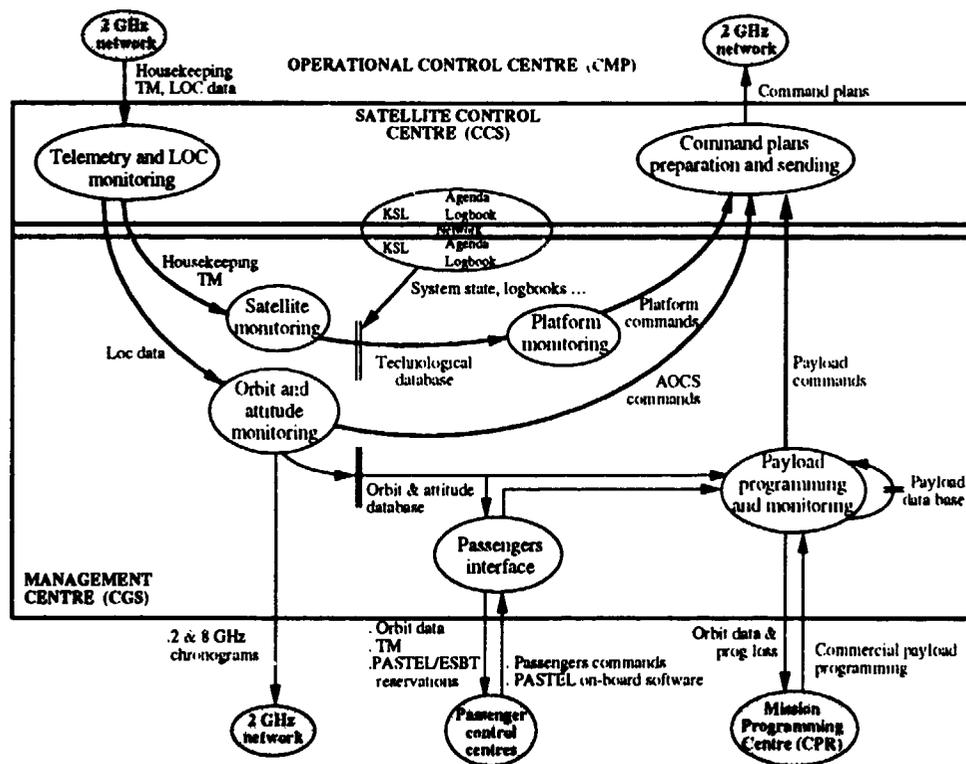


Figure 3 - Management Centre Main Interfaces

3.2. General operational environment

The SPOT4 CGS is in operation during working hours from 6 am to 10 pm. Overnight, it should be possible to carry out the operations automatically. The main operating environment of the CGS is the Agenda, presented at SpaceOps'92 (see ref.1).

The Agenda fully supports the automatic execution of the satellite monitoring and control daily plan (no human intervention).

In fact, a CGS daily program involves the execution of 100 tasks, whose average duration is 4 minutes and whose maximum execution time is 20 minutes.

3.3. Flight acceptance phase

This operations phase requires the execution of specific treatments on the flight acceptance system configuration (platform and payload configurations set up by operations engineers),

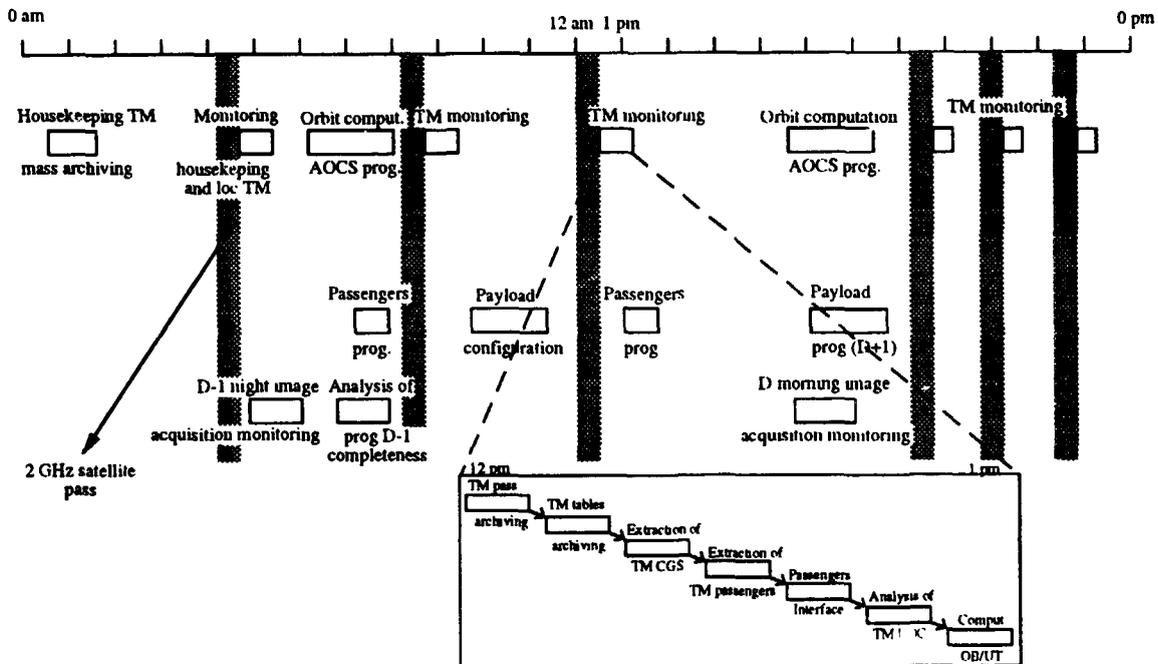


Figure 4 - CGS Daily Activity Plan

This is particularly the case of non critical tasks , run at night, under the autonomous control of the Agenda. The importance of this scheduler also resides in the fact that it can be stopped at any moment to enable the operator to take control on the sequence of operations. Figure 4 shows the first level of a typical daily work program at CGS.

in addition to the routine operations of the CGS.

To configure the platform consists in defining and sending specialized control sequences over the platform equipments. A devoted MMI supports the acceptance tests. This MMI uses the platform configuration status, stored in the satellite data base.

These tests are prepared in the operational qualification phase, thanks to the satellite simulation means. They are then stored in a library for further reuse in a given operation plan. A dedicated MMI allows the modification of those predefined controls, according to the results of the analysis of the housekeeping telemetry.

The payload is configured in order to perform the technological tests and calibration of the payload instruments and equipments. The imaging capability acceptance tests programming is based on the graphical representation of the test ground areas as shown in figure 5. This efficient graphical programming offers strong guarantees of safety and reliability for critical acceptance tests : all operational constraints are integrated in the MMI logics (eg. operational set-up, instrument performance limits, field of view, forbidden sequences).

This is a large improvement in ergonomics and security of the payload programming in technological mode, compared to the previous SPOT generation.

3.4. Routine operations phase

This operations phase is characterized by the maximum automation of the satellite control and monitoring actions. The routine operations are supported by the Operations computer through the Agenda, the Preparation and Evaluation computer being reserved for exceptional actions such as specific queries on the housekeeping telemetry or technological programming for further payload investigation. Significant enhancements have been implemented for SPOT4 CGS, as described in the next sections.

3.4.1. Satellite orbit manoeuvres computation

The satellite manoeuvres computing chain is entirely automated according to the following concepts.

After tracking data acquisition, the orbit is restituted and predicted ; these results are analyzed to check that the predicted values are within the range of acceptable values for the orbit parameters.

If the predicted orbit falls out of the normal range, the satellite manoeuvres are computed according to the rendez-vous concept which smoothes the

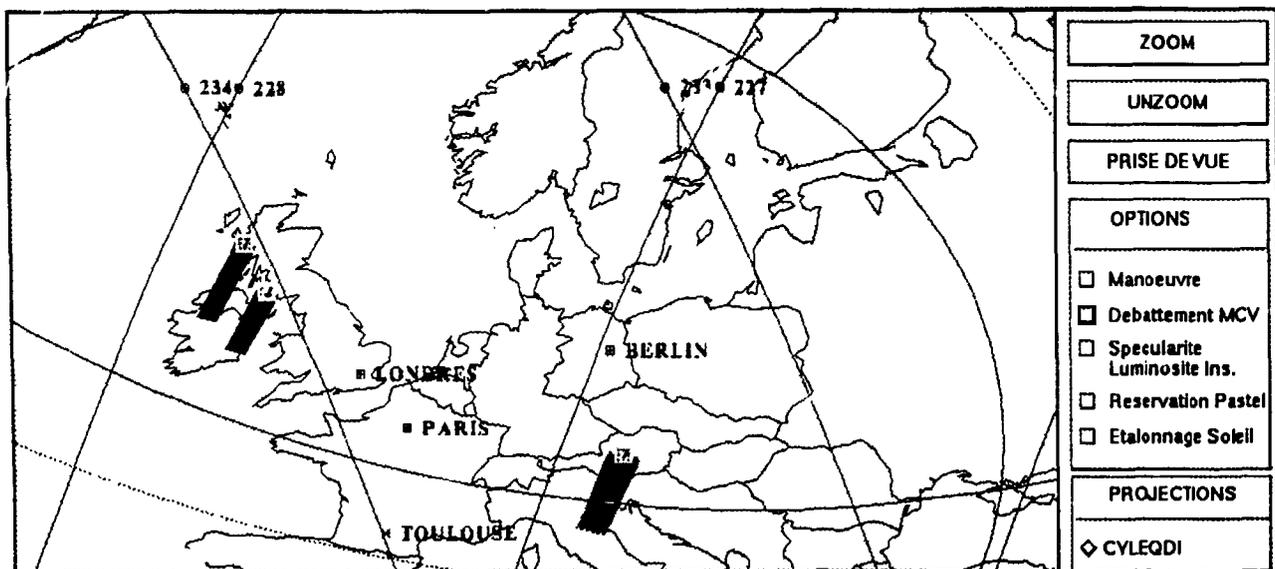


Figure 5 - Payload Acceptance Tests Programming

parameters evolutions and ensures their return to a normal range.

This new computer based strategy minimizes the number of successive orbit correction manoeuvres.

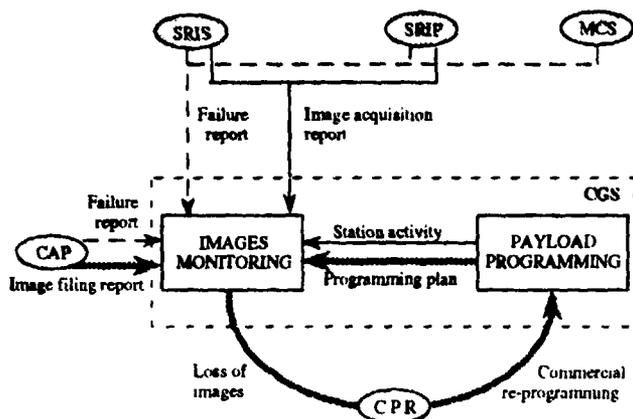


Figure 6 - Control of Image Acquisition Loop

3.4.2. Control of image acquisition loop

The image acquisition loop performs the automated monitoring of HRVIR programming, from image acquisition requests at CPR to image archiving within the Image Ground System.

The main goals of this controlled loop are to detect and report the losses of images and therefore it interfaces most of the elements of the ground segment as shown in figure 6 :

- the CPR which manages the user's requests,
- the CGS sub-system which elaborates the payload commands,
- the SPOT and PASTEL image reception stations (SRIS, SRIP),
- the SPOT image archiving center (CAP),
- the ARTEMIS satellite control center (MCS).

The correlation of the planned and achieved activities gives the operator the image loop status. This sub-system generates quantitative measures on the reliability of the ground segment and helps in the necessary reprogramming to satisfy the commercial operator's needs.

3.4.3. Parameterization based on satellite configuration

Many CGS operational tasks are parameterized by the satellite reference configuration which is stored in the satellite database under responsibility of the satellite manager. At CGS, a subsystem in charge of satellite configuration maintains an evolutionary version of this information according to operational needs (eg. update of the standard monitoring parameters thresholds).

After validation, these evolutions are centralized by the satellite manager and placed under control for future use by operational tasks.

This mechanism ensures the system operation security (fully controlled and formalized satellite configuration).

3.5. Anomaly mode

As part of CGS anomaly recovery procedures, various mechanisms support efficiently the necessary operational analysis in order to resume the execution of the system. The analysis is conducted by satellite and payload engineers.

A significant feature is the computerbased behaviour diagnosis of the flight dynamics sub-system.

When anomalies occur during the satellite orbit determination, like the divergence of successive orbital restitutions, a dedicated MMI displays :

- the probable failure causes,
- a check-up list for every cause,
- the actions related to a specified check-up step.

The MMI also gives access to graphical representations of orbit data and flight dynamics parameters.

Therefore, after detailed analysis, the engineer implements the proposed manual actions for recovering the nominal context of operations.

4. SYNTHESIS

Following the technological success of the first SPOT satellite generation in the mid 80's, the SPOT system has gained important commercial grounds, meeting the requirements of a large number of users throughout the world.

The need for images has grown considerably, and the economic stakes consequently evolve.

The periodic upgrades of the Earth Observation Systems are motivated by the aim to offer the users perennial services at competitive prices.

Due to its new characteristics (eg. levels of standardization and automation), the CGS is one of the basic components of the future ground control system of CNES remote sensing satellites.

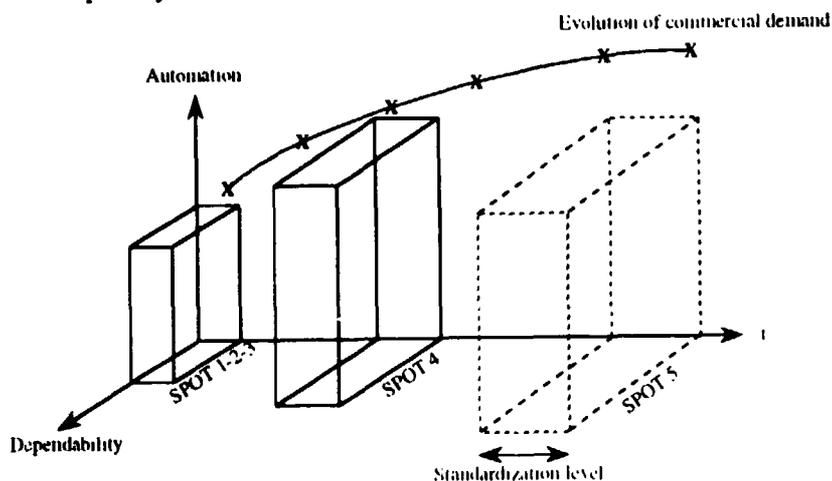


Figure 7 - Evolution of SPOT Main Operational Features

As specified in the introduction, the SPOT4 generation has to offer great improvements both, at technological and operational levels.

To achieve these ambitious goals, the SPOT4 CGS implementation relied on an industrial organization which benefitted from the experience gained on the SPOT 1/2/3 generation in the fields of advanced system engineering, development methodology, technology and quality assurance and control.

The SPOT4 CGS project gave way to the advanced rationalization of the definition and realization of the Management Centre components. These industrial products will ease further reuse and evolutions as sketched in figure 7.

REFERENCES

- 1 - PACHOLCZYK, P. 1992
SPOT satellite family.
In SpaceOps 1992
- 2 - ZAOUCHE, G. 1992
SPOT4 operational control centre.
In SpaceOps 1992
- 3 - FRATTER, I. 1992
Agenda : a task organizer and scheduler.
In SpaceOps 1992

**COSTS OPTIMIZATION FOR OPERATIONS CONCEPTS OF SMALL
SATELLITE MISSIONS**

Jean-Michel Oberto
Matra Marconi Space

Paper Not Available

Nickel Cadmium Battery Operations and Performance

Gopalakrishna Rao, Jill Prettyman-Lukoschek¹, Richard Calvin², Thomas Berry³,
Robert Bote⁴ and Mark Toft⁵
NASA/Goddard Space Flight Center
Greenbelt, Maryland 20771

The Earth Radiation Budget Satellite (ERBS), Compton Gamma Ray Observatory (CGRO), Upper Atmosphere Research Satellite (UARS), and Extreme Ultraviolet Explorer (EUVE) spacecraft are operated from NASA's Goddard Space Flight Center (GSFC) in Greenbelt, Maryland. On-board power subsystems for each satellite employ NASA Standard 50 Ampere-hour (Ah) nickel-cadmium batteries in a parallel configuration. To date, these batteries have exhibited degradation over periods from several months (anomalous behavior, UARS and CGRO {MPS-1}); to little if any, EUVE) to several years (old age, normal behavior, ERBS).

Since the onset of degraded performance, each mission's Flight Operations Team (FOT), under the direction of their cognizant GSFC Project Personnel and Space Power Application Branch's Engineers has closely monitored the battery performance and implemented several charge control schemes in an effort to extend battery life. Various software and hardware solutions have been developed to minimize battery overcharge. Each of the four sections of this paper covers a brief overview of each mission's operational battery management and its associated spacecraft battery performance. Also included are new operational procedures developed on-orbit that may be of special interest to future mission definition and development.

INTRODUCTION

The Earth Radiation Budget Satellite (ERBS), Compton Gamma Ray Observatory (CGRO), Upper Atmosphere Research Satellite (UARS), and Extreme Ultraviolet Explorer (EUVE) spacecraft are operated from NASA's Goddard Space Flight Center in Greenbelt, Maryland. Each satellite, except ERBS, employs the Multimission Modular Spacecraft (MMS) bus, which includes the Modular Attitude Control Subsystem (MACS), the Propulsion Module (PM), the Command and Data Handling Subsystem ((C&DH) - which incorporates the On-Board Computer (OBC), the Earth Sensor Assembly Module, and the Signal Conditioning and Control Unit (SC&CU)), and the Modular Power Subsystem (MPS). The ERBS spacecraft uses several, but not all, of the MMS bus features.

¹LORAL Aerosys - EUVE Flight Operations

²Martin Marietta Services Inc. - UARS Flight Operations

³AlliedSignal Technical Services Corp. - CGRO Flight Operations

⁴AlliedSignal Technical Services Corp. - ERBS Flight Operations

⁵Jackson & Tull Chartered Engineers - Space Power Applications Branch / Support Contract

The Power Subsystem, in general, comprises all power control, power distribution and all other related hardware. It contains the McDonnell Douglas Electronics Systems Company (MDESC)-supplied MPS, the Solar Array (SA) equipment and three NASA Standard 50 Ah Nickel-Cadmium batteries. Figure 1 presents the power subsystem topology. Table 1 lists the major Power Subsystem components and their functions.

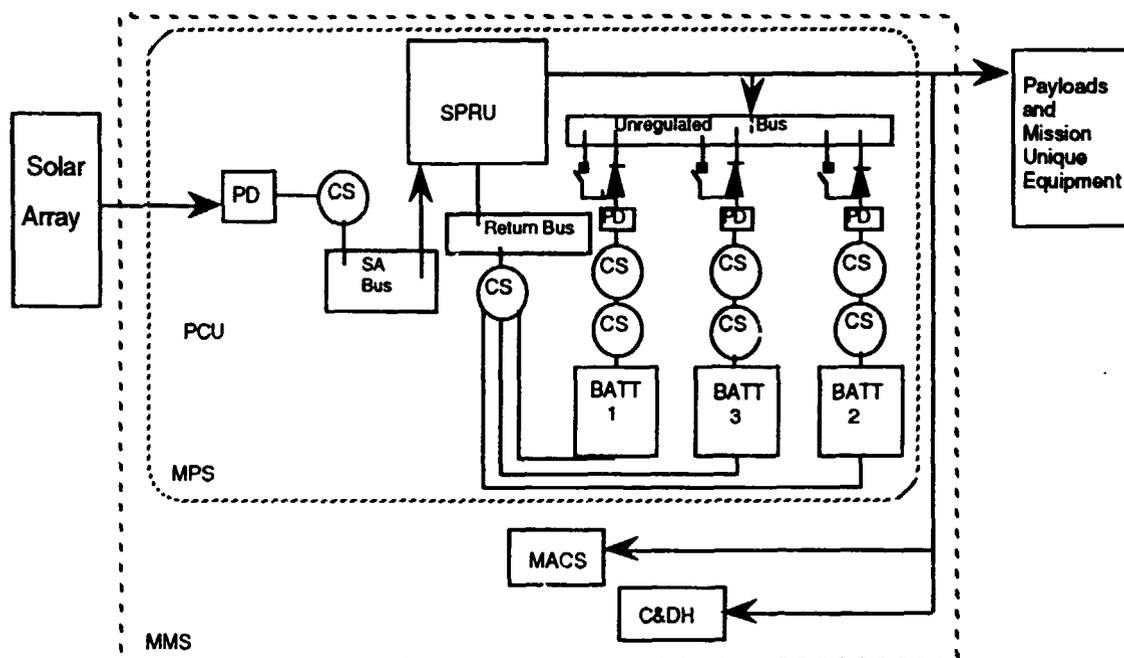


Figure 1. Power Subsystem Block Diagram

TABLE 1. Power Subsystem Components and their Functions

SPRU	Controls bus voltage and battery charging.
BPA	Provides redundant fusing for internal MPS loads.
Batteries	Provide power during spacecraft eclipse periods and supplement SA during peak loading.
Solar Array	Provides power for instrument loads and battery charging during spacecraft sunlit periods.

The MPS receives commands from the OBC which control its on-orbit battery charge modes. These modes and their operations are summarized in Table 2.

There are two to three NASA Standard 50 Ampere-hour (Ah) Nickel-Cadmium (NiCd) Spacecraft Batteries in each MPS. The NASA Standard 50 Ah NiCd batteries were manufactured and tested by McDonnell Douglas Corporation of St. Louis, using 22 serial-connected 50 Ah NiCd battery cells from Gates Aerospace Batteries (formerly General Electric Battery Business Division).

Table 2. MPS Charge Modes and their Operations

Peak Power Tracking (PPT)	While batteries are charging to the specified voltage limit, the SPRU draws maximum power from the SA so that the batteries charge on all current not required by the load bus.
Voltage Limited (Voltage/Temperature Mode, VT)	When battery voltages reaches the limit determined by one of eight user-selectable VT levels, battery charge current is tapered to maintain that voltage.
Current Limited (Constant Current Mode, CCM)	When selected by external command, battery charge current is controlled to one of three levels (0.75, 1.5 or 3.0 amps); the VT limit remains active as a backup.
Safehold	When an OBC fault is detected, the SPRU is commanded to PPT and a pre-selected VT level dependent on which VT was in effect at the time of the fault. CCM is inhibited. This mode remains in effect until reset by external command.

On all of the spacecraft, for each battery, there is hardwired circuitry that compares the sum of the first 11 series-connected cells to the sum of the remaining 11 series-connected cells. The result of this comparison is designated as the half-battery differential voltage. The first indication of possible problems with these batteries was a differential voltage exceeding 100 millivolts on charge, followed shortly thereafter by non-zero values on discharge. This initial divergence often arose after periods of increased overcharge and/or reduced discharge. It was then usually followed by a divergence in battery load-sharing (on charge and discharge), recharge (C/D) ratios, battery taper currents and battery temperatures.

Since the onset of degraded performance, each mission's Flight Operations Team (FOT), under the direction of their cognizant GSFC Project Personnel and Space Power Application Branch's Engineers has closely monitored the battery performance and implemented several charge control schemes in an effort to extend battery life. Various software and hardware solutions have been developed to minimize battery overcharge. Each of the four sections of this paper covers a brief overview of each mission's operational battery management and its associated spacecraft battery performance.

EARTH RADIATION BUDGET SATELLITE

Mission Summary

The Earth Radiation Budget Satellite (ERBS) was deployed into a 57-degree inclination LEO orbit from the Space Shuttle Challenger in October 5, 1984. The geometry of the orbit causes the angle between the orbit plane and the ecliptic plane (Beta angle) to change from 10 degrees to 170 degrees. At Beta angles equal to 90 degrees, the spacecraft executes a 180-degree yaw turn to keep its fixed SAs facing the sun.

Besides the two 50 Ah NiCd batteries and the SPRU, the Electrical Power Subsystem includes two redundant Ampere-hour Metering Units (AHMU). Batteries are charged via PPT until the VT limit is

reached, then the charge current tapers. When the AHMU's register 100% SOC, the SPRU fixes the charge current at 2 amps (approximately 1 amp per battery). Two other levels of CCM are available in the SPRU, a 7 amp level and a -5 amp level. At eclipse, the SPRU resets for the next orbit. VT 6 was initially selected for all Beta angles, and battery C/D ratio was used as the charge control parameter for switching from VT control to CCM.

Battery Management

For the first five years in orbit, battery performance was nominal. C/D ratios during this time were between 1.17 and 1.25. In September, 1989, half-battery differential voltages began to rise from the nominal value of 40 millivolts. By July, 1990, battery 1's half-battery differential voltage had risen to 220 millivolts, and battery 2's half-battery differential voltage had risen to 460 millivolts. The half-battery differential voltages reached a peak when the high power TDRSS (Tracking and Data Relay Satellite System) transponder was on during eclipse. The FOT reduced the duration of TDRSS events, and eventually turned off the TDRSS transponder during eclipses. The reduced power load decreased the half-battery differential voltages. Also at this time, it was discovered that battery 1 was carrying more of the spacecraft load. The current differential was .96 amps on discharge, and .72 amps on charge. Meanwhile, the CCM's 2 amp level had increased (apparently from component drift) from 1.00 amp to 1.50 amps for battery 1, and from 0.99 amp to 1.10 amps for battery 2.

The poor load-sharing and the increase in CCM current began to overcharge battery 1. The lowering VT level to 5 in January 1992 worsened the load-sharing. The VT level was further reduced to 4 in July 1992; but battery 2 was not fully charged at this VT level.

The first cell failure in battery 1 occurred in August 1992. The VT level was immediately lowered to 3, but the load-sharing continued to worsen. The C/D ratio for battery 1 was above 1.2, while the C/D ratio for battery 2 was less than 1. It became evident that the two batteries could not be charged together using straight V/T control. To reduce the overcharge on battery 1 (without undercharging battery 2), battery 1 was removed from the charge bus early in each charge period, then reconnected. This brought the C/D ratios to within 0.535 of each other. The VT level was raised one level after battery 1 was taken off-line so that the two batteries would be within 2 volts of each other at the time of reconnection. This required numerous commands from spacecraft memory, and became difficult when the frequency of memory bit upsets (a separate problem) increased.

The next approach was to disconnect battery 2 during eclipse and reconnect the battery at sunrise. This provided a 0.7 volt diode voltage differential in favor of battery 2 during eclipse. Battery C/D ratios were brought to within 0.307 of each other. This method was used until September 1992, when a second cell failed in battery 1. Battery 1 was immediately commanded off-charge. Battery 2 carried the entire spacecraft power load, with the charge control method reconfigured for a single battery. In attempts to stabilize battery 2, the SPRU was gradually raised to VT 5 to give battery 2 a C/D ratio of 1.10. The performance parameters of battery 2 returned to pre-1989 values. A charge scheme based on switching VT levels according to Beta angles was soon implemented. Spacecraft loads, namely heaters, were used to control the battery C/D ratio.

Eventually, however, one cell in battery 2 failed in June 1993. The previous charge control method was continued at lower VT levels. A second cell failed in July 1993. The SPRU was set to its lowest VT level and commanded into the 2 amp CCM (actual measured value: 2.7 amps) to achieve the desired C/D ratio of 1.10. The time to switch from VT control to CCM was manually calculated on the ground, and the

commands uplinked into spacecraft memory. In spite of continued high currents from PPT, this new method appeared to produce nominal battery performance.

During this time, it was found that the 7 amp CCM level had drifted to a value of 11.4 amps. The 11.4 amp level was preferred as the means to charge battery 2 without the high PPT charge currents.

Beginning in August 1993, the FOT began manually regulating battery charging by placing the battery in CCM and switching among the three CCM levels available within the SPRU. Battery management is accomplished entirely from spacecraft memory, and is based on ground calculations. All spacecraft charge controls and safeguards are disabled. Battery 2 is charged at the beginning and end of the sunlight period at the 2.7 amp level. During the middle of sunlight, the battery is charged at the 11.4 amp level. The duration at the 11.4 amp level is set to produce a C/D ratio of 1.02. Four commands per orbit are required to charge the battery in this way. The command times are manually calculated using predicted sunrise and sunset events and battery load. Since the battery load changes each orbit depending on Beta angle, spacecraft loads, and SA output, the duration at the 11.4 amp level can be adjusted every orbit (1.5 hours).

During full-sun periods, the -5 amp level is used to 'exercise' the battery. The negative current forces discharge on the battery, artificially increasing the power load. Reducing overcharge is seen as the best method to prevent another cell failure.

Operation of one battery containing two shorted cells continues in full support of the remaining scientific instruments. Battery temperature is between 2 and 5 degrees C, voltage is maintained between 24.00 and 29.66 volts, and the C/D ratio is kept between 1 and 1.02. The CCM charge control method, although labor intensive and totally reliant on down-linked data, is able to effectively control battery C/D ratio. The only difficulty in maintaining battery C/D ratio stems from the constantly changing power load (based on Beta angle and instrument status), and the time lag between real-time and off-line analysis of data. This method of charge control is expected to continue even in the event of another cell failure in battery 2.

COMPTON GAMMA RAY OBSERVATORY

Mission Summary

The Compton Gamma Ray Observatory (CGRO) was launched aboard the Space Shuttle Atlantis on April 5, 1991 and was released into a circular orbit of 450 kilometers. The Observatory orbits the Earth with an inclination of 28 degrees and a nodal precession rate of about 3 degrees per day. The length of spacecraft daylight varies from 57 to 64 minutes per 93 minute orbit. CGRO is the heaviest civilian payload ever launched by a shuttle, weighing 35,000 pounds.

CGRO's Electrical Power Distribution Subsystem (EPDS) consists of two, rotatable (single-axis) SAs and two MPS. The MPS-1 and MPS-2 launch configuration was VT 5 with a load imbalance of less than 100 watts. Post-launch performance for both MPS-1 and MPS-2 was nominal.

To date, MPS-2 batteries have shown excellent well matched performance. Battery temperature, load-sharing, half-battery differential voltage, and C/D ratios are well within their expected operational ranges. Battery DOD has been 10 - 17%. Both VT 5 and VT 6 charging schemes have been used. For DOD'S in the 10 - 14% range, VT 5 is utilized. DOD'S in the 14 - 17% range require VT 6.

In the April - June 1992 time period, MPS-1 batteries 1 and 2 showed significant signs of divergence in half-battery differential voltage, temperature, net overcharge and load-sharing. Higher than nominal overcharge rates became the leading suspect for excessive battery degradation conditions. During this period, plans were developed to reduce consumption upon MPS-1 by switching mission critical components to MPS-2. By July of 1992, MPS-1 battery 2 was progressively degrading; half-battery differential voltage reached saturation (greater than 700 millivolts) on several occasions. Finally, battery 2's temperature soared to over 30 degrees Centigrade near the end of charge of one orbit, and the battery was commanded "off-charge" on July 16, 1992. MPS-1 was then operated with the two remaining on-line batteries. Attempts to reduce net overcharge by commanding MPS-1 to lower VT levels were ineffective. Consequently, a new plan was implemented to reduce battery net overcharge .

MPS-1 Battery Management

The use of CCM (the lowest SPRU-commandable value of 0.75 amps) and VT 3 was chosen to minimize overcharge. This battery management technique involved commanding the SPRU, through the OBC's Stored Command Processor (SCP), to CCM at the beginning of spacecraft sunrise. While in CCM, the two remaining on-line batteries trickle charge at the 0.75 amp rate until SA temperatures approach a maximum, which occurs approximately fifteen minutes after spacecraft sunrise. After this user-defined time interval, the SCP commands the SPRU to VT 3.

While in VT 3, MPS-1 enters PPT mode, drawing maximum power from the SA and providing power to recharge the batteries. The batteries taper charge until a user-defined instantaneous net overcharge threshold is achieved. Upon reaching this threshold, the SPRU is again commanded to CCM (0.75A) via a Relative Time Sequence (RTS) activation and remains in CCM until the end of spacecraft daylight. Net overcharge is calculated by the OBC's PMON Processor using battery 1's current. Accumulated battery charge and discharge values are subtracted to calculate the net overcharge for each battery. Net overcharge data are then reported to the OBC's TMON Processor which manages battery charge modes, based on a user-defined instantaneous net overcharge threshold (presently 0.4 Ah). Once that threshold is exceeded, TMON takes action by activating an RTS that commands MPS-1 to CCM. This battery management charging scheme has been in use since February of 1993 and has been highly effective in minimizing further battery degradation due to overcharge.

UPPER ATMOSPHERE RESEARCH SATELLITE

Mission Summary

The Upper Atmosphere Research Satellite (UARS), designed, built, integrated, tested and operated by NASA and Martin Marietta, is a Low-Earth orbit (LEO), Earth-observing spacecraft which was launched via the Space Shuttle Discovery on September 12, 1991 and deployed three days later. The mission orbit is a 96-minute, circular orbit inclined 57 degrees to the Equator with a 585 Km height. This allows stratospheric sensors to observe up to 80 degrees in latitude (North and South) and provides near total global coverage. The full range of local times at all geographic locations is viewed every 36 days. The spacecraft batteries were specified to operate in low-Earth orbit up to 20% Depth of Discharge (DOD), between 21 and 35 volts for a nominal 36 month mission. Thermal vacuum testing revealed nominal battery performance prior to launch. The spacecraft power system was designed for a maximum 1600

Watts (orbital average), 786 Watts of which was reserved for the instrument load. The spacecraft maximum load has been approximately 1350 Watts with instrument loads of approximately 450 Watts.

Nominal battery performance was observed over the first four months of spacecraft operation. The first evidence of anomalous battery performance was observed in January 1992, after the first maximum beta angle (low DOD or "Full Sun" period). Since then, the FOT has monitored and managed battery performance by adjusting solar array offset angle, conducted periodic deep discharges, and controlled battery recharge ratios.

Battery Management

Due to the cyclical variation of the orbit Beta angle (Beta angle is defined as the angle between the orbital plane and the Earth-to-Sun line), caused by the 57 degree orbital inclination and orbital geometry, the SA power collection, the spacecraft loading, and the battery charge and discharge profiles are not constant. The Beta angle variation changes SA night periods (in addition to the normal seasonal changes) from a maximum eclipse of 36 minutes at zero degree Beta, to a minimum of zero minutes at Beta angles greater than 66 degrees. This has prompted the FOT to develop more aggressive Power Monitor (PMON) software to actively control battery performance. The use of Constant Current Mode (CCM) was employed as a means to minimize overcharging of battery 1 while ensuring battery 2 and 3 reach 100 percent State-of-Charge (SOC). When battery 1 reaches a preset charge to discharge (C/D) ratio in the PMON software, PMON configures to CCM and charges the batteries at 0.75 amps until spacecraft night. By adjusting solar array offset to maintain a constant peak charge current and employing CCM, the FOT has effectively limited overcharging of the battery and improved overall battery performance. Day to day battery operations require monitoring of the battery voltages (including half-battery differential), current sharing, SOC and the spacecraft Beta angle.

In addition to the aforementioned software enhancements, the FOT has also implemented a new charge control strategy to address constantly changing Beta angle. When the battery DOD is between 18 to 20 percent (low Beta angle) and the end-of-night (EON) Load Bus Voltage (LBV) approaches 24.8 volts, the MPS is operated at VT 5 with CCM to obtain C/D ratios between 1.04 to 1.05 on battery 1. When the battery DOD is between 15 to 18 percent and the EON LBV approaches 24.8 volts, the MPS is operated at VT 4 with CCM to obtain C/D ratios between 1.04 to 1.05 on battery 1. When the battery DOD is less than 10 percent (high Beta angle) and the temperature delta between battery 1 and 2 approaches 5 deg. C, the MPS is operated at VT 3 with CCM to obtain a C/D ratios between 1.04 to 1.05 on battery 1. Operating the CCM switch to maintain battery 1's C/D ratios between 1.04 and 1.05 has aided in improving charge acceptance and load sharing between all batteries.

Both battery load-sharing and battery temperatures are good indicators of battery performance which may identify the most efficient battery and the weakest battery. For example, battery 1 has had the greatest half-battery differential voltages and the highest temperatures. It frequently accepts the most charge current while providing the least discharge current, and hence is identified as the weakest performer.

In addition, the weakest performer has been the battery receiving the greatest overcharge. The battery charge method in place for the early part of the mission -- charging at VT 6 to a system C/D ratio of 1.00, then switching to VT 5 (resulting in total C/D ratios of 1.1 - 1.25) may have overcharged the batteries.

Aggressive management of overcharge has been the most effective method of stabilizing and improving battery performance. Battery temperatures, delta temperatures, and load-sharing during charge and discharge have all trended back to more nominal behavior.

Battery exercise also helps to limit overcharge during low load (high beta angle/minimum spacecraft night) periods. Adjusting the SA offset to achieve a power negative condition, allowing the batteries to "spiral down" in SOC for several orbits, exercises the batteries during those low load periods. The result is a DOD of 12-18% at least once per day over a week when the DOD's would normally be much smaller or zero.

Deep discharges have been performed during the bi-annual Full-Sun periods in an attempt to improve battery performance. UARS utilizes these very low load (0-6% DOD) intervals to condition the batteries through low rate, deep discharges (up to 40% DOD) followed by low-rate recharge. This activity is also aimed toward maintaining and/or boosting EON LBV.

EXPLORER PLATFORM/EXTREME ULTRAVIOLET EXPLORER

Mission Summary

The Explorer Platform/Extreme Ultraviolet Explorer (EP/EUVE) spacecraft is a LEO satellite launched by the United States Air Force on a McDonnell Douglas Delta rocket on June 7, 1992. The spacecraft orbits at an altitude of 517 kilometers with an inclination of 28 degrees. The spacecraft length of day varies from 58 to 68 minutes due to the spacecraft's orbital precession of -6.7 degrees/day with respect to the Earth. The Explorer Platform was designed to accommodate a variety of remote sensing, LEO missions requiring solar, stellar or earth pointing over its mission life of 10 years. The payload instruments and equipment can be exchanged during shuttle-based servicing missions. The current EP primary mission payload, EUVE, operates continuously, providing a consistent and stable loading profile on the spacecraft power subsystem.

EP/EUVE's power is provided by a modified MPS that is rendered unique by its inclusion of a heat pipe along the battery baseplate. Solar power is provided by 2 SA Wings, which are rotated by 2 mission-unique solar array drives (SAD). These drives are primarily commanded by flight software, with manual commanding available as required.

As a result of battery anomalies observed on the CGRO and UARS spacecraft in 1991 and 1992, the EP FOT began to implement new modes of operation to enhance the cycle life of the batteries.

Battery Management

The EP/EUVE spacecraft uses a combination of several battery controls to maintain a consistent battery performance. These controls include thermal regulation of the batteries, CCM at orbital sunrise and at battery full-charge, and SA offsets.

Battery temperature regulation was implemented to maintain a specific battery temperature operating range by TMON Processor Control. This can be performed efficiently on the EP spacecraft because the heat pipe maintains a stable thermal environment between all three batteries. TMON samples the battery

baseplate temperature and commands the battery heater thermostat bypass on and off to maintain the baseplate temperature between 5°C and 8°C. This method of operation was introduced five months after launch. In a trial period just prior to this, the baseplate was maintained at 2°C minimum. At launch, the batteries had been thermostatically maintained between -2°C and 0°C.

CCM for the cold array case was implemented to reduce the high current from the SA to the batteries when the arrays are cold (at the beginning of each orbital day). The current operational goals limit the inrush current to less than 20 amps per battery. This is implemented by an Orbit Time Processor (OTP) Command flag which trips approximately 2 minutes prior to the beginning of each orbital day. It commands the 3.0 Amp CCM for a user-defined duration, then resets the SPRU to VT control. The original operational implementation, for 10 minutes following orbital sunrise, was introduced on February 3, 1993. There were subsequent experimental implementations of 15, 20, 25 and 30 minutes following orbital sunrise. The present operational mode, for 10 minutes following orbital sunrise, was re-implemented on March 12, 1993.

CCM for the hot array case (at full charge) was implemented to maintain C/D ratios between 1.02 and 1.07 to minimize battery overcharge. When the batteries reach full charge, the SPRU commands 0.75 Amp constant current mode to maintain a trickle charge on the batteries. The C/D ratio goal is based on the assumption that when the battery reaches 100% SOC at a specified 0.98 PMON battery charge efficiency, the C/D ratio is approximately 1.02. TMON commands CCM when the state of charge on any battery reaches 100% for 2 consecutive counts of the TMON processor (≈ 32 seconds). The original operational implementation, based on just battery 3 reaching 100% SOC, was introduced on March 15, 1993. The present operational mode, based on any battery reaching 100% state of charge, was implemented on January 3, 1994.

VT changes have been performed twice thus far in the EP/EUVE mission. Both changes were made in an effort to minimize overcharging of the batteries. VT 5 was lowered to VT 4 on launch day when the C/D ratio was approximately 1.3, and further lowered to VT 3 on May 5, 1993, when the C/D ratio was approximately 1.1.

The SAs are maintained at a 40 degree effective offset to the sun. This offset was introduced to provide a thermally stable environment for the SA based on the specular reflection problem identified during the thermal envelop testing conducted in August of 1993. Prior to this, the solar array drives remained powered off and fixed at 90 degrees with respect to the $-X_{ACS}$ axis on the spacecraft. Following the in-orbit-checkout of the SADs in July of 1993, the SAs were maintained manually at a 40 degree offset until a flight software patch could be developed to maintain the user-defined offset angle. The current flight software management of the offset angle was begun on November 29, 1993.

The EP/EUVE spacecraft management combines these techniques into a generic spacecraft orbit. The batteries remain in VT 3 control with the battery temperature regulation. The 3.0 Amp CCM occurs for 10 minutes at orbital sunrise, followed by VT control to full charge, then 0.75 Amp CCM until orbital night. This generic power management orbit successfully limits battery overcharge, thus extending the life of the EP/EUVE mission.

CONCLUSIONS

Degraded performance has been observed on several NASA missions employing 50 Ah NiCd spacecraft batteries. Each mission's Flight Operations Team (FOT), along with their respective GSFC Project Personnel and engineers from GSFC's Space Power Application Branch, has closely monitored the battery performance and implemented several charge control schemes in an effort to extend battery life. Various software and hardware solutions have been developed to minimize battery overcharge, and implemented with success. New operational procedures continue to be developed on-orbit. These new procedures may have application in the management of other spacecraft batteries, and may also serve as useful design considerations for future spacecraft power systems.

GLOSSARY

CCM	Constant Current Mode	PMON	Power Monitor
C/D	Charge/Discharge	PPT	Peak Power Tracking
DOD	Depth-of-Discharge	RTS	Relative Time Sequence
EON	End-of-Night	SA	Solar Array
FOT	Flight Operations Team	SAD	Solar Array Drive
LBV	Load Bus Voltage	SOC	State-of-charge
LEO	Low-Earth Orbit	SCP	Stored Command Processor
MPS	Modular Power System	SPRU	Standard Power Regulator Unit
OBC	On-board computer	TMON	Telemetry Monitor
		VT	Voltage-temperature

MISSION OPERATIONS MANAGEMENT

David A. Rocco
Loral AeroSys
7375 Executive Place
Seabrook, MD. 20706

ABSTRACT

Redefining the approach and philosophy that operations management uses to define, develop, and implement space missions will be a central element in achieving high efficiency mission operations for the future. The goal of a cost effective space operations program cannot be realized if the attitudes and methodologies we currently employ to plan, develop, and manage space missions do not change. A management philosophy that is in synch with the environment in terms of budget, technology, and science objectives must be developed. Changing our basic perception of mission operations will require a shift in the way we view the mission. This requires a transition from current practices of viewing the mission as a unique end product, to a "mission development concept" built on the visualization of the end-to-end mission. To achieve this change we must define realistic mission success criteria and develop pragmatic approaches to achieve our goals. Custom mission development for all but the largest and most unique programs is not practical in the current budget environment, and we simply do not have the resources to implement all of our planned science programs. We need to shift our management focus to allow us the opportunity make use of methodologies and approaches which are based on common building blocks that can be utilized in the space, ground, and mission unique segments of all missions.

INTRODUCTION

Over the last several decades the space program has moved from an unbroken series of spectacular successes to a disquieting number of stunning failures. These failures have affected all participants in the space community: DoD, NASA, NOAA, and the commercial sector. On the surface there appears to be no common thread: booster failures; kick motor failures; unsuccessful shroud separations; component level failures; or operator error at the command console.

We seem to be back on the road to success. The Hubble Servicing Mission and GOES 8 launch have broken the streak of recent failures, but have we really solved the underlying problems that have been causing our recent failures?

The space community, like government and industry in general, has become a victim to a system of management that has become mired in bureaucracy and inefficiency.

TOTAL MISSION MANAGEMENT

The first, and possibly most important, step in redefining mission management, is the development of an integrated management approach. In our current organizational environment there are simply too many levels of management, too many discrete organizations, and a diluted system of responsibility, authority, and accountability.

This type of organizational structure fosters inefficiency, duplication of effort, convoluted lines of communications, and in the final stages of a mission, cost and schedule overruns or total mission failures.

Placing a satellite into orbit and conducting mission operations is an immensely complex task in its own right. Adding in additional levels of confusion and complexity that are a function of over management just makes a difficult task harder to accomplish and adds unnecessary risk to the program.

A typical DoD or NASA mission possesses three major management tiers:

1. Program Management
2. Project Management
3. Mission Management

Below these major tiers are the subsystem level management groups that oversee the design and implementation of mission components and functionality. This multi-tiered approach lends itself to inefficiency, redundancy, and duplication of effort. Each lower tier of management is larger than its preceding tier and adds to the bureaucracy, extends lines of communication, and dilutes authority.

The only way to eliminate this problem is to redefine the management organization. While the three levels must continue to exist, the numbers of personnel and the functions performed must change radically.

Program Management

Program Management must continue to exist at the agency level. The Program Level is responsible for overall budget, schedule, and interagency coordination, but these must be the *only* functions that Program Level

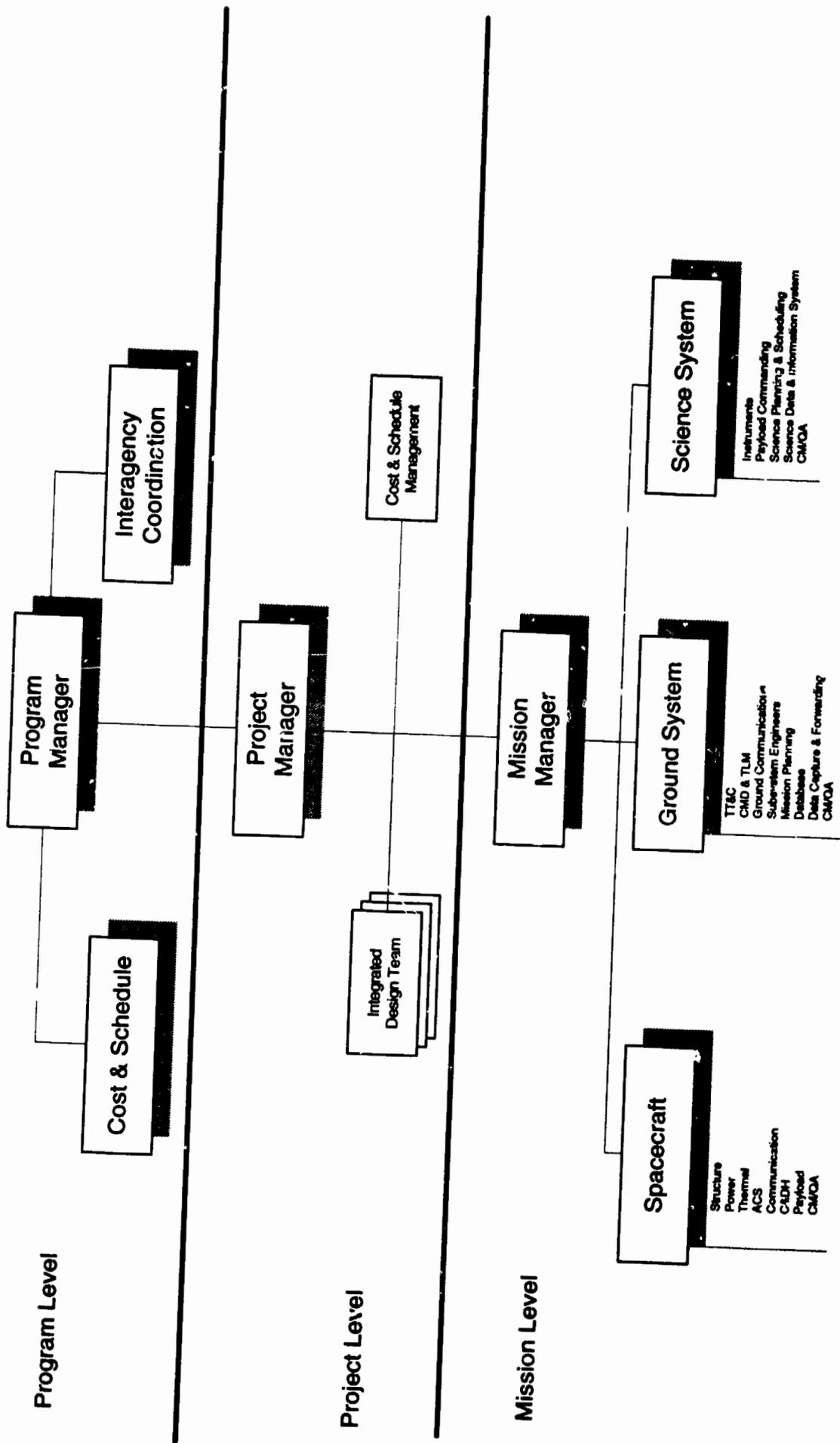
management performs. Micro managing the spacecraft, ground segment, or science compliment should not be the concern of this level of management.

Project Management

Project Management should continue to exist at the implementing center, but the focus of Project Management should change drastically. Project Management should shift its level of activity from overseeing the overseer of implementation to serving as the spearhead for planning mission operations. This planning should be performed with a core team of representatives from the space, ground, and science communities from the very beginning of the mission planning process.

With the major mission segments participating in an integrated initial mission planning process directed by the Project Manager, problems that are normally identified late in the implementation phase can be rectified or even avoided early in the mission development process.

Mission design should be the operational focus of Project Management, with cost and schedule management as a secondary responsibility. An organization tasked with this responsibility would significantly shrink the personnel requirements of the Project Level. The mission design itself should be driven by what is most practical in terms of meeting the science mission objectives and allow the scientific and ground system considerations to drive the design of spacecraft subsystems as opposed to our current method of building the mission around the platform. Figure 1 depicts the proposed organization structure.



Mission Management Organization
July 14, 1994

Most missions that actually achieve orbit and successfully complete the early orbit checkout phase tend to outlive their effective design lives by several hundred percent. The stretching of the on-orbit operations phase of the system lifecycle tends to make Operations & Maintenance (O&M) one of the most expensive elements of the mission. For example, on the Hubble Space Telescope, the cost to place the spacecraft into orbit with its supporting ground system was approximately \$ 2.1 billion. The O&M budget estimated in 1990 was \$ 200 million per year. With a fifteen year on orbit life, the cost of O&M will exceed the cost to launch by 50%.

Other missions which have exceeded their planned lifetime such as NIMBUS-7, ICE, IUE, ERBS, IMP, Solar Max, and Landsat 4 & 5 have exceeded this O&M cost factor by several hundred percent as depicted in Figure 2.

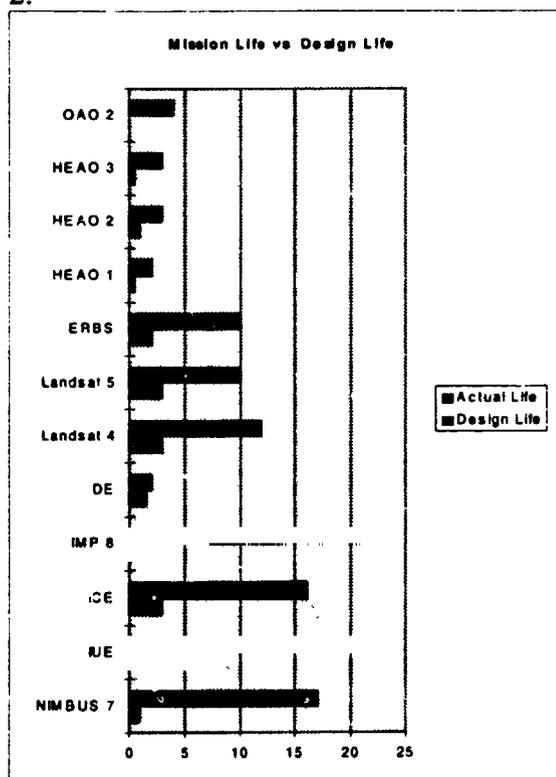


Figure 2 Mission Life vs. Design Life

By allowing the science and ground system elements to drive the design of the spacecraft, and by updating the technology that is used to control the spacecraft and process the mission data, significant reductions in O&M costs can be realized. The fact that these savings can be real and significant over time are borne out by the high level of interest in low cost mission operations concepts such as JPL's LOCOMO and GSFC's Renaissance programs.

This macro level of mission design and sustaining support is where the Project Level should concentrate its efforts.

Mission Management

Mission Management which currently lives as a small component of the Project Level, and the major component of the on-orbit level of management should shift its focus from a mostly on-orbit organization to the management of the mission implementation as well as conducting day-to-day mission operations.

By placing responsibility and authority for the implementation into the hands of the organization which must live with the results of the final system, two key outcomes will materialize:

1. A system development monitored and managed by the actual users will result in a system that is designed with operations in mind.
2. The operations managers become a true stakeholder in the total mission system and an organization that can blame no one but themselves for a poor or overly complicated system.

METHODOLOGY

With an organizational structure in place which has the authority and accountability to make major design decisions, the primary methods required to create a successful design are:

1. A systems approach to the mission.
2. An understanding of what technology is available that can support a low cost mission design.
3. A clear vision (operations concept) of how the mission will be conducted.

Systems Approach

Under a systems approach, the mission is viewed as a total system that consists of five major components:

1. A science objective.
2. A management approach.
3. A spacecraft and instrument suite.
4. A ground support system.
5. A mission operations plan.

These components exist as individual threads which are intertwined to form a common cord of mission design. As a system, any changes to any given thread will have some impact on the overall mission design. As a system these threads must function in concert to achieve the end goal of a space mission that meets its scientific objectives for a reasonable cost.

Technology

When a clear vision of what the mission is intended to look like is developed, the integrated design team must evaluate the available technology and determine what components or approaches will best meet the

requirements for the total mission. From a technology standpoint the following questions must be answered:

1. How can technology be used to lower mission risk while reducing overall costs?
2. Where will technology take us in terms of spacecraft, ground systems, and support infrastructure?
3. How can operations concept developers use evolving technology to lower O&M costs?
4. How can we plan and design for tomorrow's missions when the state-of-the-art is advancing so rapidly?

The intelligent use of technology has to be an integral element of operations management's philosophy. We are beginning to see this happen as concepts from Total Quality Management (TQM) move from a buzz word phase into actual implementation. Integrated product teams are becoming more common, and in some agencies are officially tasked to develop designs driven by a low risk and low cost operations model.

Technology is also important in terms of consolidating operations to achieve budget goals. The USAF and NOAA are currently heavily involved in planning for a converged polar meteorological program where a single spacecraft type and single ground control element operate a mission to serve both civil and military users.

Spacecraft Trends

The spacecraft itself can become a major means of reducing both cost and risk to the total mission design. New generation On-Board Computers (OBCs) are capable of providing 256K of memory, coupled with micro-processor controlled instrument and spacecraft subsystems, a capability exists to

build very high levels of autonomy into the spacecraft itself. The addition of products into the spacecraft such as Global Positioning Satellite (GPS) receivers can provide a spacecraft capable of generating its own on-board ephemeris, performing fine attitude determination (2 receiver/4 antenna configuration), and configuring itself for ground contacts. All of this can be done now, with greater accuracy than is currently provided by ground or TDRSS based tracking. It can also be done at a fraction of the staffing levels we currently need to perform these services on the ground.

The questions that need to be asked at the design phase are:

1. Is this capability required for this mission?
2. Will this capability save me money and reduce risk over the total lifecycle of the mission?

If the answers to either of these questions are yes then a cost/benefits analysis must be conducted to determine:

1. If these capabilities are needed to ensure mission success and reducing risk.
2. How much money can be saved during on-orbit operations by spending a more on the spacecraft.

This may make life more complicated for the spacecraft designer, but the spacecraft designer is only one player in the mission systems.

Ground Systems

Ground system design and capabilities have matured at the greatest rate because the ground system is not constrained by the environmental requirements the spacecraft

must withstand. Ground system technology is also directly tied to computer hardware, software, and networking technology, and we have the ability to access the ground system on a daily basis. Although the potential capability in this area has improved significantly, the implementation of this technology has lagged.

The centralized ground system support architecture was designed and implemented in the 70's using a mainframe based approach. This approach made sense because an economy of scale could be achieved when many missions shared a common service. However, advances in ground system technology have made the cost savings of the 70's a cost sink in the 90's.

Existing Commercial-Off-The-Shelf (COTS) hardware and software have the ability to reduce or eliminate our reliance on large institutional support elements which have extensive O&M requirements. Advances in ground system telemetry front-end processors have reduced the workload now performed by Pacor to the level of a few programmable cards which perform all tasks from bit synch through Reed Solomon correction. A two GPS configuration on the spacecraft itself can reduce support requirements from the Flight Dynamics Facility (FDF) from daily staffing to launch and ascent support only.

In the final analysis the ground system can be reduced to four major components:

1. A tracking facility.
2. A communications segment.
3. A control center.
4. A science operations center.

This approach provides a control center capable of directly providing Level 0 data and telemetry directly to a science operations center. The key infrastructure support element in this scenario is a reliable communications infrastructure.

Science Operations Centers (SOC)

The availability of inexpensive multi-processor workstation technology has an unlimited capability to reduce the costs associated with science data processing and product generation in terms of both the computer resources required to perform the tasks, and the science operations staffing levels needed to control and monitor the product generation process.

With science data and supporting telemetry being provided directly to the SOC by the flight control center, a multiprocessor product generation environment can allow science product operations to be reduced to a single shift activity, and at the same time minimize the physical facilities and personnel requirements in the SOC.

Operations Concepts

The final element in redefining operations management is the development of mission operations concepts that will allow automation and smart technology to provide the majority of the "cradle-to-grave" monitoring and support tasks for on-orbit missions. How this task is handled can have a considerable impact in reducing O&M costs. These tasks are now performed by implementing round the clock staffing. To cover a nominal mission day, a staffing factor of 4.0 persons per position is required to provide the *minimum* level of staffing needed to provide real-time spacecraft services. In the typical control center this factor is

applied to the Shift Supervisor, Command Controller, Ground Controller, and Payload Evaluator.

Using approaches such as compressed health and safety telemetry schemas, on-board ground contact configuration capability, and exception reporting, can significantly contribute to reduced (50-66%) control center staffing requirements.

The same types of multiprocessor technology recommended for use in the SOC combined with COTS statistical analysis software can be employed in the area of spacecraft subsystem analysis. Traditionally this function is performed using custom developed software, and resides on either a dedicated machine, or is resident within the command and telemetry processing system.

This newer technology approach provides scalability and portability that does not currently exist in off-line ground systems tasks, and reduces the operational load on the real-time system. These off-line tasks; such as mission planning and scheduling, subsystem level telemetry analysis, and long term performance trending lend themselves to this type of solution because they are normally Monday through Friday day shift tasks which do not require sustained levels of time-critical performance.

This scalable approach also allows the addition of increased capability to be achieved by using board level components and cross compiling existing software as opposed to adding new workstations or personnel into the control center to meet new requirements. In its most advanced phase, this architecture can conceivably provide multiple satellite support from a single operations center.

CONCLUSION

We must begin to embrace the mission as a comprehensive system, not as a series of discreet components which are pulled together to and literally beaten into a configuration to perform a unified task.

With proper levels of planning and the support of high level agency management, a macro-level mission approach can be developed that will allow resources to be re-directed into new missions. As a result of organizational downsizing on a mission level project, we can minimize some of the confusion and develop clear lines of communication, authority, and responsibility.

In the final analysis people will always be the most expensive component of any mission. Any personnel resources that can be eliminated from a mission provide two benefits:

1. A real cost reduction for the current mission.
2. A resource which can be applied to a new mission which up to now have not been able to secure the resources required to move from the concept into the implementation phase.

REFERENCES

- Barter, N. (1992) TRW Space Data, TRW Space & Technology Group, Redondo Beach, CA.
- Davis, E. (1993) A New Paradigm for Very Low Cost Mission Operations, JPL, Pasadena, CA.
- Larson, W. and Wertz, J. (1992) Space Mission Analysis and Design, Second Edition, Microcosim Inc., Torrance, CA.
- Lillesand, T. and Kiefer, R., Remote Sensing and Image Interpretation, Second Edition, John Wiley and Sons, New York, NY.
- NASA GSFC, (1994) Mission Requirements and Data System Support Forecast, Greenbelt, MD.
- NASA GSFC (1994) Renaissance, Greenbelt, MD.

GRTS Operations Monitor/Control System**Richard A. Rohrer****Allied Signal Technical Services Corporation
One Bendix Road
Columbia, MD 21045****ABSTRACT**

An Operations Monitor/Control System (OMCS) was developed to support remote ground station equipment. The ground station controls a Tracking Data Relay Satellite (TDRS) relocated to provide coverage in the tracking system's zone of exclusion. The relocated satellite significantly improved data recovery for the Gamma Ray Observatory mission. The OMCS implementation, performed in less than 11 months, was mission critical to TDRS drift operations. Extensive use of Commercial Off The Shelf (COTS) hardware and software products contributed to implementation success. The OMCS has been operational for over 9 months with no significant problems. This paper will share our experiences in OMCS development and integration.

INTRODUCTION

An increase in tape recorder error rates onboard the Gamma Ray Observatory (GRO) spacecraft necessitated alternative approaches to data gathering for the GRO mission. The project resorted to collecting science data using full period, in view, Tracking and Data Relay Satellite (TDRS) return link services. The TDRS system could not track the GRO spacecraft in the zone of exclusion. In addition, view periods are frequently restricted by spacecraft body blockages. Using the two available spacecraft, TDRS West and TDRS East, the project could retrieve only 50% of its science data.

To increase GRO viewing opportunities, NASA decided to implement a Southern Hemisphere TDRS ground station and move a TDRS over the Indian Ocean. The GRO Remote Terminal Subsystem (GRTS) Operations Monitor/Control System (OMCS) was developed in response to the requirement to provide means to remotely control and monitor the Southern Hemisphere ground station located at the Canberra Deep Space Communications Complex (CDSCC) at Tidbinbilla in the Australian Capital Territory.

The driving requirements for the OMCS were:

- Equipment monitor and control - The main function of the OMCS is to monitor the remote ground station equipment and provide status updates within 5 seconds. It was also required to control all critical functions for the ground station equipment for configuration and fault detection/failure.
- Provide operator workstations at geographically diverse locations - The main TDRS system control center is located in White Sands, New Mexico. It was decided that the TDRS satellite controllers would operate the station in Australia remotely from the United States with CDSCC personnel providing on site operations support for the first year. The White Sands location was designated the Extended TDRS Ground Terminal (ETGT) and the CDSCC location was designated the Remote Ground Relay Terminal (RGRT).

- **Minimize operator workloads** - The OMCS should provide a simple graphical user interface to reduce the amount of operator keyboard type-ins. It should also provide preloaded configurations to allow one button operation to reduce operator workload since the operator would also be the TDRS satellite controller with other operational responsibilities.
- **State vector to the antenna subsystems** - The OMCS was required to provide a method of entry, verification, and transmission of the TDRS and GRO state vectors to antenna control computers at RGRT since there was no other method of obtaining the vectors.

The factors that would act as constraints were:

- **Multiple interfaces** - Most of the equipment purchased for the ground station had never been integrated by ATSC before. About 20 new interfaces had to be documented, developed and tested.
- **Real time integration** - Due to the intense development schedule, equipment was to be shipped directly to Australia and integrated at the site.

In summary the OMCS had to be implemented in less than 7 months from project start using equipment to be dropped shipped to its final destination. Quite a challenge.

DEVELOPMENT APPROACH

Meeting the very aggressive schedule required a new approach to developing the system. The decision was made to use COTS products to the maximum extent possible. Several COTS approaches were investigated, but Allied Signal Technical Services Company (ATSC) had implemented a monitor and control system for another customer using an industrial control system. A trade study of current industrial control systems was currently being performed for another project and the results were used to aid in the selection. The advantages of most industrial control systems is a well defined man-machine interface (MMI) and configuration simplicity, i.e. it does not require programming, but uses a database to support the equipment interfaces. The real advantage of an industrial control system was that it allowed rapid development which was key to the success of the OMCS implementation.

The product chosen was the TIS4000 system developed by Tate Integrated Systems located in Owings Mills, MD. The TIS4000 is a distributed data acquisition and control system. The TIS4000 system is database driven and has been designed with a flexible architecture and a modular, distributed construction. This allows it to be configured for a wide range of applications. The basic architecture is based on a "client-server" structure with Motorola 68000 series microcomputers performing the real-time control and processing operations and RISC workstations providing the MMI functions. All of these components are connected via high-speed LANs using TCP/IP networking. The advanced workstation MMI uses the UNIX operating system and the X-Window environment. The MMI provides a full graphical operating environment for operators, as well as a programming and applications development environment for engineers.

The TIS4000 is database driven, i.e. parameters to be monitored or controlled are entered into a database. The database is downloaded into real-time computers, which perform all parameter gathering, limit checking, and alarm notification. The operator displays are

created on the workstations using a graphics editor and then connected to the real-time parameters. This allows parallel development of the database and displays, reducing development time. Each equipment interface can be verified as it is completed and changes easily made. Many of the commercial systems evaluated were based on this type of architecture.

OMCS HARDWARE

Figure 1 provides an overview of the system architecture. At the CDSCC, two workstations were placed at Deep Space Station 46 (DSS46), where the equipment is located and one at Signal Processing Center 40 (SPC40), the main operations area. Two workstations were placed at ETGT, one in the TIC and one in the TOC. An additional workstation was located in the GSFC Network Control Center (NCC) for state vector entry, user services, and administrative functions. Two real-time computers referred to as Input Output Controllers (IOCs) were located in DSS46 and interfaced to the ground station equipment. The operational entities were connected locally by an Ethernet local area network and standard 9.6 kbps lines were used to interconnect the geographically diverse locations. Each workstation can see all of the data provided by the IOCs.

The workstations are standard SUN Sparc IPX workstations with 32MB of memory, an internal 425MB disk and 19" color monitor. During implementation an additional 425MB external hard disk was added. Hewlett Packard LaserJet IV printers were provided. Standard Motorola VME computer components including a 68030 CPU with 16MB of memory and four communications interface cards configured to meet the requirement of 16 serial lines on each. S band Telemetry, Tracking and Control (TTC) functions were allocated to one of the real-time computers. Ku band user services were allocated to the other real-time computers. This arrangement provided some fault tolerance. An Ethernet was installed in Australia and standard bridge/routers are used to extend the LAN to White Sands, NM and Washington, DC.

Figure 2 shows the interfaces to the various RGRT equipment.

OMCS SOFTWARE

Figure 3 illustrates the OMCS software architecture. The COTS products are 1.1, the real-time database processor; 1.2 and 1.3, TCP communications stacks; 1.4, the display manager; 1.5, the display editor; 1.9 and 1.10, device drivers; and 1.12, dBase IV used for building the real-time database and generating reports. The ATSC developed items are 1.6, the state vector entry and transfer; 1.7 and 1.8, configuration macros; 1.13 and 1.15; configuration processes; and 1.14; the checkpoint process.

The TIS4000 vendor developed, under subcontract, the serial driver required to interface the ground station equipment. Due to the schedule this appeared to be the most expeditious method of completing a critical portion of the work.

ATSC engineers working with NASA engineers developed the databases and graphics based on interface information provided by the vendors. Personnel from White Sands provided operations input to the process. Some pieces of equipment were staged at the ATSC facility before shipment to Australia. When the OMCS was shipped in June 1993 confidence was high that successful on site integration would be possible. Two ATSC engineers spend three months on site in Australia completing integration. Additional testing was carried out remotely using the workstation in the NCC, reducing travel

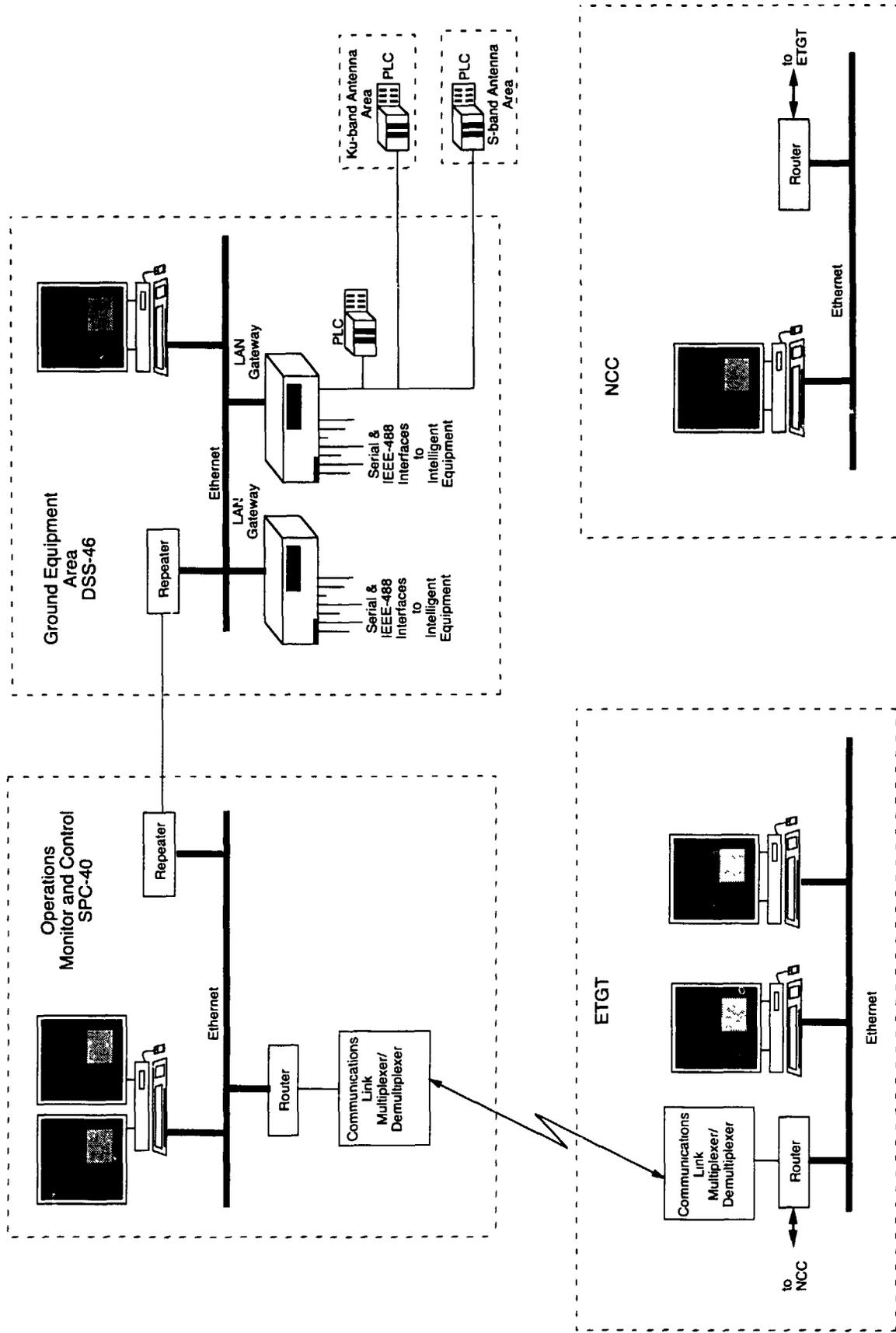


Figure 1: OMCS Systems Architecture

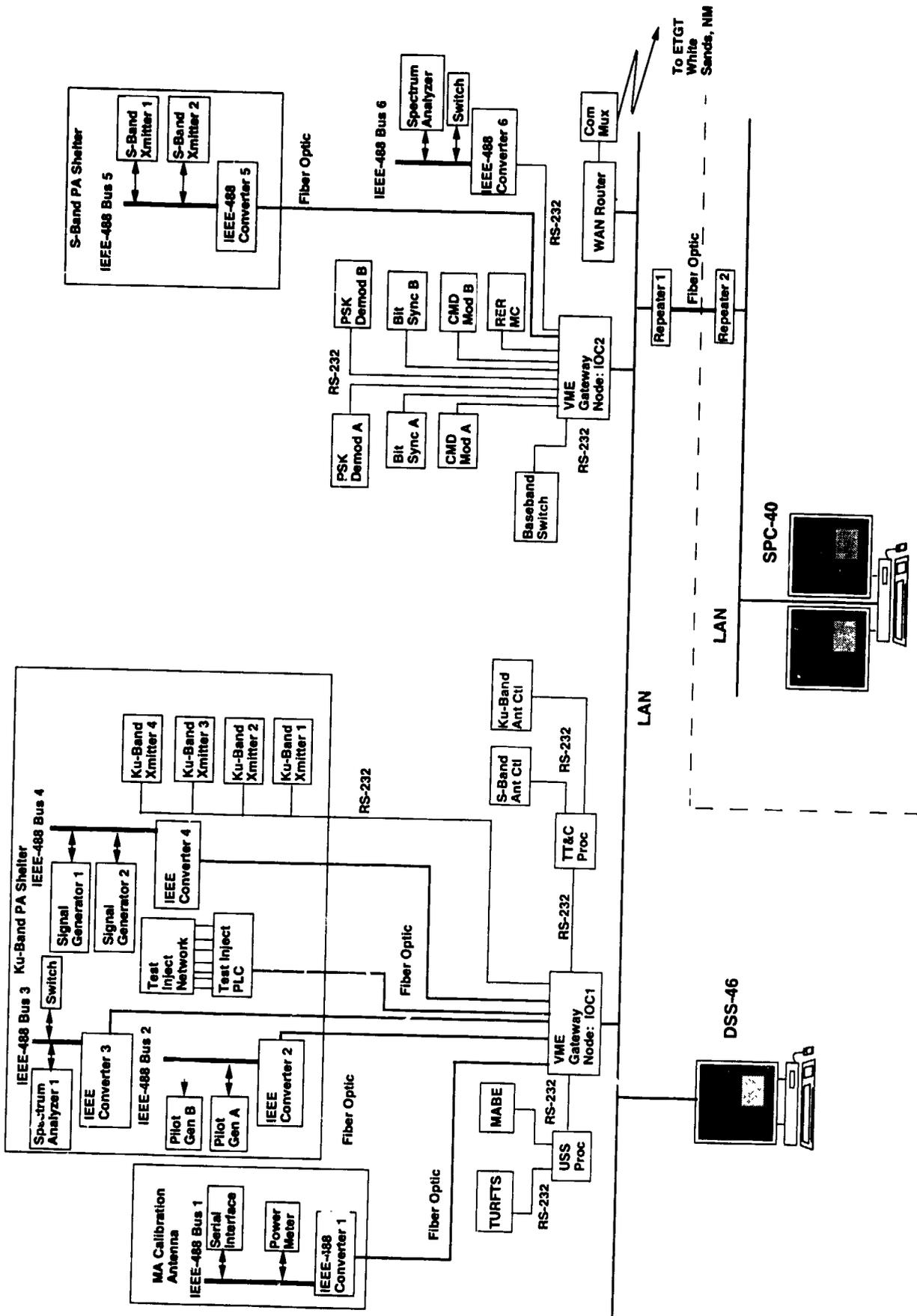


Figure 2: OMCS Equipment Interfaces

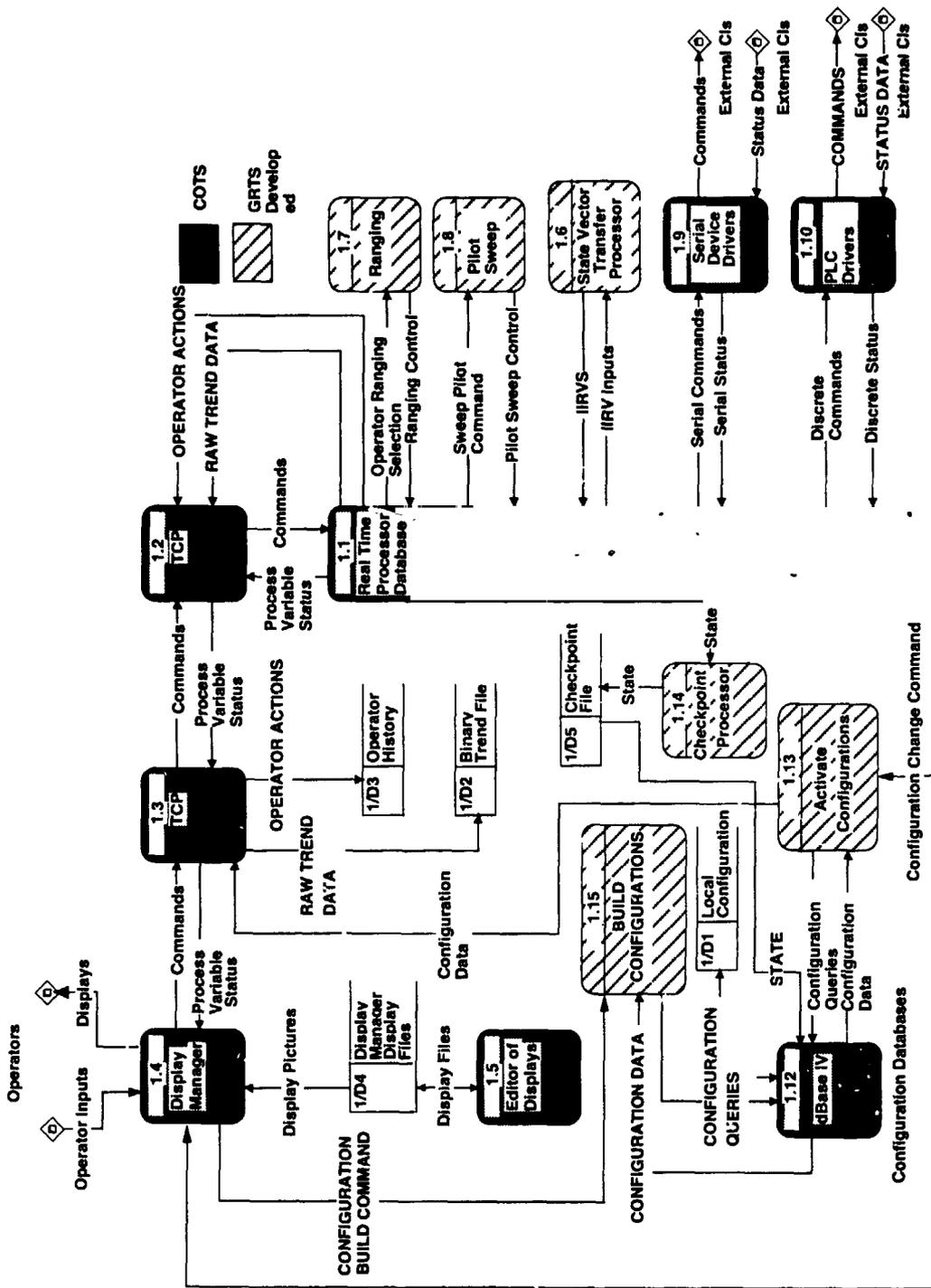


Figure - 3: OMCS Software Context Diagram

requirements. Currently, software upgrades are performed remotely from the NCC workstation.

OMCS CAPABILITIES

The OMCS presents operators with a window into the station. The system works by a point and click method, with operator keyboard entries reduced to the absolute minimum. Normal operations can be performed from a single station overview screen, but the operator has the ability to move to lower levels of detail on any specific piece of equipment if the need arises. Most detailed equipment screens mimic the actual front panel of the equipment, so there is no need for operator retraining on the equipment. If a user can operate the equipment from the front panel, he can operate the equipment from the OMCS.

The OMCS is not a fully automated system, nor is it schedule driven. It requires an operator to initiate and approve activities. A decision was made early in the development program not to attempt full automation until more operational experience was acquired. Therefore, the operator is required to acknowledge alarms and take corrective measures, configure for TTC by selecting active and backup strings, start and stop the uplink carrier, start and stop ranging, and switch in redundant equipment if necessary. The operator also must configure user services and perform Multiple Access (MA) system calibrations. These operator interactions are not burdensome for such a small TDRS ground station.

Automatic configuration sequences referred to as "macros" were developed to reduce operator workload. All normal configurations are performed by the use of "macros". These are predefined routines that set the station up in a predetermined configuration. The use of the macros allows one-button operation. Typically a macro does nothing more than duplicate all steps an operator would perform if configuring the system manually. It checks the appropriate equipment status and initiates control actions in the correct sequence. The macro informs the operator if a piece of equipment is not available, incorrectly configured, or faulted. The operator can then take the appropriate action.

Since the major configuration functions have been "automated" in the form of macros, this provides the first steps towards higher levels of automation if desired. Nothing in the current implementation of the OMCS precludes expanding to full automation or even adding an expert system helper.

CONCLUSIONS

The implementation of the OMCS was a success, but not a trivial undertaking. Many problems had to be conquered. Some caused by ground station equipment, some caused by the COTS software chosen, and some by the implementers understanding of the problems.

This effort represents the second in which a COTS industrial control package was used. The OMCS system implementation was much easier and successful because ATSC has refined the requirements for a COTS system from the previous project. The next project will be easier and less costly to implement because of the experience gained on GRTS.

INTERNATIONAL SPACE STATION ALPHA USER PAYLOAD OPERATIONS CONCEPT

Ronald A. Schlagheck
 Mission Operations Laboratory
 Payload Mission Integration
 Marshall Space Flight Center, Alabama

Elaine F. Duncan
 Payloads Project Office
 Space Station Operations & Utilization
 Marshall Space Flight Center, Alabama

William B. Crysel
 Boeing Defense & Space Group
 Missiles & Space Division
 Huntsville, Alabama

James W. Rider
 Teledyne Brown Engineering
 Space Station User Operations
 Huntsville, Alabama

ABSTRACT

International Space Station Alpha (ISSA) will accommodate a variety of user payloads investigating diverse scientific and technology disciplines on behalf of five international partners; Canada, Europe, Japan, Russia, and the United States. A combination of crew, automated systems, and ground operations teams will control payload operations that require complementary on-board and ground systems.

This paper presents the current planning for the ISSA U.S. user payload operation concept and the functional architecture supporting the concept. It describes various NASA payload operations facilities, their interfaces, user facility flight support, the payload planning system, the on-board and ground data management system, and payload operations crew and ground personnel training.

This paper summarizes the payload operations infrastructure and architecture developed at the Marshall Space Flight Center (MSFC) to prepare and conduct ISSA on-orbit payload operations from the Payload Operations Integration Center (POIC), and from various user operations locations. The authors pay particular attention to user data management, which includes interfaces with both the on-board data management system and the ground data system. Discussion covers the functional disciplines that define and support POIC payload operations: Planning, Operations Control, Data Management and Training. The paper describes potential interfaces between users and the POIC disciplines, from the U.S. user perspective.

(This work was performed, in part, under contract to NASA.)

1.0 USER PAYLOAD OPERATIONS

Figure 1-1 depicts the overall user payload operations environment and its relationship to the integration functions for ISSA [1]. Typically, users become interested and involved in many activities interspersed with the activities listed below as "payload operations." To the user, these other activities may have far more value than the listed "payload operations" activities. However, the authors associate these other activities with results of successful payload operations. For the purposes of this paper, the term *payload operations* refers to activities required to:

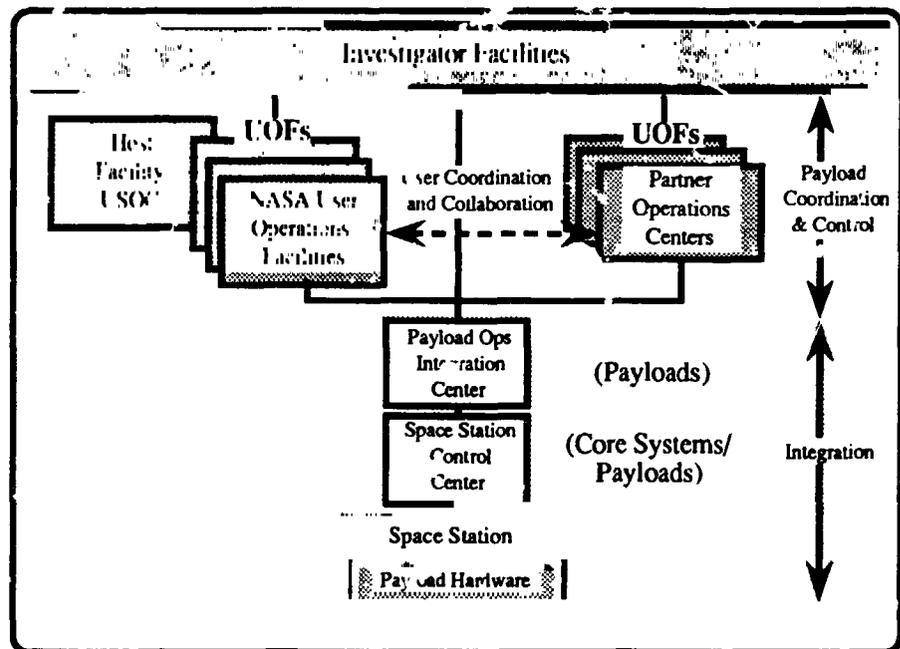


Figure 1-1. Overall Operations Concept

424

- Plan/maintain equipment to perform its intended function
- Observe safety and equipment performance
- Monitor station resources required and/or consumed
- Analyze actual "on-orbit" performance versus objectives
- Adjust or alter equipment performance
- Manage data generation and collection
- Collect and preserve results
- Prepare equipment/samples for return to the ground

1.1 FACILITIES AND INTERFACES

The following sections describe locations and some of the management interfaces involved in user payload operations. Users in all locations operate their payloads through the ISSA Payload Operations Integration Center (POIC). (Section 2)

1.1.1 User Operations Facilities

The NASA Office of Life and Microgravity Science and Applications, and the Office of Advanced Concepts and Technology, are considering development of facilities from which users may conduct payload and science operations. Initially, NASA will develop these User Operations Facilities (UOFs) to support and operate major on-board payload facilities that support more than one user. NASA will develop UOFs at Lewis Research Center, Cleveland, Ohio, for the fluids and combustion facility; Johnson Space Center (JSC), Houston, Texas for life science and biotechnology facilities; MSFC, Huntsville, Alabama for microgravity science facilities; Ames Research Center, Moffett Field, California, for non-human life sciences and the centrifuge facility [2]; and at Langley Research Center, Hampton, Virginia, for commercial and technology facilities. Figure 1-2 depicts the distributed user facilities infrastructure.

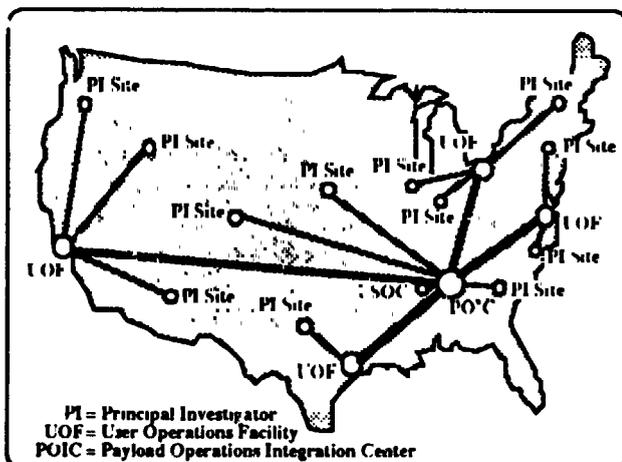


Figure 1-2. Distributed User Operations

1.1.2 United States Operations Center

The ISSA Program covers development of the United States Operations Center (USOC), collocated with the POIC in Huntsville, AL. This facility will provide floor space and operations consoles from which user teams may conduct payload operations. The ISSA Program took advantage of POIC development requirements in choosing both the location of the USOC and its processing and communications equipment.

To support the POIC, the ISSA Program developed capabilities to monitor payload health and status displays, process messages, command payloads, and conduct limited payload systems analyses. The program will make these capabilities available to USOC users and to UOF users who install compatible equipment, at no additional cost to the program. UOF developers have shown extreme interest in obtaining these capabilities.

At present, the only personnel the USOC may provide for user flight support will provide support for user data flow and data products, and for engineering support for the EXPRESS racks and pallets. EXPRESS stands for Expedite Processing of Experiments to Space Station. For payload operations, users will interface with the POIC either directly, or through *user-provided* operations integration teams within the USOC or UOF.

1.1.3 Remote User Facilities

Users may locate anywhere, from commercial facilities and university laboratories to NASA payload development centers. Some Space Station users desire to control laboratory experiments on-orbit from the user's home facility. If a user plans to control or execute real-time or "near real-time" activities defined in Section 1.0 as "payload operations" activities, from his home facility, the user should plan a constant interface with POIC, either directly or through a UOF. The POIC will require this interface to address safety concerns and assure that the payload's operation does not interfere with other payloads. No real technical obstacles exist to implementing operations from user home facilities, but users may incur some implementation costs.

Figure 1-3 [3] shows three remote payload operations scenarios, involving payloads of varying complexity and an experimenter operating from a remote home site.

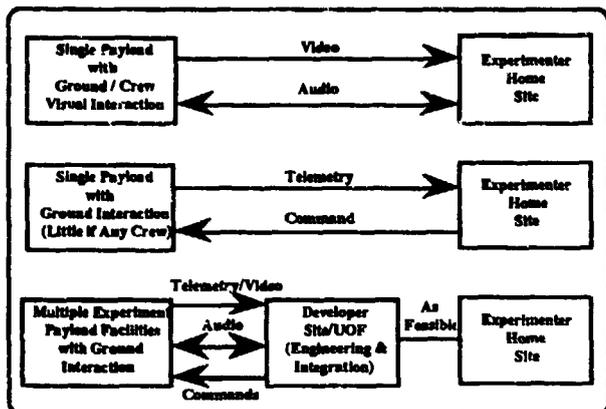


Figure 1-3. Payload Remote Operations Scenarios

1.1.4 Interfaces among U.S. User Facilities

Figure 1-4 [4] depicts interfaces users should consider in preparing for payload operations. Not all users will require all interfaces, but a careful check of each area will increase the effectiveness of payload operations planning.

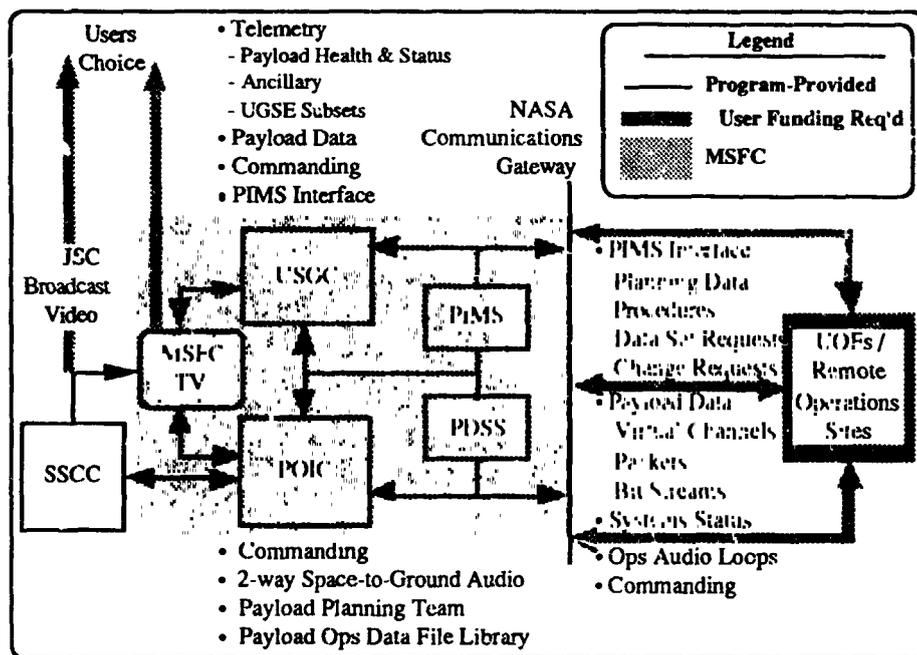


Figure 1-4. Potential User Interfaces

UOF-provided support should include:

- Planning information, procedures, change request interface
- Payload data interface
- Compliance with NASA Automated Information Security
- Payload Operations Personnel to interface with the POIC
- Training on UOF systems and capabilities as required
- Support for integrated training and simulations

1.2 PAYLOAD PLANNING CONCEPT

The Payload Planning System (PPS) supports distributed planning concept [5]. Users may define their requirements to varying levels of detail, and according to the kind of planning center support available. The distributed planning concept recognizes both individual user planning and user-developed user planning centers. The concept includes centralized payload planning, probably within the POIC. Figure 1-5 depicts the planning concepts [6], and the following describes user involvement in distributed payload planning.

For *centralized planning*, users submit very detailed planning requirements to the POIC. The POIC will schedule payload activities and distribute the results for user review and analysis. Users will provide final detailed requirements based on the POIC-provided schedule. The POIC will prepare detailed schedules and generate payload planning products.

For *user windows*, the POIC distributes envelopes of time periods and station resources available to individual users. User requirements for flexibility in scheduling form the basis for these windows. The user window may accommodate a single payload activity, or a choice of activities. Users schedule operations within the window and prepare their detailed schedules

For *planning center envelopes*, the POIC distributes envelopes to user-developed user planning centers based upon planning center defined gross scheduling requirements. User planning centers may schedule payload operations activities

for the users served. User Planning Centers may integrate the requirements of the users served, and produce a plan by which these users will operate. The POIC will make Payload Planning System software available to user planning centers at no additional cost to the program. To ensure crew safety and non-interference with other payloads, the POIC will review and approve user planning center envelope plans.

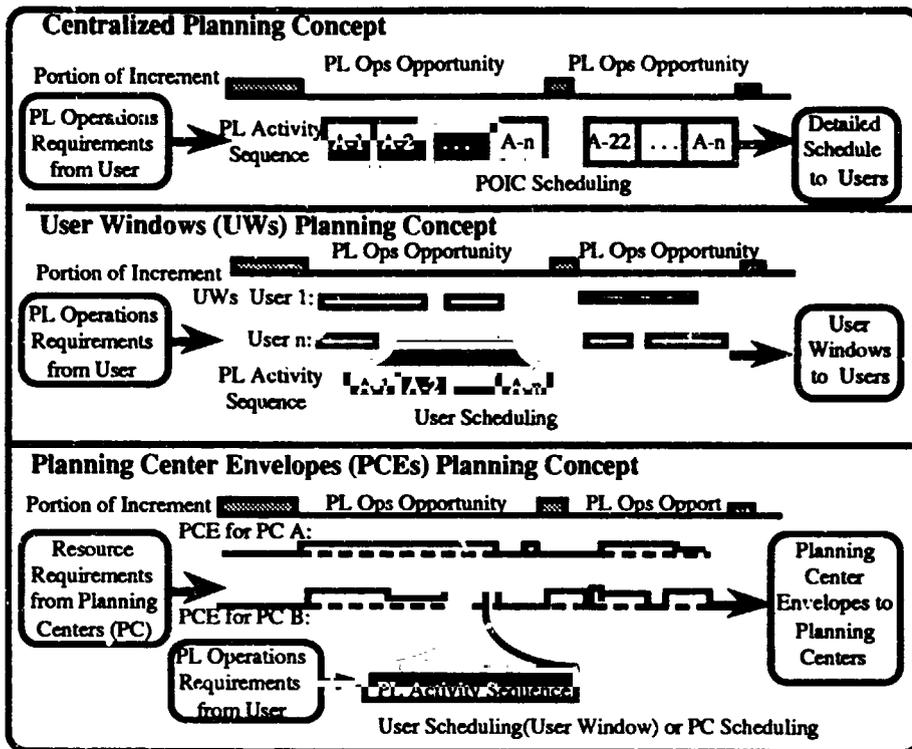


Figure 1-5. Planning Concepts

The POIC will integrate all user payload planning, regardless of where it was done, into a single integrated payload operations plan and provide this plan to the station systems planning center. The system planning center will incorporate the payload plan into an overall station operations plan. System and payload planners will resolve discrepancies and provide users with access to an agreed upon plan. Users will review and request changes to this plan through the Payload Information Management System (PIMS) interface.

1.3 ON-BOARD DATA SYSTEM CONCEPT

Designers refer to the ISSA data management system as the Command and Data Handling System. This system provides hardware and software computational resources to, (1) support ISSA core systems command and control; (2) support the payload users; and (3) provide services for flight crew and ground operations [7].

The Command and Control computers, implemented with Multiplexer / Demultiplexers (MDM) as the primary hardware device, constitute the highest tier of the architecture. They provide the point of control for subtier systems, payloads, and International modules. MDMs gather sensor and effector data through standardized analog and

digital instrumentation interfaces and provide command and control of subtier elements through Mil-Std-1553B and RS-422 serial data buses. The Command and Control computer controls the Communication and Tracking subtier element equipment. This equipment provides the on-board audio and video, uplink and downlink, extra-vehicular, and orbiter communication [7].

Uplink and downlink communications use the Tracking and Data Relay Satellite System (TDRSS). The communications design implements uplink audio communication, and core systems downlink, with a single fault-tolerant S-band communication system limited

ed to 72 kbps for the uplink and 192 kbps for the downlink. The S-band system provides all command uplink and file transfer and all operations safety data downlink. The current design also provides a zero-fault-tolerant Ku-band communication system, limited to 50 Mbps to downlink payload telemetry. The Ku-band system downlinks all payload data and on-board video generated by the internal video systems. A communications outage recorder will capture data during communication outages with the ground. The POIC can schedule this data for playback or downlink at a future time. All communication to and from the ISSA will meet the Consultative Committee on Space Data Systems (CCSDS) telemetry standard.

The payload subtier element interfaces to the Communications and Tracking subtier element for use of uplink, downlink, and video services. The Payload MDM serves as the "command and control" computer for the Payload data architecture and point to point/bus communication media for payload data transfer. The payload subtier provides command/control media, high rate data communications media (<100 Mbps), medium rate data media (< 10 Mbps) and multiple rack-to-rack communications media (≤ 10 Mbps).

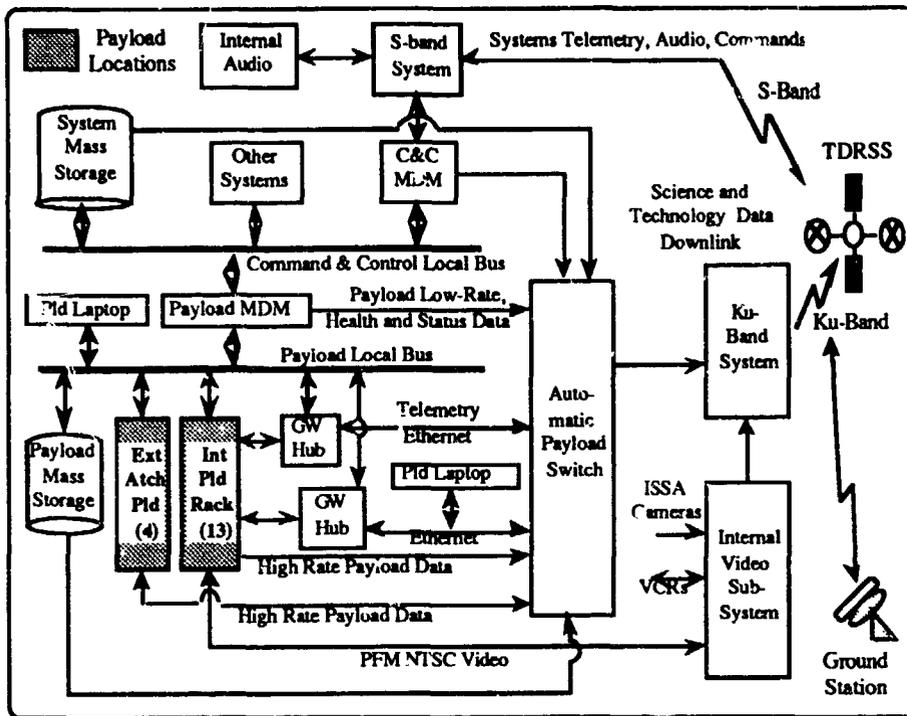


Figure 1-6. On-Board Data Architecture

Figure 1-6 depicts the on-board data architecture. The payload MDMs (one primary and one cold backup) provide overall U.S. payload complement command/control and monitoring functions. They provide the interface with the Command and Control computers for resource allocation, passing of payload safety parameters and receipt of ground commands. The design configures payload MDMs as remote terminals on the Command and Control MDM 1553B bus. They serve as the bus controllers on each of the four (4) 1553B payload local buses. The payload MDMs have high rate data link interfaces to the automated payload switch which provides a path for downlink of health and status, ancillary, and low rate telemetry data to the Ku-band downlink.

Payload local buses, implemented with Mil-Std-1553B buses, provide command and control, data distribution, and limited low rate telemetry interface for U.S. payloads. These buses cover internal payload racks (one interface each for 13 locations) throughout the U.S. Laboratory and the external payload locations (one interface at each of 4 locations). The payload local buses provide the U.S. laboratory with portable computer ports, the primary crew interface to the payload network for payload management. The payload local buses recognize all payload locations and support devices as remote terminals

An 802.3 payload ethernet routes medium rate payload telemetry data. Each payload rack has an ethernet interface controlled by a gateway/hub. The gateway/hub controls polling of the network and medium rate telemetry routing to the Ku-band downlink through the automated payload switch.

A separate ethernet (802.3) with its own gateway/hub, available at each rack, provides payload rack-to-rack communication, and payload portable computer data transfer where resource efficiency precludes using payload local buses. This ethernet can also be used for downlink, if required.

The automated payload switch / high rate data link system routes U.S. high rate payload data on board. Each payload rack and each attached location has two fiber optic serial digital interfaces (high rate links) with an automated payload switch. The automated payload switch provides optical switching of inputs to the High Rate Frame Multiplexer for downlink and can also switch inputs to other rack locations to accomplish rack-to-rack communication using the high rate data links.

Payloads use the ISSA internal video system to transmit pulse frequency modulated National Television Standards Committee standard video between payload racks, or to the Ku-band system for downlink.

As described above, the on-board data system provides a variety of interfaces to meet the diversity of requirements expected from payload users. Users will select the interface with the on-board data system that meets the user's overall data requirements. Table 1-1 summarizes the interfaces.

Table 1-1. P/L Data Requirements/Interfaces

Requirements/Interface	Mil-Std-1553B	Telemetry Ethernet	R-to-R Laptop Ethernet	High Rate Links
Command and Control	X			
Laptop Support	X		X	
<1 Mbps telemetry	X	X		
<10 Mbps telemetry		X	X	X
<50 Mbps telemetry				X
Rack-to-Rack Comm			X	X

MSFC receives the broadcast from the TDRSS Ground Terminal. PDSS ignores the video data in the Ku-band data stream and strips out the payload data to a Virtual Channel Data Unit format, a CCSDS packet format, or a Bitstream Protocol data unit format, depending on individual user requirements. PDSS distributes this payload data including health and status of the payloads, required ancillary data, and science and technology data to payload users as defined by previous agreements between the payload user and the ISSA program. The ISSA Program provides distribution to the United States Operations Center. The payload user having the specific distribution requirement will provide distribution from MSFC

to the UOFs or any other remote site. Users can implement this requirement using existing NASA networks or may rent commercial networks. PDSS also receives S-band data and provides it along with predefined Ku-band data (e.g. payload health and status data) and required ancillary data to support the POIC.

The program will provide a common user interface to payload users located at a User Operations Facility or at a remote site, provided the facility has elected to fund and equip the required communication capability and has POIC-compatible workstations. This interface will provide payload users

access to command shells, program databases, planning and procedures data input and output.

The POIC will distribute audio to payload users located at the USOC or at an appropriately equipped UOF or remote site. Audio will include required space-to-ground loops and ground-to-ground payload operations loops from the POIC. The facility hosts for the UOF or remote site will determine required internal facility audio loops.

Johnson Space Center will distribute downlinked video, using existing NASA networks, to the NASA Field Centers. Users not already equipped to receive this video need to identify their video requirements and negotiate for required support.

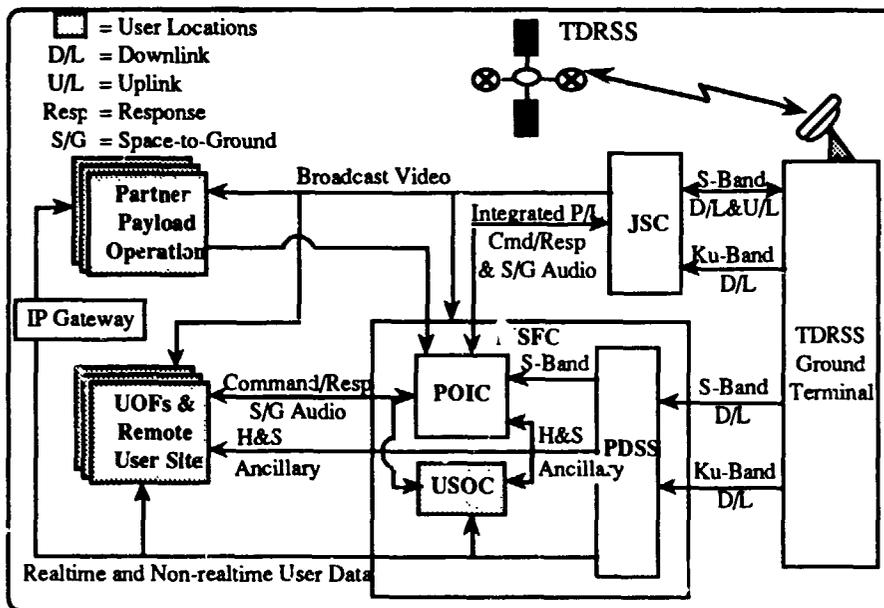


Figure 1-7. Ground Data Architecture

1.4 GROUND DATA SYSTEM CONCEPT.

Figure 1-7 shows the ground portion of the ISSA data system [8]. Telemetry in CCSDS format passes through the Tracking and Data Relay Satellite System (TDRSS). The TDRSS Ground Terminal simultaneously rebroadcasts received data to the Johnson Space Center (JSC) and to the Marshall Space Flight Center (MSFC), using a domestic satellite. JSC captures S-band data for systems operations, and strips video data from Ku-band to rebroadcast using existing NASA video networks. MSFC provides the primary distribution point for all payload data.

The Payload Data Services System (PDSS) [9] at

Because of advances in network technologies, the user has many affordable choices for interfaces to receive data from payloads on-board ISSA. Users will select the ground data system interface that meets the user's overall data requirements.

1.4.1 Payload User Database [4]

Payload users will provide information in response to Payload Analytical Integration activities during pre-increment definition. Payload user inputs gathered during this process will become part of a database which will provide a single controlled source from which the integration and operations elements may draw information.

During pre-increment activities, the POIC integration team will use the payload database to identify payload user command and telemetry requirements. Integrators will incorporate this information into a command and telemetry data base, accessible for execution through the POIC.

1.4.2 Telemetry

The ISSA Program manages and controls the on-board and ground data systems to support payload operations. This includes control of the on-board Command and Data Handling System, Communications and Tracking System operations, distribution, and ground data services supporting payload telemetry.

User responsibilities include submitting appropriate telemetry requirements during pre-increment integration planning for telemetry operations planning and performance.

2.0 PAYLOAD OPERATIONS INTEGRATION FUNCTIONAL DISCIPLINES

Space Station payload operations integration involves two major levels of responsibility. The first deals with *payload independent* functions primarily concerned with overall station management and mission success (e.g. station systems and crew safety). The Space Station Control Center at JSC controls this function.

The second integration level addresses *payload dependent* operations integration and payload operations success. The POIC executes this role and manages the payload complement in both planning and real-time command and control tasks. This function insures that the overall payload integrated plan operates within the constraints of

1.4.3 Data Reduction and Analysis [4]

Payload user responsibilities include data reduction and analysis. PDSS processing includes only that processing necessary to deliver data to payload user facilities/locations in the form that the payload user input the data to the on-board data management system. PDSS time orders the data and removes redundant information. The POIC processes payload operations data such as health and status of payloads, ancillary, and limited station systems monitoring data. If payload users require payload data reduction and analysis for critical, time-sensitive operational decisions, they should use the most reliable and expeditious systems available. An "operations rule" to provide the POIC with default decisions if data does not become available in a timely manner, should accompany each decision for which the user needs to reduce or analyze data.

1.4.4 Data Archiving [4]

Payload user responsibilities include data archiving. PDSS does not intend their temporary "short term storage" to serve as an archive. The POIC does not plan to archive any of the payload operations data processed for the POIC. If payload users want to archive payload health, status, and safety data generated for the conduct of operations, users must record that data as part of a user data stream. Johnson Space Center will archive selected audio and video data generated to support payload operations, but the user should consider making required audio and video part of the user's archive.

available station resources and systems limitations. The POIC functional discipline teams listed below [10] work with user operations/facilities teams as described in the following sections:

- Payload Planning
- Operations Control
- Data Management Process
- Training

2.1 PAYLOAD PLANNING

Payload planning consists of operations requirements definition, tactical planning, pre-increment planning, and execution planning. Planners require

payload user interfaces during all planning phases.

Payload users provide parameters that define the payload operations requirements to support strategic and tactical payload utilization planning. These parameters provide a source of operations requirements for execution level payload planning. Payload users interface with the payload planners to ensure adequate interpretation of payload planning requirements.

The POIC will coordinate with the users for planning requirements. Based on user payload planning inputs, program guidelines, payload resource allocations, and resource availabilities from the Space Station Control Center, POIC payload planners develop an integrated payload plan. Users assess whether the operations plan satisfies their functional objectives, permits payload-to-payload output data correlation, and provides expected resource utilization. Users also assess whether their ground systems, facilities, and communications networks can support the proposed payload operations.

Following pre-increment planning, users participate in execution level payload planning. This encompasses development of payload plans and products required to support and execute system and payload activities on-board and on the ground during a given period of the increment. In coordination with the Increment Payload Operations Planning Group, which will consist of all the users for a particular execution payload complement, users may submit planning or operations change requests which the POIC will use in developing execution planning products. User responsibilities include developing and updating procedures and on-board stored commands used as planning products to implement updated plans.

2.2 OPERATIONS CONTROL

The operations control team executes the payload plan. When a station or payload contingency changes the plan, the operations control team assesses and alters the plan to continue safe operations until they can re-establish nominal operations. Operations control and support activities include monitoring payload health, safety, security, payload commanding, data flow, crew communications, payload procedures maintenance and coordination, and contingency and anomaly resolution support.

When anomalies interrupt planned operations, the POIC will coordinate options with the user operations facilities. Operations controllers will consult with the users and payload planners when real-time replanning involves short-term redistribution of space station resources.

All user operations facilities and the POIC monitor payload execution, and support the flight crew. Users update their respective payload commands and procedures as required. The POIC requires a user interface for contingency support as a result of either a payload, station system, or ground system anomaly. The POIC may need users to provide payload input to resolution and replan. The POIC will coordinate payload user requirements to communicate with the on-board crew.

2.3 DATA MANAGEMENT PROCESS

The data management function responsibilities include coordination, integration, and control of the ISSA payload data systems as part of operations integration. The station data systems include the Command and Data Handling, and Communications and Tracking operations in support of payload data and video flow, and telemetry management including distribution and ground systems and services. Ground system requirements, video development, and data integration management teams perform this function. Payload data management planning personnel schedule and assess end-to-end data systems capabilities to support payload planners and operations control teams. This activity provides inputs to the planning process as it relates to end-to-end data systems, and schedules on-board Command and Data Handling, and Communications and Tracking utilization.

The POIC may require a single point of contact at each UOF to interface with the POIC Data Management Team for flight support data management functions. This point of contact will represent the UOF during "pre-pass briefings" (the periodic operations status briefings given by the Data Management Team to cover data activities), and during contingency and trouble-shooting activities within the data distribution system.

2.4 TRAINING

ISSA training philosophy builds flight and ground crew unity by joint training and simulations on payload related activities. For payload-specific crew training, particularly on scientific or techno-

logical principles, Payload Development Centers or other specific user sites will do the training. The training concept promotes initial familiarization training and later detailed payload-specific training on the routine operation of each individual payload at user facilities, under user supervision.

User payload training responsibilities include flight crew and ground support personnel introductory and familiarization training. Investigators will provide crew operations training on individual experiments. The program will provide multiple experiment integrated training at the Payload Training Complex for integrated payload operations and skill maintenance training. The Payload Training Complex is part of the Space Station Verification and Training Facility which will provide a "whole station" environment to exercise station-wide systems/payload and ground/air operations in a series of simulations involving crew, users and ground controllers. Additional training of ground

controllers constitutes an integral part of payload operations product development. Users participate throughout to ensure training meets user objectives.

The ISSA Program will insure adequate crew training to meet objectives set jointly with the users. Early discussions between users and payload integrators will define the level of fidelity of user-provided payload simulators and user involvement in individual crew training. Users should consider simulator and training requirements for both payload-specific training and integrated payload training.

User responsibilities include defining payload simulator requirements and providing simulators. User responsibilities may include providing instructors during individual payload training. Users will participate in simulations among operations and training elements.

3.0 SUMMARY

The operations approach developed by the ISSA program and its payload users consists of operations infrastructure and corresponding end-to-end data systems that will allow an effective means to produce quality science and technology results. Based on proven experience from Spacelab and other shuttle payload missions, this approach allows evolution to *highly distributed* operations for a variety of remote users in a variety of operations locations.

With the evolving changes in station flight system designs over the past few years, this concept has

remained fundamentally constant. The current approach has the necessary foundation and *flexibility* to meet the expanding user operational needs into the next century. The system must adapt to changes in user requirements including experiment complexity, planning priorities, and physical operational location.

Implementing the most *user-friendly* concept possible, within the constraints of decreasing budgets, constitutes one of the future's biggest challenges for NASA and its international partners.

REFERENCES

1. Griner, C.S. "Space Station Freedom Payload Operations Concept/Responsibilities," MSFC Mission Operations Laboratory/EO-01, Briefing, December, 1992.
2. Kreider, G. AIAA-93-4241, "The Ames User Operations Facility for Nonhuman Life Sciences Payloads on the Space Station," AIAA Space Programs and Technologies Conference and Exhibit, Huntsville, AL, September 21-23, 1993.
3. Golden, H. "U.S. Payload Distributed Operations," MSFC Mission Operations Laboratory/EO-01, Briefing to the Space Station Science and Applications Advisory Subcommittee, July 7, 1994.
4. NASA D683-35070-1, "U.S. Users Operations Concept Definition Document," Coordination Copy, June 30, 1993.
5. Maxwell, T. "Mission Planning System (MPS) Operations Concept," MSFC EO-45, Advanced Planning Branch, Briefing to MPS "Delta" Program Requirements Review, May, 1994.
6. NASA D683-10686-1, Mission Planning Operations Concept Definition Update: Final Report, January 6, 1994.
7. NASA D684-10500, "Command and Data Handling Architecture Description Document," March 16, 1994.
8. Holloway, T. "SSF Data Flow Operations Orientation," MSFC EO-32 Data Management Branch, Briefing, August 4, 1992.
9. MSFC-DOC-2130, "Payload Data Services System (PDSS) Operations Concept Document," MSFC, 1993.
10. NASA D683-43010-1, "Payload Baseline Operations Integration and Training Plan", Volume 1, April 29, 1993.

SHARED MISSION OPERATIONS CONCEPT

Gary L. Spradlin, Richard P. Rudd, and Susan H. Linick
 Jet Propulsion Laboratory, 4800 Oak Grove Drive, Pasadena, California 91109

Abstract

Historically, new JPL flight projects have developed a Mission Operations System (MOS) as unique as their spacecraft, and have utilized a mission-dedicated staff to monitor and control the spacecraft through the MOS. NASA budgetary pressures to reduce mission operations costs have led to the development and reliance on multimission ground system capabilities. The use of these multimission capabilities has not eliminated an ongoing requirement for a nucleus of personnel familiar with a given spacecraft and its mission to perform mission-dedicated operations.

The high cost of skilled personnel required to support Projects with diverse mission objectives has the potential for significant reduction through shared mission operations among mission-compatible projects. Shared mission operations are feasible if:

- i. the missions do not conflict with one another in terms of peak activity periods,
- ii. a unique MOS is not required, and
- iii. there is sufficient similarity in the mission profiles so that greatly different skills would not be required to support each mission.

This paper will further develop this shared mission operations concept. We will illustrate how a Discovery-class mission would enter a "partner" relationship with the Voyager

Project, and can minimize MOS development and operations costs by early and careful consideration of mission operations requirements.

Objective and Overview

The objective of this article is to describe a shared mission operations concept that provides for concurrent mission operations of two deep space Projects both utilizing a single MOS originally developed by the Voyager Project, but modified to accommodate shared support of a Discovery Project.

The Voyager Project is an existing JPL-operated interplanetary mission. Basically, the Voyager Project proposes to modify the Voyager MOS to enable shared operations support of a Discovery Project. The Discovery Project will benefit from savings achieved by avoiding development and operation of its own unique MOS. The Discovery Project will be responsible for costs associated with adapting the Voyager MOS for Discovery Project use, and for adding capabilities not part of the Voyager MOS baseline. For example, the Voyager Project is now operating in an extended cruise posture, has a well understood trajectory, and a fully developed and stable set of mission plans, thus Navigation and Mission Planning functions are not actively supported by the Voyager Project, are not part of the Voyager MOS baseline, and must be added for Discovery Project support.

Management Structure and Flight Team Organization

A Memorandum of Understanding (MOU) signed by both Project Managers will provide the basic ground rules governing shared flight operations, funding issues, resource utilization, priorities, conflict resolution, etc.

Development of the shared MOS will be undertaken by the Voyager Mission Director and the Voyager Flight Teams. The Voyager Mission Director will lead the effort to develop the shared MOS, and in this role, report to the manager of the Discovery Project. Figure 1 depicts the Discovery

organizational structure shown in Figure 2 is organized into three process-oriented teams - the Uplink Team, the Downlink Team, and the Navigation Team.

Uplink Team Description

The Uplink Team performs all functions required to generate spacecraft event sequences and send commands to the spacecraft. Two basic processes- the sequence generation process and the real time command process, provide the mechanism for accomplishing these functions.

The uplink process begins with the collection of spacecraft activity requests (science and

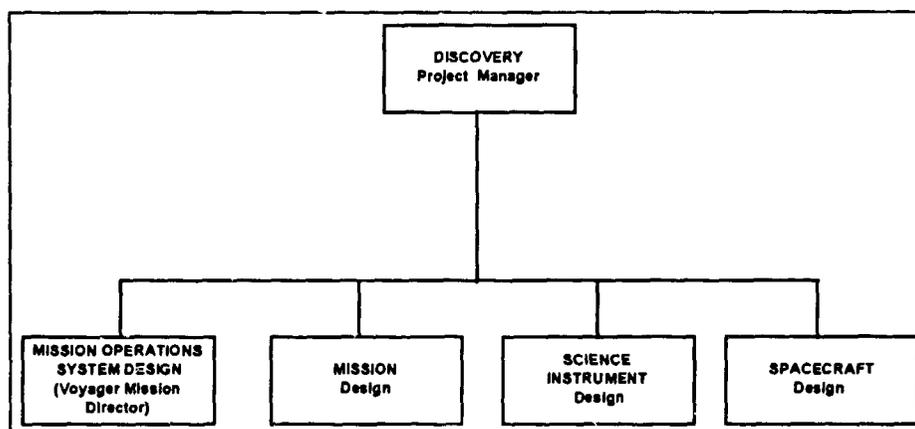


Figure 1

Project development organization, and illustrates the relationship of the Voyager Project Mission Director to other elements of the Discovery Project development organization.

Figure 2 illustrates the relationship of the Voyager Mission Director to the Managers of both the Voyager and Discovery Projects, and expands the organizational structure beneath the Mission Director to reflect the supporting flight teams and staff, and to indicate where staff additions would be required for support of a Discovery mission. The MOS

engineering), which are combined into a conflict free sequence design (a timeline of sequence events). Based on the sequence design, a time ordered listing of spacecraft events is generated and a detailed simulation and validation performed. A ground command file is then generated which is conditioned, validated for correctness and then segmented into separate ground command files for radiation to the spacecraft.

There are three types of sequences the Uplink Team builds and/or updates: mini, overlay and baseline. The mini sequence is composed

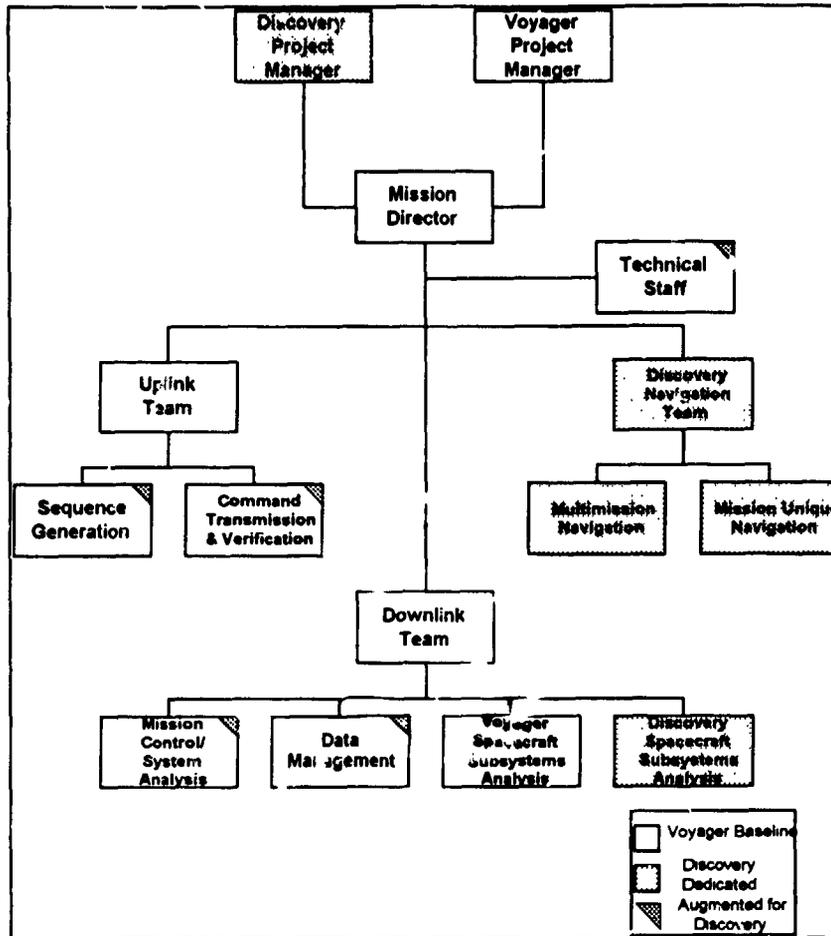


Figure 2

of one or more stored commands which are required to respond to an unplanned event or anomaly. The overlay sequence consists of non-repeating or special science and engineering observations. The Uplink Team's primary sequencing function is building these overlay loads. The baseline sequence is a looping sequence aboard the spacecraft consisting of repeating science and engineering activities. The baseline sequence operates autonomously. Modifications can be made to the baseline sequence at pre-determined restart points.

The Uplink Team is also responsible for all real-time command operations. Real-time commanding is used to load spacecraft event sequences and modify the on-board spacecraft

configuration and/or the executing sequence. These operations consist of generating the real-time command request, coordinating and reviewing the request, negotiating the Deep Space Network (DSN) coverage for uplink/downlink, generating the command file, transferring it to the DSN, and monitoring the uplink/downlink.

The Discovery Project sequence development process will be similar to Voyager Project's where possible so as to use established procedures and interfaces.

Downlink Team Description

The Downlink Process begins at the spacecraft where state, status, and instrument observation samples are integrated into a

formatted data stream for transmission to ground-based receiving stations. The Downlink Process ends with delivery of committed data products to science investigators.

The Downlink Team is responsible for the capture, conditioning, and delivery of science and ancillary data committed by the project to experimenters, as well as, all data required for monitoring the status of both the Voyager and Discovery Project spacecraft.

The Downlink Team also provides analysis of spacecraft and science instrument performance and health. This team evaluates spacecraft and instrument status against expected performance and initiates recovery actions for all spacecraft failures. The Downlink Team will provide inputs for the uplink process as necessary to generate engineering calibration and performance data needed to evaluate spacecraft performance and health, and will provide any necessary spacecraft state and status data to predict spacecraft behavior.

In support of the Discovery Project, the Downlink Team will support concurrent development of spacecraft and Ground Data System (GDS) capabilities during the period preceding launch. This will include: support to test and demonstrate spacecraft and ground system compatibility; development of capabilities needed to display and evaluate spacecraft and instrument performance and health; support for development of fault protection algorithms and other programmable spacecraft capabilities; definition of spacecraft alarm limits and recovery procedures; and other activities necessary to assure knowledge of the state and status of both the Voyager and Discovery spacecraft and instruments.

Navigation Team Description

The Discovery Navigation Team estimates, predicts, and controls the spacecraft trajectory and updates the planetary and satellite ephemerides. Navigation personnel for systems engineering, orbit determination, maneuver analysis, optical navigation support, trajectory analysis, and software maintenance will comprise the Discovery Navigation Team. Radiometric orbit determination analysis and related operations support will be provided by the Multimission Navigation Team under funding by the Telecommunications and Mission Operations Directorate (TMO). Discovery Project funding will provide for trajectory analysis, maneuver design and analysis, and optical navigation support functions.

Mission Scenario

As mentioned earlier, the Voyager Project is operating in an extended cruise phase. Operations are routine and consist of daily spacecraft contacts for science and engineering data collection and spacecraft performance monitoring. Sequence operations are based on use of a baseline sequence composed of repeating spacecraft activities. Non-repeating or time-varying activities are controlled by periodic transmission and execution of overlay sequences.

Discovery project mission operations will consist of multiple mission phases beginning before launch with System Test/Pre-launch Operations. Typical mission operations phases, operation activities, and mission phase duration are:

System Test/Pre-launch Operations

Support of system test and pre-launch operations consists of:

- i. Generating all commands to be executed by the Discovery Project spacecraft during system test and pre-launch operations using the Sequence System.
- ii. Monitoring subsystem and instrument telemetry data in real time for test support and evaluation purposes using the operational GDS and Mission Support Area (MSA).

This support will require that the GDS, including Sequence System development, be complete prior to the start of system test and that the combined Voyager / Discovery flight team be staffed at near launch-operations level prior to the beginning of system test support. This includes having the remote science locations connected to provide test data to the science teams for instrument checkout.

Flight team activities during this phase will also include personnel test and training for launch and near earth operations.

Launch And Near Earth Phase

DSN support for the first three weeks is assumed to be continuous 24 hour per day coverage using 34 meter stations. The second three weeks will require 1-2 passes per day using 34 meter stations resulting in 8 to 16 hours per day coverage.

Spacecraft telemetry data are monitored during launch, parking orbit, interplanetary injection, and spacecraft separation. Following separation and initial spacecraft acquisition by the DSN, spacecraft telemetry data are monitored for subsystem and science instrument checkout purposes, radiometric navigation data are acquired, and the post-injection trajectory estimated. A Trajectory Correction Maneuver (TCM) is designed and executed correcting any launch injection errors. Additional radiometric navigation

data are collected and processed to confirm a successful trajectory correction, or to design an additional cleanup TCM if necessary.

Following a complete spacecraft checkout and resolution of any subsystem or science instrument abnormalities, the spacecraft is configured for cruise operations. The cruise configuration should be established around L+3 weeks with the next three weeks devoted to characterizing spacecraft performance in the cruise configuration. At nominally L+6 weeks, cruise begins with reduced DSN tracking support consisting of either one 8 hour pass per week or two 4 hour passes per week using 34 meter stations.

Cruise - Spacecraft Health Monitoring And Maintenance

The spacecraft health monitoring and maintenance phase includes the time period from L+6 weeks to encounter/orbit injection-6 months. DSN support during this phase is nominally one 8 hour or two 4 hour passes, (34 meter stations), per week for spacecraft health monitoring. Navigation requirements may result in additional tracking passes being required.

Spacecraft control will be via a long-term baseline type of sequence that is augmented with periodic overlay sequences. The baseline sequence contains antenna pointing information for maintaining communications with the ground and any spacecraft events that are repetitive and can be planned in advance. Overlay sequences consisting of non-repetitive activities will be generated and transmitted to either Voyager or Discovery spacecraft on a schedule consistent with mission requirements and Uplink Team staffing; nominally this will be no more frequent than once every three months for any Voyager or Discovery spacecraft. Anomaly responses or special events will be handled by

mini-sequences generated as required. This baseline/overlay sequencing strategy will be maintained throughout the cruise phase until 6-months before the start of the Discovery Project encounter/orbit operations.

Flight team staffing during the cruise time period is minimized by maximum utilization of shared flight team personnel. With the exception of periodic detailed spacecraft checkout, the entire Discovery Project effort during cruise is spacecraft health monitoring and maintenance, and navigation. While the combined flight team staff needs to maintain a knowledge base in each spacecraft subsystem area for normal cruise operations, the ability to respond quickly to spacecraft anomalies will be limited by this minimum staffing approach. In the event of a significant spacecraft anomaly, a link will be established to the spacecraft contractor's facility for support of the contractor's spacecraft team. The spacecraft team will provide diagnostic support and will be responsible for recommending recovery actions.

Encounter/Orbit Insertion Preparation

The test and training phase for the Discovery mission includes the time period from encounter/orbit insertion-6 months to encounter/orbit insertion-4 months. DSN support will increase as final preparations for the start of the encounter/orbit insertion sequences are implemented and increased navigation support is necessary for starting the ephemeris updates of any early encounter sequences.

During this final portion of spacecraft cruise, preparations for the start of the Discovery Project encounter/orbit operations are completed:

- The acquisition and processing of navigation data is increased for sequence updates and approach TCMs.
- Final checkout and configuration of the spacecraft for the start of encounter/orbital operations is accomplished.
- Training exercises are conducted to verify and refine flight team operational readiness.
- Sequences are updated and loaded onboard the spacecraft.

Encounter (& Gravity Assist Flyby)/Orbital Operations

Flight team activities during this period are directed at:

- Acquiring the planned science data.
- Maintaining spacecraft health.
- Achieving the trajectory knowledge and control to provide the viewing conditions necessary to successfully accomplish the planned science observations (includes necessary TCMs).
- Updating the sequences based on the latest trajectory information.
- Capturing, processing, and delivering science data that are downlinked during the encounter operations.

In the special case of a gravity assist flyby without any science data acquisition, the operations emphasis is on the acquisition and processing of navigation data and the execution of approach TCMs. Following the flyby, there will be additional navigation data acquisition and processing for a post-encounter TCM to correct trajectory errors resulting from the flyby.

Mission Operations System Description

The MOS is defined to be the collection of systems (hardware and software), personnel, facilities, and procedures required to remotely monitor and control a spacecraft and deliver data products to users. The MOS may extend to a remote scientific investigator's site to support science instrument control and scientific data delivery. However, the MOS does not include data processing elements, personnel, or procedures utilized by scientific investigators for scientific data analysis.

The baseline Voyager MOS is composed of a GDS, two flight operations teams, and a collection of fully demonstrated operating procedures. Voyager management actively pursues a continuous improvement process to assure that the operational MOS is based upon the latest available technology, and incorporates new tools and processes as they become available. This continuous improvement process has kept the Voyager Project at the cutting-edge of mission operations engineering, and has resulted in the Project pioneering the operational use of new multimission capabilities, often becoming the prototype user for new capabilities developed for future flight projects.

The systems that comprise the Voyager Project GDS are Telemetry, Command, Sequence, Spacecraft Analysis, Data Records, Tracking, Monitor and Control, and Simulation. The TMO Directorate provides multimission subsystems that constitute significant portions of the GDS.

Development of the Discovery MOS will require modification of the Voyager GDS to add the capability to process Discovery spacecraft command and telemetry data. This will be done in a manner that minimizes the

addition of new hardware, and exploits existing software capability.

GDS Design and Development

The design of the Discovery Project GDS will be approached from the perspective that a minimum cost design will maximize use of existing capabilities. This implies that requirements will be imposed on the Discovery Project spacecraft data system design to avoid unnecessary incorporation of new data format definitions, spacecraft clock design, decommutation schemes, etc. that would cause significant rework of current ground system capabilities. A design team of ground system developers and Discovery contractor spacecraft engineers will be tasked with identifying and developing a set of minimum impact requirements and detailed design specifications that will likely require some compromise on both sides of this interface. A process for concurrent spacecraft data system-GDS design will maximize communications and shorten the development life cycle.

The Discovery GDS will include:

- a telemetry front-end processing capability providing: recovery from lost, noisy and disorganized data; detection and removal of data handling and transmission artifacts; removal of redundant data; distribution of data for display, analysis, and storage.
- a sequence development, validation, and command generation and transmission capability.
- a science data processing capability including: full-capability science data processing providing for data manipulation, editing, enhancement, archival storage, remote access and retrieval, Experiment Data Record production, and Planetary Data System

hand-off; or as preferred by the experimenter, quick access to science data processed only to eliminate recoverable gaps and add any required ancillary data.

- a spacecraft navigation capability providing: pre-launch tracking requirements analysis; conventional radiometric navigation; maneuver design and analysis; when required, optical navigation.

Recommendations for Low Cost Mission Operations

MOS Development

- i. Minimize life cycle costs by maximizing re-use of existing capabilities. This is feasible if: the sharing Projects do not have missions that conflict with one another in terms of peak activity periods; a unique MOS is not required; and greatly different skills are not required to operate each mission.
- ii. Design the spacecraft data system to meet existing ground system interfaces, and avoid requiring unusual data formats, data modes, derived parameters, etc.
- iii. Foster a concurrent MOS-spacecraft engineering process. Utilize a simulation capability to develop and demonstrate ground system/spacecraft interfaces and compatibility as the spacecraft evolves. Extend the GDS to the investigators home institution, and to spacecraft developer facilities. These same ground system nodes will later serve to support delivery of science data, support inputs to the uplink process, and support spacecraft subsystems analysts in event of a spacecraft anomaly.
- iv. Build ample margin into the spacecraft subsystems. This includes adequate onboard storage to avoid frequent and mandatory data playback, adequate

power margin such that science instrument power sharing will not be required, adequate telecommunications link margin to avoid reliance on scarce large aperture ground antennas, etc. There will be trade-off decisions that affect spacecraft margins, and it must be kept in mind that the smaller these margins become, the more complex and costly operations will become.

Flight Operations

- i. Develop a sequencing strategy that is compatible with your partners. Voyager Project will utilize a repeating baseline sequence with periodic overlay sequences. Do not plan a strategy that either requires a separate sequencing capability, or drives operations costs by adding complexity to the sequence process.
- ii. Utilize extensive cross training of personnel to provide increased availability of operations support personnel to both missions with a minimum of additional staffing.
- iii. Minimize cruise activities, such as cruise science. Use onboard autonomy to reduce tracking coverage required for spacecraft performance and health monitoring.

Conclusion

Flight projects that do not have overriding requirements for unique MOS capabilities or for standalone operations can reduce operating costs by reusing rather than inventing, and by partnering with compatible projects to share MOS expenses. Voyager Project is actively pursuing this concept with a Discovery-class partner, and is planning to demonstrate the practicality and cost benefit of our shared operations concept.

Mars Pathfinder Mission Operations Concepts

Francis M. Sturms, Jr., William C. Dias
Albert Y. Nakata, Wallace S. Tai

Mars Pathfinder Project
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, CA 91109-8099

Abstract

The Mars Pathfinder Project plans a December 1996 launch of a single spacecraft. After jettisoning a cruise stage, an entry body containing a lander and microrover will directly enter the Mars atmosphere and parachute to a hard landing near the sub-solar latitude of 15 degrees North in July 1997. Primary surface operations last for 30 days.

Cost estimates for Pathfinder ground systems development and operations are not only lower in absolute dollars, but also are a lower percentage of total project costs than in past planetary missions. Operations teams will be smaller and fewer than typical flight projects.

Operations scenarios have been developed early in the project and are being used to guide operations implementation and flight system design. Recovery of key engineering data from entry, descent, and landing is a top mission priority. These data will be recorded for playback after landing. Real-time tracking of a modified carrier signal through this phase can provide important insight into the spacecraft performance during entry, descent, and landing in the event recorded data is never recovered.

Surface scenarios are dominated by microrover activity and lander imaging during 7 hours of the Mars day from 0700 to 1400 local solar time. Efficient uplink and downlink processes have been designed to command the lander and microrover each Mars day.

The work described in this paper was carried out by JPL, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

Mission Overview

Mars Pathfinder will be launched on a Delta 7925 during a 30-day period beginning December 6, 1996. The flight system consists of a cruise stage and an entry stage (Fig. 1). The cruise stage is jettisoned just prior to entry into the atmosphere directly from the approach trajectory. Inside the entry body are the lander and parachute, retro-rocket, and air bag deceleration systems. The lander is a tetrahedron (Fig. 2), with a base plate and 3 petals covered with solar cells. A microrover rests on one petal. The mission uses a short Type I transfer trajectory and is targeted for a constant landing date on July 4, 1997 at 19.5° N, 32.8° W in the Ares/Tiu Valles outflow channel into Chryse Planitia. Primary surface operations, lasting 30 days, consist of rover technology experiments, and imaging, alpha-proton-X-ray spectrometry and meteorology science. A lower activity extended mission may last for up to a year.

Costs

As a Discovery mission, Mars Pathfinder costs are capped at \$150M in FY92\$ for development. This translates to \$171M in real year dollars. In addition, the technology program contributes \$25M for rover development and operations. The Mission Operations and Data Analysis budget from launch+30 days to End of Project is \$14M. Tables 1-2 show cost data for the major systems and details of the Ground Data and Mission Operations System budgets. At a total of \$10.9M and 6%, the ground system development budget is smaller in both absolute dollars and as a percentage of total project costs than previous missions.

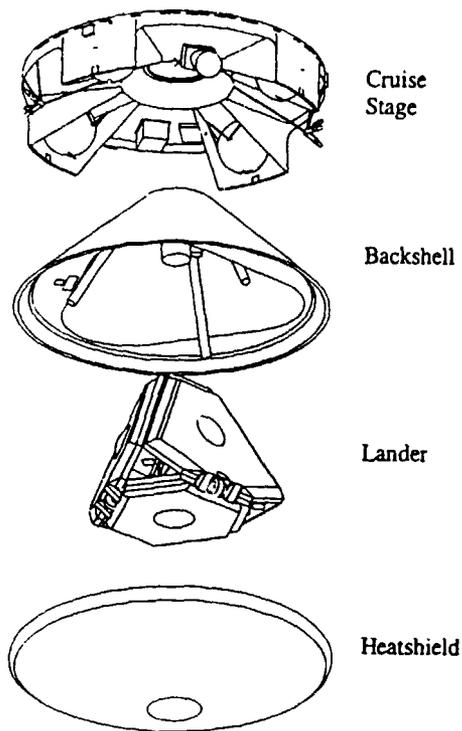


Fig 1. Flight System in Launch Configuration

Table 1. Major System Obligations, \$M

FY	94	95	96	97	98	Total
Proj Mgt	2.5	2.0	1.5	0.4	0.0	6.4
MD&Ops	3.1	4.4	4.1	1.1	0.0	12.7
Flt Sys	37.7	41.1	15.2	3.5	0.0	97.5
Sci&Inst	6.1	4.7	2.7	0.8	0.0	14.3
Reserve & other	4.6	21.4	13.9	0.2	0.0	40.1
Dev Total	54.0	73.6	37.4	6.0	0.0	171.0
Rover	6.0	8.0	6.7	2.0	0.0	25.0
(total includes \$2.3M in FY93)						
MO&DA				8.0	6.0	14.0

Table 2. Ground System Costs, \$K

FY	94	95	96	97	98	Total
Mgt	170	199	209	70	0	648
Sys Engr	421	470	459	75	0	1425
Loan Pool	80	150	150	35	0	415
H/W	605	560	71	0	0	1236
S/W	541	845	566	0	0	1952
Ops Supt	43	178	189	78	0	488
Reserve	158	50	25	0	0	233
GDS Dev	2018	2452	1669	258	0	6397
Mgt	153	228				381
Reqt&Des	116	0				116
Exp Team	150	422				572
Eng Team	178	557				735
Test-Train			1524	875		2399
Reserves	56	105	155	0		316
MOS Dev	653	1312	1679	875		4519
GDS+MOS	2671	3764	3348	1133		10916
Mgt				1404	1072	2476
Exp Team				1995	2529	4524
Eng Team				2825	1225	4050
GDS Sup				860	503	1363
Reserve				916	671	1587
MO&DA				8000	6000	14000

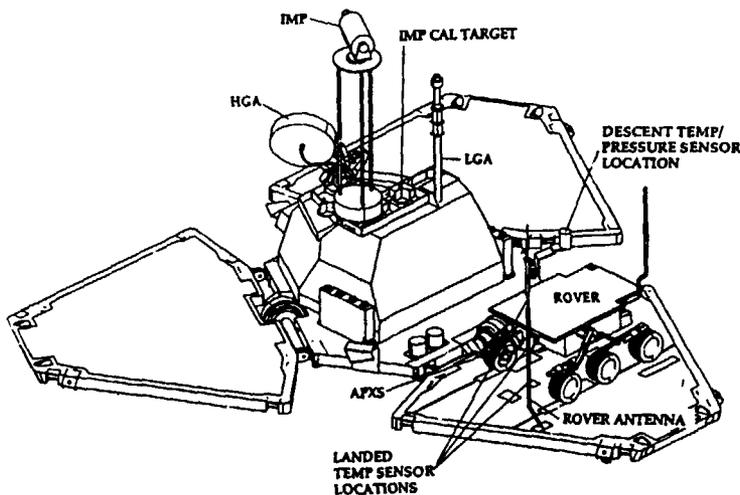


Figure 2. Lander in Deployed Configuration

Organization

The operations organization for Mars Pathfinder is smaller and has fewer teams than typical JPL flight projects. The organization, as illustrated in Fig. 3, consists of a Project Office, a Mission Director, and 5 operations teams: Experiment Team, Engineering Team, MOSO Support Team, GDS Maintenance, and DSN Operations Team. These teams report to the Mission Director, who in turn reports to the Project Manager, who heads the non-operations Project Office. The staffing numbers shown in Fig. 3 denote the maximum level planned for operations. This organization is in place in October 1995 at the beginning of ATLO (assembly, test, and launch operations).

The Experiment Team provides operations support for rover operation and science and technology experiments during surface operations phases. This team includes science investigators and rover technology experimenters when they are performing operations tasks.

The Engineering Team conducts mission planning, sequence generation, flight system performance analysis, navigation, and real-time flight control and commanding tasks that insure safe operations and achievement of mission objectives. This team also supports ATLO with planning and development of test sequences for test and training.

Support is provided by MOSO (Multimission Operations Systems Office) for data system operations, data administration, and image processing. In addition, MOSO maintains the baseline GDS capabilities at no cost to the Project.

The Project adaptations to the baseline GDS are maintained by Project GDS personnel.

The DSN (Deep Space Network) Operations Team provides the interface between the Project and the DSN for obtaining network coverage for commanding and telemetry receipt at no cost to the Project.

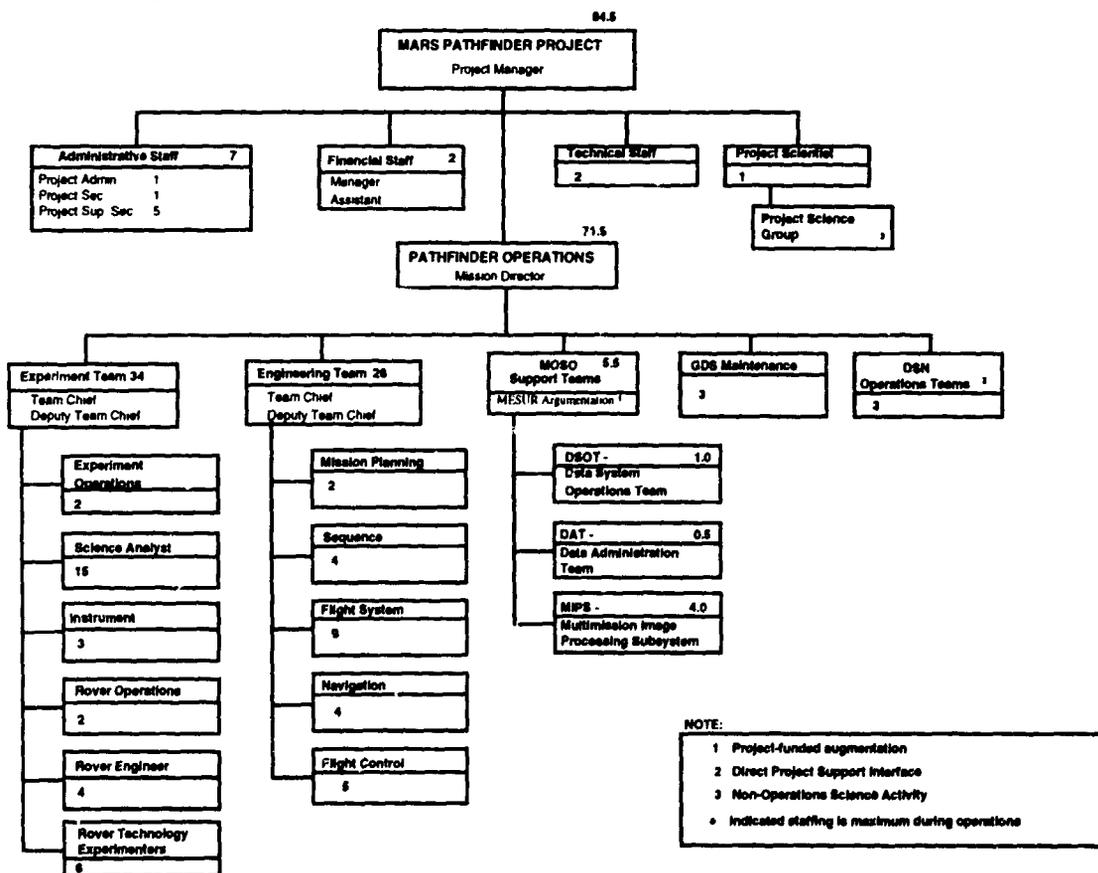


Figure 3. Mars Pathfinder Operations Organization

Enabling Characteristics

Low operations costs are enabled by system characteristics established from an end-to-end perspective and concurrent engineering of the flight and ground systems: **a. Acceptance of more risk as a Class C mission** enables the use of more autonomous capabilities onboard and thus enhances flight system operability. For ground operations, multiple levels are not needed for review and approval. **b. No cruise science and limited surface instruments.** **c. Large GDS inheritance** enables the project to build a GDS at minimum cost. **d. Better flight system operability.** The flight system is designed with emphasis on operability. Sample design decisions based on the operability view are: (1) A simple spin-stabilized flight system (2) Large flight computer margins - The provision of large computer resources (i.e. a 32-bit RISC processor with processing power up to 20 MIPS, sequence memory size at 4Mbytes, data storage capacity at 128Mbytes, and system backplane bandwidth \geq 10Mbytes/s) negates the need for allocating the above resources during mission operations. (3) No external storage devices (such as a digital tape or solid state recorder) - only RAM and EEPROM are used for storing flight software and data - offering better flexibility and simplicity for uplink operations. (4) On-board sequence memory management simplifies the uplink process. (5) On-board autonomous capabilities - includes closed-loop monitoring and control of the thermal and power condition, lander high gain antenna Earth pointing, flight system mode control, a demand-driven rover-lander interface scheme, and autonomous rover traverses. (6) Asynchronous data-driven telemetry handling scheme which makes the telemetry data collection, data recording, data retrieval, and data downlink processes totally decoupled from each other. The MOS no longer has to design, test, and schedule the telemetry modes as in other planetary missions. (7) Priority downlink of telemetry data so that high priority operations data can be downlinked ahead of lower priority data. **e. No complex navigation data types** -only Doppler and ranging data are used for orbit determination. Thus not only the process of flight path estimation but other activities, such as sequence development, are greatly simplified.

Cruise Scenario

The cruise mission phase begins at separation of the flight system from the launch vehicle upper stage and ends with the turn to entry attitude at Mars arrival -1 day. The initial cruise sequence continues from the launch load, and includes spin down, attitude stabilization, telemetry acquisition and the first of two complete flight system health and status checks (Fig. 4). For launch on the opening day of the 30-day launch period, 8 cruise sequence loads are planned with the first 7 about 4 weeks and the 8th about 1 week in duration. The 8th load contains the second and last health and status check. No other experiment activity is planned for cruise.

Four trajectory correction maneuvers (TCMs) are scheduled. Navigation is based on two-way Doppler and range data. TCM-1 removes launch injection errors and most of the aiming point bias necessary for Planetary Protection. TCM-2 corrects execution errors of TCM-1. The 3rd TCM targets the flight system for Mars atmospheric entry and TCM-4 corrects execution errors of TCM-3. Delivery accuracy on the surface is about 200 km downtrack and 100 km crosstrack (3 sigma).

The sequence load strategy for launch delays is to maintain the first three cruise sequences as designed with TCM-1 at L+30 and TCM-2 at L+60 near the beginning of loads C0002 and C0003. Cruise load C0004 will be shortened with each launch delay and will be deleted if the delay approaches 4 weeks.

The uplink process allocates three weeks for cruise sequences (4 working days for planning, 8 days for generation, 1 day for final updates, and 1 day for commanding). The allocation of three weeks to generate a four week sequence provides the margin to enable the mission operations teams to perform the following additional tasks while minimizing the requirement for extended work hours: (1) characterize and respond to flight system anomalies, (2) participate in test and training exercises and certification of team members for surface operations, and (3) design, generate and update entry, descent and landing, and nominal surface operation sequences.

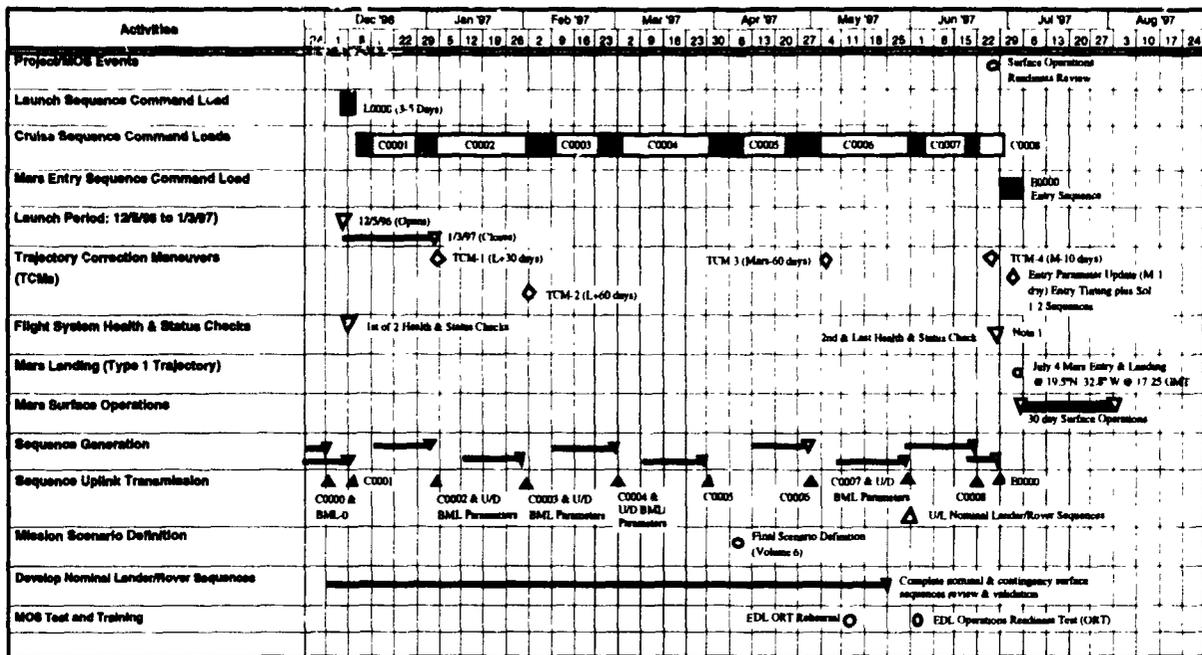


Figure 4. Mars Pathfinder Cruise Timeline

EDL Scenario

The entry, descent, and landing (EDL) phase of the Pathfinder mission begins one day prior to Mars arrival and ends with the touchdown of the lander on the Mars surface. An EDL sequence, a sol 1 and sol 2 sequence, and a contingency sequence (covering the entire surface operations phase in the event landing damage prevents normal operations) are uplinked prior to jettison of the cruise stage. The EDL operational scenario is characterized by the following:

- a. **Continuous DSN coverage** is provided through a 70-M DSN station.
- b. **Continuous real-time engineering telemetry monitoring** of the flight system state up to parachute deploy.
- c. **Continuous carrier tracking** to obtain flight system state information concerning key EDL events. Real-time tracking and recording of carrier signals are performed at the DSN station. Real-time display of frequency spectrum through a Spectrum Signal Indicator (SSI) gives some visibility into EDL status. Telemetry acquisition will not be possible after

parachute deploy due to large varying angles between the flight system low gain antenna boresight and Earth. MOS will obtain knowledge of critical events using the following two mechanisms: (1) Determine the deviation from the nominal entry profile by measuring the line-of-sight velocity using a recovered Doppler frequency profile. (2) Analyze the transition of amplitude modulated carrier signals to determine the modulation index changes. These changes are commanded by the flight system to obtain a carrier suppression of 0 or 6 dB (and perhaps other levels) upon the occurrence of key events in the EDL sequence. They provide critical state information to the MOS. Figure 5 depicts a potential strategy for obtaining telemetry and EDL state information.

- d. **Autonomous execution of on-board EDL activities** by the flight system. These autonomous actions, e.g. cruise stage separation, chute deploy, heatshield release, lander release, RAD firing, are controlled by the flight software based on pyro event timing parameters in the EDL sequence. This means that no real-time ground control of the flight system is possible after cruise stage separation.

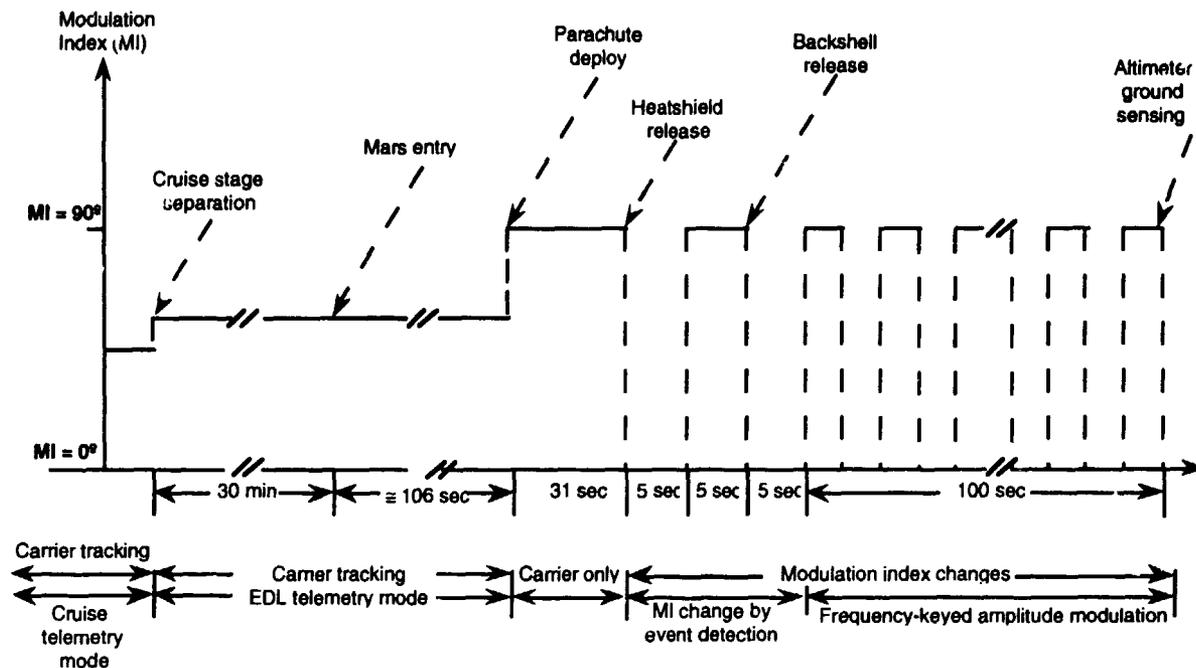


Figure 5. EDL Telemetry and State Information Acquisition Strategy

Surface Scenario

Most JPL spacecraft operate in the relatively well-understood environment of deep space. Pathfinder, however, must land and operate on a largely unknown planetary surface. The tilt of the lander with respect to the Sun, Earth and local horizontal all affect battery charging, communications and rover maneuvering operations. The amount of atmospheric dust affects solar panel response and amount of battery heating required. Nearby rocks or features might block the Sun or rover exit paths. Lander orientation in azimuth with respect to the Sun changes the time of day when various amounts of power states are reached and when pictures can be taken. These are all factors which can be statistically surmised but not known in advance. They make prediction of activity sequences and mission data return time tables more problematic than for most other types of missions -- even if the lander were to work perfectly.

Nominal lander performance and surface scenarios are based on an optical depth, τ , of 1.0 and an adverse lander tilt of 15° . Rough estimates indicate a probability of about 82%

that these values will not be exceeded. They are bad enough to significantly limit activity schedules. Contingency scenarios will be ready if conditions are better or worse.

We hope to accomplish the basic mission quickly, since thermal cycling could end the mission early. As shown in Table 5, much of the imaging is done on sol 1 and stored. This is because the data acquisition scheme includes imaging as much as possible early and storing the compressed data in memory, in case anything goes wrong with the camera later. The plan is to complete the basic rover mission in a week.

To account for a range of environments, as well as for possible hardware problems, the project has adopted a policy of maintaining some number of both "nominal" and "contingency" scenarios. These scenarios are to be negotiated before landing to reduce decision times.

Table 3 shows a range of activity schedules and milestones for three different example conditions: optimistic, reduced, and loss of High Gain Antenna (HGA). These scenarios are generated in a system of spreadsheets

which include formulas for battery charging and discharging, data compression, and engineering data acquisition, and tables for DSN coverage, solar array input, activity schedules, rover and lander power modes, and data rates. As shown, each scenario can be reported against a schedule for achievement of formal mission success milestones.

Data acquisition and data return projections for an "optimistic" nominal mission are shown in Tables 4 and 5. Among the optimistic assumptions is the amount that can be achieved on sol 1, as shown in Table 3.

The EDL and sol 1 sequences (as loaded before entry) run until telemetry can be received and the lander can be commanded from Earth, at which point sol 1 activity can be modified if necessary. Rover deployment is enabled from the ground based on downlinked images. The operations plan for each sol thereafter is to command the lander and rover in the Mars

morning just after receipt of important overnight telemetry to confirm acceptable status. Depending on the amount of solar power available and lander energy balance, up to 5 hours of additional telemetry is obtained during the rest of the sol for rover and lander status and science data. The nominal communications period is from 0700 to 1400 LST. During the other 17 hours each sol, 5 hours are allocated for telemetry analysis, and 15 hours for replanning and a highly automated generation of the sequence for the next sol, leaving a 2 hour margin. This process repeats each sol for the rest of the 30 sol prime mission.

Operations after the 30 sol prime mission continue similarly, except that the data rates will go down by 75% when the project switches to 34-meter DSN stations. The data rate continues to drop to a low of 150 bits/s in June 1998, as the Earth-Mars distance increases.

Table 3. Mission Activity Milestones for Optimistic, Reduced, and LGA Missions

	"OPTIMISTIC" NOMINAL MISSION				"SLOWED DOWN" NOMINAL MISSION				MISSION ON LOW GAIN ANTENNA			
	Sol when First Achieved	on Sol 1	in Week 1	First Month	Sol when First Achieved	on Sol 1	in Week 1	First Month	Sol when First Achieved	on Sol 1	in Week 1	First Month
Mission Success Level	-	70%	90%	100%	-	50%	70%	100%	-	50%	70%	100%
Spacecraft Deployments												
Partially Deployed	1	-	-	-	1	-	-	-	1	-	-	-
Fully Deployed	1	-	-	-	1	-	-	-	2	-	-	-
Rover Ramp Deploy	1	-	-	-	2	-	-	-	2	-	-	-
Downlink												
EDL	1	-	-	-	1	-	-	-	1	-	-	-
Health	1	-	-	-	1	-	-	-	1	-	-	-
Imaging for Planning	2	-	2	2	3	-	1	2	6	-	1	2
Science	1	-	-	-	1	-	-	-	6	-	-	-
Rover Activities												
Deployment	1	-	-	-	2	-	-	-	3	-	-	-
Traversing	1	2M	15M	150M	2	-	10M	75M	3	-	10M	20M
Image Lander (incl close-ups)	1	1	3	>20	2	-	2	3	1	-	1	1
Image Landing Path	1B	-	-	10	1B	-	-	5	-	-	-	-
Image Terrain	1	-	3	>20	2	-	3	10	3	-	1	3
Soil Mechanics	2	-	4	8	4	-	2	4	7	-	1	2
Soil APX's	1	1	1	3	2	-	1	2	7	-	1	1
Rock APX'S	3	-	2	6	5	-	1	3	9	-	-	1
Stereo Pans	9	-	-	3	15	-	-	1	-	-	-	-
Materials Tests	1	1	7	30	1	1	7	30	3	-	3	10
High Gain Antenna												
Deployment	1	-	-	-	2	-	-	-	-	-	-	-
Point : 4.7° (1200 bps)	1	-	-	-	2	-	-	-	-	-	-	-
Point : 3.5° (2200 bps)	8	-	-	-	10	-	-	-	-	-	-	-
Lander Camera												
Deploy Imager Mast	1	-	-	-	2	-	-	-	3	-	-	-
Mission Success Pan	1	1	1	1	1	1	1	1	3	-	1	1
Mid-Res Science Pans *	1	1	2	2	2	-	2	2	-	-	-	1
Color Science Pans *	1	1	2	2	3	-	1	2	4	-	-	1
High-Res Science Pan *	2	-	4	6	6	-	1	3	-	-	-	-

* Sol numbers indicated are for data acquisition.
Downlink of much of the panorama data occurs on later sols.

Table 4. Optimistic Surface Mission Data Acquisition Timetable (Mbits)

DATA CATEGORY	LANDED MISSION WEEK				Data
	1	2	3	4	Totals
APXS Data					
Rock	0.2	0.1	0.1	0.1	0.6
Soil	0.1	0.1		0.0	0.2
Lander Imager Data					
Monochrome Science Pans	169.9		74.2		244.2
Eight-Color Pans	166.4				166.4
Mission Support Pans	26.2				26.2
Other Science Imaging	3.3	3.3	3.3	3.3	13.1
Rover Support Imaging	9.4	1.7	10.7	2.4	26.2
Rover Activities					
Mission Support Imaging	3.6	0.6	17.7	9.8	39.7
Science Imaging	1.4	1.4	0.7	1.4	4.8
Technology Imaging	6.9	2.8	0.4	1.2	11.4
Technology Experiments	3.6	1.8	0.9	1.0	7.4
Rover Engineering	0.9	2.0	1.6	1.2	5.7
Meteorology	0.7	0.7	0.7	0.7	2.8
Weekly Data Acquisition Totals	392.6	24.5	110.3	21.1	548.5

Table 5. Optimistic Mission Data Return Timetable (Mbits)

Time Frame	Experiment Data				Engi- neering
	Raw	Com- pressed	Downlink	Storage	Downlink
Sol 1	998.5	58.7	13.1	45.7	0.2
Sol 2	937.3	242.9	9.8	278.8	0.3
Sol 3	188.3	75.8	10.4	344.2	0.3
Sol 4	13.6	4.5	5.5	343.2	0.3
Sol 5	12.5	4.4	6.6	340.9	0.3
Sol 6	2.4	1.5	12.7	329.7	0.3
Sol 7	12.8	4.2	14.1	319.8	0.3
Week 1	2165.4	387.8	72.2	319.8	1.9
Week 2	126.8	39.5	116.1	243.2	2.0
Week 3	380.8	115.7	136.2	222.7	2.0
Week 4	94.1	31.1	127.6	126.1	2.0
Weeks 1-4	2767.2	574.1	452.1	126.1	7.9
Daily Average	98.8	20.5	16.1	32.6	0.3

3. Planning Tools		Page 451
MM.3.a	Towards a New Generation of Mission Planning Systems: Flexibility & Performance <i>A. Gasquet, Y. Parrod, A. De Saint Vincent</i>	453-460 51
MM.3.b	Generic Mission Planning and Scheduling: The AXAF Solution <i>O. T. Guffin, M. Newhouse</i>	461-466
MM.3.c *	A PC Based Tool for Mission Plan Production <i>Jean-Pierre Joli, C. Yven</i>	467-471
MM.3.d	Towards a Class Library for Mission Planning <i>Olivier Pujo, Simon T. Smith, Paul Starkey, Thilo Wolf</i>	473-480
MM.3.e	Autonomous Mission Planning and Scheduling - Innovative, Integrated, Responsive <i>Charisse Sary, Simon Liu, Larry Hull, Randy Davis</i>	481-488
MM.3.f *	A Mission Planning Concept and Mission Planning System for Future Manned Space Missions <i>Martin Wickler</i>	489-497

* Presented in Poster Session

TOWARDS A NEW GENERATION OF MISSION PLANNING SYSTEMS: FLEXIBILITY & PERFORMANCE.

A. Gasquet, Y. Parrod, A. De Saint Vincent

MATRA MARCONI SPACE
31 Rue des cosmonautes - Z.I. du Palays
31077 Toulouse Cedex - France

Tel +33 61.39.61.39
Fax +33 62.24.77.80
Email : gasquet@osiris.matra-espace.fr

ABSTRACT

This paper presents some new approaches which are required for a better adequacy of Mission Planning Systems. In particular, the performance, flexibility and genericity issues are discussed based on experience acquired through various Mission Planning systems developed by Matra Marconi Space.

Key Words: Mission Planning, Knowledge Based Systems, Flexibility & Performance.

INTRODUCTION

The increasing complexity of modern spacecrafts, and the stringent requirement for maximizing their mission return, call for a new generation of Mission Planning Systems. Indeed, this complexity has several impacts on the :

- adequation of the specific planning & scheduling methods;
- performance problems;
- compliance with the eventual evolutions of the mission itself.

This paper presents the main lessons learned by Matra Marconi Space from several projects on Mission Planning, showing the benefits of advanced software techniques. They are illustrated by systems developed by Matra Marconi Space.

PROBLEM DEFINITION

The term "Mission Planning" is used to refer to the process of planning and scheduling all

activities and operations of the space segment (spacecraft platform and payload, e.g. power sub-system for the platform, optical instruments and tape recorder for the payload) and the ground segment (ground station activities, payload data processing and product dissemination) associated to a given mission.

The main inputs to the Mission Planning System are a set of requests of the following types :

- Spacecraft platform operation;
- End User request (e.g. observation requests for an Earth observation satellite);
- Other types of ground segment activities (data processing, dissemination, etc).

The main outputs of the Mission Planning System are the *Service Utilization Plan* for satellite End Users, the *Final Operations Plan* uplinked to the space segment. Additional outputs include ground segment activities plans. From an operational point of view, the whole process is decomposed in the two following phases :

• Generation of the Operations plans : This phase is performed off-line and deals with the acquisition of User Requests and the detailed planning and scheduling of all space / ground operations. It includes :

- The generation of the Preferred Exploitation Plan (PEP),
- The integration of this first plan with the activities required by the Operations team for house keeping manoeuvres, and the production of the final "executable" plan.

• Execution of the Operations plans : Once the whole planning and scheduling process has

been completed, a schedule is available for execution and transmitted to the execution environment. During execution, monitoring is performed to control the evolution of the mission and detect eventual anomalies. If any disturbance on the current schedule occurs during its execution, rescheduling may be required and performed locally by the mission control center. If the rescheduling fails, a replanning session is entered on the Mission Planning System. Examples of anomalies include resource shortage (e.g. electrical power drop, unavailable ground station), activity execution failure (constraint violation, unexpected result), and changes in the satellite status due to some contingency (automatic or manual plan interruption, unexpected state transition).

ADVANCED TECHNIQUES

Based on experience learnt from past developments and current studies, both on operational Mission Planning systems and on advanced prototypes, three main areas which can be improved using advanced techniques (e.g. Artificial Intelligence) can be identified :

Algorithmic Performance

The Mission Planning problem is generally characterized by an intrinsically high combinatorial complexity, reflecting the complexity of the spacecraft itself and the numerous utilization constraints related to the resource usage, the inter-instruments constraints and the mission operational constraints. Taking the example of the Earth Observation missions, the planning process (typically performed on a daily basis) has to select an "optimal" set of candidate observation requests to be executed in the next day, among a set of pending requests which may be of the order of thousands. The generation of all the possible scenarii cannot be performed in a reasonable time. It is thus necessary to find powerful algorithmic techniques to deal appropriately with that complexity, in order to optimize as much as possible the utilization of the satellite, while taking into account the constraints on the available computing time.

Matra Marconi Space has conducted an internal study on this problem in order to evaluate the applicability of advanced algorithmic techniques on the planning & scheduling of an Earth Observation spacecraft. Generally, this type of spacecraft raises a complex planning & scheduling problem due to the high number of potential requests that can be submitted and also the hard operational constraints having strong impacts on the feasibility of the resulting plan. Thus, the objective of the study was to optimize as much as possible the use of the satellite resources with an acceptable response time taking into account the following points :

- On one hand, the combinatorial problem due to the high number of requests to be scheduled makes the determination of a good solution difficult in a reasonable time (large space of potential solutions to be explored);
- On the other hand, the complexity of the spacecraft due to the management of tape recorders, the strategy used for ground station dump operations and the constraints imposed by the capabilities of the instruments (e.g. transitions between image acquisitions) makes the determination of one feasible plan a time consuming step.

The activity performed in 1993-94 lead to the definition and implementation of a planning algorithm applied to the SPOT4 mission planning problem using an iterative and "any-time" optimisation strategy [1]. This approach is characterized by two phases :

- Phase 1 : Determination of a first plan (without optimization) based on a simple heuristic strategy. This phase is considered as an initialization phase being responsible for the determination of a first potential solution.

- Phase 2 (The anytime phase) : The algorithm starts a loop which explores the initial plan elaborated in Phase 1 and then optimizes this plan. This operation is done by iteratively removing some requests and inserting new requests according to heuristics driving the plan evolution toward a better plan quality. In order to avoid looping in the remove / insert process, all generated plans (up to several thousands) are stored and each new plan is checked against the history of the already generated plans. At any time, the "current plan" is defined as the best solution at hand,

with respect to the plan quality criteria as specified operationally.

This algorithm was integrated into a mission simulator for analysis on real problems. Testing has been performed using operational scenarios and the analyses conducted during the testing phase have allowed to demonstrate the following advantages of the approach :

- It tackles the problem globally, optimizing the solution with respect to the whole set of constraints, instead of handling separately the different constraints (this latter approach based on filtering mechanisms, by nature, always leads to sub-optimal solutions);
- A first plan can be made available at the end of the first phase, in a very short time;
- The initial plan is improved regularly and solutions are available at any time (Several plans of approximately the same "quality" are available);
- The flexibility of the iterative approach allows late insertion into the plan of new requests, which is an important advantage from an operational point of view.

This approach thus proved to be quite successful; furthermore, it is general enough to be reusable for other planning and scheduling problems. Further developments in this area now concern the application of these techniques to a new observation satellite.

Flexibility

The lifetime of spacecrafts and the duration and complexity of the projects call for highly flexible and evolutive planning systems, enabling users to adapt the planning system to the evolutions of the planning problem. Indeed, the following cases can be envisaged :

- evolution of the spacecraft (e.g. degradation of the available power, degradation of the recorder capacity, equipments out of order, ...): The definition of the spacecraft model reflecting the capabilities of its main components must be modifiable all along the mission since it influences the planning constraints related to the space segment.

- evolution of the ground segment : Modifications of ground segment may impact on the planning problem by adding or modifying constraints related to the ground segment capabilities. For instance, the post-processing of received data may be improved by new computer characteristics enabling the possible processing of more requests.

- evolution of planning strategies : The feedback of the mission is generally a source of experience that can be used to improve the spacecraft utilization and to better fulfill the objectives of the mission. This implies a lot of modifications on the planning & scheduling strategy to be used. This is particularly true at the early beginning of the exploitation.

In conventional Mission Planning System, information is more or less hard-coded, making changes and corrections difficult. For instance, the evolutions of conceptual information concerning strategies for resolving conflicts cannot be modified by the operator and requires software modification. In order to solve this problem, Knowledge Based Systems (KBS) have a more declarative approach which brings a high degree of flexibility in the system.

The following systems can be mentioned to illustrate this approach :

- PlanErs, dedicated to Mission Planning;
- Optimum, a more generic project planning & scheduling system.

PlanErs :

PlanErs [2], [3] is a mission planning system developed by MMS (France), CRI (Denmark) and AIAI (University of Edimburgh) for the European Earth Resource Observation satellite ERS-1. It has been developed during an ESA R & D project from 1987 to 1990. Its first objective was the modelling of the planning & scheduling process in order to optimize planning strategies (usage of recorder, record / dump strategy and selection of the ground station dedicated to the dump operation, priority mechanism between requests in order to cope resource shortage, etc). It is implemented in Common Lisp on top of the

KEE [4] development shell which provides an object-oriented programming environment and graphic functions.

One of the main features of the system is the use of a high level, user accessible formalisms for representing the different areas of the planning knowledge.

The object oriented model of the satellite, the rules used for expressing planning constraints and strategies, and the associated syntactic editors, provide the users with an easy-to-use environment enabling them to modify the internal planning knowledge, for instance on the following aspects :

- operational constraints related to instrument usage (e.g. maximum usage per eclipse) : these rules have been frequently modified during the system experimentation in order to optimize instrument usage as well as power consumption;
- transition modes for instrument . An example is the following rule.

From Mode Measurement_1 to Mode Measurement_2

- *Goto Mode Standby_1 during 10 seconds*
- *Goto Mode Standby_2 during 20 seconds*
- *Goto Next_Mode*

- rules defining the IDHT (recorder) strategy. These rules have been one of the main problems raised by the ERS-1 application. The challenge was to define a concept enabling to change interactively strategies concerning the transition between IDHT modes in order to optimize the recorder capacity as well as ensuring a good coverage of global zones, taking into account priorities. Due to the numerous events to be taken into account in the definition of these transitions (orbits, eclipses, ground stations, precise timing between events, transition duration), a specific rule formalism had to be designed. Using the syntactic editor, end-users have been allowed to modify the IDHT behaviour, modifying the chain of transitions according to the context. A specific effort has been made during the experimentation phase in order to increase the readability of this formalism, and in

particular to define a clear set of parameters to be taken into account during IDHT planning.

- power conflicts resolution rules : these rules are used when conflicts are detected on power usage. Here too, the difficulty was to define a set of parameters (e.g. Depth Of Discharge over the orbit N) and generic actions (reject a request, reduce a request, ...) to be taken into account during power verification and conflict resolution.
- other parameters : finally, the system includes a set of parameters characteristic of the planning constraints, such as the transition duration, the power consumption per instrument modes, the precise tape position table for the recorder, the available power from solar arrays,... All these parameters are user editable.

The flexibility offered by the system was originally limited to the transition rules but was extended during experimentation to cover operational constraints as the users identified the numerous possibilities offered by this feature. The possibility for the user to modify on-line various constraints and conflict resolution strategies, and see immediately the effects on the plan generated by the system, was a preponderant argument to the PlanErs usage.

Figure 1 describes the Man Machine Interface of PlanErs.

Thanks to this approach, the PlanErs system has been used in 1991-1992 by the European Space Agency (ESA) as a Mission Analysis tool for interactively simulating the impact of various strategies and constraints on the mission output of the satellite. PlanErs also allowed to demonstrate a high benefit of Knowledge Based System techniques to deal with the problem domain evolutivity thanks to the very modular and declarative representation of the different types of knowledge involved in the scheduling problem.

PlanErs is going to be reused for the ERS-1 and ERS-2 mission analysis at ESA / ESRIN.

Optimum :

Optimum [5] is a generic purpose planning and scheduling system that has been designed to handle complex problems in which plan quality, resource optimization and plan progress monitoring are key issues. Interactivity of the system enables the user to assess different planning scenarios and to take a decision in real-time. It was originally an R&D project for ESA/ESTEC developed in 1991-1992. Consolidated by Matra Marconi Space in 1993, it is now used for planning integration activities of the Ariane 4 Vehicle Equipment Bays. It is implemented in Common Lisp + the CLOS object system.

The comparative advantage of OPTIMUM, with respect to classical project planning systems, is its ability to capture information which describes the underlying logic of the plan, instead of using pre-defined sequences of activities. This allows the system to :

- verify the logic of the plan built or updated by the user;
- provide a rich formalism to describe the constraints of the domain;
- schedule activities and resolve resource conflicts.

Figure 2 describes the Man Machine Interface of OPTIMUM.

Genericity

The need to reduce mission-specific software development costs requires to develop Generic Mission Planning functions, from which a mission-specific Mission Planning system can be derived at low cost. In this case, the use of an object oriented representation for both the spacecraft model and the definition of the planning and scheduling methods participate to the genericity of the planning system by offering a more natural and reusable decomposition of the planning & scheduling world and of the methods governing the planning process.

GMPF :

This issue is addressed in the Generic Mission Planning Facilities (GMPF) project [3] which is currently performed by Cray Systems (UK) and Matra Marconi Space (France) for the European Space Agency (ESA/ESOC). The objective of this study is to analyze the commonalities between the large variety of Mission Planning Systems dedicated to specific missions and, by identifying the plan elements and the planning and scheduling process required by several types of mission, to define a common planning & scheduling kernel which can be easily customized to specific missions. The GMPF project should contribute to the definition of the new generation of Spacecraft Control Center (SCOS II) which is conducted by ESA / ESOC.

The envisaged types of missions are :

- Observatory Missions: The spacecraft has one main instrument. End Users are allocated observing time windows during which they have dedicated usage of the instrument.
- Survey Missions: The spacecraft has a single or a small number of payloads. The spacecraft and payload are normally operated by a centralised agency on behalf of a number of End Users who request specific observations that are planned according a high level mission definition.
- Multi-Instrument Missions: The spacecraft has a number of independent experiments, each provided by a separate Principal Investigator (PI). The platform is operated by a centralised agency but PIs are responsible for operation of their experiments, submitting requests to the control centre.
- Telecommunication Missions: The spacecraft has a number of transponders to provide communications between ground stations (fixed service) or between another spacecraft and ground (data relay service). The spacecraft and its payload are operated by a centralised agency on behalf of the End Users. Transponders communication channels are allocated to Users.

The result of the GMPF study will be the definition and prototyping of :

- an objects library defining all the planning & scheduling elements and methods. These objects can be later reused or customized (by subclassing) for a specific application.
- a set of tools used to customize the library for a given application. These tools include :
 - a User Interface Builder based upon an existing commercial tool and complemented by dedicated widgets specific to Mission Planning functions. It is used to define the User Interface dedicated to the Mission Planner.
 - a Library Browser used to navigate in the classes hierarchy and dedicated to the software developer to pick up software components to be used in a specific application.
 - a Mission Specific Information Editor used to defines all the parameters which are normally fixed for the whole mission but can evolve due to modification of the space / ground segment.
 - a Rule / Constraint Editor used to provide the Mission Planner with the capability to define and edit rules and constraints using templates (e.g. syntax driven editor). This tool is used during the mission lifetime.

At the current stage, the definition of the users requirements for the GMPF library / toolset has been performed leading to a first specification of the main object classes (and attached informations) to represent data (plans, schedules, activities, etc) and knowledge (constraints, planning strategies) relevant to the planning process.

This project will be completed at the end of 1995, and will lead to the implementation of a prototype of the GMPF and of a mission specific demonstrator.

CONCLUSIONS

In this paper, we have presented three main areas where advanced software techniques can contribute to solve the requirements raised by

Mission Planning systems : performance, flexibility and genericity. Addressing these issues in future Mission Planning Systems is a major effort necessitated by the growing complexity of space systems in order to combine performance and flexibility without impacting on the global cost. Since this last aspect is becoming more and more crucial, the genericity issue is one of a major concern of space companies and agencies.

REFERENCES

- [1] A. Gasquet : "*AI embedded in CPH*" Final Report, February 1994 (internal document), Matra Marconi Space.
- [2] J. Fuchs, A. Gasquet, B. Olalainty, and K. Currie : "*PlanErs-1 : An Expert System for Generating Spacecraft Mission Plans*", 1st International Conf. on Expert Planning Systems, June 1990, Brighton, UK.
- [3] J. Fuchs, A. Gasquet, Jean Muller, Pierre Viau : "*PlanErs : Experimentation and Users Feedback*", Artificial Intelligence and Knowledge-based systems for space, May 1991, ESTEC, Noordwijk, The Netherlands.
- [4] IntelliCorp Corporation : "*KEE 3.1 Users Manual*", Vol 1, IntelliCorp 1989.
- [5] Y. Parrod : "*OPTIMUM - Review of System Functions*", MMS technical report, 1993 (internal document).

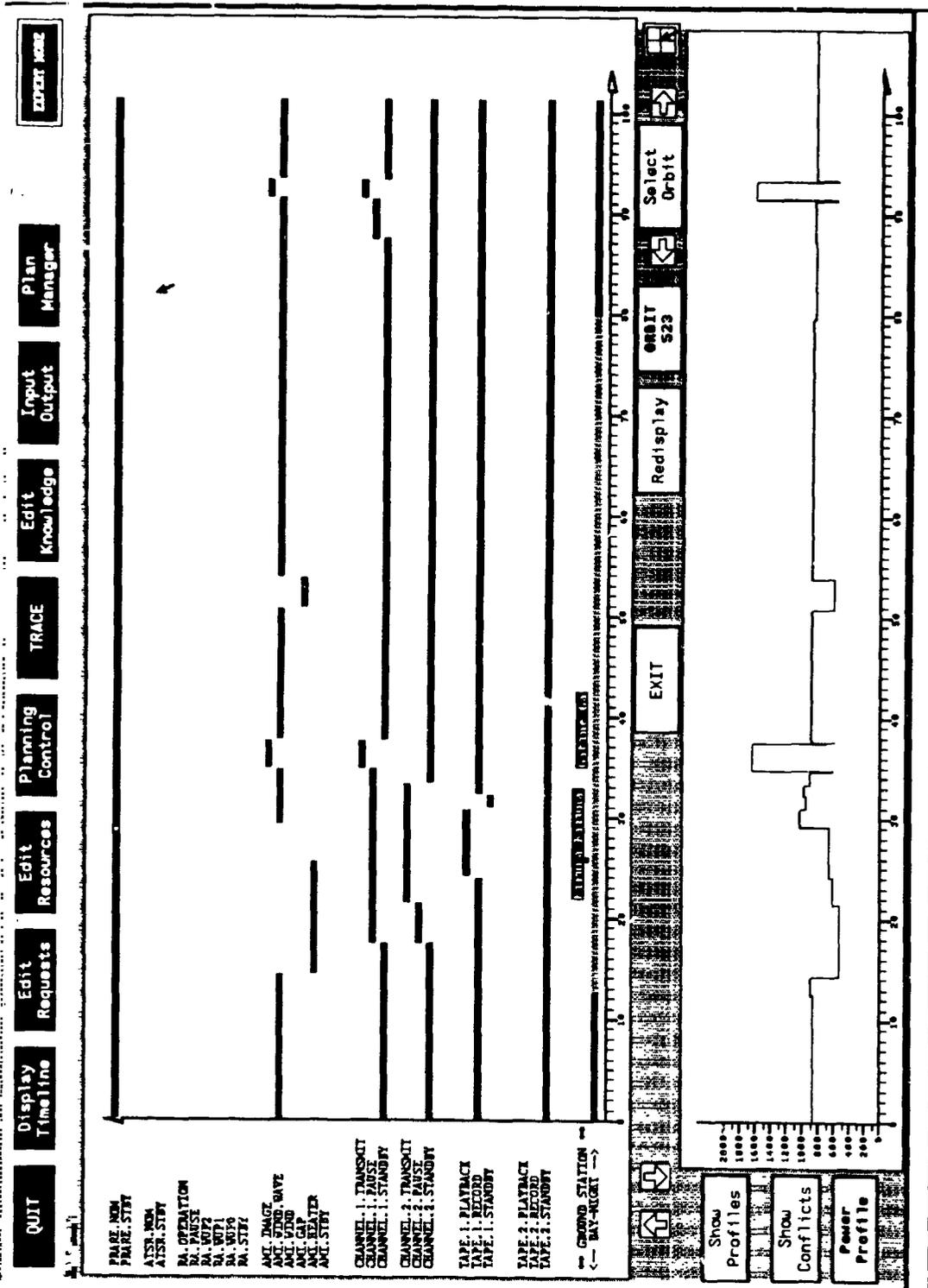


Figure 1 : PlanErs Timeline representation. The various modes of the instruments are represented in the top window and the resource consumption (power in this case) is available in the bottom window.

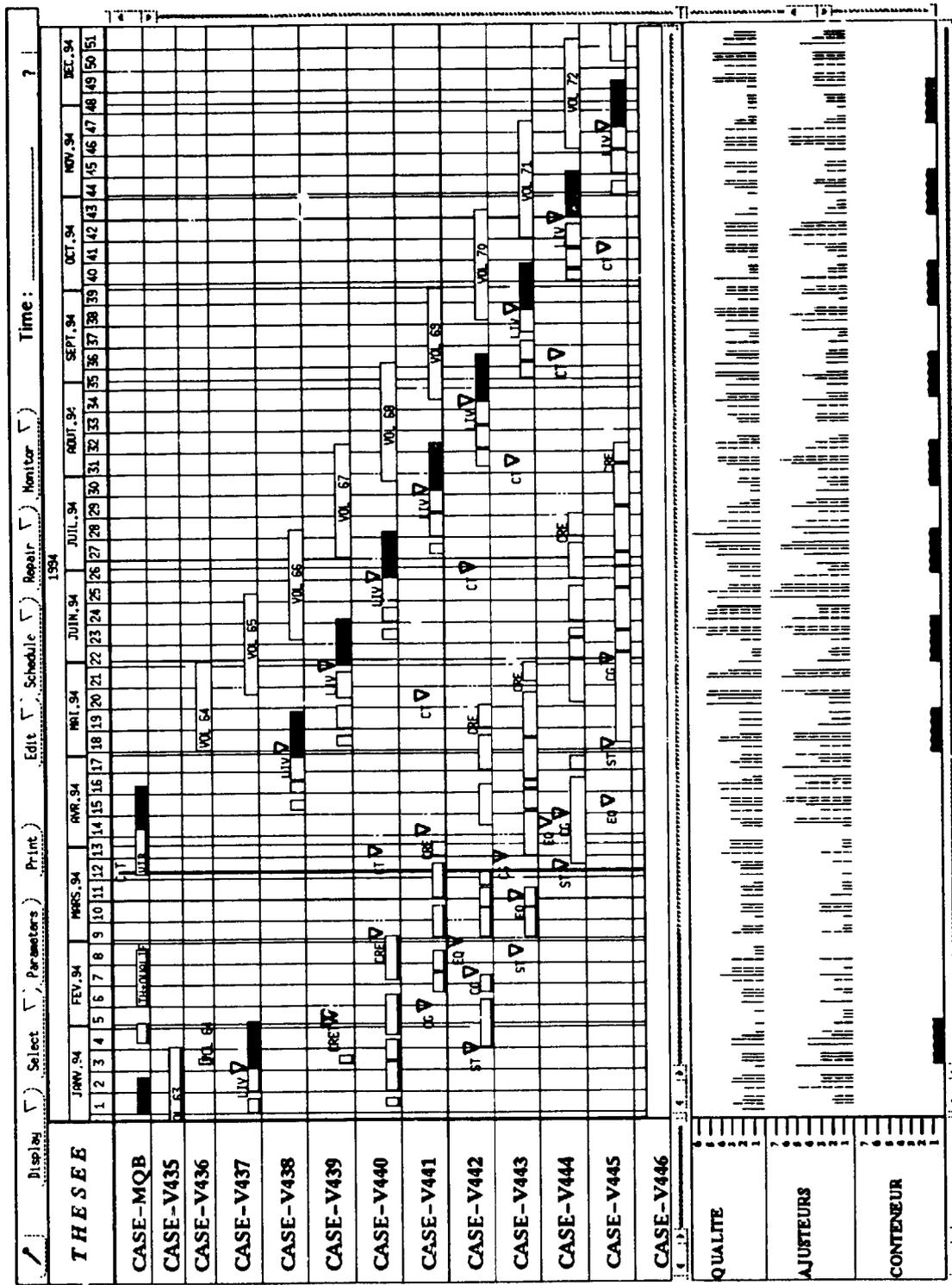


Figure 2 : Optimum Gantt representation. Activities are represented in the top window and resources consumption profiles are shown in the bottom window.

GENERIC MISSION PLANNING AND SCHEDULING: THE AXAF SOLUTION

O. T. Guffin
National Aeronautics and Space Administration
Marshall Space Flight Center, AL, U.S.A.
and M. Newhouse
Computer Sciences Corporation

ABSTRACT

During SpaceOps 92 the idea of generic mission planning concepts for space astronomy missions, that could be applied to future missions in order to simplify software development, was introduced. It was proposed that mission planning systems could be decomposed into functional elements that could be standardized and then organized into optimal functional flows for each individual mission. In addition, it was further suggested that these flows themselves could be reduced to a small set of possibilities by describing them in terms of generic mission type, such as manned, unmanned, high orbit, low orbit, etc. The Advanced X-ray Astrophysics Facility (AXAF), planned for launch in the latter part of '98, represents the first application of this idea on an unmanned mission. This paper examines the AXAF Mission Planning and Scheduling concept in light of the generic system theory. Each functional element is evaluated according to AXAF characteristics and requirements and then compared to its generic counterpart. Functional flow considerations are then derived from the overall AXAF mission planning concept to determine the viability and sensitivity of the generic flow to actual requirements. The results of this analysis are then used to update the generic system concept and to define the level of commonality and core system components that are practical to achieve across multiple missions.

INTRODUCTION

The recent emphasis on smaller, cheaper and faster satellite development has led to a corresponding reduction in ground support system funding. This trend manifests itself not only in control center hardware architecture, but in software system design as well. Several control centers already exist that support multiple missions and it is expected that this will in the future be the norm. A natural extension of this philosophy is a concomitant thrust by ground system designers to devise generic on-line support software and, to a lesser extent, the off-line software used for spacecraft operations and control. The latter, especially, has been more difficult to bring about because of unique science instrument and satellite characteristics and (unlike common control center development) different designers are involved in each project. In the case of AXAF, great emphasis has been placed on generic on-line software and extensive reuse of existing off-line software elements. Simple reuse of appropriate routines, however, is not enough to produce a software system that will be useful for more than one mission; it also requires careful consideration of flexible design features, functional modularity and functional flow. The benefits of a generic system are reduced costs, easier maintenance and updates, reduced user training, and analytical tool spin-offs.

THEORETICAL COMPONENTS

During SpaceOps 92 the idea of a generic mission planning and scheduling system for space astronomy missions was introduced. The theoretical basis for this idea was determined by examination of past and existing systems spanning over 20 years. By comparing similar

functional elements in each of these systems, the authors were able to define a set of functions common to every system, although the specific implementation and packaging of these functions varied widely with the passage of time and the peculiarities of each project. The eight resulting theoretical components of the generic system are listed in Table 1 along with a brief definition of each.

Table 1. Generic Mission Planning Functions

MISSION PLANNING FUNCTIONS
<ul style="list-style-type: none"> • Observation & Engineering Request Processing Receive, check and edit observation and engineering requests • Orbital Mechanics Generate all ephemeris, environmental and geometric data • Guide Star Selection Select guide/aspect stars for each observation • Scheduling Schedule science and spacecraft activities • Editing Modify and revalidate existing schedule • Communications Planning Determine communications opportunities • Spacecraft Management Generate detailed, chronological list of spacecraft activities to implement schedule • Flight Operations Team Support Display and tabulate mission planning data required for flight operations support

AXAF CONCEPT COMPONENTS

Since SpaceOps 92, two new missions have begun development of their respective mission planning and scheduling systems along the lines of the generic model. Astro-2, a manned Spacelab flight, will reuse much of the Astro-1 software with improvements in the schedule editing, guide star selection and flight operations support areas. The other mission, AXAF, belongs to the unmanned world and is one of the four satellites in the Great Observatories program. It too will reuse much software from previous missions and its off-line software design will emphasize modularity and independence of functional elements.

Although the AXAF Mission Planning and Scheduling system design is in the early prototype stage, a recognizable structural outline of process flow, and the features included in each functional module are emerging. The elements composing this concept and their interaction are depicted in Figure 1. Notice that some of the functional titles in the flow diagram are different than those listed in the generic concept, and that the "packaging" is not always the same. These variances, however, are not detrimental to the generic theory. Specific titles for each function will vary from project to project. What really matters is that each function remain essentially the same regardless of what it's called. As was mentioned

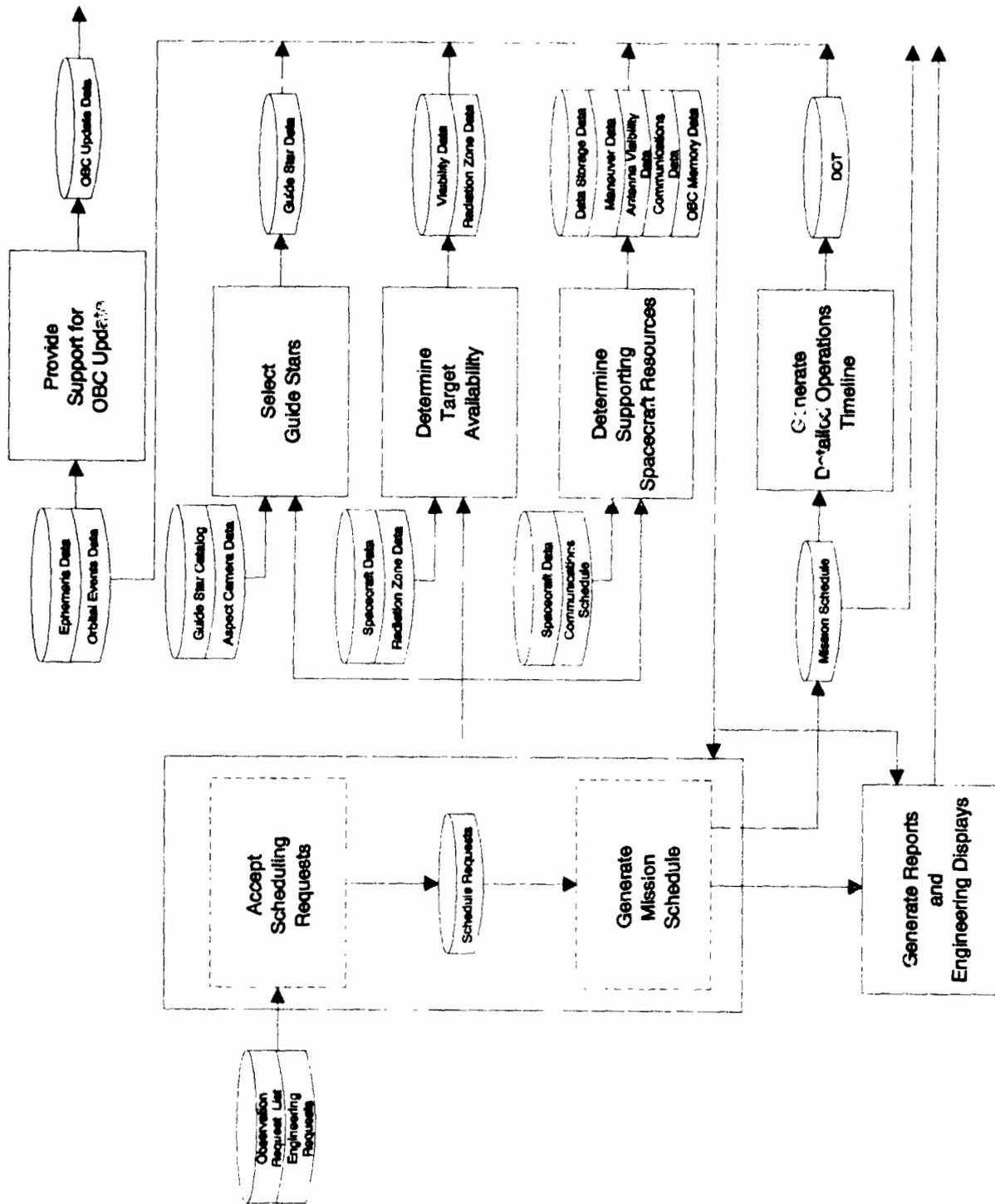


Figure 1. AXAF Mission Planning and Scheduling Functional Flow

in the original paper, it is likewise acceptable to package functions together as needed by specific missions, so long as each function maintains its modularity and standalone capability. The reverse process of splitting subfunctions into separate packages, as is the case in the AXAF solution, is also permissible with the same stipulation.

In the AXAF generic solution, the process described above was used liberally. The scheduling, editing and communications planning functions, for example, have been packaged together for convenience due to their close relationships. This allows the user to interact with these functions as needed without having to create intermediate products and migrating between applications windows. The spacecraft management function for AXAF is called the Detailed Operations Timeline (DOT), but otherwise exactly matches the theoretical generic element. The name itself derives from the fact that the DOT contains a complete chronological list of all activities at the mnemonic level and is the final mission planning product that feeds directly into the Command Management System (CMS).

One of the most difficult to define elements of the generic system is what was called (for lack of a more definitive name) "Flight Operations Team Support." In terms of functionality, this element differs from the other elements in that it doesn't have its own unique computational niche; i.e., it is not part of the essential data flow required to operate the spacecraft. It consists instead of information produced in the other elements, but organized and presented in formats suitable for Flight Operations Team support. The AXAF concept has clarified this function considerably by creating a support module called the Interface and Support Software (ISS),

formulated by combining selected subfunctions of the Orbital Mechanics element with spacecraft environmental and orientation displays. In conjunction with appropriate scheduler displays, Flight Operations team personnel will be provided with all the mission planning information needed to conduct flight operations.

The advantages of this approach are that duplication of planning tasks and products can be minimized, and that ISS data and displays, which are also needed by other off-line software systems (attitude determination and spacecraft analysis), can be more easily shared. As a generic element, this solution works well because the selected orbital mechanics subfunctions and environmental displays, such as ephemerides and ground tracks are independent of schedule and spacecraft complexities.

A listing of the subfunctions included in each element of the AXAF concept is presented in Table 2.

Table 2. List of Mission Planning Functions/Subfunctions

Accept scheduling requests
Accept and validate observation requests
Generate engineering requests
Provide edit and override capability
Generate mission schedule
Provide optimal observation ordering
Provide timeline editing tools
Validate schedule
Perform guide star selection
Determine target visibility and availability
Check bright object constraints
Check object occultations
Determine spacecraft roll constraint
Check thermal constraints
Check radiation zone constraints

Table 2. List of Mission planning Functions/Subfunctions (Continued)

Check orbit day/night constraints
Determine supporting resource requirements
Calculate data storage requirements
Determine power requirements
Calculate spacecraft maneuvers
Calculate solar array position
Determine LGA visibility
Determine OBC memory availability
Determine uplink and tracking contact needs
Generate DOT
Translate observation schedule to DOT
Provide edit and override capability
Provide support for OBC updates
Generate reports and engineering displays
Display spacecraft activity timeline
Provide processing and error log displays and reports

AXAF FUNCTIONAL FLOW CONSIDERATIONS

Modularity/Standalone Capability

Because of its similarity to the HEAO-2 and Hubble missions and the unique mission planning features developed for them, the AXAF mission planning requirements were written with a strong emphasis on functional modularity and standalone operation. It is therefore not surprising that the resulting design approach also gives great importance to these considerations. Standalone operation and modularity greatly facilitate the reconfiguration of software data flows in response to flight contingencies, and minimizes maintenance costs.

Flow Sequence

The independence of mission planning and scheduling functional elements

and the flexibility required of the scheduler module dictate the fundamental flow sequence of the AXAF Mission Planning and Scheduling concept. This fundamental principle is that all constraint calculations related to spacecraft ephemerides are completed before the scheduling process begins. The separation of orbital mechanics and scheduling functions in this manner allows independent development of each discipline and prevents coding entanglement that makes software maintenance difficult. The body of support data generated also facilitates troubleshooting analyses in contingency situations and reordering of functions as mission conditions change.

Another fundamental principle of the AXAF design concept is the clean division of the schedule generation function from the spacecraft management function. The former is concerned with determining what the schedule of activities will be, while the latter comprises all the spacecraft support (such as appendage movement) required to implement the schedule. Breaking the mission planning process at this point allows review of the spacecraft schedule by science and flight operations personnel before proceeding with the generation of detailed spacecraft functions and commands. Since communications networks require support requests 3-4 weeks prior to execution, mission schedules must be completed long before command generation is necessary. Thus the production of mission schedules as separate entities from the Detailed Operations Timeline simplifies schedule review and editing and reduces control center workload.

CONCEPT REFINEMENT

Based on the AXAF prototype concept, the generic mission planning and scheduling concept needs little

refinement. As mentioned earlier, the only element in the original concept that needed more definition was Flight Operations Team Support. This problem appears to be satisfactorily resolved in the AXAF solution. By putting together subelements of the orbital mechanics function that are independent of schedule with environmental and spacecraft geometric displays, a much more definitive element is formed. In the authors' opinion this refinement improves the focus of this function.

In terms of process flow, further concept refinements can be realized by associating the communications planning function with the scheduling element instead of spacecraft management. This accounts for the scheduling of contacts based on engineering request selection criteria and facilitates schedule editing.

CONCLUSIONS

After comparing the AXAF mission planning and scheduling design concept with the generic concept, it appears that the generic model is valid, and that it can reasonably be expected that most future designs will comprise generic elements. The AXAF experience also suggests that orbit type is not as strong a design driver as previously thought. Before cancellation of the AXAF-S project, sufficient concept evaluation had been done to assure that a single mission planning and scheduling system could support both the highly elliptical, high orbit AXAF-I and the low polar orbit AXAF-S.

As a result of the AXAF design work, the authors believe even more strongly that a generic system for space astronomy missions is well within reach for unmanned missions, and much of this system can be used for manned missions as well. The mission planning elements that have the best chance of forming a "core"

system for all missions include (1) orbital mechanics, (2) observation and engineering request processing, (3) communications planning and (4) flight operations support. Once this core system has been standardized, the other functions can be incorporated one subfunction at a time. Eventually, this emphasis on generic systems will pay many dividends in the future by reducing software development and maintenance costs, simplifying user training and possibly even influencing spacecraft design.

REFERENCES

1. Guffin, O.T. and Onken, J.F. 1992. "Generic Mission Planning and Scheduling Concepts for Space Astronomy Missions." In SpaceOps 92, 2nd International Symposium on Ground Data Systems for Space Mission Operations. Pasadena, California.
2. Newhouse, M. and Guffin, O.T. 1994. "Mission Planning and Scheduling Concept for the Advanced X-Ray Astrophysics Facility (AXAF)." In AIAA Space Programs and Technologies Conference. Huntsville, Alabama: AIAA 94-4641.

A PC BASED TOOL FOR MISSION PLAN PRODUCTION

Authors: Jean-Pierre JOLI, Clet YVEN

Address: CNES - ET/EO/ER/MP
18 avenue Edouard Belin
31055 TOULOUSE CEDEX
FRANCE

Abstract

A satellite positioning is managed according to a MISSION PLAN (MP) which provides, on a minute accuracy basis, a chronological list of events and associated actions to be performed. This tool, called MM2, is designed under WINDOWS environment.

EXCEL is used to provide the MP itself. A VISUAL BASIC process then translate it into a graphic symbolic representation called Flight Plan (FP).

During operations, MM2 is also used to log the actual event dates and/or dated OPS MANAGER live comments.

*Key words:
Operations Management.*

Introduction

The MP is redacted and mainly used by the OPERATIONS MANAGER (OPS) to conduct operations.

To be safe it must be qualified during the simulation phase

To be useful it must be **up to date**.

This implies an important OPS workload when updating is handily managed.

Definition of a tool aiming to reduce human participation to only design tasks was then started.

It resulted in the following main specifications.

The tool must:

- Be PC based,
- be run under WINDOWS environment,
- only use "on-shelf" software,
- accept input data in "character type" files,
- allow easy adaptation to various spacecraft's and tracking networks constraints,
- allow a quick delivery, within basically 5 at least 10 minute, of a tuned issue,
- provide partial or complete plan without operator intervention when production process is started,
- allow, during operations, actual event dates and/or OPS live comments recording.

An updating strategy was also chosen.

General conventions

On a time point of view, in MP, all events are related to a main time reference which is booster lift-off.

MP is split down into a collection of time slices, roughly corresponding to the spacecraft physical orbit, called "orbit" and named by a mnemonic.

An orbit has its own time reference, itself related to the main.

Each orbit event refers to this orbit reference through a main Count-Down (C/D). If necessary, secondary C/D can be set-up.

All times are in UTC.

Events are either information to give or action to do.

An event can be either

- "simple", when it needs only one MP line to be completely described, or
- "complex" when it is an organized lists of sub-events. Flight Control Procedures (FCP) and ranging session (LOC) are complex events. In a complex event, each sub-event has its own duration and is time related to previous and following sub-events. It is assumed to begin at time 0:0:0.

Entities involved are clearly named

Application description

MM2 work is organized as follow :

- Tailoring of input data (TXT files),

- for one orbit: Merging, processing, sorting, formatting and print of results. When many orbits processing is requested the process is repeated.
- VISUAL BASIC processing to draw FP,
- Use in operations. All along, actual time and OPS live comments are logged. When orbit is completed an "as run" issue is produced.

Tailoring and creating input data

As show in figure N° 1 here after, some input data files are available from external entities. They are supposed to be in a text format allowing direct Excel input. If not, a text pre-processing is necessary and can be done with a text editor, Winword for example

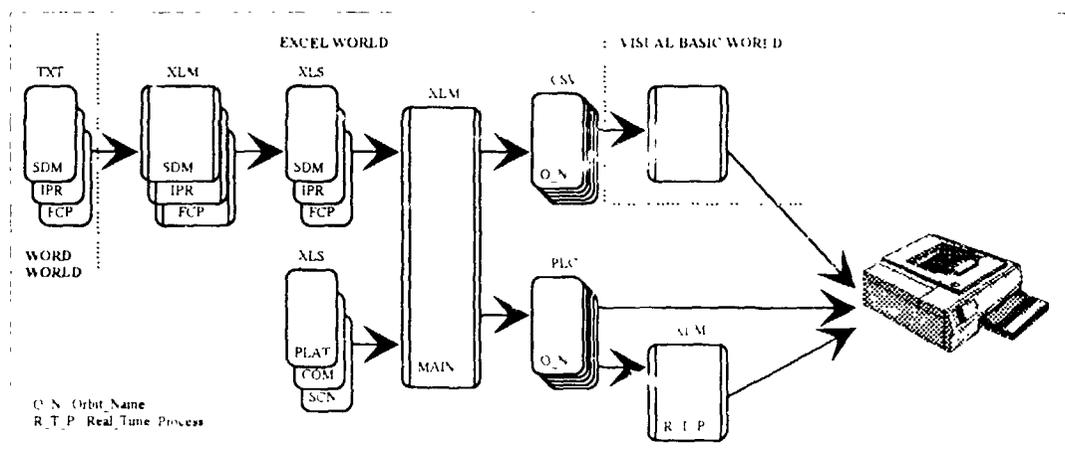


Figure N° 1 General process organization

When under Excel, some complementary treatments are applied (tailoring) They mainly consist in:

- Shifting the right data to the right column,
- deleting not relevant lines,
- naming all significant data area to allow easy access later on

Due to the fact that these data are supplied in various formats, three Excel specialized routines have been developed to make them comply with Excel main process input specifications.

- One for data coming from Flight Dynamics Center (tracking stations and sensors visibility's, eclipse periods, apogee date, etc.) witch deliver a file called SDM,
- one for data coming from Operational Orbit Center (interference predictions) witch deliver a file called IPR,
- one for data coming from Satellite Team (flight control procedures) witch deliver a file called FCP

Some other necessary files are internally setup under Excel. They are :

- PLAT, in witch are stored general time references and orbits data base (ODB).
- COM in witch are stored pre-defined comments (witch are complex events), a model of orbit and page banner and the general GO/NOGO sheets.
- SCN in witch are stored the orbit scenari A scenario describes work to do during a given orbit.

This last file is made out of dated lines witch can be:

- Free comments,
- reference to pre-defined comments (stored in the COM file),
- reference to FCP (stored in FCP file).

MAIN processing

Excel main program is working as follow :

First, read from PLAT file general time references and name of orbit to process.

Then by mean of the orbit name:

- Get from ODB:
 - The orbit reference name,
 - the orbit reference time,
 - the orbit "main operation to perform",
 - the first page number of orbit in MP.
- get from SDM and IPR files, data area related to orbit,
- process SCN one line at a time to:
 - Directly copy free comments to MP,
 - get and insert, from COM and/or FCP

Fore. UTC		True UTC	Main C/D	Action	Step description & comments	Sec. C/D
1994, Jul.21						
00 40			+01 59		S130-3 - Reconfiguration register	
				FDC	Display "SSH monitoring" (G6) on video page 4	
00 42			+02 01		S130-4 - Delete Time-Tag	
00 43			+02 02	OPS	Announce end of S130 procedure	
				OPS	Announce beginning of S140 - SOLAR ARRAY PARTIAL DEPLOYMENT OPS TM pages S14SAPDP EPS1 S140-1 - Initial configuration	Duration 00 30
00 57			+02 16		S140-2 - Solar array partial deployment	
				FDC	Display "Sun reference coefficients for EAM" (M2) on video page 2	
				FDC	Provide SSR with "Sun direction prediction" (L2) by Fax	
				FDC	Send to SCC "Sun reference coefficients for EAM" (F1)	
01 11			+02 30		S140-3 - Trickle charge	
01 13			+02 32	OPS	Announce end of S140 procedure	
				OPS	LOC sequence with Perth	
				SAT	Set the ranging "ON" onboard the spacecraft	
				OPS	Request NOC to start a 5 mn full LOC sequence at Perth	
				NOC	Report beginning of LOC sequence at Perth	

NO_GTO_1

Spacecraft Acquisition and Configuration

Page 205

Figure N° 2 · Mission Plan page example

to MP, pre-defined comment or FCP.
 In this last case, COM or FCP execution time is updated according to SCN specified time

- sort MP on a chronological basis,
- Update time taking into account last known date and time,
- set main C/D,
- save MP in appropriate format for later Visual Basic processing,
- Format to give it, its definitive look as shown in figure N° 2.

VISUAL BASIC processing

At the end of the Main processing of an orbit, a specific file is created by Excel, and stored in a ".CSV" format. This file contains all the information needed to generate the FP.

The application developed under Visual Basic then allows to position and draw the elements of the MP on a time pattern as shown in figure N° 3.

One is able to get like this a quick (2 minutes for one orbit) and accurate graphical representation of the MP

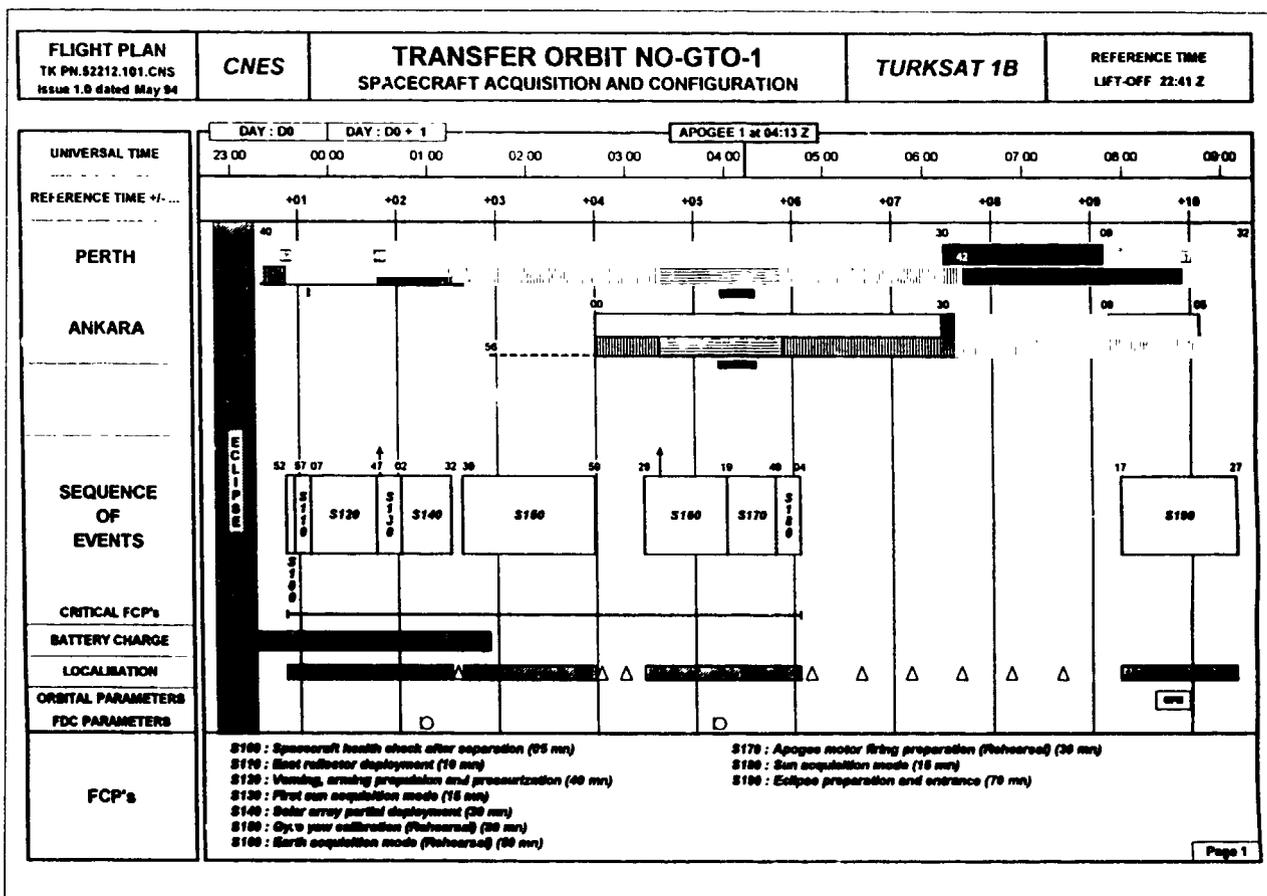


Figure N° 3 : Flight Plan page example

Real time processing

During operations a separated Excel program provides the following opportunities by simple click on the appropriate icon:

- Read PC clock and store sample in the right format at MP appropriate place,
- Insert dated lines logging live comments. These comments can be either input from the PC keyboard or pre-defined.

- when needed, finish the logging process and supply the "as run" issue.

Hardware environment

A 386 PC based configuration with a 5 Mbytes RAM, a 120 Mbytes mass memory and a laser printer is able to produce MP and to run it during operations.

However, at CNES TOULOUSE control center, due to presence of a concurrent Windows telemetry processing application on OPS workstation, we use a 486 based PC

Software environment

Software environment is quite basic

- MSDOS 5.0,
- WINDOWS 3.1,
- EXCEL 4,
- VISUAL BASIC 2,
- WINWORD 2,

Using MM2

When a project starts, first work is to "adapt" MM2 to the new environment

That means

- Select the appropriate language,
 - tailor PLAT file according to positioning strategy and time,
 - tailor COM file according to tracking network to be used and GO/NOGO format to apply,
 - create the SCN file according to Spacecraft Operations Handbook (SOH) and general constraints,
 - as soon as input data format is known and if necessary, modify the tailoring routines
- When input data are available setup the Excel SDM, IPR and FCP files,

At this time, MM2 is ready to supply a MP and FP first issue which will be used as support for Simulation and Rehearsal Phase (S&RP)

We can notice that this previous work, which can be important, is usually done during calm periods.

Some updates, mainly concerning FCP and SCN, are done during S&RP. At the end MP and FP are qualified

SDM data, taking into account last predictions for blinding or eclipse problems, is usually issued two weeks before launch. MP and FP are once more updated

Since this time each update has to be quickly delivered (within 5 minutes).

According to update strategy, at a given time, only the next orbit update is mandatory.

Complete update, if necessary, can be slightly delayed

As a consequence, an update of the first orbit is issued as soon as the actual launch date is known. It must be available before first spacecraft telemetry acquisition.

Then and if necessary, an orbit by orbit update can be initiated taking into account new orbit data as soon as they are available. This allows accurate following of maneuver dispersions.

Conclusion

First use of MM2 was for HISPASAT 1B positioning. This spacecraft was spin and S band controlled in transfer. The MP was issued in English.

Since, MM2 has been adapted without any major difficulty, for TURKSAT 1B, which is 3 axis and KU band controlled.

Today, adaptation to TELECOM 2C is in progress. In this case the main change is that MP will be issued in French

This clearly demonstrates the flexibility of this tool

On an efficiency point of view, at this time, we only have experienced slight deviations from nominal launch and maneuver performance. All goals were then reached.

However, we are presently reflecting on an "assistance to design" program which could allow improved performance as well as coherence controls in case of major problem requiring a quick and complete MP reorganization

TOWARDS A CLASS LIBRARY FOR MISSION PLANNING

Olivier Pujo¹
Simon T. Smith², Paul Starkey², Thilo Wolff¹

¹European Space Operations Centre (ESOC)
Robert Bosch Str. 5
D-64293 Darmstadt

²Cray Systems Ltd.
DAS House
Quayside
Temple Back
UK-Bristol BS1 6NH

Abstract

The PASTEL Mission Planning System (MPS) has been developed in C++ using an Object-Oriented (OO) methodology. Whilst the scope and complexity of this system cannot compare to that of an MPS for a complex mission one of the main considerations of the development was to ensure that we could re-use some of the classes in future MPS. We present here PASTEL MPS classes which could be used in the foundations of a class library for MPS.

Key words: Mission Planning, Object-Oriented, Class Library

Introduction

PASTEL is an experimental optical terminal to be flown on as a passenger on SPOT-4, the earth observation spacecraft developed and operated by the Centre National d'Etudes Spatiales (CNES). A

corresponding optical terminal will be flown on the ARTEMIS spacecraft, which is part of the Data Relay and Technology Mission programme (DRTM) of the European Space Agency (ESA). PASTEL will have a separate Mission Control System (MCS), which will be operated by ESA. The PASTEL Mission Planning System (MPS) is part of the MCS and has been developed in C++ using an Object Oriented (OO) methodology.

In ESA, the area of Mission Planning is one in which the use of generic systems has been considered only recently, in sharp contrast to spacecraft control systems for which ESA has been using configurable multi-mission systems for nearly twenty years. ESA's mission planning systems have been project specific development and there has been very little carry over of the expertise, tools or software from one project to another. A recent ESA study showed that there are areas of commonality between different mission planning systems (ESA, 1992). The use of traditional software technologies (FORTRAN, PASCAL,

functional decomposition...) for MPS development has certainly been one of the hindrances to provision of re-usable components.

One of the strong claims of OO approach is the possibility of re-use. Re-use in an OO system is usually implemented through the Class Library concept, where a Class Library is a collection of general purpose components that can be used as a basis for further refinement on specific projects. The concept is similar to that of standard graphics or numerical libraries, with the important difference that by using the OO principle of inheritance the behaviour of the components can be modified (to add, remove or alter features).

Whilst the scope and complexity of the PASTEL MPS cannot compare to that of a mission planning system for a complex science or earth observation mission, it is used in this paper to highlight certain possibilities for re-use. One of the steering factors in the development of the PASTEL MPS was to try to ensure that some of the classes would be re-usable in future mission planning systems. An immediate motivation being potential re-use in other areas of the DRTM programme.

We start with an introduction to the PASTEL mission. Then we look at various aspects of the PASTEL MPS: the development process and object class hierarchy. A discussion on the re-use potential of the PASTEL MPS Timeline and Reservation Plan area follows.

PASTEL and the SILEX experiment

PASTEL and its counterpart terminal, OPALE, mounted on the ARTEMIS satellite

form the SILEX (Semiconductor Inter-Satellite Link Experiment) mission which will be used to downlink high rate data generated by SPOT's optical camera, using ARTEMIS as a data relay. For technological purposes PASTEL will also be able to point to stars.

The PASTEL terminal will be operated by ESA from the PASTEL Mission Control System (MCS) located in the ESA Redu station. Control and monitoring information will transit through the SPOT-4 Control Centre, located at Toulouse, in a cross-support scenario. The planning of the SILEX experiment is under the responsibility of the PASTEL MCS, which will coordinate with the SPOT-4 Control Centre and the ARTEMIS Control Centre.

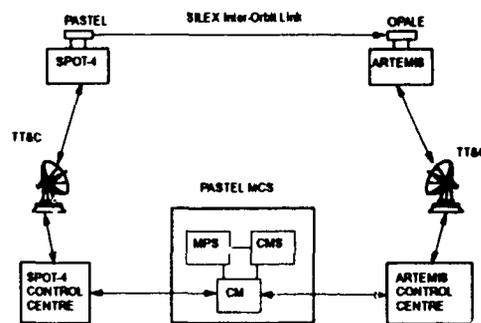


Figure 1: SILEX experiment and PASTEL MCS

As shown in Figure 1, the PASTEL MCS comprises three principal subsystems, (1) a Control and Monitoring System, (2) a Mission Planning System, and (3) a Communications Monitor. Within the PASTEL MCS, the MPS is in charge of all planning activities.

PASTEL MPS

The PASTEL MPS main functions (ESA, January 1993) are:

- (A) to allow the MPS operator to coordinate the production of the Reservation Plan, (which defines the periods in which PASTEL can communicate with OPALE, and the periods where star tracking can be performed by PASTEL),
- (B) to produce an Operations Timeline, containing all the details including telecommands of the operations to be scheduled from the PASTEL MCS, under MPS operator control.

The Reservation Plan holds SILEX communications sessions, which are also called "windows". At the first stage of the planning process, the communications sessions are called visibility windows and are derived from flight dynamic information provided by SPOT-4 Control Centre. The following steps involve a number of iterations between PASTEL MPS, SPOT-4 CMP and ARTEMIS MCS to allow each centre to reserve or cancel the windows according to their operational constraints.

PASTEL MPS development

PASTEL MPS has been developed using C++ and OO methodology following the Object Modelling Technique (Rumbaugh *et al*, 1991). A traditional waterfall life cycle process model (ESA, 1991) was adopted with the following adaptations. The user interface was prototyped at an early stage. The design documentation was simplified: a unique design document replaced the traditional Architectural Design and Detailed Design Documents. And finally integration of components was performed from very early design stages. The overall effort for

the development was in the area of 30 man-months, and 24000 lines of codes were produced.

The object-oriented approach was primarily adopted for this development in view of the potential re-use of it in the frame of the ARTEMIS MCS development to support the scheduling of OPALE. PASTEL MPS will be the first OO system delivered in ESOC for operational usage.

PASTEL MPS Object Classes

The overall object classes hierarchy for the core of PASTEL MPS is provided in figure 2 (ESA, June 1993). Two parallel structures appear in this hierarchy: the Reservation Plan and the TimeLine. For simplification purposes, this figure does not cover the class hierarchy for the user interface objects, which were introduced to dissociate application objects from the user interface. We will first provide a short outline of the Reservation Plan and the Timeline before discussing their potential re-use.

The Reservation Plan (and its associated user interface objects) contains, from a user perspective, the list of all windows for a planning period (typically of five weeks). Each window has a status which determines whether the corresponding communication session is reserved or cancelled.

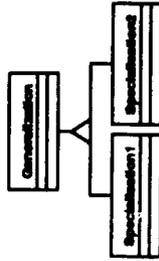
The operator interacts with the Reservation Plan through a specific display, called the Reservation Plan Mode Display, which consists of two areas:

- The Reservation Plan Index, which allows the operator to navigate through the weeks and days of the

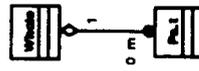
Legend:



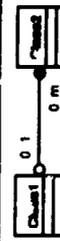
Class-&-Object:



Generalisation-Specialisation



Whole-Part



Instance Connection

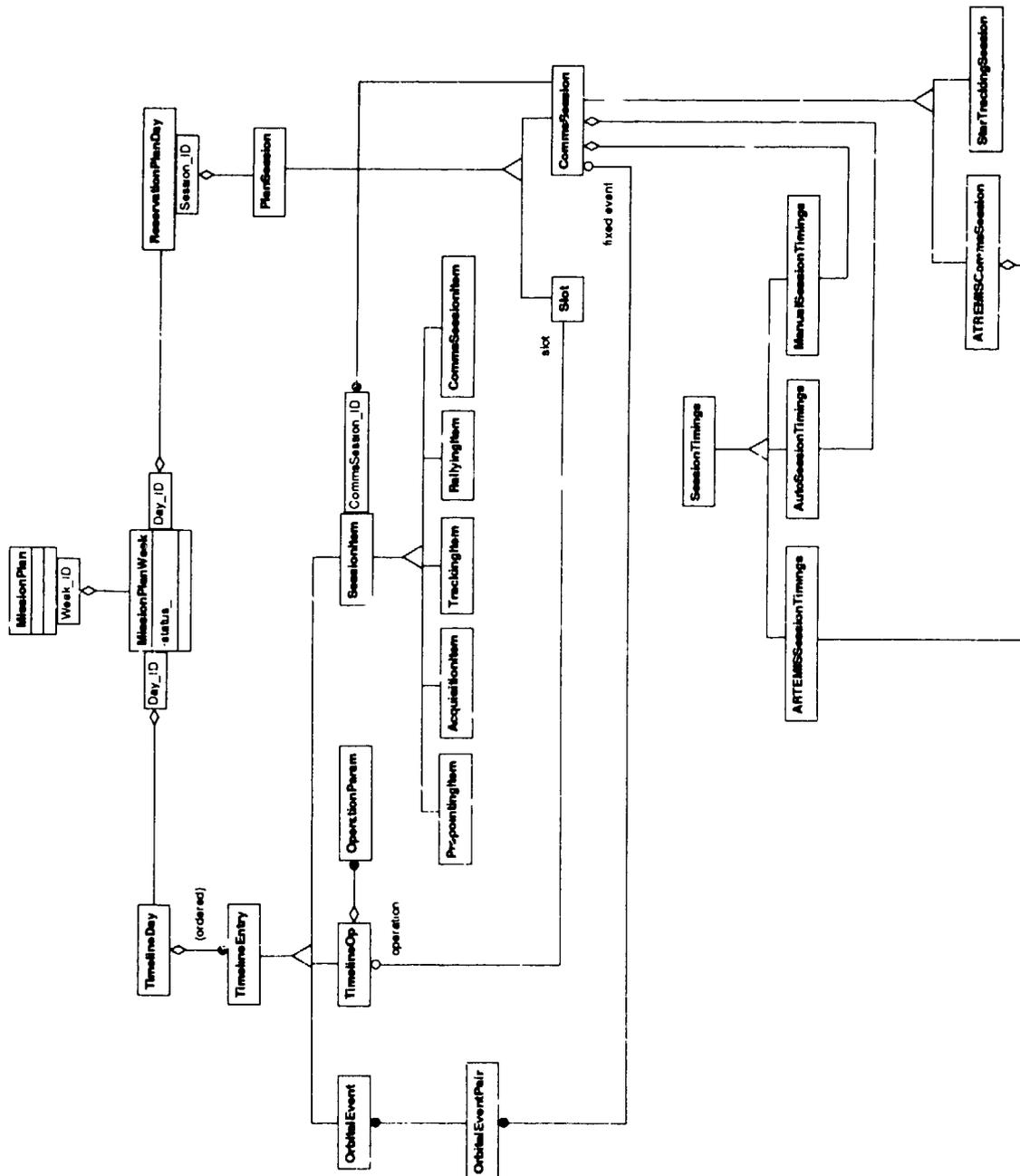


Figure 2: Overall Object Hierarchy

Plan and to select the window to work with.

- The Window Display, which displays all information relevant to a single window and provides the operator with a set of options for controlling the planning of the window.

A parallel data structure to the Reservation Plan, the Timeline, is available to the operator from the start of the planning cycle. The Reservation Plan and the Timeline provide two different views of approximately the same information describing the operations to be scheduled on-board. Where, in the Timeline, the information is formatted in templates close to command sequences, in the Reservation Plan, the information is formatted in templates which are closer to the intuitive user perception of the planning. In other words, the Reservation Plan provides a macroscopic view of the planning, whilst the Timeline provides a microscopic view.

The Timeline contains mainly sessions items, which describe the operations to be performed to establish a communications session. The Timeline may also contain other operations, such as Laser diodes calibrations. In principle the operator is free to enter operations into the Timeline at any stage, although in practice most of the operations will probably be entered at a late planning stage once the Reservation Plan has been more or less finalized.

Re-use of PASTEL MPS classes

The PASTEL MPS is simple compared to that of other ESA mission planning systems for the following reasons:

- it operates within a fixed set of resources and constraints,
- scheduling tasks are handled manually,
- the communications sessions scheduled result in a fairly fixed pattern of operations in the Timeline,
- it is an off-line system, i.e. there is no requirement to perform real-time re-scheduling of the mission.

Despite these restrictions, the areas covered by the PASTEL MPS are equivalent to parts of more complex planning systems. Thus some of the classes identified in the PASTEL MPS Object hierarchy can be considered for re-use.

Timeline

The most promising area for re-use in the PASTEL MPS is certainly the Timeline area. It includes several classes: TimelineDay, TimeLineEntry, SessionItem, OrbitalEvent, TimelineOp, etc., and for each of these classes we foresee potential for genericity. However, we will focus in our discussion on the timeline itself, which is implemented in PASTEL MPS in the TimelineDay.

One fundamental component of any spacecraft mission planning system is the timeline. The timeline is the basic structure to store information required to plan a mission:

- scheduled operations such as time tagged operations to be executed on-board the spacecraft or at the ground station,

- events pertinent for spacecraft operation such as eclipse entry and exit, ground station visibility period.
- spacecraft on-board status changes such as instruments mode switching or on-board tape recorder activity.

PASTEL MPS *TimelineDay* class is an interesting starting point because it highlights very generic features, namely:

- time-ordered list with protected insert mechanism (in order to force entries to be inserted in chronological order),
- support of heterogeneous list items, i.e. all elements forming it do not necessarily belong to the same class,
- support of active display filtering, i.e. it is possible to select list items of the selected types.

The Timeline is in essence a chronological list of items, which correspond either to operations to be executed on-board or to events such as eclipse times, etc. Whenever an item is added to the list, it is essential to check that this is done according to a correct time order.

To meet the genericity objective a timeline clearly needs to support a mix of objects. There are some good reasons to distinguish between timeline inputs such as operations or orbital events. We use the OO inheritance mechanism to solve this problem as shown below in figure 3. By defining *operation* and *eclipse* as subclasses of a more general class, *timeline_entry*, we can construct a timeline with entries of type *timeline_entry* provided that the programming language supports the use of the subclasses *operation*

and *eclipse* in place of the general *timeline_entry* contained in the class definition for timeline.

Editing the timeline is, by nature, a highly interactive task and the support of active display filtering is a common requirement. In fact any combination of elements types should be displayable. This has been implemented simply by adding a dedicated attribute, *type*, to the *timeline_entry* class, which is then used by the mission timeline user interface object to implement the active filtering.

Reservation Plan

Another potential area of re-use in the PASTEL MPS is the Reservation Plan area. On the overall object hierarchy (figure 2), the parallel between the classes *TimelineDay* and *ReservationPlanDay* is quite striking. They are formed respectively of *TimelineEntry* and *PlanSession*, which are generic classes for parallel structures such as *SessionItem* and *CommsSession* or *TimelineOp* and *Slot*.

Schematically, one could say that the

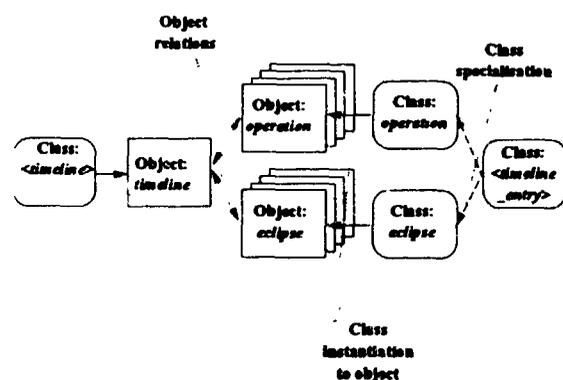


Figure 3: Mixing objects within the timeline

Reservation Plan is used in early planning phases, while the Timeline is used in the last

planning phase. However, there is no general rule forbidding the operator to edit the Timeline at an early stage or the ReservationPlan at a late one. In fact, they both contain more or less the same information and what distinguishes them is more the way in which an operator wants to interact with them.

The Reservation Plan is geared more to specific planning aspects of PASTEL; it holds e.g. information such as terminal constraints, which are useful only to compute communication sessions timings, or window status attributes (reserved|cancelled|...), which are not held in the PASTEL timeline parallel structure.

In order to keep the Reservation Plan manageable, three levels of hierarchy are defined: week, day and window. This breakdown maps the events managed in the Reservation Plan and the planning cycle. Although it is clearly specific to PASTEL, it could be very simply generalized through the OO inheritance mechanism and/or by renaming some classes.

This breakdown is also useful to provide external users with some "snapshots" of the Plan or to update the Plan according to new data from external users. This is performed in PASTEL MPS by specific callbacks and methods, which could be re-written for any application. It is anticipated that the definition of external user interface is in any case specific to each mission. All that a generic mission planning class library needs to provide is the anchor points to these external interfaces, which are provided, in PASTEL MPS, in the methods of the `ReservationPlanItems` (`ReservationPlanDay`, `CommsSession`...).

Finally, the possibility to manage various versions of the Reservation Plan is believed to be of interest to a number of missions.

PASTEL MPS allows two versions of the Reservation Plan to co-exist, one being the reference and the second corresponding to an update generated when receiving inputs from external users. The two versions can be compared and the operator can select the update to apply to the reference Reservation Plan. This feature is used only for temporary purposes but could be used in a broader way to allow multiple operators working concurrently.

To summarize, the following features are, we believe, quite generic in the PASTEL MPS Reservation Plan area:

- the breakdown of a reservation plan into smaller units, which is a mandatory requirement to keep the plan manageable;
- the requirement to provide to external users "snapshots" of the plan to synchronize planning activities.
- the configuration management of several plan versions.

Conclusions

At ESOC a number of object-oriented developments have recently taken place or are in progress, PASTEL MPS being one of the very first. Whilst the prime objective of the PASTEL MPS development was not to provide a generic class library for mission planning, there was a strong motivation to achieve some genericity in view of the potential for re-use on ARTEMIS and DRS satellites.

It has been shown that there is a good expectations that certain PASTEL MPS classes can be re-used. The Timeline and Reservation plan areas seem very promising starting points. The on-going Generic Mission Planning Facilities for Operations Study should confirm these expectations by consolidating some aspects of these PASTEL MPS classes to make them more generic and by using them to model other ESA mission planning systems.

It is anticipated that the result of this work is fed into SCOS II, ESA's new infrastructure for spacecraft control, which is currently under development and which is being built as a C++ class library for spacecraft control.

frame of ESOC Contract 10694/93/D/IM(SC) - Study on Generic Mission Planning Facilities for Operations - performed by Cray Systems Ltd. and Matra MarconiSpace.

References

ESA (1991) ESA Software Engineering Standards, ESA PSS-05-0 Issue 2.

ESA (1992) Generic Mission Planning Toolset, ESA Report CR(P) 3600.

ESA (1993, January) PASTEL Mission Control System Software Requirements Document.

ESA (1993, June) PASTEL Mission Control System Architectural Design Document.

J. Rumbaugh *et al.* (1991) *Object-Oriented Modelling and Design*. Prentice-Hall International .

Acknowledgements

The PASTEL MPS was developed by Cray Systems Ltd. under an ESOC contract 8793/90/D/IM. The re-use of the PASTEL MPS classes is currently under study in the

AUTONOMOUS MISSION PLANNING AND SCHEDULING--INNOVATIVE, INTEGRATED, RESPONSIVE

Charisse Sary
Simon Liu
Computer Sciences Corporation

Larry Hull
NASA/Goddard Space Flight Center

Randy Davis
Laboratory for Atmospheric and Space Physics
University of Colorado at Boulder

ABSTRACT

Autonomous mission scheduling, a new concept for NASA ground data systems, is a decentralized and distributed approach to scientific spacecraft planning, scheduling, and command management. Systems and services are provided that enable investigators to operate their own instruments. In autonomous mission scheduling, separate nodes exist for each instrument and one or more operations nodes exist for the spacecraft. Each node is responsible for its own operations which include planning, scheduling, and commanding; and for resolving conflicts with other nodes. One or more database servers accessible to all nodes enable each to share mission and science planning, scheduling, and commanding information. The architecture for autonomous mission scheduling is based upon a realistic mix of state-of-the-art and emerging technology and services, e.g., high performance individual workstations, high speed communications, client-server computing and relational databases. The concept is particularly suited to the smaller, less complex missions of the future.

INTRODUCTION

NASA's scientific spacecraft are unique and valuable resources, so it has always been an important part of mission operations to assure

that the time a scientific spacecraft spends in space is utilized as fully as possible in making observations and conducting experiments. To achieve this, most NASA missions plan their scientific activities well in advance; convert those plans into formal spacecraft and instrument schedules on a daily, weekly or monthly basis; and then generate and uplink the commands needed to carry out the scheduled activities.

There are two principal types of mission scheduling problems for NASA. The first type arises when a spacecraft must perform a large number of activities in serial fashion. An example is the Hubble Space Telescope (HST). There are always hundreds of proposed observations in the queue for the HST, and typically only one observation can be made at a time. HST schedulers must select the observations to be supported and then lay them out as single thread of activities. The problem is complicated further by the fact that an experiment may require several observations: if the HST is scheduled to look at a particular target today, then it may also be committed to viewing the target on future occasions as well. Serial scheduling problems are well known (they occur in many terrestrial applications), but they are inherently difficult and time consuming to solve. Developers of automated schedulers for space missions that must handle this kind of problem tend to concentrate on devising

algorithms that increase scheduled observing time while reducing the processing time needed to generate the schedule.

The second type of mission scheduling problem is where a spacecraft can perform a number of major activities in parallel. An example is the forthcoming Earth Observing System (EOS) AM satellite which will carry instruments that can conduct their observing programs simultaneously and more-or-less independently of one another. It has long been recognized that this kind of parallel scheduling problem allows for a distributed solution. Investigators, responsible for each instrument on a spacecraft, generate the schedule for their own instrument. These detailed instrument plans can be collected and combined with a plan for spacecraft housekeeping activities to form a master schedule that can then be checked for conflicts or resource over-subscription.

Since 1986, the Data Systems Technology Division at Goddard Space Flight Center (GSFC) has been investigating scheduling issues relevant to GSFC missions through analysis, prototyping tasks, and testbeds. Recent work has concentrated on EOS and studies of planning and scheduling in a distributed environment. Because the scheduling of observations by most EOS spacecraft falls into the parallel scheduling category described above, the EOS project decided to sponsor an EOS Planning and Scheduling Testbed project during 1992-1993, to explore issues associated with distributed instrument scheduling.

The EOS Testbed was successful in demonstrating that distributed planning and scheduling is feasible for a project like EOS. Several important problems were discovered, but not resolved, however. For example, it proved difficult to keep all of the nodes' scheduling activities synchronized. The scheduling process required substantial coordination between personnel at all nodes. Even when nodes coordinated there were problems, such as nodes not having the most

up-to-date ephemeris data available for use in their scheduling.

An interesting result from the EOS Testbed was that conflicts between instruments were usually best resolved by making the instrument investigators aware of the problem and letting them work it out for themselves. To aid in conflict resolution, it would have been useful for investigators to be able to see schedules for instruments other than their own (a feature that the EOS Testbed did not provide). As the testbed progressed the need for a "central scheduler" became less clear. Ideally, every scheduling node—not just the central scheduler—would have access to all information needed for scheduling, and every node would be able to view the spacecraft schedule and any instrument schedule. The ability to detect constraint violations and conflicts, and the potential to automatically resolve simple conflicts, are important capabilities for a distributed scheduling system. However, these functions need not be implemented within a central scheduler.

An autonomous mission scheduling concept has been developed that may eliminate the problems noted above. As shown in Figure 1, separate nodes exist for each instrument and one or more operations nodes exist for the spacecraft. Central to this concept is one or more databases that make needed information available to all nodes. For example, the most up-to-date ephemeris data is always available in a database. Similarly, all nodes have access, via the database(s), to the most current schedules for the spacecraft and for all instruments. All scheduling system transactions become transfers of information to or from a database, using a standard query language (SQL). The schema of a scheduling database is flexible and easy to modify, so new information can be added as needed.

Along with the database approach, the autonomous mission scheduling concept proposes a client-server architecture for a distributed scheduling system. Services, like resource tracking, conflict detection and

conflict resolution, can be invoked by a scheduling node as needed. Distributed scheduling may be one of the first opportunities to actually apply the client-server architecture to space mission operations.

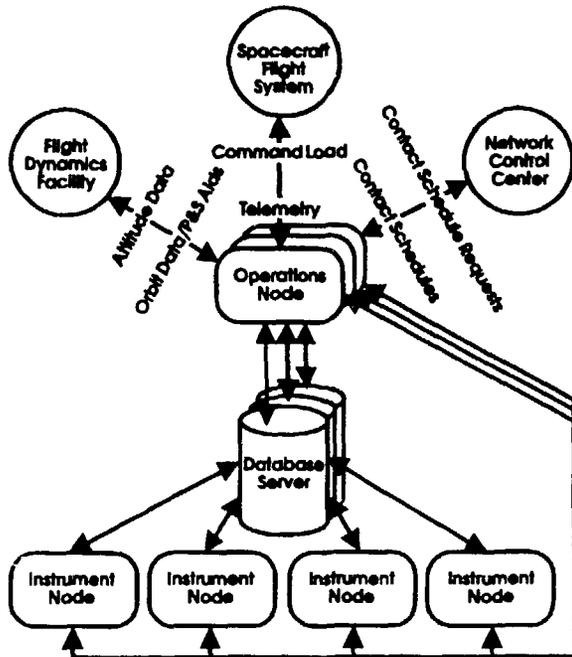


Figure 1. Decentralized and Distributed Scheduling

We believe that, even with the trend toward smaller and simpler spacecraft, distributed scheduling systems may provide new and exciting capabilities. For example, multiple investigators can independently schedule the use of a single shared spacecraft or instrument, or simultaneous observations from multiple spacecraft.

The autonomous mission scheduling operations concept supports key features of the Reusable Network Architecture for Interoperable Space Science, Analysis, Navigation, and Control Environments (Renaissance), a new approach to the development and operation of Mission Operations and Data System Directorate (MO&DSD) ground data systems. This approach avoids technical obsolescence and

facilitates hardware and software reuse by using generic components to support science and mission operations. With generic, reusable components, ground data systems will be rapidly and inexpensively built by tailoring components for each new mission. Each ground data system will consist of a number of physically independent, possibly geographically distributed nodes. These nodes would operate together and participate in coordinated planning, scheduling, and commanding using client-server computing and standards-based open systems.

ARCHITECTURE

The autonomous mission scheduling architecture is distributed with application functionality and data partitioned between workstations (clients and servers) connected to local area networks (LANs). Autonomous mission scheduling functions are allocated to components or nodes, and nodes are integrated together to produce a ground system for a target mission. Many different ground system architectures are possible by integrating different combinations of functions and nodes. A typical autonomous mission scheduling architecture is illustrated in Figure 2.

In this architecture, a Mission Operations Center (MOC), the database server, the Flight Dynamics Facility (FDF), and the Network Control Center (NCC) are all located at GSFC. Since a Science Operations Center (SOC) is remote, the MOC and SOC do not share telemetry processing and state vector determination functions. The FDF located at GSFC, provides orbit and attitude planing and scheduling aids. The NCC, located at GSFC, provides network scheduling data to the MOC and remote SOC. A specialized node, a database server, at the MOC, receives and stores this data. Nodes store planning, scheduling, and commanding data on the database server, and may access other nodes' planning, scheduling and commanding data of interest as well. Nodes can access a database

server whether they are remote or not, the only difference being in the kind of network interface used; remote nodes access the database server through a wide area network (WAN) and local nodes through a LAN. The database server node also detects inter-instrument and instrument-spacecraft exceptions, and notifies affected nodes to begin negotiations in order to resolve the exception. GSFC nodes communicate with one another through a LAN, while the remote SOC communicates with GSFC nodes through a WAN.

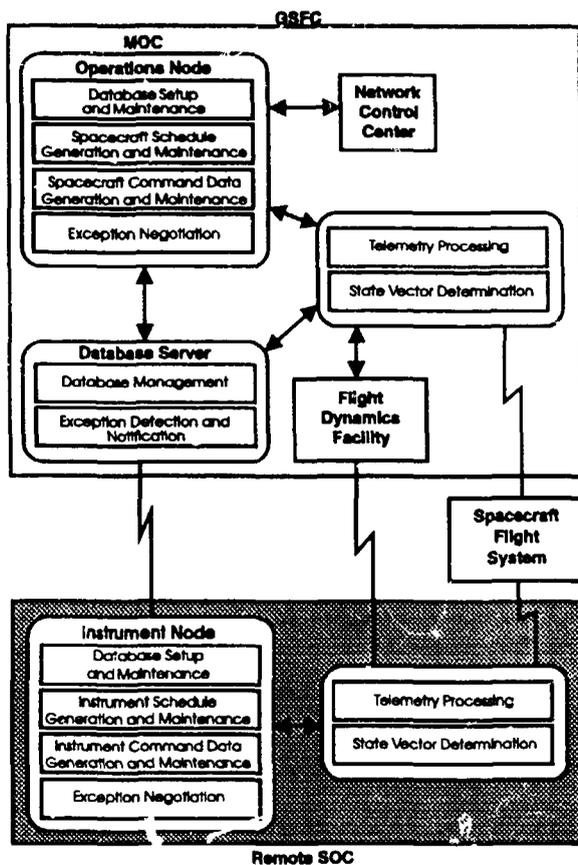


Figure 2. Architecture

The Instrument Node, the Operations Node, and the Database Server Node share several functions. The Database Setup and Maintenance function enables remote or client nodes to access the database server for common planning, scheduling, and commanding data. It stores network

schedules, received from the NCC, on the database server, and notifies nodes when this data is initially available. It also maintains the node's local database, which contains data not useful or accessible to other nodes.

The Schedule Generation and Maintenance function generates and stores, on the database server, coordination and operation constraints and activity definitions for the instrument or spacecraft. This information describes nominal operations and planned unique operations and will be used by the database server to detect exceptions later. This function plans and schedules resources to support spacecraft or instrument operations (e.g., scientific observations, calibrations, maintenance), generates and maintains spacecraft or instrument schedules, and stores these schedules on the database server. It designates, as a part of each scheduled activity, the appropriate commands or command sequences to invoke an activity. This function accesses the database server for planning and scheduling data, including data received from the NCC, network resource support schedules, coordination constraints, and activity definitions.

The Command Data Generation and Maintenance function stores instrument or spacecraft command definitions on the database server. Command definitions are used to generate command data and to detect command exceptions. This function extracts the appropriate command or command sequence from command definitions, inserts the necessary parameters, creates the node command data, and stores this data on the database server. It converts composite (instrument and spacecraft) command data to binary, creates a network packet, and uplinks command data to the spacecraft during a Tracking and Data Relay Satellite System (TDRSS) contact. This function also extracts real time command data from the database server, converts command data to uplink format, and uplinks the result when specified to the spacecraft for execution when received onboard. It resolves commanding exceptions,

validates and verifies command data, and maintains command history.

Deviations from normal behavior or unexpected situations are *exceptions*. The Exception Negotiation function coordinates and negotiates with other nodes to resolve exceptions, following the receipt of a message indicating that an exception has occurred.

The Database Management function, provided by a commercial Database Management System (DBMS), manages planning, scheduling, and commanding data stored by nodes. This includes insuring that the data is stored, modified, and accessed correctly, that the security and integrity of the data is maintained, and that distributed, concurrent, reliable, and efficient access is provided.

The Exception Detection and Notification function notifies nodes when new data is available, checks schedules and command data for exceptions, creates a message describing the exception, and forwards the message to affected nodes.

CONCEPT

Long Term Planning

Long term mission planning establishes mission objectives in an overall science operations plan and a long term spacecraft operations plan. Long term mission planning begins with the project scientist and principle investigators producing a long term science plan for the instrument complement. The flight operations team uses this long term plan to develop a corresponding long term plan for spacecraft operation.

With the NASA mission model evolving from a small number of large missions to more numerous but smaller, less complex missions, both the long term science plan and the long term spacecraft operations plan are expected to be relatively brief and to cover largely routine operation, observation, maintenance,

and calibration activities. The long term science plan also includes planned, unique operations such as contingency and emergency activities and details concerning coordinated activities and observations.

Based on the long term plan, scientists and flight operators define and store information in the database. The information includes inter-instrument and spacecraft coordination constraints; activity definitions which depict normal operations; command definitions which specify commands, command sequences, and parameters for activity execution; and operation constraints to maintain the health and safety of instruments and spacecraft subsystems.

Initial Scheduling

A large number of instruments have repetitive data acquisition cycles. These natural cycles are not necessarily the same for all instruments on a given mission, and some instruments do not have such cycles, e.g. targeting instruments. Nevertheless, instruments with natural repetitive data acquisition cycles find it easiest to plan and schedule instrument activities within these cycles.

The objective of initial scheduling is to define instrument and spacecraft operation, observation, maintenance, and calibration activities for a given interval. Initial instrument scheduling is done at the SOC and initial spacecraft subsystem scheduling is done at the MOC. All participants in initial scheduling may access available planning and scheduling information in the database. Intra-instrument conflicts are detected and resolved locally at each node. Inter-instrument and instrument-spacecraft conflicts are detected and resolved as described in the next section. The results of initial scheduling are stored in the database.

In the past, for large missions, initial scheduling was used to define requirements for communications resources and services

requested from the NCC. For future smaller missions, the initial schedule will largely be used to detect exceptions. For the less complex missions of the future, requests for communications resources and services are expected to be routine, repetitive, and largely independent of the mission schedule.

Exception Handling

In the past, planning and scheduling systems monitored the scheduling process continuously to detect exceptions. For autonomous mission scheduling, exceptions are detected when the potential arises. An exception does not necessarily have to be an error but is something that requires attention. Exceptions are detected by software and may require special handling. *Exception detection* is checking for and determining that an exception has occurred. *Exception notification* is informing nodes that an exception has occurred. *Exception handling* is responding to a notification and resolving an exception once notified. With this approach, once an exception is detected, it is handled before a major problem arises.

Exceptions can be schedule or command data exceptions. The three types of exceptions are:

- operator errors such as failing to produce information by a deadline or storing incorrect information.
- deviations from normal operations which may or may not be erroneous. An example of a deviation is a late change which is not preplanned and uses leftover available resources. Deviations do not necessarily create conflicts.
- resource, constraint, intra-instrument, and inter-instrument conflicts.

When an event occurs, exception detection is invoked. Two events that trigger exception detection are:

- operator actions such as adding to, deleting from, or updating the database.
- deadlines for performing an action or receiving data such as missing a deadline for receiving an initial schedule.

If an exception is detected, an exception notification message is generated and sent to the nodes involved. If more than one node is involved, one node is given primary authority for resolving the conflict. The responsible node may be:

- The owner of the activity that contributes the most to the conflict.
- The owner of the most critical or most important activity.
- The involved node that has the most restrictive operation constraints.

Upon receiving a notification message, nodes analyze exception data contained within the message, resolve any internal errors, deviations, or conflicts, and negotiate with other nodes, if necessary, to resolve inter-instrument or instrument-spacecraft conflicts. Exception handling, at any node, is expected to be performed manually by an operator or automatically with user agents. Automation will be introduced gradually based on operator need and software maturity. Using exception history, user agents can be developed to handle exceptions that have occurred previously and are likely to recur. A unique user agent is defined for each exception. The initial system automatically handles only a few exceptions and contains only a few user agents. As the system matures, it is expected to handle more exceptions and to contain many user agents.

With user agents, the automation level can change dynamically depending on operator workload, level of expertise, and preference. When an exception occurs, the system automatically invokes the appropriate user

agent to handle the exception. However, operators still have final authority over decisions made. They can override the user agent operating at a node and direct the node to do something different than it would have chosen automatically. Also, if an exception occurs that the system cannot handle, operators become involved. Human operators may want or need to negotiate among themselves to resolve exceptions using the telephone, electronic mail, or other methods.

Final Scheduling

Final scheduling is the last step in the planning and scheduling process. The final schedule is an executable, exception-free, composite schedule of instrument and spacecraft operation, observation, maintenance, and calibration activities for a given time interval. Final scheduling is the process of incorporating the results of the exception handling process, and any changes that have occurred including late changes or targets of opportunity, in the initial schedule. Targets of opportunity are phenomena of interest that cannot be predicted, are often short-lived, or are changing rapidly. As throughout the scheduling process, final instrument scheduling is done at the SOC and final spacecraft subsystem scheduling is done at the MOC. The results of final scheduling are stored in the database where last minute inter-instrument and spacecraft-instrument conflicts can be detected and resolved as described above. Changes are permitted as long as there is ample time to handle them, they do not cause an exception, and they can be accommodated within the communications resources and services obtained from the NCC.

Commanding

The objective of commanding is to direct the spacecraft and instruments to perform scheduled or other required activities. Commanding involves generating, uplinking, storing, and executing command data. There

are three major levels of commanding: normal commanding, contingency commanding, and emergency commanding.

Normal commanding directs the spacecraft to perform scheduled spacecraft and instrument activities. Command data is stored in the database so that exceptions can be detected and resolved. When and how often command data is generated varies by mission. Command data is generated from scheduled activities. Each SOC is responsible for its own instrument command generation while the MOC is responsible for spacecraft subsystem command generation. The MOC is responsible for assembling the instrument and spacecraft command data and uplinking the composite command data set to the spacecraft during a communication link.

Spacecraft and instrument constraints are defined prior to launch and stored in the database. The MOC and SOCs validate all spacecraft and instrument command data before it is uplinked by the MOC. They also verify that command data was received onboard completely, correctly, and in sequence, and that command data was stored and executed properly. All onboard command data is verified by evaluating the appropriate return-link housekeeping and engineering parameters. The MOC and SOC maintain their respective command history archives.

Contingency commanding directs the spacecraft to perform contingency spacecraft and instrument activities, possibly due to late changes or targets of opportunity. Since most contingency activities are preplanned, the associated command data can be stored in the database. If no preplanned command data is available, the responsible node must generate the command data in sufficient time so as not to subject the mission to undue risk. When accepted, the schedule is updated, and a new command data set is generated and uplinked at the appropriate time.

Emergency commanding directs the spacecraft to perform spacecraft and instrument safing operations, generally in reaction to some potentially catastrophic event. Emergency commanding for the spacecraft subsystem is performed by the MOC. Emergency commanding for an instrument is performed by the SOC using the results of instrument monitoring. Whenever practical, emergency command data is preplanned and stored in the database for later use. If unavailable, the responsible node generates the command data. When initiated, emergency commands are validated and uplinked at the next available communication link. The responsible node monitors the return-link telemetry to verify the receipt and execution of emergency commands.

FUTURE WORK

We plan to prototype the concept described above, and plan to develop a representative subset of components: a planning and scheduling database at GSFC, a MOC at GSFC, and two SOC's—one at GSFC and one at the University of Colorado (CU). The command management portions of the concept will not be prototyped.

The planning and scheduling database and the CU SOC will be implemented on VAX workstations. The MOC and the GSFC SOC will be implemented on SUN 4 workstations. A commercial DBMS, SYBASE, will be used to implement the database server functionality with all nodes having SYBASE client functionality for distributed access.

The MOC and SOC at GSFC will use an enhanced Request Oriented Scheduling Engine (ROSE) scheduler. The SOC located at CU will use an enhanced Operations and Science Instrument Support Planning and Scheduling (OASIS-PS). ROSE and OASIS-PS are written in Ada and use the Transportable Applications Environment Plus (TAE+) (Century Computing, Inc., 1993) for the user interface.

REFERENCES

- Buford, Carolyn. (August 1992). *Scheduling Data Representation: Concept and Experience in Code 520*, (DSTL-92-010). Greenbelt, MD: NASA/Goddard Space Flight Center.
- University of Colorado at Boulder, Jet Propulsion Laboratory, Goddard Space Flight Center. (June 1993). *Earth Observing System Distributed Planning and Scheduling Prototype Lessons Learned Working Paper*.
- NASA/Goddard Space Flight Center. (April 23, 1993). *Future Mission Operations and Data Systems Directorate Architecture and Ground Segment Operations Concept*. Greenbelt, MD.
- Century Computing, Inc. (September 1993). *TAE+ User Interface Developer's Guide, Version 5.3*, Greenbelt, MD: NASA/Goddard Space Flight Center.
- Fahnestock, Dale. (November 18, 1993). *Renaissance, A New Approach to Ground Data Systems*. Presentation held at Greenbelt, MD.

GERMAN AEROSPACE RESEARCH ESTABLISHMENT (DLR)

Oberpfaffenhofen, D-82234 Weßling, Germany

A MISSION PLANNING CONCEPT AND MISSION PLANNING SYSTEM FOR FUTURE MANNED SPACE MISSIONS

Dr. Martin Wickler

DLR, GSOC, WT-RM-RS, Postfach 1116, D-82230 Weßling, Germany

Abstract

The international character of future manned space missions will compel the involvement of several international space agencies in mission planning tasks. Additionally, the community of users requires a higher degree of freedom for experiment planning. Both of these problems can be solved by a decentralized mission planning concept using the so-called "envelope method", by which resources are allocated to users by distributing resource profiles ("envelopes") which define resource availabilities at specified times. The users are essentially free to plan their activities independently of each other, provided that they stay within their envelopes.

The new developments were aimed at refining the existing vague envelope concept into a practical method for decentralized planning. Selected critical functions were exercised by planning an example, founded on experience acquired by the MSCC during the Spacelab missions D1 and D-2. The main activity regarding future mission planning tasks was to improve the existing MSCC mission planning system, using new techniques. An electronic interface was developed to collect all formalized user inputs more effectively, along with an "envelope generator" for generation and manipulation of the resource envelopes. The existing scheduler and its data base were successfully replaced by an artificial intelligence scheduler. This scheduler is not only capable of handling resource envelopes, but also uses a new technology based on neuronal networks. Therefore, it is very well suited to solve the future scheduling problems more efficiently.

This prototype mission planning system was used to gain new practical experience with

decentralized mission planning, using the envelope method. In future steps, software tools will be optimized, and all data management planning activities will be embedded into the scheduler.

Introduction

The proposed concept and system primarily addresses mission planning (of on-board operations) for payloads of future manned space missions. But they should be applicable to system planning and/or to unmanned missions as well. Most of the examples and expressions are taken from the world of Spacelab or Space Station (especially D-2 or APM), and most of the mission planning aspects are discussed from the MSCC point of view.

All payload mission planning activities of the First German Spacelab Mission D1 (30 October to 6 November 1985) and of the Second German Spacelab Mission D-2 (26 April to 6 May 1993) were performed by DLR at MSCC in the function of a (remote) POCC.

For D1 and D-2 a centralized mission planning concept was applied. That means that all payload relevant information and requirements were collected at MSCC, and each timeline version was generated at MSCC exclusively. The user community was involved in the timeline preparation (-data base creation or update-) but not in the timeline development itself. Up to the present, centralized mission planning concepts have normally been used for manned space missions. Many experiences gained during D-2¹, studies and ideas from NASA⁴, upcoming new requirements³, and some new (technical) capabilities were the drivers for a refined mission planning concept and a partially new system.

Mission Planning Tasks and Constraints

The mission planning activities include the generation of several versions of pre-mission timelines, the timeline replanning during a mission, and the preparation of an "As-Flown-Timeline" after a mission.

Mission planning in the context of this paper consists of developing the plan for all manned and unattended activities on board (*e.g. on board Spacelab*). The plan is written down in a document which specifies the times for performing the procedures necessary to conduct the attended experiments, and further documents, such as lists and plots of activity steps, resource profiles, and command timelines, which are produced as supplementary information needed by the control center. The plan from which all these documents are derived is called simply the "timeline".

In general, the mission planning consists mainly of three different tasks:

- Collecting and analyzing of information, availabilities and requirements
- Generation of the timeline
- Production of all necessary outputs and documentation.

The second task -timeline generation- is performed in three steps:

- Event generation (=orbit analysis, generation of an attitude timeline, computation of event on/off times)
- Experiment and/or system scheduling
- Data management (=generation of a data flow timeline)

This short description of a mission planning task flow is valid for all payload and system spacecraft operations.

The mission planning team has two main interfaces. On one side is the spacecraft (*e.g. the Shuttle including a spacelab*) with all its capabilities and availabilities together with the organisations (*such as NASA and ESA*) which offer this spacecraft and determine the operations concepts in a set of constraints and rules. On the other side are the investigators and their representative organizations (=the user community) with their requirements to perform experiments or other activities. The mission planning team attempts to fulfill the requirements

as well as possible, according to the availabilities and regulations.

New Requirements

User Requirements

In order to optimize the scientific return, the users need to do some basic mission planning functions outside the control center:

Instead of providing their inputs in the form of FO sheets (*-the generation and update of these FO sheets was very time consuming and was a major source of errors-*), the users should provide their experiment requirements and inputs in form of computer files which can be automatically processed. These files should be sent to the mission planning center electronically. Furthermore, the user community requires a certain flexibility for their own experiment planning. They require a certain degree of freedom to rearrange their experiment runs within given time slots by themselves, instead of being tied to an inflexibly fixed experiment schedule.

In addition to the gain of flexibility and autonomy, another aspect should be mentioned. Some "editing" (=data base entries and updates) and "micro-timelining" (=detailed step by step experiment configuration) tasks are shifted from the control centers to the experiment and procedure experts of the user community.

International Co-operation

Future manned space missions will require more international co-operation. These complex missions will generally require a certain decentralization of mission planning activities. (*E.g. ESA requires that all planning activities for the APM system and payload will be performed in Europe, and that different USOCs (in different countries) shall take over some basic mission planning tasks.*)

General Operations Aspects

The distribution of mission planning outputs and documentation should be performed electronically. This would reduce the reaction time to get a response from the investigator.

Future manned space missions will last longer than two weeks. The timelines must be developed, generated, and maintained in a shorter time frame than before. (*For D-2 [duration 10 days] the timeline generation process lasted up*

to three weeks, excluding data base preparation but including all documentation.) The Space Station operations concept requires a new timeline for every time increment, and requires the capability of handling mission planning activities for multiple increments simultaneously³. In contrast to centrally planned short missions, the upcoming long duration missions require that all detailed experiment knowledge (*necessary for mission planning*) is located exclusively in the user team, and not at the control center. The number of experiments of such a mission is too high, and/or the turn-around times of the payload (=the number of different experiment facilities) is too short to collect all the mission planning information in detail at a single point. Therefore an electronic interface is necessary, as well as a very fast and sophisticated scheduler.

Most of the software used for mission planning purposes during the D-2 project was placed at DLR's disposal by NASA Marshall Space Flight Center (MSFC). Most of these software tools have been in use for many years and they cannot fulfill the requirements of a modern, user friendly, and sophisticated mission planning concept.

A MSCC-specific problem was that all MSFC software was available as executables only. No software updates, modifications, or changes were possible. For a complex and flexible mission planning system, it is necessary that new or changed software requirements can be implemented as soon as possible. This requires a modular software concept, with all the software code be available at the control center or, -at a minimum- very responsive software maintenance.

The Concept

Compared to the D-2 mission, the upcoming multi-national space missions will have more exchange of information between the different space agencies on one side, and between the user community and the agencies on the other side. The flight crew will also need added flexibility in the planning and implementation of longer duration operations. Therefore, the era of Space Station payload operations requires a reassessment of traditional modes and methods of conducting payload operations⁴. However, a

(new) concept needs not only new methods, but also new hardware and software features, and new technologies.

The Concept for Decentralized Mission Planning

The Envelope Method is able to support all shades of mission operation concepts between a totally centrally organized and planned mission to a mission planned in a completely decentralized process. This proposed mission planning concept does not discuss different mission operations concepts, but proposes a feasible mission planning concept under known constraints.

To begin the discussion of a concept, especially the discussion of the Envelope Method, on a rational and practical level, some general assumptions should be presupposed:

- The concept shall support a reasonable and balanced usage of all available (spacecraft) resources.
- The concept shall lead to a higher degree of flexibility and autonomy for the user community (*compared to traditional (=centralized) methods*).
- The concept shall allow a flexible reaction on changing or modifying the spacecraft operations concepts.
- The concept shall permit a control center to implement all necessary planning, re-planning, and conflict-solving activities efficiently.
- The main rule of the "envelope game" is: Do not exceed any value of your assigned envelopes!

The Envelope Method

All aspects of a flexible and efficient decentralized mission planning concept can be covered by the so-called "Envelope Method". A decentralized mission planning concept enforces the Envelope Method (and vice versa). Therefore, decentralized mission planning with the envelope method is further abbreviated into "the envelope concept".

The resources which are shared by several users, can be distributed via resource envelopes. Resources include crew time, power, real-time data downlink, etc. A resource envelope is a time-dependent profile that defines the available amount of the resource at a specified time. An envelope should be a greater, contiguous block of a resource. Each user will get several envelopes, one for each resource. A user can plan his

activities within his resource envelopes independently from the other users. The block structure of the envelopes prevents an interlocking of the activities of different users. Envelopes are updated only by shifting, increasing, or decreasing the blocks, not by breaking them down into smaller blocks.

(Resource) envelopes are a very well suited means for information exchange between different levels of a hierarchical (mission planning) organisation structure (E.g.: POIC (at MSFC) <=> APM-CC (at MSCC) <=> Experimenter (at USOC)).

There are not only advantages to the Envelope Concept. The main disadvantage is that the efficiency of the resource usage decreases with the number of different envelopes, and decreases according to the size of the envelopes. The number of envelopes depends, on one hand, on the number of resources, on the other hand, on the number of "L3"-users (see figure 1). The efficiency of a decentrally planned timeline will never reach that of a centrally planned one². In other words, if all sharable resources (such as power, crew time, downlink and uplink etc.) are split up into several resource envelopes for the different users, it is impossible to fully exploit each resource and to fill up each unused gap of a resource. One can gain a high flexibility and autonomy of planning by using the envelopes, but one has to pay for this with a decreasing resource usage. (For more information see "Envelope Concept in detail".)

than expected, there will be less chance of impacting other experiments than in the case where the schedule is tightly packed.

Mission Planning with the Envelope Concept

Figure 1 describes the (Decentralized) Envelope Mission Planning Concept of a three level system by the Space Station-APM scenario from the MSCC point of view:

All users generate and update their mission planning inputs and deliver them in form of requirement profiles to the APM-CC. All inputs are then checked against operational constraints and integrated into the mission planning data base.

At first cut, the APM-CC develops a timeline according to the user requirements and the resource availabilities provided by level 1 (L1, overall mission management or e.g. the POIC) to each member of level 2 (L2, e.g. the APM-CC). (It is assumed that there will be different control centers which are responsible for different modules of the Space Station.) From this timeline, the resource envelopes for level 3 (L3, the users) are generated and transmitted to the users. The users plan their experiments/activities independently from each other within their assigned resource envelopes.*

The results are new or changed requirements (in form of an updated subtimeline or in form of

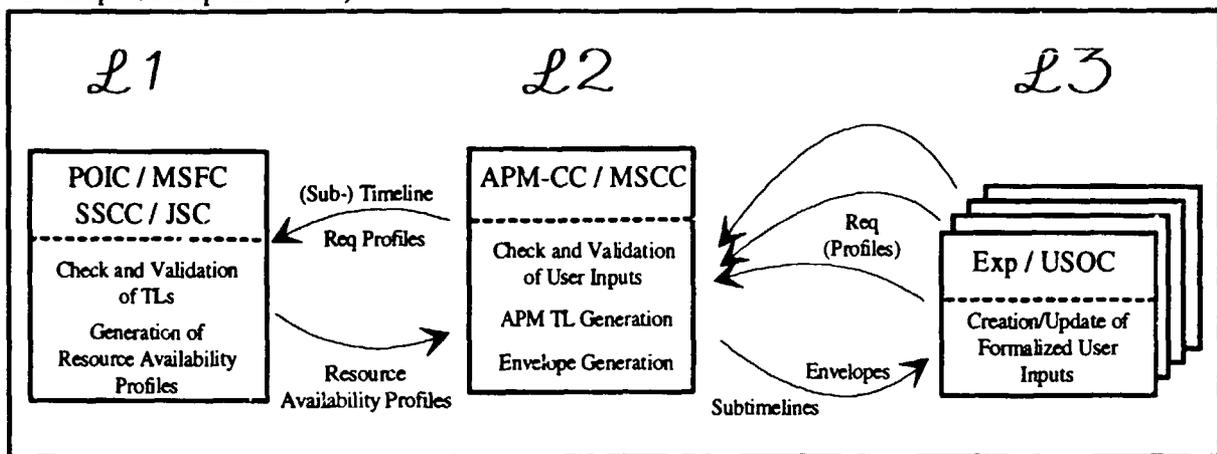


Figure 1 The MSCC Mission Planning Concept (an example of the APM scenario)

However, a loosely packed timeline is less sensitive to the problems which typically occur during a mission; if an experiment takes longer

updated requirements) which are returned to the APM-CC, where all subtimelines (or requirements) are merged into the master timeline (or data base). Each user is responsible

* Keep in mind that it is not allowed to the users to exceed any envelope value.

for updating his data base input and forwarding it to the APM-CC. For each version of the timeline, several iterations of this process with updated envelopes will be necessary to solve upcoming conflicts between different users.

The APM-CC maintains the mission planning data base, the master timeline, and the resource envelopes, and checks them against operational constraints and for conflicts. All output products are produced at the APM-CC. The co-ordination with the *L1* is performed by the APM-CC.

The above mentioned concept describes roughly the pre-mission planning scenario. It could also be used for the re-planning during a mission. However the (iteration) process has only one cycle and the user reaction time and input delivery must be fast enough to support the re-planning.

The Envelope Concept in detail

In general, several variations are possible for distributing resource envelopes:

- Envelopes for all sharable resources: All resources used by several users are distributed as envelopes.
- Envelopes for special sharable resources only: Only a few resources, which are heavily used, are distributed as envelopes. After each iteration, additional resources which turn out to be strongly in demand, and to cause conflicts, can be added to the envelope resources.
- Envelopes for special users only: Only users with activities which block out resources for a relatively long time (block usage) receive resource envelopes. The activities of the other users are planned at the APM-CC.

Having the above mentioned advantages and disadvantages in mind, the second option may be the most appropriate way to establish the envelope concept for mission planning purposes. An analysis (of *D-2*) revealed that most of the experiments could be satisfactorily scheduled by providing three resource envelopes to each experiment. These three main resource envelopes may differ from experiment type to experiment type, but they all are members of the overall set of resource envelopes (such as crew time, power, downlink and uplink capabilities, micro-g environment, and other mission dependent resources). Resources which are mandatory for a

successful experiment performance are such main resources. (E.g.: For an earth observation experiment the (three) main resources could be power, the earth target observation opportunities and the reprogramming opportunities. For a human physiology experiment the three main resources could be crew time, real-time down link and uplink capability.)

Studies ^{2, 4} demonstrated that with a decentralized envelope concept, nearly 80% of the activity time (compared with a centrally planned timeline) could be achieved. The efficiency may be a little bit higher if there are many more iteration steps of envelope updating.

It is not reasonable to distribute all sharable resources via envelopes. The resulting timeline would have an unacceptably low resource usage. If there are too many different envelopes available, not only the micro-timelining will become very difficult, but also the envelope generation (on the control center side) is very time consuming. However, distributing only a reasonable number of envelopes will unavoidably lead to some violations of operational constraints.

These assertions need a detailed discussion:

One idea of the Envelope Method is to shift the minor conflict solving concerning some heavily used resources from the control center to the users. But the user is able only to solve conflicts concerning his own experiments and concerning the distributed (*main*) resource envelopes. Because each resource envelope has the same priority for the user, and if all resources and constraints were distributed as envelopes, the user could get into trouble in the course of his internal experiment redesigning. Why? The competition (within a certain time frame) of some (independent) experiments for different resources forces the control center to create envelopes with variant shapes for each resource. (E.g.: An experiment requires for nearly one hour crew time and power. the resource envelopes for both resources may not be exactly the same.) If this phenomenon is extended to a great number of envelopes, it is possible that an experiment has a very spacious envelope for each single resource, but the intersection of all these resource envelopes forces this experiment into a completely fixed time frame!

It is possible to overcome this pressure of competition between different experiments by avoiding any parallel scheduling as long as possible. But in this case, the overall resource exploitation decreases to an unacceptable value.

The solution is to distribute only the heavily used resources via real envelopes, and to consider all other resources as free, the first approach. If any conflicts concerning these resources arise, the resulting conflict management will be done at the next higher level (e.g. L2).

In the above-mentioned example (*of the earth observation experiment and of the human physiology experiment*) both experiments have different main resource envelopes, but they could interfere by any other resource usage. The conflict detecting and solving, the rescheduling, and the generation/updating of the resource envelopes will be one of the principle tasks of a control center. (The conflict resolution between level L1 and L2 should be done in a similar manner, depending on the assigned responsibilities.)

The Concept for Distributed Mission Planning

The Envelope Concept requires a fast and uncomplicated, user friendly information exchange between the control center (*especially the MSCC for the APM control*) and the user community. Decentralized mission planning gives the user the flexibility and autonomy for his own experiment rearrangement. It gives the user the possibility to enter all his (*mission planning*) relevant data (real experiment requirements or secondary information such as experiment procedures etc.) into specific electronic data bases. Vice versa, the control center is able to electronically distribute all outputs and information to the user community.

The mission planning tasks are performed on dedicated mission planning computers. Any direct access from outside of the control center to these machines is denied, for safety reasons. Therefore, a practical electronic information exchange concept should be based on commonly available networks as the transportation vehicle and on commonly used PC's and software as the aid to enter or to read data. The recent advances in computer technology have made the concept of distributed mission planning feasible, because all the necessary hardware and software is powerful enough, and affordable for everybody,

and the network connections are no longer a problem.

The Mission Planning System

The following chapter gives an overview of all modifications and new developments necessary to fulfill the above mentioned requirements and concepts. The functions and a rough module design of the separate parts are presented, but no implementation or software details are mentioned.

The former D-2 mission planning software was mainly NASA-MSFC software. The whole system can be divided into four main software packages all needing DEC computers with VMS as the operating system. (*The four software packages correspond essentially to the above mentioned mission planning tasks: Event Generation System (EGS), Experiment Scheduling System (ESS), Data Management System (DMS) and an Interface and Output System consisting of different software modules which are necessary to receive information and to produce and forward the output plots, listings, and documents. See also figure 2*)

The Event Generation System (EGS)

The EGS is an autonomous system necessary to prepare event availability profiles for the ESS and DMS. The EGS is not affected by the new requirements, and is not involved in any new concept. Therefore no modifications or updates are mentioned here.

The Requirements Collection System (RCS)

The MSFC software does not support the distributed mission planning as described above. Therefore, a completely new software tool had to be developed. A first trade-off resulted in the decision to use as a basis a commercial relational data base with the possibility of defining graphical user interface applications. Another decision was to implement the RCS on a PC. After a market survey, a commercial relational data base was found to be the most suitable tool. The RCS is a very user-friendly tool, which allows the usage of two variant modes:

- The first mode allows the control center to design a mission dependent questionnaire.
- In the second mode, the user can enter all requirements.

The RCS offers the user window menus and mouse-sensitive fields to answer all questions; naturally it is very easy to change or update the parameters.

The implementation of the RCS could be done in three ways:

- The questionnaire and the resulting (requirements) data base can be distributed via floppy disc
- or via networks
- or the complete RCS is installed at the control center, and each user can login remotely.

These three options are not inevitably exclusive. Up to now, the first two options are possible.

The Experiment Scheduling System (ESS)

The ESS version used for D-2, especially the Experiment Scheduling Program (ESP), (and all later versions available up to now) is not able to support the decentralized mission planning with the envelope method. The main weak points of ESP are that it is not possible to receive, process (compute), or generate detailed profiles* or resource requirements, which are given as a percentage of the task duration. Additionally, the data base concept is problematic, because it is not user-friendly and its capacity is limited, the handling and the user interface are very uncomfortable, and the scheduling philosophy is too conservative to support scheduling according to the envelope method. (Scheduling according to the envelope method corresponds approximately to using fuzzy logic.)

A scheduling tool assessment identified the Science Planning Interactive Knowledge Environment (SPIKE) as the most suitable and fastest scheduling program¹.

(SPIKE is an Artificial Intelligence scheduler. It was originally designed and developed for scheduling Hubble Space Telescope operations. The development started in 1987, and SPIKE has been operational since 1990. The primary goal (of SPIKE) is to maximize scientific efficiency by optimizing the schedule and minimizing the violation of scheduling constraints. SPIKE has demonstrated its capabilities as a powerful and flexible scheduling framework with applicability to a wide variety of problems in different scientific satellite projects (e.g. EUVE, ASTRO-D).)

*A profile defines the available and/or requested amount of a resource as a step function of time.

(For detailed information about the SPIKE scheduler see ^{5,6}.)

ESP (and the corresponding data base) could not be exchanged easily with SPIKE. In a first step, SPIKE was modified to be used by unexperienced operators. *(The former user interface of SPIKE required a detailed knowledge of the programming language LISP.)* In a second step, SPIKE was imbedded into the remaining mission planning system. In a third step, SPIKE had to be modified to fulfill all operational aspects, especially with regard to the replanning capabilities¹, and an interface between the RCS and the ESS (mainly SPIKE) had to be established.

The Envelope Manipulation System (EMS)

Similar to the RCS, no EMS was available. The envelope manipulation task has several dependencies. It is influenced by the kind of mission and its payload, and by the mission operations concept as well as by the experiment requirements. Envelope manipulation is done in a separate task after the scheduling process.

Envelope manipulation in detail involves the shifting, increasing, decreasing, smoothing, and gap filling of a single resource profile. It also includes the balancing of resource profiles according to the overall (resource) availability. Therefore, the EMS needs a very comfortable graphical user interface, which allows the operator to flexibly imbed the balancing rules as external subroutines.

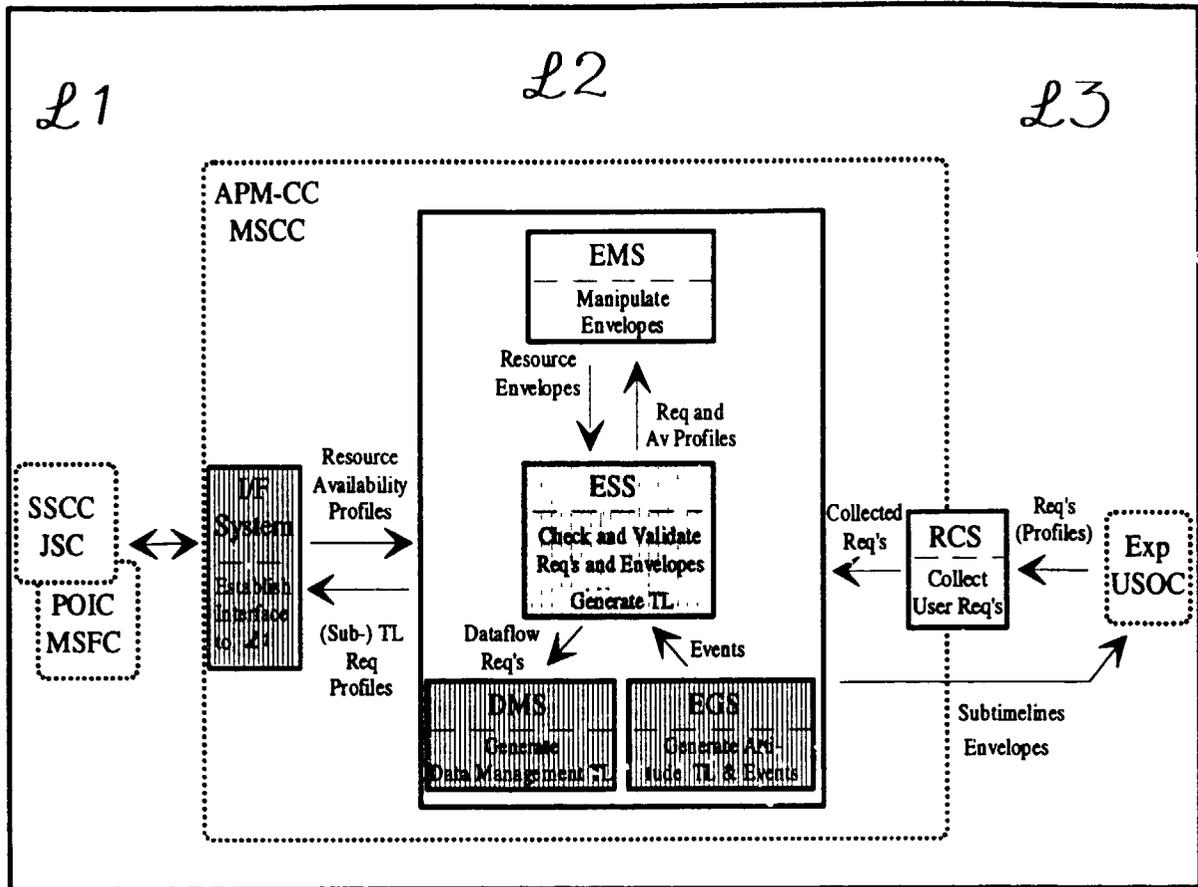
Because EMS and ESS interact together very frequently, it is advantageous to install them on the same hardware.

The EMS was developed with the aid of a commercial graphical user interface. The subroutines were developed in "C". Consequently, the EMS is now nearly independent of the hardware and the operating system.

The Data Management System (DMS)

The DMS as used for D-2 is still available. The DMS could not meet the D-2 requirements; they were performed by separate software (especially developed for D-2) or by timeline engineers and DMS operators.

For the moment no actions are completed concerning a new or changed DMS.



Legend:



Figure 2 The MSCC Mission Planning System (an example of the APM scenario)

The Interface and Output Modules

This tool has only the function of receiving and/or forwarding information (to the next higher level). This information is in detail mission dependent. The single modules are changed or updated by requests only. Therefore, a further discussion of these modules is not necessary in this paper.

Figure 2 summarizes the actual MSCC Mission Planning System and gives an impression of how all these different (sub-) systems act in combination and how they interact with externals. Following figure 1, the Mission Planning Concept and the information flow is reflected.

Results and Future Aspects

This mission planning concept and system could not be yet verified in a real mission, but the complete data base of D-2 is still available, and can be used for verifying and tuning the concept and system in detail. The RCS was tested in-house and distributed to some representative experimenters to get a feeling for the acceptance, and to get proposals for changes or improvements. The complete envelope scenario was simulated in-house with the ESS and EMS. The scheduling capabilities, the operator interface, and the performance of SPIKE satisfied almost all of the requirements.

Based on this experience, the existing MSCC mission planning prototype is able to handle the complete envelope concept with all its requirements and consequences.

To bring the mission planning prototype to a fully operational system some additional tasks remain to be done:

One main task is to design a new DMS. Two options are possible: either to develop a complete new and autonomous system, or to implement the missing functions into SPIKE.

The other main task concerns the interface and output modules. All outputs and interfaces are highly dependent on actual missions. Therefore, several output and interface modules have to be changed or to be developed in future.

(The interface for Shuttle missions already exists and will be adapted or upgraded if necessary. Interfaces to the ZUP for EUROMIR missions must be established. Finally all interfaces (*e.g. to MSFC and to JSC*) necessary for the operation of the APM must be specified and established.)

For further development of operational concepts, mainly concerning mission planning, some outcomes of D-2 and from the prototype testing should be taken into account. The timeline generation premission and the replanning during mission should be reorganized. A premission timeline should cover just the first one or two mission days. The following mission days (or crew shifts) will be planned in near real-time during the preceding day or shift. All necessary inputs for the planning must be available at the beginning of such a planning cycle, of course.

The main advantage of such a concept is that the science community is able to react very quickly to events. The science people are not forced to follow an obsolete preplanned timeline. Also, the overall premission timeline generation task could be easier. It is no longer necessary to create timelines for a whole mission (or great mission increments), only the overall resource budgeting must be managed. It is obvious that all experiment runs which are to be flown on the mission must verified, tested, and trained premission, but the time when they will be performed may be open.

Abbreviations:

APM	Attached Pressurized Module
DEC	DIGITAL Equipment Cooperation
DMS	Data Management System
CC	Control Center
EGS	Event Generation System
EMS	Envelope Manipulation System
ESP	Experiment Scheduling Program
FO	Functional Objectives
GSOC	German Space Operations Center
IBM	International Business Machines
IDL	Interactive Data Language
JSC	Johnston Space Center
MSCC	Manned Space Laboratories Control Center
MSFC	Marshall Space Flight Center
RCS	Requirements Collection System
SPIKE	Science Planning Interactive Knowledge Environment
SSCC	Space Station Control Center
TL	Timeline
ZUP	Operation Center for Russian Manned Space Flights

References:

- [1] "Mission Planning Study Document", E1-OPS-TN-009, M. Wickler (DLR), 1993
- [2] "A Test of the Envelope Method for the D-2 Mission with MIPS", D. Dietrich and M. Wickler (DLR), 1991
- [3] "Mission Planning Subsystem Specification"; COL-MSCC-DP-SP-005, A. Valentino (SSI), 1993
- [4] "Resource Envelope Concepts for Mission Planning", NASA TP 3139, K.Y. Ibrahim and J.D. Weiler (MSFC, USA), 1991
- [5] "SPIKE: Intelligent Scheduling of Hubble Space Telescope Observations", M.D. Johnston and G. Miller, in Intelligent Scheduling, ed. M. Fox and M. Zweben, (San Francisco: Morgan-Kaufmann), ISBN 1-55860-260-7, 1994
- [6] "An AI Scheduling Environment for the Hubble Space Telescope", J. Sponsler, M.D. Johnston, G. Miller, A. Krueger, M. Lucks and M. Giuliano, "Computing in Aerospace 8", (21-24 Oct 1991, Baltimore), American Institute of Aeronautics and Astronautics, pp. 14-24

Operations

1. Automation		Page 499
OP.1.a	Operations Automation <i>Charles Thomas Boreham</i>	501
OP.1.b	X-Analyst, A Generic Tool for Automatic Flight Data Analysis and Spacecraft Performance Assessment <i>Jean-Marc Brenot</i>	507
OP.1.c	Production and Quality Assurance Automation in the Goddard Space Flight Center Flight Dynamics Facility <i>K. B. Chapman, C. M. Cox, C. W. Thomas, O. O. Cuevas, R. M. Beckman</i>	509-516
OP.1.d	Graphic Server: A Real Time System for Displaying and Monitoring Telemetry Data of Several Satellites <i>Stéphane Douard</i>	517-523
OP.1.e *	Advanced Technologies in the ASI MLRO Towards a New Generation Laser Ranging System <i>Thomas Varghese, Giuseppe Bianco</i>	525-530
OP.1.f *	The Sequence of Events Generator: A Powerful Tool for Mission Operations <i>Hubertus Wobbe, Armin Braun</i>	531-537

* Presented in Poster Session

OPERATIONS AUTOMATION

Charles Thomas Boreham
 OAO Corporation
 787 West Woodbury Road
 Altadena, CA 91001

ABSTRACT

This is truly the era of "Faster-Better-Cheaper" at the National Aeronautics and Space Administration/Jet Propulsion Laboratory (NASA/JPL). To continue JPL's primary mission of building and operating interplanetary spacecraft, all possible avenues are being explored in the search for better value for each dollar spent. A significant cost factor in any mission is the amount of manpower required to receive, decode, decommutate, and distribute spacecraft engineering and experiment data. The replacement of the many mission-unique data systems with the single Advanced Multimission Operations System (AMMOS) has already allowed for some manpower reduction. Now, we find that further economies are made possible by drastically reducing the number of human interventions required to perform the setup, data safing, station handover, processed data loading, and tear down activities that are associated with each spacecraft tracking pass.

We have recently adapted three public domain tools to the AMMOS system which allow common elements to be scheduled and initialized without the normal human intervention. This is accomplished with a stored weekly event schedule. The manual entries and specialized scripts which had to be provided just prior to and during a pass are now triggered by the schedule to perform the functions unique to the upcoming pass.

This combination of public domain software and the AMMOS system has been run in parallel with the flight operation in an online testing phase for six months. With this methodology, a savings of 11

man-years per year is projected with no increase in data loss or project risk. There are even greater savings to be gained as we learn other uses for this configuration.

INTRODUCTION

The purpose of this paper is to explain what has been done to automate the operation of the Multimission Ground Data System (MGDS) at JPL. It is the further intent of this paper to explain some of the problems encountered during the systems' evolution that prevented this automation from occurring earlier.

OBJECTIVES

The implementation of JPL's automation of MGDS operations addressed seven objectives:

- [1] Automate the operation of telemetry processing for realtime operations thus eliminating all of the repeated tasks that the system controller would normally perform manually during a support period.
- [2] Accomplish the automation in a simple yet reliable fashion.
- [3] Maintain the ability for system controller intervention.
- [4] Provide automatic backup to MGDS systems in case of hardware or operating system failure.
- [5] Use a method independent of applications software and hardware platforms.
- [6] Eliminate labor-intensive operational work-arounds associated with unstable or incomplete applications software deliveries.

- [7] Give the system the flexibility to easily accommodate additional functions and/or projects.

PROBLEMS

Providers of realtime support are always interested in minimizing costs and maximizing reliability through the automation of operator tools. A series of obstacles have persisted, however, that have held back the automation process.

- [1] The cost of operations is higher than necessary because systems are frequently delivered strictly to meet budget and schedule constraints. Such a delivery is made with the absolute minimum capabilities that will meet project processing requirements, and no emphasis is placed on operability issues. Yet, the prime focus of operations groups is not that deliveries meet specific requirements. Rather, it is that deliveries produce the data products required by projects without extensive human intervention. So when deliveries are rushed in order to meet budget and schedule constraints, they lack operability and the operational costs are increased.

Further, the automation of a system is not an achievable goal if the hardware and software are not stable. Thus, the significant reduction in costs available from the automation of operations also hinges upon the operability of delivered systems.

- [2] For a variety of reasons, there is strong pressure to adopt a Graphical User Interface (GUI) strategy for all levels of applications. This type of interface is beneficial for occasional users of the system, but not for operations personnel who maintain and run the system around the clock and who understand the system's full capabilities. Operations personnel need to be able to act quickly at all levels of each application and its operating system. From an operational perspective, a punch and

click type of interface is intrusive, limiting, and cumbersome and is thus an obstacle to any type of automation that would lower operational costs.

- [3] The concept of running a system from a Sequence of Events (SOE) file with little or no human intervention is not a new one. But the implementation of this concept, too, has had its associated problems. One of these is the high frequency of changes that are applied to any given weekly SOE. Historically, these changes have had to be applied manually, forcing frequent operator intervention.

Thus, the question to be answered became: *Could the operation of the MGDS system be automated to the degree that we desired using available software and with the system design that was already online?* With a little creativity and a thorough understanding of the operational functions, this goal turned out to be achievable.

The UNIX utilities that are being applied in JPL's automation are straight forward and available to all users. The third party software programs are available on the network and once again can be accessed and used by anyone. Not only did we accomplish the objectives set out earlier; the implementation of these automated operations features resulted in an operational staffing reduction from 28 to 17 for the same data delivery workload. On an annual basis this saves JPL approximately 1.2 million dollars in operational costs.

THE ROLE OF UNIX

When considering the automation of realtime operations, we frequently tend to see large complicated software programs that cost as much as the current operators who run the systems. With the operating systems that were previously in use, this assessment would have been accurate. But with the adoption of UNIX as the operating system and with the tools and utilities that then become available, the cost of automation is within the reach of all groups.

JPL commenced its transition to the UNIX Operating System in 1986. The first version of the flight applications that ran in the UNIX environment was V7, which supported the Magellan mission, but required extensive operational work-arounds. V7 had to be monitored continuously by the realtime operations group to ensure the delivery of usable data to the project. As previously described, the stability of the realtime applications software is a key factor in successfully automating operational tasks. In our case, this needed stability was achieved in December, 1993, with the delivery of the nineteenth major version of the application software. This delivery allowed our operations staff to take advantage of the tools, utilities, and public domain software packages that are available for UNIX.

Using off-the-shelf and public domain software with a small amount of custom coding, we were not only able to achieve a high degree of autonomous operation but also to build an inexpensive, software-switched, fault-tolerant system. We never lose data due to a host system failure. This general approach can be applied to a broad variety of high reliability applications at a fraction of the cost of the special purpose fault-tolerant computing systems on the commercial market. Moreover, this solution is vendor platform independent, requiring only a UNIX operating system environment.

THE ROAD TO AUTOMATION

The reliability of delivered applications paved the way for automated control. The operations task for flight projects is repetitive and can therefore be scripted to run on a schedule. This was done on our systems by combining a seven day SOE, custom software to convert the schedule to applications directives, public domain software, and UNIX utilities.

The integration of COTS and public domain software into realtime mission-critical systems is a viable and cost-effective alternative to custom designed and developed code. The automated operational

capability described in this paper was conceived and integrated in a two month period by selected individuals in the operations group as time permitted. Parallel testing took an additional six months. *Under the automated configuration, more spacecraft data arrived at the projects' databases than under the manual system!*

WHAT IS AUTOMATED?

We maintain at least 32 applications and 10 monitoring processes on 35 remotely accessed systems. Prior to automating operations this same configuration was maintained manually. In the following paragraphs we describe details of what is currently automated in our implementation. In addition, we describe some of the specific components that we used.

At the heart of the configuration is the seven-day SOE. From this, all associated jobs are derived and submitted to the system for the full week for all monitored spacecraft. Such job schedules are disk based in UNIX and therefore remain scheduled even when the host system is brought back from a failure. This means that all scheduled jobs will still execute when the system is brought back to online status. Jobs that did not execute when the host was down have to be entered manually but all jobs are scripted and well-ordered, and can thus be resubmitted to the system easily.

screen

In the UNIX world, software follows a standard input/output protocol that previously created a major problem for application and system failure recovery. If a host system failed, the applications that were being run from that host by remote login also failed. This difficulty was resolved by utilizing screen, a public domain software program written by Oliver Laumann of the Technical University of Berlin. Here, predefined scripts start screen prior to starting the applications. Standard input and output are buffered by screen on the X terminals

harboring remote logins, so that when the host system fails, the applications continue to run. A mechanism for reattaching to the application is also provided by `screen` so that operations can be normalized once the host is back on line. Now, when the host system goes down, there is no data loss during the host's down period. All remote systems continue the processing and loading of project data during a host failure.

System Utilities

To recover from failures of the host system we have used a number of UNIX capabilities: First we have the failed host automatically reboot itself. Next, we provide that X Windows accesses a customized initialization when the host comes up. The initialization file creates all appropriate windows and remote logins that were being used prior to the failure. The host then accesses a script that reattaches its windows to the proper processes (using `screen`) which have remained unaffected despite the failure of the host. Again, downstream users of the data will not have been affected by the failure of the host system.

monitor

The system is also protected against a total hardware failure of the host system. The operations group has built a program that runs on the backup system and monitors the prime host. If a failure of the prime host occurs, a five minute timer is set on the monitoring system, and a popup window notifies the controller immediately that the prime host has failed. The controller can respond to the popup window informing the backup system to promptly usurp the duties of the prime host. If, on the other hand, the popup is not responded to within the five minutes, the backup system automatically executes the X Windows initialization file and reattaches

to the appropriate processes using `screen`. The backup host thus assumes all control and processing for the prime host. Once again, the downstream user of the data is not affected.

The problem of applications failures on the remote systems is handled by additional monitors. If an application or its remote system fails, a popup window notifies the controller so that the hardware can be substituted or the software problem can be properly handled. The popup window is activated frequently and has a very annoying beep that cannot be ignored.

expect

We use a public domain package called `expect` written by Don Libes of the National Institute of Standards and Technology. This utility is set up to acquire and update a copy of the seven-day schedule. The output of `expect` (the seven-day schedule) is piped through another piece of custom software written by the operations group. That output is a file of the scripting schedule that is to be submitted to the system. Scripts are scheduled using the UNIX utility, `at`. With `expect`, updates are made to the schedule automatically without the need for human intervention.

force

The third public domain package used is `force`, which was written by Jeff Glass of the MITRE Corporation. We use this essential utility to place the applications-level commands on their associated windows. The applications commands and their `force` directives reside in the predefined scripts that are submitted by `at` to execute at specific times according to the seven-day schedule. We have also modified the `xterm` program and updated the UNIX device directory to incorporate the use of special ttys. We dedicated these ttys to each of the X terminals being used so that the

same tty names always assign to the same windows. This was important because, while the command that force sends is guaranteed by TCP/IP to reach its destination, force knows nothing of the context of that destination. Scripts can now consistently force commands to given windows with confidence that the assumed context is valid.

CONCLUSION

The automation of the operation of our system has been accomplished with some very simple concepts and tools, using scripts, minor amounts of C programming, and public domain software. There was no significant expense involved, and the outcome has been the dramatic reduction of 11 man-years per year in the cost of operations.

**X-ANALYST, A GENERIC TOOL FOR AUTOMATIC FLIGHT DATA
ANALYSIS AND SPACECRAFT PERFORMANCE ASSESSMENT**

Jean-Marc Brenot
Matra Marconi Space

Paper Not Available

Production and Quality Assurance Automation in the Goddard Space Flight Center Flight Dynamics Facility*

K. B. Chapman, C. M. Cox, and C. W. Thomas
Computer Sciences Corporation (CSC)
Lanham-Seabrook, Maryland, USA

O. O. Cuevas and R. M. Beckman
Goddard Space Flight Center (GSFC)
Greenbelt, Maryland, USA

ABSTRACT

The Flight Dynamics Facility (FDF) at the National Aeronautics and Space Administration (NASA) Goddard Space Flight Center (GSFC) generates numerous products for NASA-supported spacecraft, including the Tracking and Data Relay Satellites (TDRSs), the Hubble Space Telescope (HST), the Extreme Ultraviolet Explorer (EUVE), and the Space Shuttle. These products include orbit determination data, acquisition data, event scheduling data, and attitude data. In most cases, product generation involves repetitive execution of many programs. The increasing number of missions supported by the FDF has necessitated the use of automated systems to schedule, execute, and quality assure these products. This automation allows the delivery of accurate products in a timely and cost-efficient manner. To be effective, these systems must automate as many repetitive operations as possible and must be flexible enough to meet changing support requirements.

The FDF Orbit Determination Task (ODT) has implemented several systems that automate product generation and quality assurance (QA). These systems include the Orbit Production Automation System (OPAS), the New Enhanced Operations Log (NEOLOG), and the Quality Assurance Automation Software (QA Tool) (Chapman et al., 1993, Chapman et al., 1994) Implementation of

these systems has resulted in a significant reduction in required manpower, elimination of shift work and most weekend support, and improved support quality, while incurring minimal development cost.

This paper will present an overview of the concepts used and experiences gained from the implementation of these automation systems.

INTRODUCTION

As part of the FDF, the ODT is responsible for processing tracking data, performing orbit determination, and generating state vectors, ephemeris data, and station contact scheduling products. The ODT makes use of the FDF's two IBM 9121/490 mainframe computers to generate its products. The jobs necessary to generate the products must be set up and executed according to schedules specified by agreements between each mission and the FDF. Jobs are executed either in batch mode using Job Control Language (JCL) or in the foreground. Products are generated daily and must be quality assured and delivered to the appropriate users. These products are used by other groups in the FDF and by outside users for generating acquisition data, spacecraft onboard computer ephemerides, and flight operations and science mission support schedules. The products are necessary for the acquisition of spacecraft by tracking sites, prediction of tracking schedules and

* This work was supported by the National Aeronautics and Space Administration (NASA)/Goddard Space Flight Center (GSFC), Greenbelt, Maryland, under Contract NAS 5-31500.

spacecraft events, and generation of spacecraft computer uploads used in navigation. Errors in the products could result in lost support and science data, missed tracking, or the loss of the spacecraft. Thus, these products and data are extremely important in the day-to-day operations and safety of the supported spacecraft. The standard support provided by the ODT in the GSFC FDF is illustrated in Figure 1.

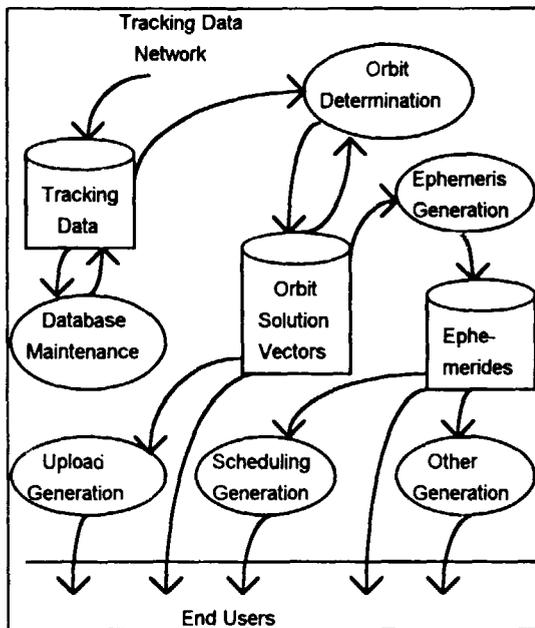


Figure-1: Orbit Determination Task Standard Support

In addition to the standard support, the ODT also performs analysis on the data and products that are generated. The analysis is performed to trend and update QA parameters, to aid in maneuver planning, and to monitor the orbital evolution of the mission. Analysis parameters include the spacecraft's semimajor axis, tracking data statistical information, and the derived coefficients used in the orbit solution. Previously, this type of analysis involved manually transcribing values obtained from job output into required reports.

In the past, the generation, QA, and delivery of products were labor intensive. Users manually edited JCL, changing up to 33 different parameters per job before submitting the JCL. The resulting output and printouts, most containing thousands of

lines of output, were hand-checked by the users to perform QA using an average of 60 to 70 parameters per product. Deliveries were performed by relying on a user's knowledge of what product went to what user. Previous to any automation, daily product generation required two to three personnel for 4 to 6 hours a day. The QA process required three to four staff personnel for up to 4 hours per day, and product delivery took two people 2 hours. Thus, the combined production, QA, and delivery processes resulted in up to 38 staff hours per day for nominal support. Not only was this process costly, but, because of the amount of time it took to generate a completed product, delivery schedules were being impacted. Also, the number of required products continued to grow as new missions, often requiring more complex support, were added (see Figure-2).

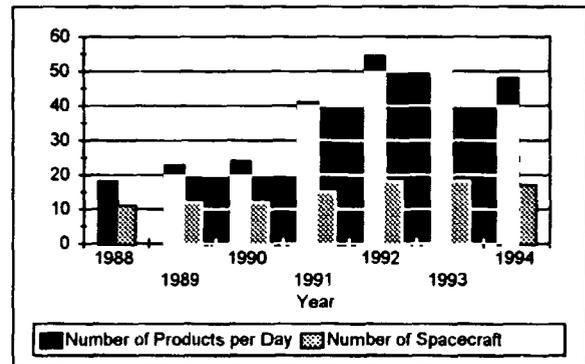


Figure-2: Orbit Determination Task Workload

In order to reduce costs, improve quality, and increase productivity, these manual processes were automated. This paper describes the ODT's product generation processes that required automation, discusses the automation tools generated, summarizes some lessons learned, and presents results and conclusions.

PRODUCTION PROCESS

Analysis of the ODT production cycle defined five product generation processes: scheduling, generation, QA, delivery, and tracking (see Figure-3). Because every ODT product passes through these steps, the emphasis was placed on the definition and execution of the processes for the entire

workload, not just on a product-by-product basis. For example, if there are 50 products in the day's worklist, to schedule, generate, QA, and deliver each product one by one would be costly. Since each process is necessary for the completion of an ODT product, these processes were targeted for automation as a means of reducing the cost of support.

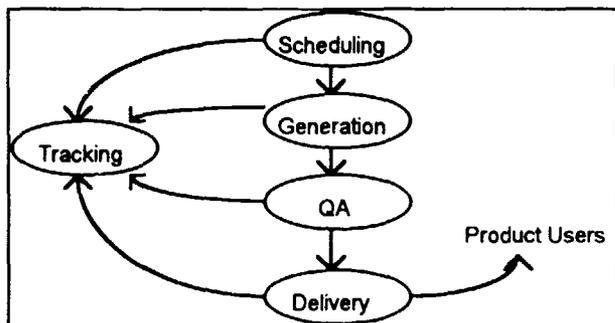


Figure-3: Product Generation Processes

The five product generation processes are described in the following subsections.

Product Scheduling

Products are generated according to support schedules determined by mission requirements and customer needs. This process is complicated because the missions have different delivery and support requirements for their products. These are specified in the Interface Control Documents (ICDs) and mission support documentation (for example, GSFC Flight Dynamics Division, 1991), and are determined through extensive analysis of the mission accuracy requirements. The current support includes 93 different product generation runs with varying schedules. Requiring users to remember an involved product schedule increases the risk of incorrect support. This process needs to be flexible enough to accommodate combinations of every possible product schedule (see Table-1). Also, the scheduling is subject to change depending on the status of the spacecraft or the requirements of the customer receiving the product. Scheduling also pertains to the various delivery methods employed after a product was generated. If a product is scheduled for generation, it may also

need to be scheduled for the various available deliveries.

Table-1: Example of Schedule Variance

Product	Schedule
EUVE Orbit Solution and Ephemeris	Every Day
HST Orbit Solution and Ephemeris	Every Other Day
UARS 8 Week Ephemeris	Every Thursday
IMP-8 Long Ephemeris	First Friday of Month

Product Generation

Product generation involves submitting the correct software with the correct input to create the end product. The products are generated by a variety of software, such as the Goddard Trajectory Determination System (GTDS) (Bleich, 1994), which is the primary orbit determination and product generation package for the ODT. Missions might have different requirements for similar products. For example, two missions may require TDRS ephemerides with different timespans. In addition, special support is sometimes necessary for product generation, such as following spacecraft maneuvers.

Setting up the product runs involves calculating and inserting proper timespans, orbital elements, force modeling, and other input into the run stream, and submitting it to the system. In many cases, input is required in different locations and formats. For example, a GTDS execution to perform an orbit determination solution, generate an ephemeris, and perform a comparison might need at least three different timespans as input.

Product Quality Assurance

QA is performed to ensure that products are free from anomalies resulting from incorrect input data, corrupt tracking data, environmental events (e.g., solar activity), human error, or spacecraft anomalies. All products are quality assured twice. During initial product generation, ODT personnel perform a preliminary QA on all products by reviewing basic parameters. Then a second group

of ODT personnel perform a detailed QA on the product. Up to 110 parameters from each product are checked against predetermined quality tolerances. Items checked include product data quality (i.e., tracking data statistics, computed or estimated values) and product data consistency (i.e., timespans, correct file names). These data are often spread throughout the output. A subset of the data items used in the QA is recorded in a permanent log to serve as a record and for analysis and trending. The tolerances used are derived from mission requirements, software specifications, and analysis. If a product fails QA, ODT personnel decide if the product should be regenerated with modified input or if the tolerances should be overridden and the product passed for delivery.

Product Delivery

Product deliveries occur in several different ways, and the workload for each delivery type is decided by the products generated and the schedule of deliveries. The delivery of the products consists of copying generated products to operational data files (promotion) and updating a delivery log to inform internal elements that products are ready for their use. It also involves transmitting or delivering products to external sources, such as Payload Operational Control Centers (POCCs) or science centers. Many of the external elements use different methods to receive their products. Transmissions take place over teletype, through Ethernet, or via the NASA Communications network (Nascom). Data may also be received as hardcopy or on a 9-track tape. Deliveries have to be carefully coordinated with each site to ensure that the proper product is delivered in the proper fashion.

Product and Event Tracking

Product and event tracking is a process that occurs throughout the entire production cycle, to satisfy the requirement to maintain a record of activities performed by both the system and the users. Such records should maintain a running account of the jobs that have been run, the products that have been generated and delivered, any anomalies that might have occurred, special requests, and shift

turnover. This process is also used to maintain key statistics and QA parameters for future analysis.

In the past, these logs were kept as handwritten or typed manual logs in many groups of the FDF. Problems with the old paper system included missing and illegible entries and the need to consult multiple logs to gain information. Also, with a paper log, only one person could efficiently read and write to it at a time, and that person must be at the same physical location as the log.

AUTOMATION OF THE PRODUCTION PROCESSES

ODT product generation activities were automated by developing several system utilities, which were created as another layer over the existing systems in use (see Figure-4). This was done because the institutional product generation software already existed, and it would have been too expensive to modify it. The systems need to handle the wide range of different product generation programs, and should be able to accommodate new programs without modification. Creating the automation separately was a cost-effective means of implementing improvements as soon as the pieces were ready. Because the generation programs execute primarily in batch mode with JCL, the automation systems deal primarily with configuring the JCL and input data to properly generate and deliver products. A menu system ties the automation systems together under a single user interface (UI).

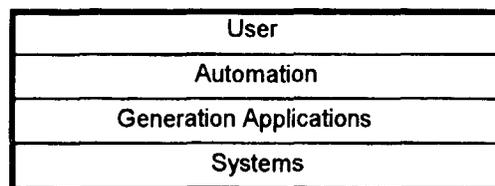


Figure-4: Relationship of Automation Layer to Applications and System

To handle ODT support variability (support schedules, timespans, satellite names, etc.), input configuration files were used to avoid the need for major system updates. Hardcoded parameters were avoided so a change in support would not

necessitate a change in the components of the automation system as well.

The automation also had to accommodate nonstandard or anomalous support. While the ultimate automation would be a total "hands-off" system, there are cases where control of the process should be returned to the user. In the ODT's case, the capability for manual intervention at key points in a process was all that was necessary. Requirements for this capability were a function of the type of support, the environment, the expected frequency of nonstandard support, and the potential impact if operations were delayed.

With the large number of jobs submitted on a regular basis, the users and system needed a means of determining whether processes have been completed. This information is required for system error detection and correction and process logging. Process status information was also useful for notifying and executing subsequent processes. Process status traceability was accomplished through log files and status file updates.

The ODT first developed the OPAS to automate the scheduling and product generation processes. Next, the delivery process was automated with the Delivery Tool. UI improvements were then made by implementing panel-driven menus and then developing the QA Tool. Each implementation resulted in further reduction in the time needed to complete a product (see Figure-5).

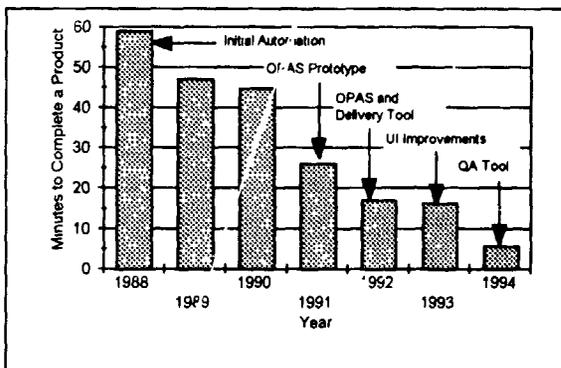


Figure-5: Implementation Dates and Effects on Average Completion Time

All of the automation utilities were developed with significant user input, especially with regard to UIs. Because of the close ties between the users and developers, the system closely reflected the user's needs.

The automation utilities for ODT product scheduling and generation, delivery, QA, and tracking are described in the following subsections.

Product Scheduling and Generation—OPAS

OPAS automates the scheduling and product generation. An original attempt at automation was implemented, refined as a newer prototype system, and then implemented as the final system now in place. OPAS makes use of a master requirements file to describe when a job is to be run, provide the updates needed for the runstream execution, and control the delivery processes. When OPAS is executed, its scheduler function creates a status file containing the list of the day's work and the status of its completion (see Figure-6). The status file becomes the link to the other sections of the automation. The OPAS generation function then sets up the jobs specified in the status file in accordance with the information in the requirements file, including date and timespan calculation. The user has the option to edit the completed runstream before execution, to aid in nonstandard support. Frequently, subsequent product runstreams may require input used from a previous setup. To support this, OPAS uses a current data file to store input needed for several jobs, which reduces the amount of user input required. Input that may be required from the previous day is stored in an a priori file. As the jobs are set up and submitted, OPAS updates its status file to indicate that the step has been completed for that product. The updated status file then serves as the notification to subsequent processes that a product is ready for the next step, such as QA or delivery. Also, because manual user setup is still available, anomalies can be easily worked around without the services of the maintenance personnel.

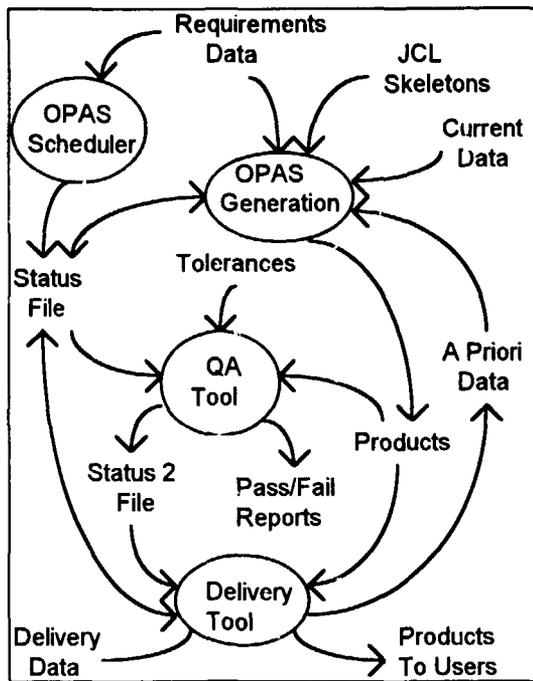


Figure-6: OPAS Functional Data Flow

Product Delivery—OPAS Delivery Tool

The ODT implemented the Delivery Tool function of OPAS to help automate the delivery processes. When executed, the Delivery Tool checks the OPAS status file for the list of the day's work for the type of delivery selected by the user (see Figure-6). It also checks the status file to see if the prerequisite steps have all been completed. The user can then instruct the system to deliver all of the products for that type or individual products. The Delivery Tool also updates the status file to indicate that delivery processes have been performed to maintain accountability. All of the UIs for the Delivery Tool functions operate in the same way where possible and allow for delivery of products that may have been generated but were not in the schedule. Information that aids in the delivery of products, such as file names and product destinations, is stored in delivery data files that are input to the Delivery Tool. The files can be easily modified to fit support requirements.

Product Quality Assurance—QA Tool

Automation of QA required that the data items to be checked be extracted from the output of the

product generation phase, checked, and reported. Because a variety of software is used to generate the products, the system could not be coded for the output of any single product. It had to be flexible and generic, with the specified data items and their locations user specified. The tolerances and the operations (i.e., =, <>, <, >, etc.) required in the process also had to be user specified.

The QA Tool is currently implemented as a prototype. The software runs instream with the product generation at the end of the batch run. It extracts user-specified data items from the product output and checks the values against user-specified tolerances (see Figure-6). Depending on the results of the tolerance checking, a flag for each data item is set to pass or fail. Reports are generated to inform the user of the results, and these take the place of the manual logging of data items for recordkeeping and analysis. More data are now available for analysis and recordkeeping. A UI allows the user to quickly ascertain the results of a particular product generation or of the entire day's work. The UI makes use of the OPAS status file, creating an updated version that indicates the pass/fail status of each product. Changes to the production software necessitates, at most, a configuration file change in the QA Tool, not a software update. *Because the user specifies in a single central location the desired data items, their locations, and the tolerances to use, the output from any existing or new software can be checked.*

Product Tracking—NEOLOG

NEOLOG is an online database implementation of the activities log that complements the accountability and tracking provided by OPAS. It allows entries to be made under several different categories and allows entries to be made from runstreams automatically or from interactive sessions with a user. Any user can access the log from any terminal, and multiple users can access the log simultaneously. All production and delivery runs in the ODT write information into the log, as do the analysts performing the work. The end result is a long-term running record of activities

and job executions that can be used for troubleshooting, analysis, and activities tracking. Typically, a log file contains up to a year's worth of entries, and previous years are easily accessible.

LESSONS LEARNED

Significant lessons have been learned from the use and implementation of product generation automation in the GSFC FDF. A key concept is the importance of analyzing the procedures involved in a process to identify repetitive and redundant user actions. Sometimes gains in efficiency are realized through simple procedural changes. Reducing and simplifying procedures also has the benefit of reducing the size of the automation. Other key lessons involve the areas of UIs, reliability, training, and requirements definition.

User Interfaces

The use of UIs to control the system requires special consideration. It is important to keep the interfaces as consistent as possible so that similar functions require similar user actions. Also key is keeping UIs logically organized and easy to use and understand; the urge to create overdone UIs should be firmly resisted. This significantly speeds user familiarization, makes the process more efficient, and reduces the chances for erroneous input. Also, UIs for individual utilities in the automation should be configurable or have the capability to be bypassed. This offers a high degree of flexibility in combining processes and eliminating the need for user input.

Reliability

Reliability is characterized by system robustness, accuracy, and ease of maintenance. The best method for achieving reliability is to keep the system simple. Thorough testing prior to implementation should be conducted to ensure robustness and accuracy. All of the systems implemented by the ODT went through thorough independent testing. By making control and data parameters configurable, maintenance is limited to file and parameter updates. Sufficient configuration management should be in place to ensure that

configurations are correct, changes are traceable, and quality controls are enforced. However, *the configuration management must not stifle quick and effective responses to problems.* In the ODT, configuration management of the automation systems is handled by personnel who also participate in the generation of products. Use of the system results in a familiarity that enhances the quick responses for changing requirements. The amount of software maintenance has been reduced significantly by the fact that most changes are now simple configuration file updates instead of coding changes. To avoid any impact that might arise on "off" days due to flawed maintenance, updates are discouraged on Fridays or any day before a holiday.

Training

For the ODT, training issues can be broken into two categories: system training and product familiarity. System training for an automation system is the same as with any other system. The users must be trained in the availability and use of the automation system's capabilities. Again, keeping the functionality of utilities and user interfaces consistent can reduce the time it takes to train users. In the case of automation, the usual resistance and mistrust of a new system by users may be heightened by the fact that many processes now occur out of view. Training and testing help, but if the system is designed to allow manual intervention as a backup, some of the resistance can be alleviated.

As processes and QA become more automated, the user becomes less involved in creating the product. This may result in reduced familiarity with the products and the generation software being used. In the FDF, this is a concern because the support for maneuvers and missions still involves a lot of manual work and analysis, requiring an in-depth knowledge of the products and support software.

Reducing automation to keep users familiar with the software and products is essentially the same as subsidizing the training budget through increased production costs. It is preferable to address the

issue with ongoing training, instead of reducing the amount of automation for production. Graphic feedback from the system may also help, as long as it does not unnecessarily add to the completion time for a product. This means that training costs and issues must be specifically addressed as efficiency is gained through automation. In the case of the ODT, familiarity with the products is maintained through analysis and special requests, as well as other training exercises. In fact, the automation is now freeing up time to perform more analysis, which improves the quality of support.

Requirements Definition

When drafting requirements for new product generation software, special consideration should be given to defining the parts of the output that truly define the quality of the product. While all of the output may be required as a product or for detailed analysis, usually smaller portions (that may be scattered throughout the output) are needed as a "quick look" to indicate the quality of a product. This information could then be provided as a condensed report that is easier to check and incorporate into other utilities. This requires that attention be paid to the potential uses and users of a particular system early in its development.

RESULTS AND CONCLUSIONS

After implementing the automation software, the ODT found that to create an effective automation system, attention must be paid to the reliability of the automation, to the training required to execute and maintain the system, to product familiarity, and to the design of software maintenance releases of product generating systems. By implementing the automation system, ODT personnel were able to make their product generation and QA more efficient (see Figure-7). Product generation time was reduced to 2 staff hours a day. QA time was reduced from an average of 12 staff hours a day to 1 to 2 staff hours, and delivery was reduced to 1 staff hour. Implementation of the automation systems allowed the FDF to provide operational phase orbit determination and navigation support more effectively for more missions, without having

to significantly increase staff or make expensive changes to product generation systems.

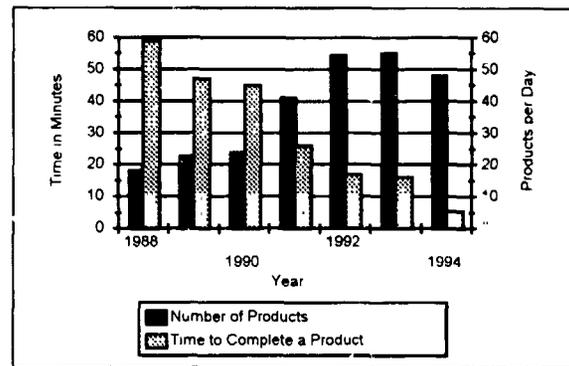


Figure-7: Workload Versus Average Product Completion Time

ACKNOWLEDGMENT

The authors acknowledge Mark Schmitt of Computer Sciences Corporation for his significant contributions to the early development of OPAS

REFERENCES

- Chapman, K., Thomas, C., and Schmitt, M (September 1993). *Orbit Production Automation System Product Generator (OPASPG) System Description and User's Guide* (Report 553-FDD-92/017R0UD0). Goddard Space Flight Center, Flight Dynamics Division, Greenbelt, MD.
- Chapman, K and Thomas, C. (March 1994). *Quality Assurance Automation Software (QA Tool) Software Requirements Specification* (Report 553-FDD-93/097R0UD0). Goddard Space Flight Center, Flight Dynamics Division, Greenbelt, MD.
- Goddard Space Flight Center, Flight Dynamics Division (October 1991). *Flight Dynamics Division (FDD) Generic Data Product Formats Interface Control Document* (Report 553-FDD-91/028). Goddard Space Flight Center, Flight Dynamics Division, Greenbelt, MD.
- Bleich, A. (February 1994). *Goddard Trajectory Determination System (GTDS) User's Guide, Revision 2, Update 7* (Report CSC/SD-85/6738R2UD7). Computer Sciences Corporation, Lanham-Seabrook, Maryland.

GRAPHIC SERVER A REAL TIME SYSTEM FOR DISPLAYING AND MONITORING TELEMETRY DATA OF SEVERAL SATELLITES

Stéphane DOUARD
CNES French Space Agency

ABSTRACT - Known as a Graphic Server, the system presented in this paper was designed for the control ground segment of the *Telecom 2* satellites. It is a tool used to dynamically display telemetry data within graphic pages, also known as views. The views are created off-line through various utilities then, on the operator's request, displayed and animated in real time as data is received. The system was designed as an independent component, and is installed in different Telecom 2 operational control centers. It enables operators to monitor changes in the platform and satellite payloads in real time. It has been in operation since December 1991.

GENERAL PRESENTATION

The Graphic Server system is a system for displaying and monitoring telemetry data of several satellites. It is based on the dynamic visualization of information on what are known as *graphic pages* (or *views*).

Logged in to a data server with which it can interact, it receives telemetry parameters in real time, interprets them and refreshes the graphic pages that the operator is currently displaying by inserting the new values. The operator therefore has access to images or views that reflect in real time the state of the satellites.

Graphic pages are made up of a background part and animated objects whose value, representation and colour vary in relation to the telemetry parameters with which they are associated. A relay, for example (an animated object) in an electrical circuit diagram (graphic page) will appear open or closed depending on the value of the corresponding telemetry parameter, and its outline colour will indicate any anomaly.

Graphic pages are firstly drawn up off-line using a graphic editor, then checked before operational use. This check serves to confirm their coherence with the satellite databases. An animation environment is then generated and acts as a medium on which the real time animation can occur.

Several different graphic pages can be displayed at the same time in real time, for one or more satellites. The rate at which the views are animated then depends on the telemetry acquisition cycle, and the operator can quickly change page due to the graphic objects built into the views.

From a functional viewpoint then, the Graphic Server system integrates both a off-line mode offering the tools used to create and check the graphic pages, and a real time mode for actually using the graphic pages, acquiring data and animating the views.

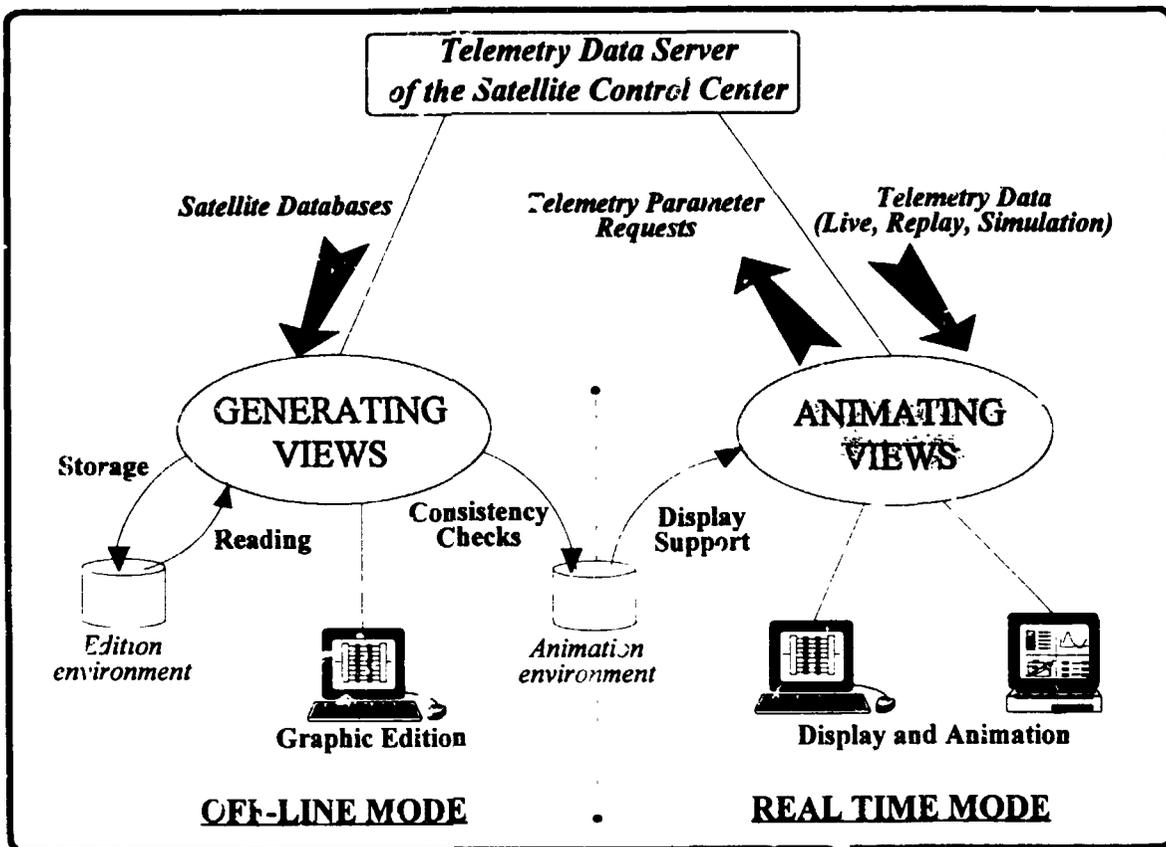


Fig.1 - Operating principles behind the graphic server system

OFF-LINE MODE: GENERATING VIEWS

The tools used in off-line mode are a graphic editor, used to create or modify graphic pages, and various utilities used to check the pages or analyze results obtained.

Creating views

Graphic pages are wholly created by the users, who thus have a wide range of freedom in the organization and representation of telemetry data, in the choice of viewpoint taken by each page (thermal, electrical, orbitography etc.) and in the synthetic degree of the detail represented. Within a page, each parameter can be shown several times in forms which may or may not be

complementary, which means that the user can have greater or more detailed information on certain parameters.

The first stage consists of formatting the content of the graphic pages, each page being able to include the following three types of objects:

- *static objects* constituting the background,
- *animated objects*, materializing in various forms the values of telemetry parameters,
- *pointable objects*, used to support operator dialog in real time mode (point and click objects).

Static objects are composed of graphic objects such as polylines, polygons, texts etc.

Animated objects include:

- alphanumeric and digital readouts used to display the raw or physical value of parameters in different display formats (binary, octal, hexadecimal, decimal and label),
- active symbols used to associate a particular graphic representation with each labelled value (e.g. relay open or closed); a maximum of sixteen such representations are allowed per parameter, each being defined by users and able to be put in a library for use with other parameters and different pages,
- scrolling curves of parameters in relation to time : these curves may optionally stop, reset from the origin or shift left (two-thirds of scale) and continue plotting when the right-hand axis is reached,
- moving symbols (e.g. dial with a needle such as a voltmeter).

Pointable objects include:

- static or dynamic tags (these objects, whose graphic representations are defined by the users, enable the user to change from one page to another simply by pointing to the object with the mouse and clicking it),
- data entry fields (these objects may be used to change page by typing in the name of the new page, satellite and display peripheral).

The second stage of edition is designed to associate telemetry parameter identifiers with the corresponding animated objects. This association is based on *simple naming rules*.

Finally, the last stage consists of "compiling" the pages that have been created so as to optimize real time performance for each page displayed.

Coherence check on views

The coherence check is carried out at one time on all the graphic pages created. The contents of each graphic page are validated with respect to the database for each satellite. If any errors are detected within a page, it cannot be used (for

those satellites where an error has been found). To correct the page, the previous stages must be repeated.

During the coherence check, a check is carried out to see that the naming rules mentioned earlier have been correctly applied. The checks are *syntactical* to confirm the existence of names and labels of telemetry parameters, and *semantic* to check that the animated objects chosen to represent each telemetry parameter are coherent with its type (an analog parameter, for example, could not be associated with an active symbol as, by its very nature, it cannot have more than sixteen values).

At the end of this phase, the user has a real time animation environment in which the telemetry data received in real time mode may be displayed.

REAL TIME MODE : ANIMATING VIEWS

Available on all the computers in the system, the Real Time application uses the animation environment created off-line and performs the graphic animation on the various display peripherals.

Acquisition of telemetry data

The Graphic Server system can manage and receive telemetry data from several different satellites at the same time. This data may correspond to *"live" telemetry*, to telemetry that has been recorded and is being played back in deferred time (*"replay" telemetry*), or even *simulated telemetry*.

The data is received in a *processed form*, and the raw value, the physical value (which may correspond to either a value or a label) and alarm status are associated with each telemetry parameter. Telemetry data is received via virtual X25 channels, each of which transmits the data for one particular satellite.

Displaying a new page automatically leads to dissemination requests being sent to the data server. The latter then interrupts the dissemination of parameters associated with the display of the previous graphic page and then transmits the new parameters needed by the graphic server to animate this new view. This principle allows operators to access almost all the telemetry parameters in terms of animation (virtual access). It does not affect the other pages displayed.

However, telemetry parameters may be systematically received and memorized by a graphic server. This capability means that when changing a page, the operator can immediately display the latest information on these parameters without having to wait for the acquisition cycle of them within the telemetry. For the Telecom 2 ground segment, for example, each graphic server in the control center receives all the parameters of a satellite, whatever the pages currently on display. On the other hand, the graphic servers in the payload control centers only receive those parameters needed to animate the pages actually displayed by the operators. This is because of the low transmission rates of the X25 links between these graphic servers and the data server of the Telecom 2 satellite control center. Like this, the operators can display all the views they want.

When a graphic server is used in a "off-line processing context" (such as telemetry replay or simulation), the systematic dissemination of all the telemetry parameters and their storage in memory by the graphic server grants the operator potential access to all these parameters in terms of display. The acquisition of at least one telemetry format and the interruption of replay telemetry or simulation, enables the user to consult whatever pages he wishes to in order to check particular points, diagnose a failure, divide up information and so on at his ease.

Display and graphic animation

Graphic animation, triggered whenever a new telemetry frame is acquired, can have different forms depending on the type of animated objects chosen to represent the telemetry parameters (cf. creating views).

Graphic animation also covers general parameters associated with each view and includes the name of the satellite and station, the number and date of the telemetry frame.

A default system of graphic representation is used to materialize parameters whose value is unknown. By this means, users quickly distinguish those parameters which, for special reasons are not received in Real Time, from parameters actually received and whose value is therefore significant.

The colour of each animated object varies in relation to the alarm status of the telemetry parameter with which it is associated (grey in the case of a telemetry drop, green when the parameter is nominal, orange or red when its status is simple or dangerous alarm). This means that any anomalies may be identified very quickly.

The number of graphic pages able to be displayed at any one time may be configured before the start of a Real Time session and may vary from one to five. The graphic peripherals may be used either in full screen mode (one page then filling the entire screen) or in quarter screen mode (four pages displayed on the screen).

Operator dialog

The user interface is the means by which a graphic page may be directed to a particular peripheral for a given satellite. The user dialog is based on the pointable objects (graphic objects able to be selected by the user) available in each view.

Data entry fields are used to type in information: name of the new page and/or name of the

satellite and/or number of the peripheral. The operator thus needs to entry data.

Static tags offer more limited functions in that their use limits the page change to the current peripheral for the same satellite. However, simply by using the mouse, this type of object may be used to change page automatically.

Dynamic tags have both the advantages of data entry fields and static tags. The operator can use them to define or modify a preselection of pages in real time. This makes calling them easier. The operator first associates the new page to be displayed, the satellite and the display peripheral with each dynamic tag. When the user next points to the tag, the corresponding page will automatically be displayed, with no need for any data entry.

FEATURES OF THE ARCHITECTURE

The Graphic Server's software includes the **ANIMATOR**® graphic software package developed and distributed by *Syseca*. This package comprises a graphic edition module, a Real Time animation module and an access library module.

The Graphic Server application uses the concepts and mechanisms of *data streams* (the arrival of data triggers off processing by tasks which themselves generate data for other tasks. Communication mechanisms are based on system V IPC). Processing systems for one data stream are independent from processing systems for another stream, which ensures continuity in downgraded mode should certain failures occur. As the application operates with multiple display stations, the failure of one of them does not interfere with graphic animation on another. Likewise, as the application also operates with multiple satellites, a problem linked to the telemetry data stream for one satellite does not perturb the processing of data streams for other telemetry. Furthermore, these mechanisms

ensure a certain extendability of the system (management of further satellites, addition of graphic workstations etc.).

The Graphic Server hardware architecture is based on Hewlett Packard HP 9000 from the 800 series.

There are three types of configuration :

- "off-line configuration" for generating views that includes a bitmap, a printer and an Ethernet link,
- "real-time configuration" for animating views that includes a bitmap, a X terminal, an optional printer and a X25 link,
- "full configuration" for both, generating and animating views (cf. Fig.2)

Hardware Architecture

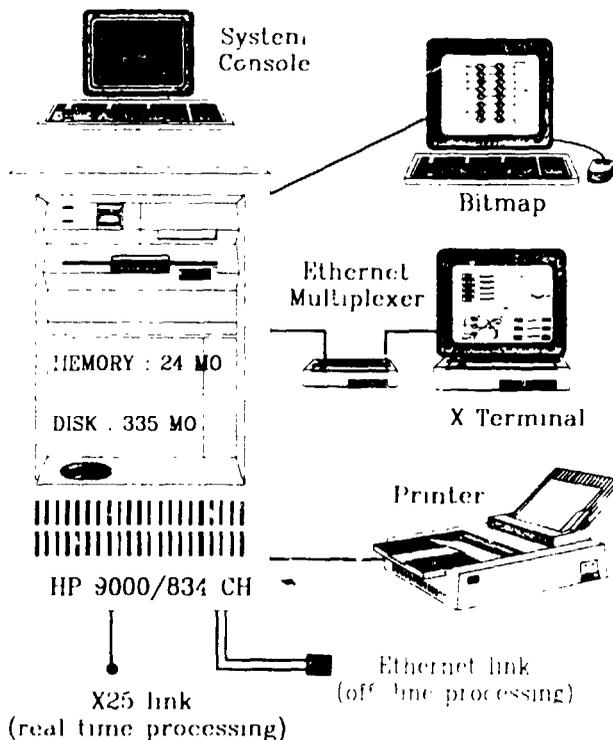


Fig. 2 - "Full configuration" of Graphic server

The views can be generated on an "off-line configuration" and then, the associated animation environment can be exported by streaming tape on others "real time configurations". This possibility allows the users to centralize the creation and the management of the views.

In off-line mode, the bitmap is used for editing views and running the utilities (consistency check, storage on streaming tape, ...). Consistency check results can be displayed on bitmap or printed. Telemetry parameters description files of the satellites are transmitted by Ethernet link via File Transfert Protocole.

In real-time mode, the bitmap is the support of the operator dialog. The views are displayed and animated on the bitmap (one "full screen" view) and on the X terminal (one "full screen" view or four "quarter screen" views). System and software messages are listed on the system console. The printer can be used to have a small logbook (some high level messages are printer as a telemetry drop warning, alarme status transition of a parameter). Telemetry data is received via X25 link and the telemetry parameter requests are sent by the same way.

OPERATIONAL VIEWS ON TELECOM 2

Constructed for the control ground segment of Telecom 2 satellites, the Graphic Server system has been operating in various operational Telecom 2 control centers since December 1991. There is a configuration reserved for the drawing up of pages. Drawn up then checked by satellite engineers, the pages are exported and finally animated on the "real time" graphic servers.

After over two years of operation, nearly *two hundred and fifty views* able to animate approximately *two thousand telemetry parameters* have been constructed according to team needs. Different categories of page have been created and correspond to special uses.

The following may be distinguished:

- *page catalogs,*
- *parameter dictionaries and specialized pages,*
- *parameter curves,*
- *functional synoptic displays of satellite subsystems (mimics),*
- *summaries of satellites in standby mode.*

Page catalogs grant rapid access to a given view. They are made up of static tags, each being associated with a particular page. Pointing to the name of a catalog page with the mouse automatically displays it.

Parameter dictionaries and specialized pages contain lists of telemetry parameter names with their raw and physical values. These dictionaries grant rapid access to a parameter (in alphabetic order), whereas specialized pages bring together related parameters needed for particular operations.

Parameter curves, used either in standby mode or during operations, are used to monitor in real time the changes in one or more parameters over time.

The functional synoptic displays of satellite subsystems (mimics) represent, in various forms, the state of the various satellite components. These pages are gradually broken down, moving progressively from a general level granting an understanding of the state of a subsystem down to a highly detailed level for specialists. The pages are linked together and the user can reach the level of representation he wants very quickly.

The summaries of satellites in standby mode are pages for general satellite monitoring. They inform the operators of any anomalies and of the main characteristics relating to the state of platforms and payloads (cf. example Fig. 3). The operator is thus kept continually informed of any alarm, its nature and its severity, and can display the most detailed views whenever he wants, so as to diagnose the origin of a failure.

All these views are detailed on [Loub1].

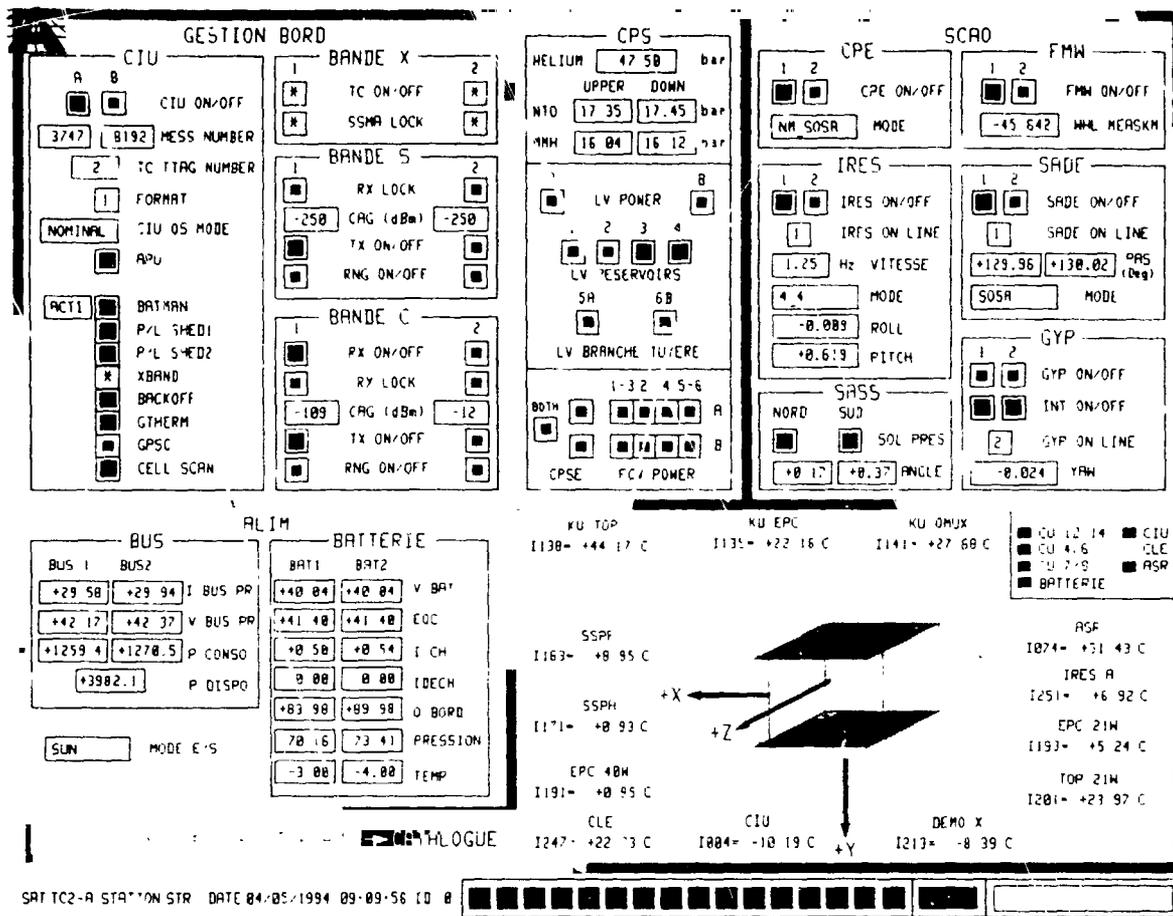


Fig. 3 - Example of a Telecom 2 operational view (150 parameters are animated)

FUTURE PROSPECTS

The graphic server system has become a vital tool for Telecom 2 operations because of its functional characteristics, its ease of use in real time mode, its graphic modelling capabilities, high-speed access to information, and the visual verification it allows on the state of satellites and their alarms. This system can also be used for control center applications and, more generally, adapted for use in *monitoring and "process" control* situations (the term "process" being taken in its widest sense, and may mean a satellite, test or simulation bench, an industrial manufacturing process etc.)

The Graphic Server system currently includes specificities peculiar to the Telecom 2 environment, mostly with respect to the mode of acquiring telemetry data and the format in which this data is disseminated. If this interface were to be made more general, the system could be put to a wide variety of uses involving the graphic display of data streams.

REFERENCES

[Loub 94] Jean-Philippe Loubeyre : "A graphic server for telemetry monitoring and procedure performing", SpaceOps 94.

ADVANCED TECHNOLOGIES IN THE ASI MLRO TOWARDS A NEW GENERATION LASER RANGING SYSTEM

Thomas Varghese
 AlliedSignal Aerospace
 ATSC/SLR
 7515 Mission Drive
 Lanham, MD 20706, USA
 E-mail: varghese@atscv1.atsc.allied.com

Giuseppe Bianco
 Agenzia Spaziale Italiana
 Centro Geodesia Spaziale
 75100 Matera, Italy
 E-mail: bianco@asimt0.mt.asi.it

ABSTRACT

Matera Laser Ranging Observatory (MLRO) is a high performance, highly automated optical and astronomical observatory currently under design and development by AlliedSignal, for the Italian Space Agency (ASI). It is projected to become operational at the Centro Geodesia Spaziale in Matera, Italy in 1997. MLRO, based on a 1.5meter astronomical quality telescope, will perform ranging to spacecrafts in earth-bound orbits, lunar reflectors and specially equipped deep space missions. The primary emphasis during design is to incorporate state-of-the-art technologies to produce an intelligent, automated high accuracy ranging system that will mimic the characteristic features of a fifth generation laser ranging system. The telescope has multiple ports and foci to support future experiments in the areas of laser communications, lidar, astrometry, etc.. The key features providing state-of-the-art ranging performance include: a diode-pumped picosecond (50ps) laser, high speed (3-5GHz) opto-electronic detection and signal processing, and a high accuracy (6ps) high resolution (<2ps) time measurement capability. The above combination of technologies is expected to yield millimeter laser ranging precision and accuracy on targets up to 300,000km,

surpassing the best operational instrument performance to date by a factor of 5 or more. Distributed processing and control using a state-of-the-art computing environment provides the framework for efficient operation, system optimization and diagnostics. A computationally intelligent environment permits optimal planning, scheduling, tracking and data processing. It also supports remote access, monitor and control for joint experiments with other observatories.

INTRODUCTION

Ever since the first deployment of laser ranging for space geodetic applications in the mid-sixties, the techniques of Satellite Laser Ranging (SLR) and Lunar Laser Ranging (LLR) have significantly contributed to the advancement of a number of scientific disciplines [Degnan, 1991; Schutz, 1992; Smith, et al., 1993]. Today a network of over 40 globally distributed systems support space geodetic efforts. The primary reason for the success and maturity of the measurement technique is the progressive use of advanced technologies as they evolved [Degnan, 1985; Varghese, et al, 1986; Veillet, et al., 1993; Shelus, et al., 1993]. The adaptation of newer technologies over the years yielded significant improvement in the instrument

performance. The quality of the SLR and LLR data has improved by two orders of magnitude during the last two decades. The accurate data over the years coupled with improved scientific understanding through measurement and modeling of phenomenon such as gravity field, tides, and the dynamics of earth's interior allows computation and maintenance of precision orbits to a few centimeters. The precise apriori knowledge of the orbit in turn permits the computation of precise acquisition and pointing vectors for tracking, thus allowing tighter target coupling of the laser beam through smaller beam divergence. The combination of precise pointing, high repetition rate laser systems and high opto-electronic detection capability has also led to vastly improved data quantity over the years.

There are, however, increased demands on laser ranging technique due to competing techniques and fiscal constraints. The future of SLR and LLR will depend on the scientific data quality as well as the cost of producing such data. High quality globally distributed measurement on a number of satellites, supporting various scientific applications, at low operational cost is a critical requirement for the future. Automation and multiple use of the facility are key aspects to be considered for the reduction and distribution of the cost.

In the global network, fiducial observatories play a fundamental role for the high accuracy measurements of geophysical properties. MLRO with its wide target coverage and ranging performance will become a part of a suite of geophysical and astronomical instruments at Matera obtaining critical measurements for a variety of applications. The targets for these measurements include satellites in earth orbits from ~200km to geosynchronous distances, the lunar reflectors (left by Apollo and Lunakhod missions) and deep space mission spacecrafts. With the significant coverage offered by MLRO together with the potential for other astronomical and optical experiments, optimal use of the observatory

during the 24 hour daily cycle is essential. The capability to configure, monitor and perform experiments in an expeditious manner without operator intervention is vital to the most effective collection of scientific data. The ability to perform intelligent decision making based on the observing conditions and the critical requirements of various experiments is a highly desirable feature. Thus, the precision, accuracy, reliability and ability to perform automated expeditious intelligent operations are emerging as the system goals a state-of-the-art system. MLRO detailed design is currently performed in the context of these emerging scientific requirements.

The system specification calls for millimeter precision and accuracy on ranging to targets as far as 300,000km. The absolute accuracy of laser ranging is limited by the measuring accuracy of the SLR instrumentation, the refraction model of the atmosphere, and the knowledge of the spacecraft optical reference to the center of mass. The spacecraft induced errors can be significantly reduced through modeling and correcting the laser data [Varghese, 1992; Minott, 1993]. The unique hardware characteristics of the ranging system can be corrected to the submillimeter level to obtain accurate range to the center-of-mass (CM) of the spacecraft. It is estimated that the atmospheric model induced errors can be reduced to the 1-2mm level using multi-wavelength ranging [Abshire, et al., 1985]. A high accuracy receiver system was developed to measure atmospheric dispersion very accurately in "real-time" [Varghese, et al., 1993]; the real-world operational performance of this receive system is currently under evaluation at the NASA 1.2meter telescope facility. If it demonstrates operational success, this feature will become part of the future millimeter system, thus solving the atmospheric model dependent problems. The ranging instrument performance is determined by the laser transmitter, opto-electronic technologies, time measurement system, telescope and the computing

technologies. Each of these disciplines is examined in detail in the current design phase to reduce ranging errors and exceed the system specifications.

SYSTEM DESCRIPTION

The laser ranging instrumentation of MLRO incorporates a number of highly desirable features [Varghese, 1992] that is expected of a fifth generation [Varghese, 1994] laser ranging system. The system and sub-system features are carefully chosen to exploit the best of currently available technology. In addition, design and integration of certain hardware components in the system is strategically scheduled to incorporate the best of evolving technologies. Major system hardware features are as follows:

- Multipurpose optical and astronomical observatory.
- 1.5 meter astronomical quality telescope with a high resolution imaging system for astronomy applications.
- Day/night laser ranging capabilities to dynamic targets in orbits of 200 km to geosynchronous distances, the moon and deep space missions.
- Design features to accommodate multi-wavelength ranging to directly measure atmospheric refraction effects.
- State-of-the-art computing and ranging instrumentation
- Easy referencing of telescope axes to external datum to further reference it to the center(CM) of the earth and the latitude and longitude.
- Hazard reduction of radiation on aircrafts using a radar.
- 10-20 Hz Operation at high laser powers; KHz operation using lower powers.
- High resolution(<2ps) time measurements of all critical times associated with various events.
- Aggregated instrument limited ranging precision of ~2mm and accuracy of ~1 mm.

The system software provides a number of highly desirable features. These include:

- Computational intelligence tools for decision making.
- Sophisticated GUI for expeditious diagnostics and operations monitoring functions.
- Autonomous operation of the system for tracking, instrument calibration and optimization.

The MLRO hardware and software modules are designed at the present time to provide an integrated framework for high performance automated operations. The hardware elements for ranging consists of the telescope, laser, transmit/receive optics, transmit/receive electronics, computing and control, timing, and safety. The 1.5meter aperture Cassegrain telescope has a pointing accuracy of ~1arcsecond and is based on a parabolic primary, hyperbolic secondary and a flat tertiary. It has a truly rotatable/removable tertiary to switch to Coude, Nasmyth or the Cassegrain focal planes for coupling to various instrumentation. The provision to "truly" rotate the tertiary mirror and position it within 1arcsecond allows easy interchange of Nasmyth and Coude foci. A state-of-the-art digital state space control system employing 32bit RISC processors for each axis control ensures smooth tracking and pointing operation while allowing self diagnostics and computer access to the telescope. The telescope jitter of <1arcsec RMS combined with the 1arcsec accuracy after star calibration allows precise tracking of distant targets. Since the observatory will be a multi-experiment research and observational site, safety measures for instruments as well as humans is given prompt consideration in the overall design of the system. The safety features include: radar, flashing warning lights, displays, alarms, video cameras, and computer-inhibited operations.

A diode-pumped picosecond (50ps) master oscillator and flash lamp pumped power

amplifiers generate ~125mJ in a 50-70 μ s pulse at 532nm to provide adequate link especially to very distant targets. This configuration is carefully chosen to address the future possibility of high duty cycle (>KHz) operation. The common Transmit/Receive (T/R) optics and the telescope transfer the laser beam to the target and also couple the retroreflected signal from the target to the detectors in the receiver system. The receive optics assembly couples the reflected light from the polarization discriminating T/R switch to the detectors after spatial and spectral filtering. The spatial filter has an adjustable field of view (FOV) from 1 to 60 arcseconds and its geometrical positioning is adjustable to accommodate defocus and decenter. The precise value will depend on laser beam divergence and background conditions. A CCD camera coupled to an image digitizer analyzes the transmitted laser beam quality; this feature is especially desirable for ranging to very distant targets. The narrow bandpass filter (0.1-0.3nm) allows tracking of the satellites/moon under high background conditions of day or night. The 1.5 meter telescope aperture and the superior optical quality of the telescope allows the coupling of the laser beam to the target at a beam divergence of 1-2arcsec with good wavefront quality. This beam divergence will be maintained for tracking all satellites whose orbits are computed and maintained precisely. The beam divergence control feature will be exercised to expand the beam divergence to accommodate prediction errors or when the initial acquisition was not successful. This is also true when the system attempts to track a newly launched satellite whose ephemeris is not known precisely. The data collected in real-time will be used to compute the short arc and propagate forward the improved real-time pointing information. An intensified CCD camera will optically track sun-lit earth orbiting satellites. It will also acquire lunar craters for ranging to the lunar retro-reflectors. These images will be processed in near real-time to permit target recognition and allow optimal guiding of the

laser beam to the retro-reflectors.

The data quality of ranging instrumentation is primarily determined by the T/R Electronics subsystem and therefore, plays a crucial role in determining the overall ranging performance. The opto-electronic detection and measurement of the time associated with each event is performed by the T/R Electronics. Special attention is taken to obtain the highest opto-electronic detection efficiencies (30%) and bandwidths (3GHz). The signal processing bandwidths will match the detection bandwidths to generate the most precise definition of the signal for time measurement process. The time and frequency subsystem is a critical part of the overall system. It provides the critical frequencies (10MHz, 500MHz) and timing (1, 10, 20, 100pps) signals from an ultrastable maser to support the generation of the high accuracy data. A multiple channel, multiple vernier event timer measures the time of occurrence of all critical events associated with each laser transmission to the target. The 28 bit event timer operating at a clock frequency of 0.5GHz measures the time from 100millisecond down to ~2picosecond. This 'local' precise time measurement is referenced to universal time (UT) within the uncertainty of UT. The optical events associated with each frame filtered spatially (1-60arcsecond), spectrally (0.1-.3nm) and temporally (~10-300ns) will provide the highest SNR for collected data. This feature is extremely useful for tracking of very distant targets with low link budget in the presence of high background count rate.

As stated earlier, the computing/control system architecture is partitioned to provide the users with the capability to perform multiple experiments/measurements. The software exercising control of the system and providing automation will be versatile in configuring the system for various applications. The emphasis of software engineering is on the ease of maintainability, upgradability and expandability. This will accommodate future expansion and all

optimal use of the system features and capability. The advanced computing environment in MLRO will permit smooth integration of all control and data related hardware functions and facilitate a very high level of automation. The software domain is divided into (1) man machine interface (MMI), (2) computing/decision making and (3) computing/control subsystems. The primary emphasis of the MMI will be to support monitoring, diagnostics and optimization of the system. The MLRO computer hardware configuration will consist of several state-of-the-art Hewlett Packard computers networked to form an efficient and effective computing environment with significant I/O capability. A VME-based real-time interfacing approach and a POSIX 1003.4 compatible real-time HP-RT operating system are special features of the real-time computing environment. The UNIX based HP-755 workstation permits a state-of-the-art man machine interface (MMI) and supports high end computing. Thus, compute-intensive applications such as GEODYN can be run with relative ease using this computing configuration. This capability is extremely useful for near real-time computing of orbits for improved satellite acquisition and pointing as well as processing the data. Currently, an a priori estimate of the orbit is used to discern the data from noise followed by statistical filtering and polynomial regression. With the ability to compute near real-time orbits from actual laser data, the filtering and data fitting processes can be implemented with greater effectiveness.

The real-time control and data related functions are addressed in the design using modern software engineering practices. Object-oriented programming techniques are conceived to facilitate speed of development as well as improve maintainability. Integrated performance monitoring of all processes constitutes a step toward identifying real-time process bottlenecks and highlight potential problems for scalability in the future. A key aspect of an automated system is also the ability to monitor the performance of the

system continuously. Device performance as well as data queue utilization, memory utilization, etc., will be included for routine monitoring.

The system performance to a large extent is monitored by numerical and statistical processing of various process parameters. For tasks involving numerical computation, conventional programming and analysis techniques offer superior speed over that of humans. However, in certain types of decision making problems, straightforward numerical computing alone is insufficient to deduce the pertinent scientific or technical conclusion. This is also true in cases where the problem is extremely complex and intuition is required for reaching decisions. If the exact rules for solving the problem is ill-defined or fuzziness exists such that conventional logic will not suffice to adequately and unambiguously define the answers to the problem, then "intelligent" decision making capability resident within the system will be an asset. Mission planning, scheduling, optimizing, sparse image and data analysis are areas where an expert system or computational intelligence tools (or their hybrids) can significantly offer help. Implementation of such tools are expected to further enhance the automation of operations and speed the evolution of MLRO towards a truly autonomous system. The availability of significant computing power is thus included in the current design of the system for the implementation of these capabilities.

SUMMARY

MLRO project is currently underway with the goal of designing a state-of-the-art system. Software and hardware architectures are carefully chosen to meet and exceed the projected specifications. The ability to perform automated intelligent tracking and ranging of dynamic targets at high accuracy will offer vastly improved capability for a number of scientific applications. The significant improvement in the quality and quantity of both SLR and LLR data will

further advance the science in all associated disciplines.

REFERENCES

- Abshire, J.B. and C.S. Gardner, Atmospheric Refractivity Corrections in Satellite Laser Ranging, *IEEE Transactions on Geoscience and Remote Sensing*, GE-23(4), 1985.
- Degnan, J.J., Satellite Laser Ranging: Current Status and Future Prospects, *IEEE Transactions on Geoscience and Remote Sensing*, GE-23(4), 398-413, 1985.
- Degnan, J.J., Applications of Laser Ranging to Ocean, Ice, and Land Topography, *Proc. of SPIE*, 1492, 176-186, 1991.
- Degnan, J.J., Millimeter Accuracy Satellite Laser Ranging: A Review, *Contributions of Space Geodesy to Geodynamics. Technology; Geodynamic Series*, Vol.25, 1993
- Minott, P.O., T.W. Zagwodzki, T. Varghese, M. Selden, Prelaunch Optical Characterization of the Laser Geodynamic Satellite (LAGEOS-2), *NASA Technical Paper 3400*, Sept. 1993.
- Schutz, B.E., Applications of SLR, *Proc. of Eighth International Workshop on Laser Ranging Instrumentation*, Annapolis, Maryland, May 18-22, 1992, NASA Conference Publication 3214
- Shelus, P.J., R.L. Ricklefs, A.L. Whipple, and J.R. Wiant, Lunar Laser Ranging at McDonald Observatory, 1969 to the present, *Contributions of Space Geodesy to Geodynamics, Technology, Geodynamics Series, Volume 25*, 1993.
- Smith, D.E., D.L. Turcotte, Editors, *Contributions of Space Geodesy to Geodynamics: Technology; Geodynamic Series*, Vol.23 and 24, 1993
- Varghese, T. and M. Heinick, Subcentimeter Multiphotoelectron Satellite Laser Ranging, *Proc. of Sixth International Workshop on Laser Ranging Instrumentation*, Antibes, France, September 1986.
- Varghese, T. and M. Selden, Towards High Accuracy (millimeter) Satellite Laser Ranging using Laser Geodynamic Satellites, *Proc., 10th Annual International Geoscience and Remote Sensing Symposium*, vol. II., 1427-1430, 1990.
- Varghese, T., Analysis of Topex Laser Retroreflector Array Characteristics, *Proc. of Eighth International Workshop on Laser Ranging Instrumentation*, Annapolis, Maryland, May 18-22, 1992, NASA Conference Publication 3214.
- Varghese, T., C. Clarke, T. Oldham, M. Selden, Streak Camera Based SLR Receive System for High Accuracy Multiwavelength Atmospheric Differential Delay Measurements, *Contributions of Space Geodesy to Geodynamics: Technology; Geodynamic Series*, Vol.25, 1993
- Varghese, T., Matera Laser Ranging Observatory (MLRO): An Advanced High Accuracy Satellite and Lunar Laser Ranging Observatory for Agenzia Spaziale Italiana (ASI), *Proc. First International Symposium on Laser and Optoelectronics Technology and Applications*, Singapore, pp. 460-465, Nov. 11-13, 1993.
- Veillet, C., J.F. Mangin, J.E. Chabaudie, C. Dumolin, D. Feraudy, and J.M. Torre, Lunar Laser Ranging at CERGA for the Ruby Period (1981-1986), *Contributions of Space Geodesy to Geodynamics, Technology; Geodynamics Series, Volume 25*, 1993.

THE SEQUENCE OF EVENTS GENERATOR A POWERFUL TOOL FOR MISSION OPERATIONS

Hubertus Wobbe
Armin Braun
DLR / GSOC
Mission Operations
82230 Oberpfaffenhofen - Germany

P. 7

Abstract - *The functions and features of the Sequence of Events (SOE) and Flight Operations Procedures (FOP) generator developed and used at DLR GSOC for the positioning of EUTELSAT II satellites are presented. The SOE and FOP are the main operational documents that are prepared for nominal as well as for non-nominal mission execution. Their structure and application is described. Both of these documents are generated, updated and maintained by a common software tool. Its main features and advantages are demonstrated. The tool has been improved continuously over the last 5 years. Due to its flexibility it can easily be applied to other projects, new features may be added.*

1 INTRODUCTION

The German Space Operations Center (GSOC) has been in charge for the positioning of various geostationary spacecraft during the last twenty years. The main operational documents for mission planning and execution of the Launch and Early Orbit Phase (LEOP) are compiled in the Operations Plan. It consists of TM/TC dictionaries, procedures for satellite, ground and flight dynamic operations as well as the Sequence of Events (SOE).

For the EUTELSAT II project a special software tool is used which allows for an easy and flexible generation of such a consistent SOE. The experience gained by the application of this tool during several missions led to the implementation of many additional features for ease of mission preparation and execution.

2 HISTORY

All the above mentioned elements of the Operations Plan were already applied for the SYMPHONIE

satellites, the first geostationary spacecraft positioned by GSOC in the mid seventies. These documents were typewritten without exception whereas last minute changes and updates appeared just as handwritten redline copies. All the timing functions as well as consistency checks in the preparation phase had to be done manually.

The TV-SAT direct broadcasting satellites were launched and positioned in the mid to the late eighties. For these spacecraft the Operations Plan documentation was produced by an electronic writing system using extracts of the original manufacturers operations handbook as prime input. TM/TC dictionaries were partially generated from the operational databases. A simple SOE generator was installed on a large scale computer, providing some limited features for mission timeline generation, mainly in the field of timing:

- time conversion from reference times into UTC
- time calculations
- consistency checks w.r.t. timing constraints
- automatic step (re-)arranging (e.g. in case of adding new steps)

The Sequence of Events was printed out in a fixed predefined format on a lineprinter.

For the DFS communication satellites, positioned during the late eighties and the early nineties, all information required for mission operations execution was incorporated in the SOE as one single applicable document including all operations procedures. This SOE was produced, printed and distributed for each mission event. Due to the complexity of this document most of the mission operations staff was supplied with special tailored extracts (e.g. subsystem, flight dynamic, ground data system extracts etc.). In order to handle such a comprehensive document a special Sequence of Events generator was developed on a mainframe computer. Main task of this software was to provide

a tool for easy and safe generation of a consistent mission sequence and the respective extracts. Since the early nineties GSOC is positioning the satellites of the EUTELSAT II series. For this project the initial philosophy was used, having the operations procedures as separate documents with the SOE as the guiding document.

The SOE generator developed for the DFS-project was found to be a very useful tool for the handling of the flight operations documentation. It was transferred to PCs and adapted to the project needs; its features and functions are pointed out in the following paragraphs. A simplified survey of the SOE generation process is depicted in Figure 1.

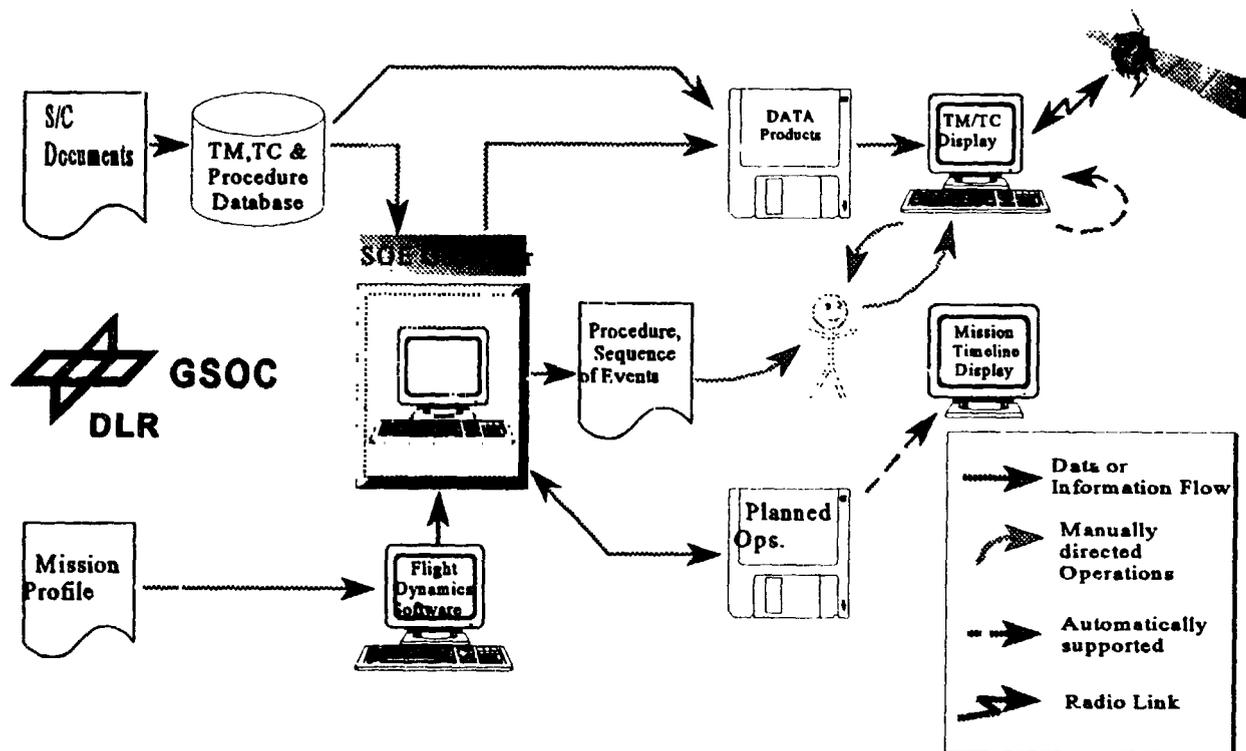


Figure 1: Sequence of Events Generation for EUTELSAT II LEOP

3 OPERATIONAL DOCUMENTATION

3.1 PROCEDURES

The satellite procedures provide detailed description of all nominal and contingency operations. They

define step by step all relevant activities including S/C surveillance, commanding and command verification to change the S/C from one stable and well defined configuration into a new one. The S/C manufacturer operations descriptions are complemented by indication of applicable reference

times, operational TM/TC mnemonics and references to the appropriate display masks as well as to the use of online supporting tools. The procedures are organized in substeps which contain only one single telecommand, telemetry verification or directives to the ground staff. Each substep can be flagged for which subsystem it is relevant or what constraints must be met. This allows to produce extracts for different user groups and to check the feasibility of the planned operation. Reference times which are assigned to all substeps allow to create a consistent operations timeline.

3.2 THE SEQUENCE OF EVENTS

The Sequence of Events is the controlling document for the execution of the mission. It coordinates all requirements and constraints arising from mission profile, S/C technology and ground network availability. The SOE provides the mission operations plan and specifies step by step all mission events and activities from launch through injection into the geosynchronous position. Activities are satellite operations, ground operations or flight dynamic operations.

As a mission timeline the SOE specifies item by item the relevant step to be performed in the mission, it indicates go/no-go decision points and orbit related information as:

- apogees, perigees, eclipses
- sensor visibilities, sun/moon interferences
- RF contact conditions
- maneuver start / stop times
- ground station visibilities
- ranging schedule

Reference is made to the respective procedures, only selected substeps of these procedures which are of general interest are reproduced in the SOE.

3.3 APPLICATION

During the whole LEOP each party of the mission operations team gets its directives from the SOE as the common guiding operational document. It is

valid for all nominal and contingency operations and is used in combination with the respective operations procedures :

- Satellite procedures for Flight Operations team
- Ground procedures for Ground Data System team
- Flight Dynamic procedures for Flight Dynamics team

According to the mission analysis the SOE is prepared before launch for the nominal positioning sequence which is for EUTELSAT II a 3-impulse strategy. Backup strategies in case of maneuver postponements are also considered in the SOE in advance in order to guarantee a save continuation of all operations in such a case. Figure 2 shows a typical maneuver tree of a 3-impulse LEOP strategy including all prepared backup cases.

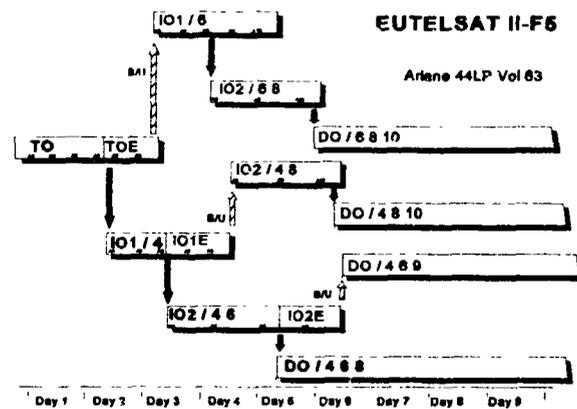


Figure 2: Typical maneuver tree for a 3-impulse LEOP strategy

Due to the fact that all online changes of the mission profile in case of launch delays, severe injection errors and ground or satellite contingencies only affect the SOE, all procedures remain valid and unchanged. The updates of mission operations documentation are minimized and can be introduced in a flexible, quick and safe way by means of the SOE generator.

4 THE SOE GENERATOR

The SOE generator software is written in high level programming languages linked to an application of a standard database software.

4.1 DATABASES

All information for flight procedures is stored in a relational database - the procedure database - containing several tables. The most important and comprehensive table contains the procedure steps. They are generated and validated with the Editor of the SOE generator. Inputs are provided by the satellite operations manual and design summary. Other tables are extracts of TM and TC databases, tables of the spacecraft attitude control modes, assignments of actions, available resources, constraints and affected subsystems.

The link from the original TM/TC databases to the procedure database ensures consistency of all these critical data. Throughout software development, mission plan preparation and mission execution the same databases are used by all parties thus minimizing effort and risks. There are no changes in this field without being driven and/or being documented by these databases.

The TM/TC databases are derived either from manufacturer delivered tables or files, complemented by information of S/C design summary and user defined information (e.g. TM/TC mnemonics). Operational products extracted from these databases are:

- TM/TC dictionaries
- all processing information for command system and TM-processor including alarm limits
- automatic TC execution verification database

4.2 PLANNING SOFTWARE

EDITOR AND VALIDATION

A comfortable menu system gives access to the various tools for handling of the procedure database.

A special editor allows to generate the procedures in a simple and safe way. Many automatic functions are triggered by single entries using information stored in the database:

- conversion of TM parameters into acronyms
- translation of TC codes into mnemonics
- functional description of TM/TC
- indication of display mask reference numbers
- generation of correct TC-datawords
- push-button queries for TM/TC codes
- suggestion of TM status parameter outputs

In addition the editor allows beside all standard functions to calculate and shift time labels and to branch into subprocedures. Each substep is automatically labelled with the date of its latest update for control purposes and automatic generation of change bars in the documentation.

Various validate functions allow to check the integrity of the database. In particular the following checks are supported:

- step records exist for all procedures
- no unnecessary step records exist for a procedure
- no step records exist without procedure header
- all codes have entries in the acronym tables
- all referenced subprocedures exist
- procedure duration matches the end time of last step
- TM/TC fields in procedures match with contents of TM/TC lookup tables

Inconsistencies found at validation may be corrected automatically in the database. Steps that require updates in the document printouts due to changes in the database are indicated by the validate utility.

The printout of procedures as a predefined database report is initialized by the editor tool. Layout changes of the output format for special extracts, testing purpose and for adaption to new missions can be freely chosen.

CHECKER AND FORMATTER

Task of the Checker and Formatter is the generation and validation of a consistent mission timeline. It directly accesses the procedure database.

Any field of the database can be selected by the user

to be implemented in the timeline. In addition the checker provides some further fields, such as absolute times and overall step number. The checker calculates absolute times for every single substep using freely definable time labels or absolute UTC time, whereas the procedures contain an internal chronology.

Orbital events which are the filtered output of the flight dynamics software are interleaved with the scheduled procedures (e.g. eclipses, acquisition of signal by ground stations, S/C geometry dependent constraints). By including all these data the SOE provides also information about margins and constraints to be considered in case of contingencies.

Various checks can be enabled in the checker's menu to confirm consistency of the generated mission sequence. These checks can be performed at procedure level, record level or for the timeline files and validate mainly following criteria:

- Chronological: making sure that start times are in chronological order
- Sequential: making sure that all elements are strictly sequential without overlapping
- Duration: checking for finite duration of records
- Constraints: checking for violation of constraints

A typical LEOP sequence consists of about 2500 steps each containing several substeps. Every single substep is verified by a number of checks like the above mentioned. To give an example: For each command activity in the mission timeline it is checked that a ground station is scheduled for uplink and no ranging is in progress. A number of checks like these are performed for every single substep counting a multiple of the 2500 main steps of a typical LEOP.

For the mission sequence printout the formatter module allows to specify output filters and sort orders to produce extracts for different parties. As for the procedures the print-layout of the SOE may be freely defined if required.

GRAPHICAL SEQUENCE EDITOR

The scheduling tasks necessary for a LEOP are relatively trivial compared to scientific missions. On the other hand some of the requirements and constraints are not absolutely fixed and may be negotiated between affected parties. In order not to overpower the SOE generator an automatic scheduler was not implemented. As a practicable approach an interactive graphical tool was introduced to manipulate the input files for the checker and formatter. All necessary information is provided in a graphical display as plot versus time. Different coloured boxes indicate start and stop times of

- station visibilities
- station schedule (prime / backup)
- satellite flight procedures
- rangings
- eclipses

In addition orbital information directly retrieved from flight dynamics software is plotted:

- apogees / perigees
- eclipses
- collinearity regions
- min/max altitude constraints
- sun/moon interferences

A supplementary window can be selected which shows schematically the S/C position and orbit geometry as well as the above mentioned orbital information. It allows to verify the regular distribution of rangings which is a non-linear function of the time. The sequence is edited and modified in an alphanumeric window at the lower part of the display. Editing uses automatic functions to a maximum extent thus minimizing the manual inputs. Entries are mainly keywords that can be converted into plane text by function key. All changes in the alphanumeric parts lead online to respective changes in the graphical presentation. By having this prompt graphical response to manual inputs eventual discrepancies can be detected and corrected immediately. Time consuming checker runs have to be performed less frequently.

The timescale of the displayed timeline can be selected in the range of 30 minutes for detailed analysis to the whole mission duration (about 2 weeks) for general overview.

Jumping to selected parts of the sequence may be done either by scrolling with cursor keys or by selecting the respective branch of the maneuver tree which is provided on a special display page.

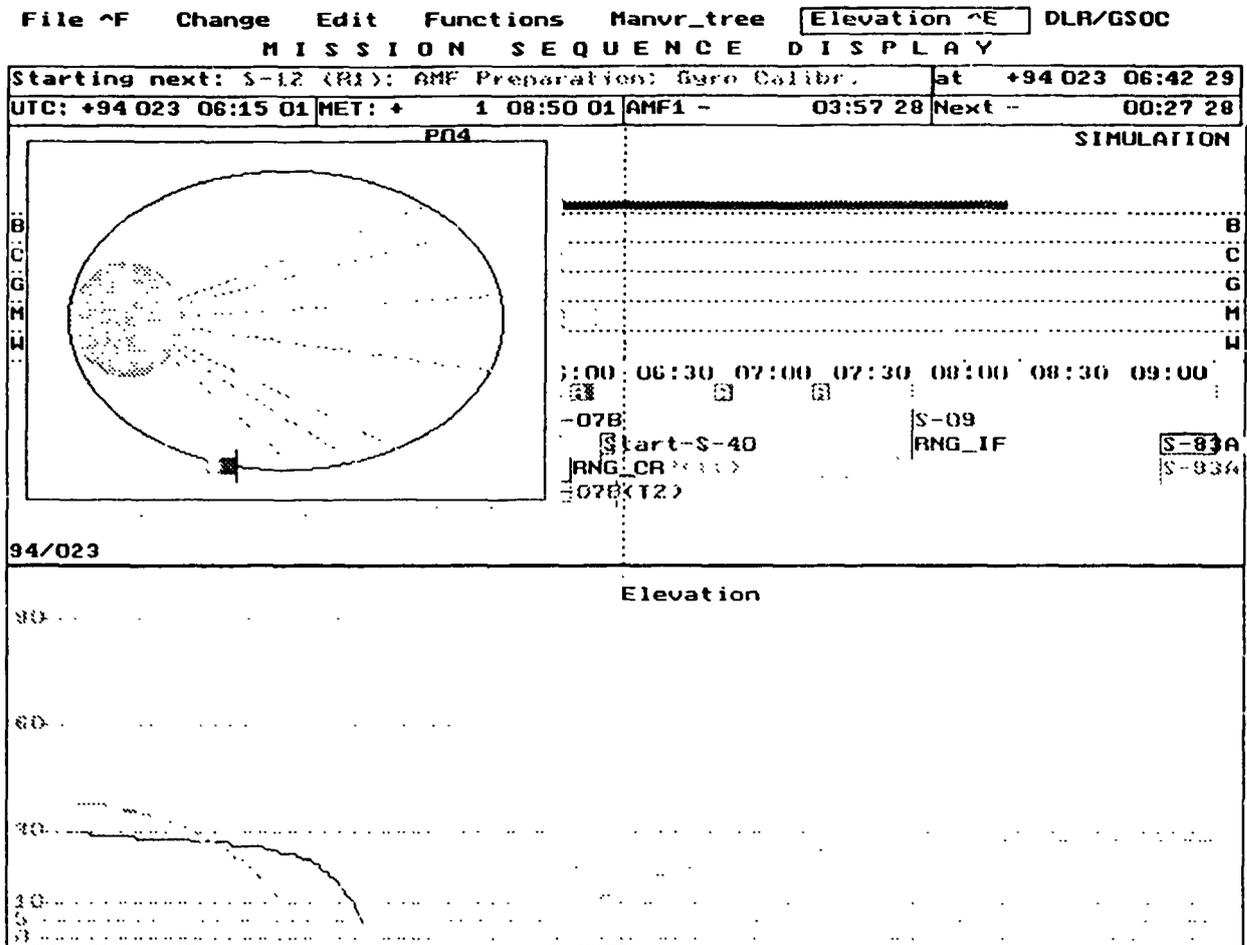


Figure 3: Graphical sequence editor. The station elevation angles, the S/C position and orbit geometry are optional windows.

4.3 SEQUENCE GENERATOR OUTPUTS

Main products of the Sequence of Events Generator are printouts of the Flight Operations Procedures and the Sequence of Events as described above. These may be printed completely or as special extracts. Additionally various outputs are generated making use of its features and the immense amount of information stored in the affected databases.

- Configuration Matrix

For each dedicated step in the mission timeline the SOE generator can determine the nominal status of the S/C configuration. The respective telemetry values can be output in a configuration matrix file which is linked to the online telemetry processor. On request the processor compares this matrix with the actual S/C data and indicates any deviations. Configuration check files are prepared.

in advance for each procedure as reference for start-up configuration checks as well as for major steps in the mission for go/no-go decisions.

- TC file generation:

The SOE generator provides a function to generate files of all telecommands for each single procedure. These files can be loaded directly into the command system where they may be released either manually or automatically with the radiation time defined by the SOE generator.

- Mission Timeline Display:

During the missions it was found that the Graphical Sequence Editor output provides an excellent general survey of the actual mission sequence. Therefore the display is projected onto a wall screen throughout the whole mission. The graphic is permanently updated by the actual time and shows all information related to orbit, station visibilities and procedures as described earlier. The actual status, history and future activities are displayed in a range corresponding to the selected resolution. Orbit related information is automatically updated by input files of the latest orbit predictions. In case of deviations from the original schedule the amount of expedite or delay can be entered and the graphic is shifted accordingly.

5. SUMMARY

The Sequence of Events Generator is a powerful instrument to prepare and guide all LEOP operations of geostationary spacecraft. This database oriented tool ensures easy, quick and safe generation of consistent mission operations documentation. In addition it provides some useful operational features and outputs. The flexibility of the application software and database structure allows to implement it also for other missions than geostationary positioning. At GSOC it has already been adapted for scientific missions as ROSAT and EXPRESS.

Operations

2. Command and Control		Page 539
OP.2.a	Designing an Autonomous Environment for Mission Critical Operation of the EUVE Satellite <i>Annadiana Abedini, Roger F. Malina</i>	541-546
OP.2.b	Increases in Efficiency and Enhancements to the Mars Observer Non-stored Commanding Process <i>Robert N. Brooks Jr., J. Leigh Torgerson</i>	547-558
OP.2.c	SCL: An Off-The-Shelf System for Spacecraft Control <i>Brian Buckley, James Van Gaasbeck</i>	559-568
OP.2.d	Mission Operations Data Analysis Tools for Mars Observer Guidance and Control <i>Edwin P. Kan</i>	569-576
OP.2.e	Generic Trending and Analysis System <i>Lori Keehan, Jay Reese</i>	577-582
OP.2.f *	A Graphic System for Telemetry Monitoring and Procedure Performing at the TELECOM S.C.C. <i>Jean Philippe Loubeyre</i>	583-590
OP.2.g	Small Scale Sequence Automation Pays Big Dividends <i>Bill Nelson</i>	591-597
OP.2.h	Safety Aspects of Spacecraft Commanding <i>N. Peccia</i>	599-605
OP.2.i	The Development and Validation of Command Schedules for SeaWiFS <i>Robert H. Woodward, Watson W. Gregg, Frederick S. Patt</i>	607-614

* Presented in Poster Session

538

DESIGNING AN AUTONOMOUS ENVIRONMENT FOR MISSION CRITICAL OPERATION OF THE EUVE SATELLITE

Annadiana Abedini, Roger F. Malina
 Center for EUV Astrophysics
 University of California
 2150 Kittredge St.
 Berkeley, CA 94720-5030

Abstract—Since the launch of NASA's *Extreme Ultraviolet Explorer (EUVE)* satellite in 1992, there have only been a handful of occurrences that have warranted manual intervention in the *EUVE* Science Operations Center (ESOC). So, in an effort to reduce costs, the current environment is being redesigned to utilize a combination of off-the-shelf packages and recently developed artificial intelligence (AI) software to automate the monitoring of the science payload and ground systems. The successful implementation of systemic automation would allow the ESOC to evolve from a seven day/week, three shift operation, to a seven day/week one shift operation.

First, it was necessary to identify all areas considered *mission critical*. These were defined as:

- The telemetry stream must be monitored autonomously and anomalies identified.
- Duty personnel must be automatically paged and informed of the occurrence of an anomaly.
- The "basic" state of the ground system must be assessed. Monitors should check that the systems and processes needed to continue in a "healthy" operational mode are working at all times. Network loads should be monitored to ensure that they stay within established limits.
- Connectivity to Goddard Space Flight Center (GSFC) systems should be monitored as well: not just for connectivity of the network itself but also for the ability to transfer files.
- All necessary peripheral devices should be monitored. This would include the disks, routers, tape drives, printers, tape carousel, and power supplies.
- System daemons such as the archival daemon, the Sybase server, the payload monitoring software, and any other necessary processes should be monitored to ensure that they are operational.
- The monitoring system needs to be redundant so that the failure of a single machine will not paralyze the monitors.
- Notification should be done by means of looking through a table of the pager numbers for current "on call" personnel. The software should be capable of dialing out to notify, sending email, and producing error logs.
- The system should have knowledge of when real-time passes and tape recorder dumps will occur and should know that these passes and data transmissions are successful.

Once the design criteria were established, the design team split into two groups: one that addressed the tracking, commanding, and health and safety of the science payload; and another group that addressed the ground systems and communications aspects of the overall system.

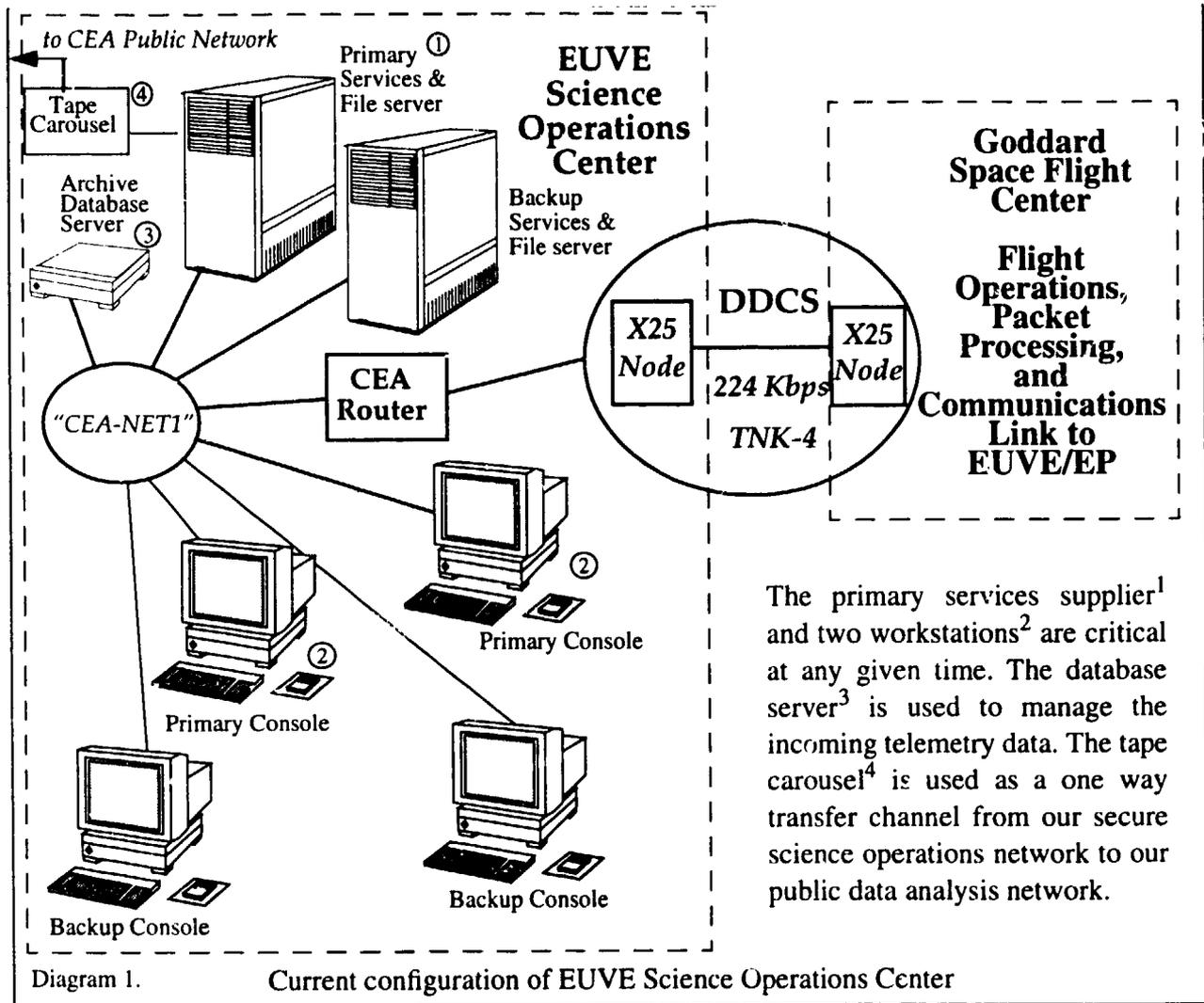
INTRODUCTION

In June, 1992, NASA launched the *Extreme Ultraviolet Explorer (EUVE)* satellite (Bowyer & Malina, 1991). The science payload for *EUVE* was designed and built at the University of California, Berkeley. The operations center (ESOC) for the science payload is located at the Center for EUV Astrophysics (CEA) at UC Berkeley.

The current method used for monitoring the *EUVE*'s science payload is a program called "soctools," which was developed at CEA. This tool displays numerical tables that change color and, in some cases,

give audible output when a monitored value goes out of or back into limit. Soctools is dependent upon the presence of a human who watches the printed displays as the values change. This situation requires 24 hours per day, 7 days per week staffing of the ESOC.

Though many aspects of the operations have been automated to some extent,¹ the monitoring of the EUVE/Explorer Platform (EP) science payload, as well as the monitoring and reconfiguration of the ground systems of CEA's secure science operations network,² is done manually.



The primary services supplier¹ and two workstations² are critical at any given time. The database server³ is used to manage the incoming telemetry data. The tape carousel⁴ is used as a one way transfer channel from our secure science operations network to our public data analysis network.

At the time that the ESOC was originally designed, the systems and software, which could allow a robust, fault-tolerant computing environment, were prohibitively expensive. With the desire to acquire a 99% delivery of data to CEA, the current configuration of the ESOC contains redundant computing and network hardware to reach this goal at a more reasonable cost. In the case of a failure of any of the integral parts of this system, it is possible for the payload controller or a hardware support person to

1. The telemetry reception and storage as well as the transfer of the telemetry from the ESOC network to a mass-storage optical jukebox on the science data analysis network are all done by means of automated software programs.
2. CEA's public science data analysis systems and network are physically separate from the science operations network.

remove a failed device from operation and quickly introduce a similar device into operation in its place.

In the above diagram, the *services supplier* is responsible for providing the processes that monitor the reception and storage of incoming telemetry data. The services supplier also provides the localized storage space used by the system utilities, login directory space needed by the payload controllers and other ESOC supporting staff, and temporary storage for the incoming telemetry.

The *backup services supplier* stands ready to be reconfigured if a problem arises with the primary system. The backup services supplier is also available for development and data analysis when it is not needed in the role of primary services supplier.

Similarly, the configuration of the database server has been planned such that one of the *backup console* workstations can be easily reconfigured in case of a controlling CPU failure. However, the tape carousel, which has proven to be one of our weakest links, has no *hot spare*. For this reason, we have set aside enough disk space to store several days of telemetry. In a situation where the carousel is down for a longer period of time than the disk space could accommodate, there is even a *sneaker-net*¹ plan that outlines a procedure for moving the telemetry from the "CEA-NET1" to our public network by hand, if necessary.

As noted above, after the statement of criteria had been defined, the initial design team was split into two parts, one to focus on the ground systems and communications, the other to focus on the monitoring of the science payload. The latter of these groups later divided the design efforts into health and safety monitoring of the payload, and the commanding of the of the payload. The team has postponed addressing, in depth, a design plan that would focus on the automation of commanding.²

THE SCIENCE PAYLOAD

The team evaluated several commercial and NASA funded packages intended to monitor satellite telemetry. The software package RTworks® was chosen. This commercial package contains generalized tools that can be customized to fit the specific needs of a project. It contains tools for building user displays as well as the capability to be "taught" about processes and to initiate other processes when an anomaly has been detected.

Initially, the team selected a set of six critical engineering monitors for autonomous monitoring using RTworks. The payload controllers captured their procedural knowledge into flow charts, which were then transformed into data-flow diagrams,³ as the controllers worked with a small team of programmers.

The lengthy process of creating and reworking these diagrams requires many iterations. However, the resulting diagrams not only give an accurate representation of the detection process but provide the programmers with an accurate starting point.

After both the controller and the programmer were satisfied with the visual representation of the anomaly detection process for each of the monitors in question, the result was programmed into RTworks' inference engine, creating EUVE specific extensions we call Eworks.

As the telemetry is received, Eworks actively monitors the telemetry data stream and if an anomaly is detected, a process is initiated which will notify someone of the occurrence. Currently this is done by means of initiating a program which pages the on-call controller.

Eventually, we would like to add diagnostic capabilities that would then allow autonomous response to

1. Sneaker-net is an industry buzz word that describes the process of hand carrying information from one location to another, or from one system to another, usually on a tape or floppy disk.
2. The team hopes to resume efforts in the area of automated commanding once confidence has been gained in the use of the monitoring software.
3. Flow charts may be adequate for the documentation of step by step procedures, but are not always the best means for representing a detailed picture of a complex, interactive system.

some of the anomalies. Since the software contains the ability to initiate other programs when an anomaly is detected, having the software invoke a diagnostic process that would then initiate a corrective action sequence, would be the logical next step.

THE SCIENCE OPERATIONS CENTER GROUND SYSTEMS

Monitoring

The team also investigated several software packages that could be used to monitor the availability and capacity of disks, system and network services, and critical processes. The team selected the commercial software package Sun NetManager® (SNM) for use for this task.

Like RTworks, in addition to its monitoring capabilities, SNM possesses the ability to initiate a corrective action sequence when an anomaly is detected. This means that SNM can be configured to either notify someone of the anomaly or take corrective action as appropriate.

The software can be configured so that corrective action is taken when disk usage of critical areas exceeds a specified limit, or when critical system or network services become unavailable. In example, if the primary services supplier becomes unavailable, the SNM software can start the lost services on the backup services supplier without human intervention. It can also initiate a process that will page the on-call hardware person and notify them of the loss of the primary server.

Similarly, if a critical disk area is filled above a prescribed limit, SNM can initiate a program that will clean up the area and remove files that are no longer needed.

Systems configuration

Although the automated software can simplify the overall monitoring process, the interdependence of the systems and peripherals are still a point of failure that would require human intervention in many cases. For this reason, suggestions have been made that would reduce these dependencies.

The current configuration utilizes multi-processor computer servers that share the tasks of providing disk storage to the systems on the network as well as processing power. The recent introduction of a networked version¹ of the redundant array of independent disks² (RAID) disk technology allows us to resolve the problem of having to duplicate disk storage areas and user accounts on both of the services suppliers.

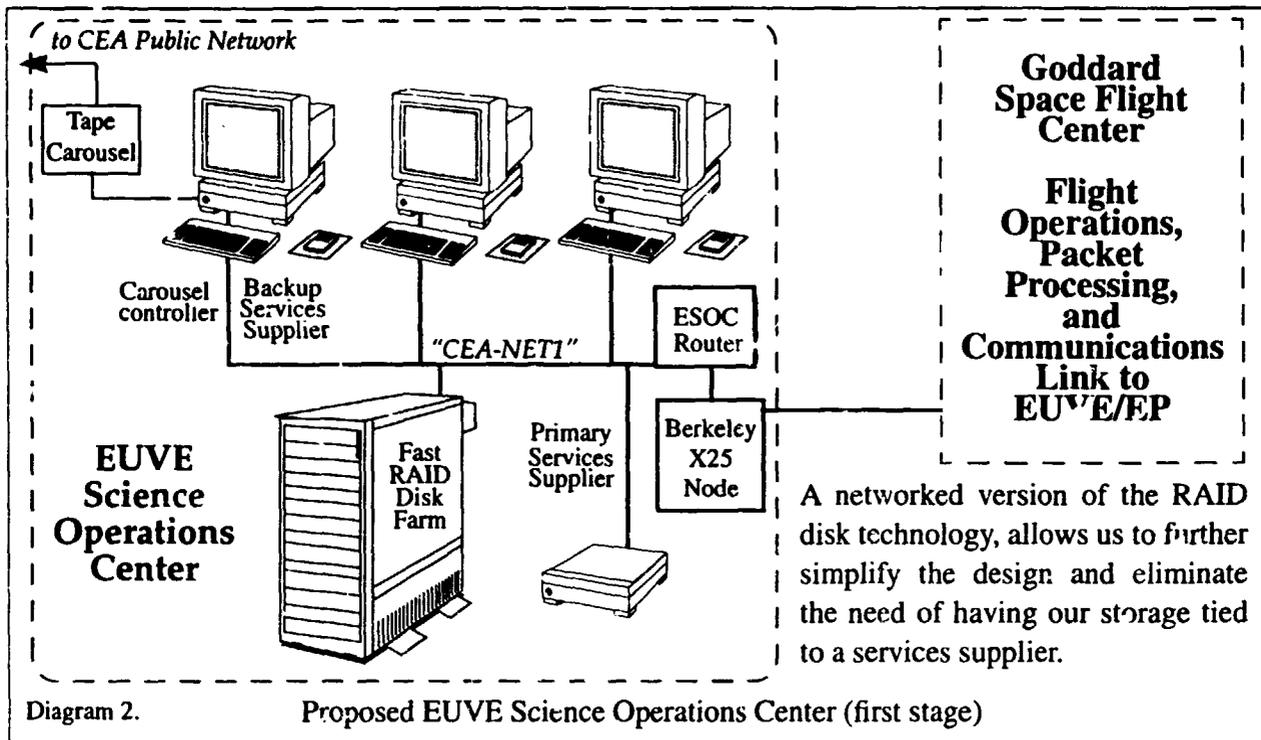
The RAID disk array provides highly reliable, fast disk storage to all of the systems on the network. With the utilization of disk stripping, a *warm spare* disk and a backup power supply, the system provides a hands-off, fault-tolerant storage solution. This unit also allows us to resolve many of the issues that set the requirement for human intervention in the transition from one disk and services supplier to another.

To date, our experience has shown that when there is a problem with the file server, which is currently supplying services to the ESOC, most of the systems in ESOC lock up and often require rebooting. This happens when the disk space that was being supplied by the services supplier becomes unavailable.³ Since this RAID disk is attached to the network and not any single system, the RAID disk eliminates the need to reboot systems. If a system supplying services fails, the disk area is still available to all of the other systems on the network. Thus, there is no network lockup and no interruption of services.

The elimination of the network deadlock allows the monitoring SNM software to initiate a program that

-
1. The FAServer 1400 from NAC is the first Unix RAID disk array that is not tied to another CPU.
 2. The description of RAID disk technology is beyond the scope of this paper. Please contact your hardware vendor for more specific information regarding this technology.
 3. If the disk was being actively accessed and the server does not come back on line, the workstation is locked into a disk-wait state.

will autonomously start any lost services on an alternate system and to dial out to notify hardware or software support personnel of the occurrence.



Having freed ourselves of the need for file server class systems, we can consider moving our processing needs to some of the fairly inexpensive, high powered, multi-processor workstation class computers. A computing system such as the Sun SparcStation 10 can supply an adequate level of computing power for the tasks of providing the daemons that oversee the reception and storage of telemetry, and monitoring the ground systems and science payload.

With the purchase of additional network cards, the NAC FAServer, RAID disk array, can allow up to four simultaneous network connections in order to supply the disk space to those networks. It, however, does not route information *between* those networks. The FAServer simply allows the users on all networks to be able to view the data stored on the disk array. The FAServer also allows for restricting access from a given network if desired. This means that the RAID disk array can be located in the access restricted computer room with direct login allowed only from its console. All other access to the unit would only be to the drives directly and the information stored there. This access could be restricted to read-only if desired.

This in mind, we could further simplify the configuration and allow the elimination of the tape carousel if we utilize the network RAID box feature that allows it to span multiple networks. In this alternate configuration the telemetry data, which is written to the disks as it comes in from the satellite, can be made available to the science data network as read-only information, and there would be no direct access to the secure network, nor opportunity to alter the data being provided by that network.

CONCLUSION

As confidence is gained in both the hardware and software being introduced into the ESOC, we will relax the staffing requirements, which are currently needed to ensure a smooth running environment.

Additionally, CEA is currently involved in the testbedding of diagnostic software packages from Jet Propulsion Labs and NASA Ames Research Center, as stated earlier, that will facilitate the autonomous resolution of predictable anomalies. Once the anomaly diagnostic systems are mature, we hope to start utilizing these techniques in the diagnosis of detected anomalies as well as add autonomous resolution of diagnosed problems.

ACKNOWLEDGMENTS

We thank M. Montemerlo, P. Friedland, D. Korsmeyer, and D. Atkinson for their collaboration on the work of new technology infusion into the EUVE Project. We would also like to thank the members of the CEA staff who are involved in the design and implementation of these operations improvements.

The work described here is funded by NASA contract NAS5-29298 and NASA Ames contract NCC2-838.

REFERENCES

- Bowyer, S., & Malina, R. F. (1991). The EUVE Mission. *Extreme Ultraviolet Astronomy*, 397-408.
- RTworks, Talarian Corporation, 444 Castro St., Suite 140, Mt. View, CA 94041, (415) 965-8050.
- Sun NetManager, Sun Connect, 2550 Garcia Ave., Mt. View, CA 94043, (800) 241-CONX.
- FAServer 1400, Network Appliance, Corp., 295 N. Bernardo Ave., Mt. View, CA 94043, (415) 428-5100.
- SPARCsystem 10, Sun Microsystems Computer Corporation, 2550 Garcia Ave., Mt. View, CA 94043, (415) 960-1300.

Increases in Efficiency and Enhancements to the Mars Observer Non-stored Commanding Process

Robert N. Brooks Jr.
Mission Profile and Sequencing Section
Mars Observer
Planning and Sequencing Team Chief

J. Leigh Torgerson
Spacecraft Systems Engineering Section
Mars Observer
Deputy Spacecraft Team Chief for Operations

Abstract - The Mars Observer team was, until the untimely loss of the spacecraft on August 21, 1993, performing flight operations with greater efficiency and speed than any previous JPL mission of its size. This level of through-put was made possible by a Mission Operations System which was composed of skilled personnel using sophisticated sequencing and commanding tools.

During cruise flight operations, however, it was realized by the project that this commanding level was not going to be sufficient to support the activities planned for mapping operations. The project had committed to providing the science instrument principle investigators with a much higher level of commanding during mapping. Thus, the project began taking steps to enhance the capabilities of the flight team. One mechanism used by project management was a tool available from Total Quality Management (TQM). This tool is known as a Process Action Team (PAT).

The Mars Observer PAT was tasked to increase the capacity of the flight team's non-stored commanding process by fifty percent with no increase in staffing and a minimal increase in risk. The outcome of this effort was to, in fact, increase the capacity by a factor of 2.5 rather than the desired fifty percent and actually reduce risk. The majority of these improvements came from the automation of the existing command process. These results required very few changes to the existing mission operations system. Rather, the PAT was able to take advantage of automation capabilities inherent in the existing system and make changes to the existing flight team procedures.

• **Jet Propulsion Laboratory/California Institute of Technology**
4800 Oak Grove Dr.
Pasadena, CA 91109

The work described in this paper was carried out at the Jet Propulsion Laboratory/California Institute of Technology under contract with the National Aeronautics and Space Administration.

This paper will describe in detail the enhancements recommended by the PAT for the non-stored command generation process on Mars Observer. This will be contrasted with the process used by the flight team prior to implementation of these improvements. Finally, there will be a discussion of the applicability of the techniques devised by the PAT for enhancement of the non-stored command process to present and future projects.

INTRODUCTION

The Mars Observer project had as its goal the complete mapping of the Martian surface in several spectral regions. Some areas were to be mapped in extremely high resolution. This was going to be accomplished by following a flight and operations strategy which used the following design principles.

- The spacecraft would be a relatively simple device which would act as an orbiting platform from which to perform remote sensing of the planet's surface and atmosphere.
- The spacecraft would be placed in a low altitude (378 km), near circular, near polar orbit.
- The science instruments would be Nadir pointed with the remote sensing science instruments mounted on a rigid platform.
- Any and all instrument articulation would have to be performed internal to the instrument and be of a non-interactive, non-interfering nature.
- All control of the instruments was to be managed and commanded by the remotely located science instrument teams. The JPL flight team was to be a "passive" through which commands moved, but were not interfered with.
- The flight team staffing was only normal working hours.

These six basic design principles were intended to reduce complexity of operations, increase the autonomy of the Principle Investigators over their instruments and, ultimately, reduce costs by reducing flight team workload and staffing requirements. Unfortunately, a multitude of factors influenced the designers of the operations processes and true autonomy was not attained at the time of launch in 1992.

Though the thrust of this discussion is not to elaborate on these factors, it should be sufficient to point out that, at the time of launch, all were legitimate concerns and, therefore, causes for conservatism on the part of the operations designers.

However, after launch it was discovered that many of the aforementioned concerns were no longer problematic. Steps had been taken by various parties to mitigate the problems and a less conservative approach was deemed appropriate. In addition, it became abundantly clear to management, the science teams and the operations team that the level of science commanding necessary to accomplish mission goals was not going to be possible given the conservative operations techniques used by the flight team. A totally new approach would be necessary to satisfy these needs.

The tool which project management decided to use for accomplishing this goal was a standard tool available from Total Quality Management (TQM). This tool is called a Process Action Team (PAT). The PAT assembled by the project manager was charged with determining the best method for increasing efficiency and through-put of the processing of Non-interactive Non-stored Commands (NINSC). This paper will discuss the concept of a PAT,

describe the original NINSC process as it existed at launch and the streamlined NINSC commanding process which resulted from the deliberations of the PAT. Finally, a brief discussion of the application of these operations strategies to future projects will be given.

ORIGINAL NON-INTERACTIVE NON-STORED COMMAND PROCESS

The Mars Observer spacecraft design allowed for command execution immediately upon receipt or for the storage of a series of time-tagged commands that would autonomously execute at the appropriate time. These stored commands were referred to as "sequences," and the spacecraft was capable of simultaneous execution of several stored sequences.

As the Mars Observer spacecraft normally flew with one or more stored sequences on board and executing, non-stored commands were scrutinized carefully to assess the possibility of adverse interaction with current sequences, spacecraft configuration or power and thermal conditions.

The spacecraft was specifically designed to minimize the interaction of the science instruments with the power, thermal or dynamic states of the spacecraft bus. A small number of payload commands could cause the power consumption of the payload suite to significantly increase and these were deemed "Interactive" commands. The majority of the payload commands were "Non-Interactive," and the design intent was to allow the science instrument operators maximum freedom to send non-interactive commands to their instruments in real-time without submitting command requests for scrutiny by the flight team, as was necessary in the case of interactive payload commands.

These were termed "Non-Interactive Non-Stored Commands," or NINSC's.

A basic innovative concept behind the Mars Observer operations strategy was that the science teams were located at their home facilities, with command requests and science instrument data communicated electronically through computer networks. A central Project Data Base (PDB) was established at the JPL facilities in Pasadena, with appropriate security measures in place. Each science team had electronic access to current spacecraft health and status data, science data downlinked from the spacecraft, and a repository for placing files that contained NINSCs they wished sent to their instruments. Each science team had their own secure database "bin" for command requests and science data.

There were two parts to an instrument command. Part one was the binary file or files containing the actual commands to be sent to the spacecraft, and part two was the command request which detailed the purpose of the commands, the desired time of transmission, or, if several files needed to be sent in a specific order at certain times, a radiation plan for the Mission Control Team (MCT) to follow. The science team would put these items in the PDB, and notify the Experiment Representative at JPL via FAX, telephone call or E-Mail that a command request was ready for processing.

Processing these requests involved the steps summarized in figure 1. The command file containing the commands for the science instrument to execute had to be

- a. Checked for valid instrument ID and opcodes.
- b. Merged with spacecraft commands which would pass the

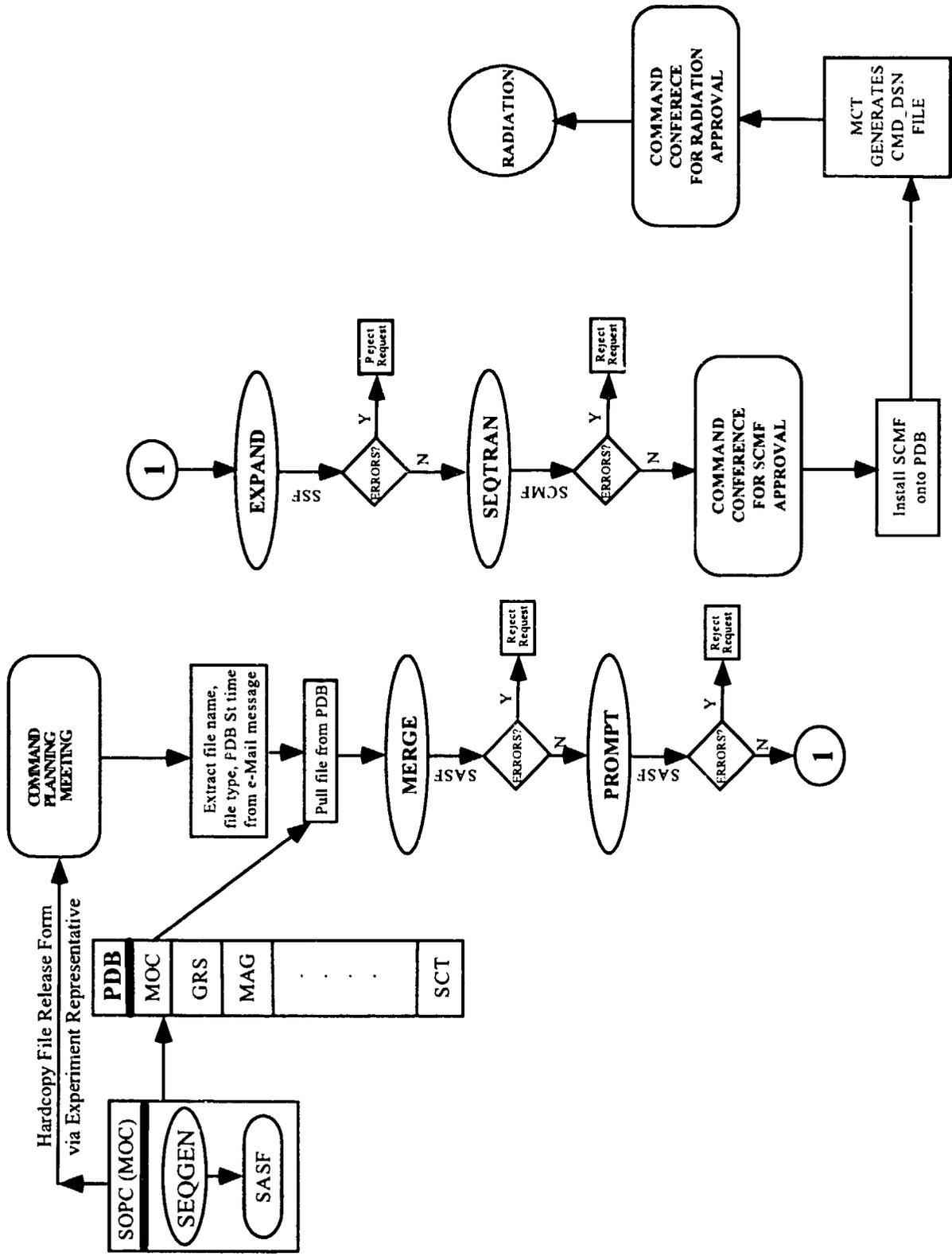


Figure 1: Original NINSC Process

- payload commands through to the appropriate instrument.
- c. "Wrapped" with a header which provides information to the DSN about which spacecraft to send the command to and at what time.
 - d. Converted to the actual binary file to be sent to the DSN for radiation.

Each of these steps were conducted by different people and several separate pieces of software were required to generate the intermediate files and reports. To limit development costs, much of the software used was taken from other projects and modified to suit the needs of Mars Observer, resulting in a multi-stage process.

With each of these steps there was much paperwork generated, manual Quality Assurance (QA) operations to insure that errors were caught and management scrutiny to see that the commands were indeed non-interactive. In parallel with this process, a series of meetings were conducted to sign off the QA process, coordinate with the Mission Control Team (MCT) on when the commands were to be sent, and to apprise the flight team of the intended command activity.

This process embodied the conservatism necessary to avoid problems which might be brought on by inappropriate commanding, and served the project well for the first few months of Mars Observer flight operations. It was, however, far from the "real-time" commanding expected by the science community, and the process promised a significant workload during mapping, where as many as six NINSC requests per day were expected. Extrapolation to the mapping scenario showed that the original NINSC

process would have taken 34 work-hours per day and produced 120 items of paperwork per day.

PROCESS ACTION TEAMS

The basic concept behind a Process Action Team (PAT) is that the owner of some process assembles a group of people familiar with the process to study it in detail and then to recommend ways to achieve a set of specific objectives and measurable goals with respect to that process. The PAT uses a formal methodology, and has both a schedule to adhere to and a set of deliverables. A facilitator from outside the project is brought in to aid in objectivity, and a Quality Council panel of senior managers (some from outside the project) periodically reviews the work of the PAT.

The Mars Observer (MO) Uplink PAT was established by formal charter by the project manager, and had the task of reevaluating the uplink process and to establish revised procedures to fulfill several objectives, including:

- Improved responsiveness to science command requirements
- Increased command volume without risk
- Streamlining of the entire uplink process.

These improvements were to be made without any increase in command-processing workforce, and as a goal, the resulting process was to provide at least a 50% increase in command generation capacity by the existing workforce.

The PAT was to deliver a defined set of products which included revised project policies, procedures, forms, interface

agreements and any other documentation necessary to describe and control the revised uplink process.

The activities of a PAT are conducted in a structured, 4-part methodology described by the acronym "FADE", which stands for "Focus", "Analyze", "Develop" and "Execute".

The Focus phase is to decide on exactly what the problem is, and to narrow the focus of the team's work so as to avoid attempts to either solve too much or solve the wrong problem. The result of the Focus phase was a Problem Statement which described the current state of the uplink process, the impact to the customer, and the desired state. The MO Uplink PAT focused on the NINSC process.

At the completion of each phase, the Quality Council reviews and approves the work of the PAT before the commencement of the next phase. This is to avoid the possibility of designing a solution to a problem which, in the eyes of management, may not exist.

The Analyze phase is designed to investigate and quantify the process to shed light on just where the problem areas are. The phase involves deciding what data are necessary, collecting these data to baseline and identify trends, and to finally determine which factors are the most influential. The MO Uplink PAT studied the NINSC process, and did a detailed accounting of the time and energy required to complete each step of the process and determined what "value-added" there was for each step or process output.

During the Development phase, the improvements to the process are developed. These improvements include not only a new process to implement, but also an

implementation plan to smoothly transition from the old process to the new. The MO PAT found paperwork and reports generated which had no "customers", found several areas where inexpensive automation could replace manual checks, and identified new command categories which would allow achievement of science objectives without increasing either risk or team size.

The final phase is to Execute the solutions defined in the Development phase. The first step is to obtain management and team support for the solutions - a task made infinitely simpler by the objective data and thorough methodology of the preceding three steps. Next is to implement the new process, and to monitor its effectiveness using the same metrics and methods used in the Analyze phase. In the case of the MO Uplink PAT, management and team acceptance of the new process was obtained, some of the new procedures were implemented and monitored, but the unfortunate loss of the spacecraft prior to mapping precluded a full evaluation of the new process.

The following section details the new NINSC process recommended by the MO Uplink PAT.

DESCRIPTION OF RESULTS

The final outcome resulting from the deliberations of the MO Non-interactive Non-stored Commanding Process PAT was a set of recommendations which would increase the through-put for Non-interactive Non-stored Commands from the current one hour or more per command file to a maximum of fifteen minutes per file. This increase in efficiency was to be accomplished by altering the existing process in three specific ways.

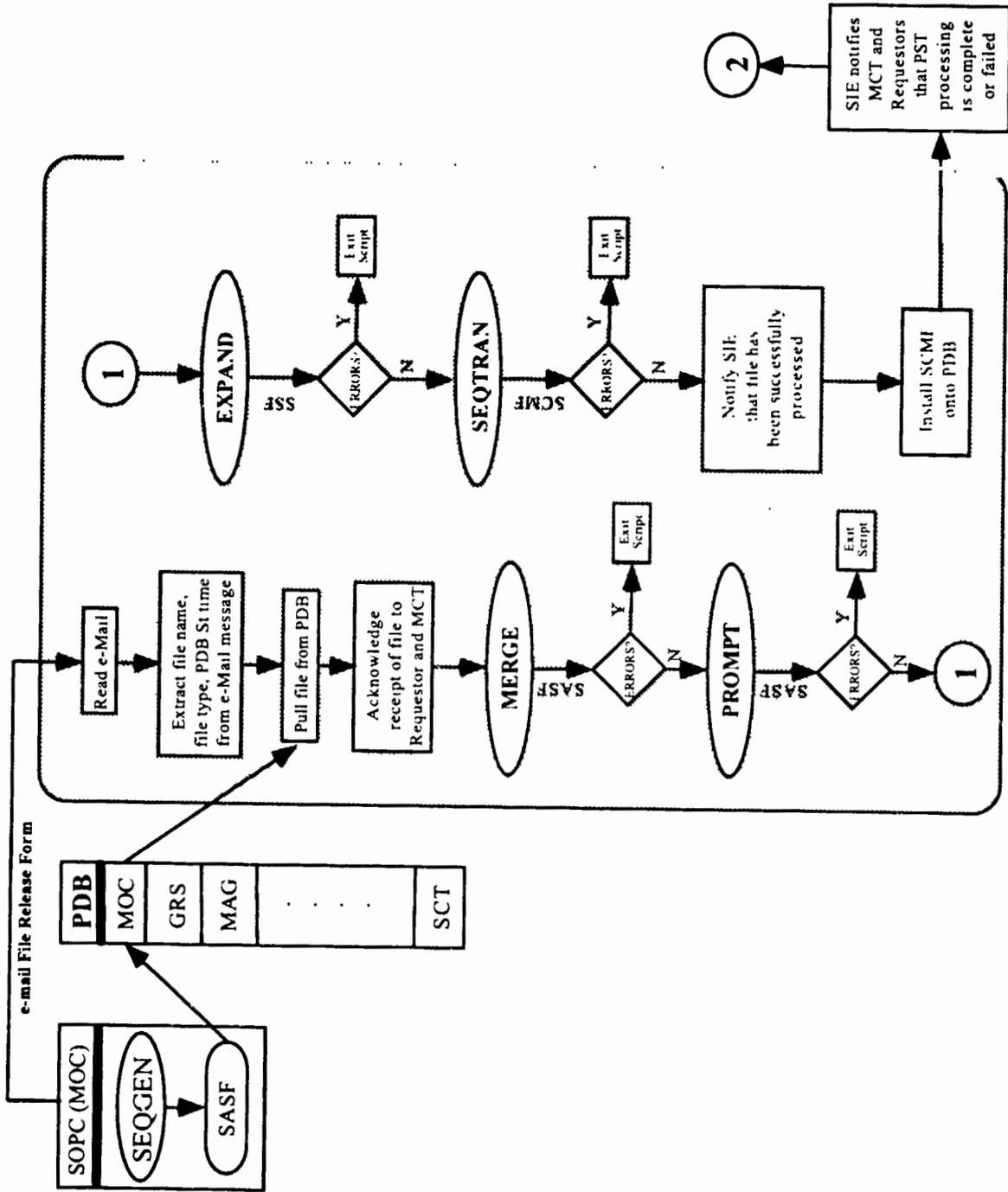


Figure 2: Planning and Sequencing Team (PST) Script

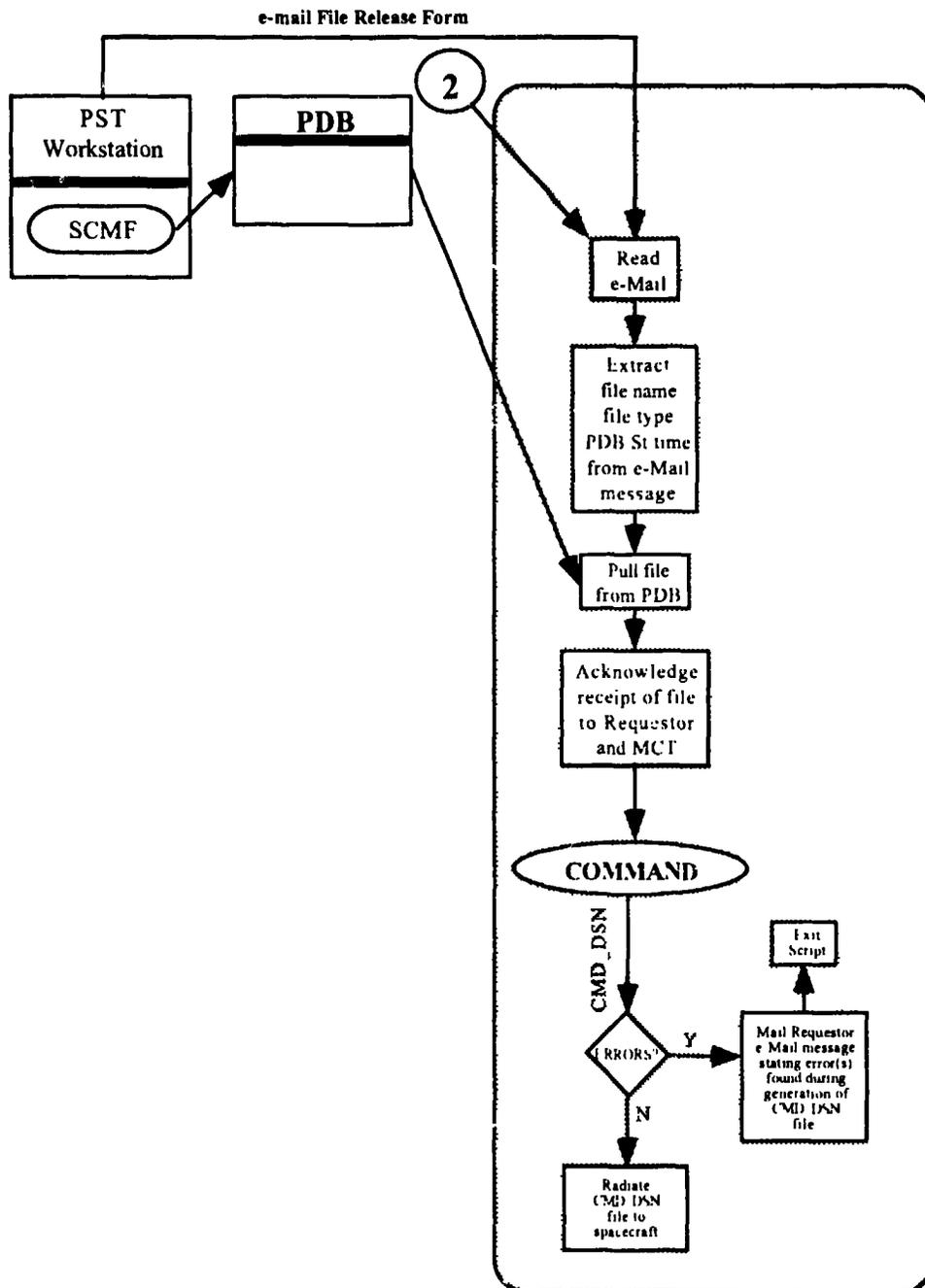


Figure 3: Mission Control Team (MCT) Script

The first problem identified by the PAT as hindering the processing of NINSCs was excessive management scrutiny of the command requests. This scrutiny was felt to be necessary to prevent erroneous commands from being sent to the science instruments. The elements of the command request which were scrutinized included purpose of the requested commands and correctness of the data contained in the request. After some study, the PAT found that such intense scrutiny was totally unnecessary. This was based on the fact that the spacecraft and science instruments had been built so that such commands could not compromise spacecraft health or safety. Furthermore, much of the syntactical checking was already being performed by the ground software system and, therefore, did not need repeating by management. The PAT therefore recommended that all such scrutiny of NINSCs be stopped.

Another problem which was identified by the PAT was excessive amounts of paperwork associated with this type of commanding. Every command request processed required between ten and twenty pages of paper, depending upon the number of commands in the original request. Completion of this paperwork became an intense burden on the flight team. The PAT recommended that NINSCs be exempt from the large amounts of paperwork associated with other types of commanding.

This leads to the third change recommended by the PAT. At the time of launch all NINSCs had been classified together as one large group.

Flight team and management procedures treated all of these commands with equal conservatism and caution. However, as the flight team gained more experience flying the spacecraft, they found that approximately 85% of these commands were genuinely non-interactive in the truest sense of the word.

These commands required no spacecraft resources or significant ground resources. This led the PAT to recommend that a new class of NINSCs be defined which required no coordination beyond any incorporated within the file as it was submitted by the requester. Their processing was to be heavily automated and very rapid. This new class of commands would be referred to as Express commands.

The automation of the Express NINSC process was fundamental to the successful increase in efficiency. This automation would be accomplished by using two scripts written in UNIX, PERL and awk. These scripts were divided along team functional lines. The Planning and Sequencing Team (PST) used a script which would execute all necessary and appropriate software, automatically checking each file for errors as it was processed. After each file had completed its PST processing, it would be retrieved by the Mission Control Team (MCT) using their script and processed into a CMD-DSN file for radiation to the spacecraft. What follows is a detailed description of the Express NINSC process as implemented on Mars Observer.

DETAILED DESCRIPTION OF FINAL IMPLEMENTATION

The EXPRESS NINSC command process would begin with each requester who required commanding installing their request Spacecraft Activity Sequence File (SASF) onto the PDB in the appropriate PDB bin. At the same time that the requester installed their SASF(s) onto the PDB, they would send an e-Mail "File Release Form" (FRF) to both the PST and the MCT. These two tasks were to be completed by 10:00 am Pacific time for the file(s) to be considered for same day processing.

Flight team processing of Express NINSCs required very minimal human interaction (at only the beginning and end points of the scripts). This interaction was of a process management and instigation nature. Actual file processing, execution of sequencing software and error checking were performed internally by the script. Figures 2 and 3 are graphical representations of the Express NINSC process.

Beginning at 10:00 am Pacific time every weekday, the PST would instigate execution of the EXPRESS NINSC script. This instigation would be authorized by the Sequence Integration Engineer (SIE) and actual script execution initiated by the Software Operations Engineer (SWOE). Each file would be processed by the script, one file at a time in the order that the e-Mail file release forms were received by the PST, until processing was complete.

The script would begin by reading the e-Mail FRF submitted by the requester. This FRF adhered to a specific format and contained data necessary to verify file origin and location. The script extracted from the FRF all of the above described data. The script used these data to extract the SASF from the PDB and install this SASF onto the PST workstation being used to process NINSCs. The script then sent an e-Mail acknowledgment of receipt of the SASF to the requester and the MCT. This acknowledgment allowed these two groups to track the status of those files being processed.

The script executed the MERGE software. This software correlated requesting group and destination instrument. The latter was accomplished by comparing the file type provided in the FRF with the instrument OPCODE provided in the SASF.

The script would then execute a general purpose error detection program. This piece of

software used other program's runlogs as input to check the success of those runs. In this case, it used the MERGE program's runlog as input.

As is obvious from figure 2, during execution of other parts of the script other program's runlogs would be used as input for this program. Any errors detected during execution of this software caused immediate exit from the script and a failure message, containing file name and failure details, to be sent by e-Mail to the SIE. The SIE then determined which was the best resolution of the error. At the discretion of the SIE, this may have included rejection of the file or contacting the requester to help in correction of the error. In any case, an erroneous file was not guaranteed same day readiness for transmission to the spacecraft.

This was followed by the script executing the PROMPT software, which would verify syntax, data field value limits and SASF format, the EXPAND software, which converted the SASF into a Stored Sequence File (SSF). The SSF can be thought of as the "source code" for the commands requested in the SASF. This SSF was used as input to the SEQTRAN software in the next step and finally the script would execute the SEQTRAN software. This software converted the SSF generated by EXPAND in the previous step into an Spacecraft Message File (SCMF, the actual binary representation of the data in the original SASF).

Upon successful completion of all preceding steps in this script, the script would notify the SIE that the file had completed processing and would automatically write the SCMF for the file to the PDB.

The final step of PST processing was the responsibility of the SIE (not the script). This was the notification of the requester and the MCT by e-Mail that the file completed

processing and was available on the PDB. This e-Mail message contained a PST FRF. This FRF was formatted in a specific way and contained information needed by the MCT to begin their processing.

The PST would repeat the above steps for each file for which an FRF was received, until all files submitted for that day had been processed.

Immediately upon receipt of the PST e-Mail File Release Form (FRF), the MCT would initiate its script to process SCMFs into CMD_DSN files (the files which is formatted to be transmitted through the Deep Space Network). The first step in this script was to retrieve the e-Mail FRF and extract the SCMF file name and other pertinent data. The script would use the information provided by the PST FRF to extract the appropriate file from the PDB. The script would then verify the file's authenticity. The script then executed the uplink window computation software to determine the available uplink windows for the file being processed.

After determining all available uplink windows in the preceding step, the script would execute the COMMAND software, which converted an SCMF into a CMD_DSN file. Though an SCMF does contain the actual bits to be loaded onto the spacecraft, it is not properly formatted so that it can be radiated through the Deep Space Network (DSN). The COMMAND software formats each SCMF and produces a CMD_DSN file.

As was the case with the PST script, the MCT script checked the COMMAND runlog for errors encountered during execution. Any errors detected in the runlog would cause immediate termination of the script and a failure message, containing file name and failure details, to be sent by e-Mail to the MCT member responsible for running the script. The

MCT member would then determine which was the best resolution of the error. At the discretion of this MCT member, this may include rejection of the file or contacting the PST or requester to help in correction of the error. In any case, an erroneous file was not guaranteed same day readiness for transmission to the spacecraft. If no errors were found during the above check, then the MCT script would queue the CMD_DSN for radiation to the spacecraft at the time determined by the uplink window computation software above.

Upon successful completion of all preceding steps in this script, it would notify the responsible MCT member that the file had completed processing and would automatically write the CMD_DSN to the PDB for archival purposes.

The final step of MCT processing would be carried out by the responsible MCT member (not the script). This would be the notification of the requester by e-Mail that the file completed processing and was queued for radiation. This e-Mail message contained an MCT FRF. This FRF was formatted in a specific way and contained information which unambiguously identified the CMD_DSN file. The MCT repeated the above steps for each file for which an FRF was received from the PST, until all files submitted for that day had been processed.

APPLICATION OF RESULTS TO FUTURE FLIGHT OPERATIONS

The results of the Uplink Process Action Team promised broad application to other non-stored processes used by Mars Observer as well as to other JPL flight projects, both current and future. In fact, experience from Mars Observer indicates that risk is actually reduced when these types of commands are not scrutinized but rather the process by which they are

generated is scrutinized and verified and then is automated in such a manner as to prevent circumvention unless approval is given.

In general, present missions can benefit from these results by scrutinizing and analyzing their processes and identifying all unnecessary (little or no value added) 'human interaction' steps. These steps should then be eliminated if possible or automated when still needed. Prime candidates for this type of automation would include checking of printouts for errors and 'checking' of paper forms for errors. The latter of these two items represented an enormous amount of time spent by managers on MO which slowed down the process. Few if any errors of these types were ever encountered for the NINSCs processed.

Future missions can benefit from this effort by accepting the precept that rigorous analysis of processes and automation of these processes leads to increased efficiency and, hence, either increased productivity or decreased staffing levels. Mitigation of risk is accomplished by scrutinizing and validating the automation tools before they are used in operations. In the case of Mars Observer, the tools in question had been used in actual flight operations for several months and had been well validated. In addition, the team procedures used to define the NINSC process had been well practiced and, when necessary, modified or corrected to eliminate error sources. Finally, the tools used in this processing had been developed in a 'modular' sense and to allow command line control of all software elements. These two characteristics of the software permitted the operations teams to modularize their procedures and break them down into easily understood and automated functions.

REFERENCES

Planetary Exploration Through Year 2000, A Core Program: Mission Operations, A Report by the Solar System Exploration Committee of the NASA Advisory Council, U. S. Government Printing Office, 1986.

Varghese, Philip (Team Lead), et al, Final Report of the Mars Observer Uplink Process Action Team, Mars Observer Project Document, 19 November 1993

Quality Action Teams, Workbook for Team Members, Organizational Dynamics Incorporated, 1990.

SCL: An Off-The-Shelf System For Spacecraft Control

By: Brian Buckley & James Van Gaasbeck
Interface & Control Systems, Inc.
1942 South Dairy Road
Melbourne, FL 32904

Abstract

In this age of shrinking military, civil and commercial space budgets, an off-the-shelf solution is needed to provide a multi-mission approach to spacecraft control. A standard operational interface which can be applied to multiple spacecraft allows a common approach to ground and space operations. A trend for many space programs has been to reduce operational staff by applying autonomy to the spacecraft and to the ground stations.

The Spacecraft Command Language (SCL) system developed by Interface and Control Systems, Inc. (ICS) provides an off-the-shelf solution for spacecraft operations. The SCL system is designed to provide a hyper-scripting interface which remains standard from program to program. The spacecraft and ground station hardware specifics are isolated to provide the maximum amount of portability from system to system. Uplink and downlink interfaces are also isolated to allow the system to perform independent of the communications protocols chosen. The SCL system can be used for both the ground stations and the spacecraft, or as a value added package for existing ground station environments.

The SCL system provides an expanded stored commanding capability as well as a rule-based expert system on-board. The expert system allows reactive control on-board the spacecraft for functions such as Electrical Power Systems (EPS), thermal control, etc. which have traditionally been performed on the ground. The SCL rule and scripting capability share a common syntax allowing control of scripts from rules and rules from scripts. Rather than telemeter over-sampled data to the ground, the SCL system maintains a database on-board which is available for interrogation by the scripts and rules. The

SCL knowledge base is constructed on the ground and uploaded to the spacecraft.

The SCL system follows an open-systems approach allowing other tasks to communicate with SCL on the ground, and in space. The SCL system was used on the Clementine program (launched January 25, 1994) and is required to have bi-directional communications with the Guidance, Navigation and Control (GNC) algorithms which were written as another task. Sequencing of the spacecraft maneuvers are handled by SCL, but the low-level thruster pulse commands are handled by the GNC software. Attitude information is reported back as telemetry, allowing the SCL expert system to inference on the changing data. The Clementine SCL Flight Software was largely re-used from another Naval Center for Space Technology (NCST) satellite program.

This paper will detail the SCL architecture and how an off-the-shelf solution makes sense for multi-mission spacecraft programs. The Clementine mission will be used as a case study in the application of the SCL to a "fast track" program. The benefits of such a system in a "better, cheaper, faster" climate will be discussed.

Introduction

In 1988, the Naval Center for Space Technology (NCST) and Interface and Control Systems, Inc. began development of a spacecraft controller for a "black" program. Due to the political climate at the time, the requirement was levied for 180 days of autonomous operation for the satellite. Since then, the politics have changed, but the system which was designed and prototyped showed a great deal of promise and was funded for development even though the 180 day autonomy requirement was discarded. ICS has evolved the concept for Spacecraft Command Language (SCL) over the years and has developed a spacecraft flight control system which is

innovative in its approach to ground and space standardization.

The SCL system provides an embedded control system software package for the spacecraft which uses a rule-based expert system. This A.I. technique was prototyped and found to be awkward to use for the day to day operations of a satellite. We found that adding a high-level scripting capability integrated with forward chaining rules provides a powerful alternative to the traditional approach to spacecraft command and control. The SCL system is based on a Hyperscripting language which can be extended to meet the mission unique aspect of each spacecraft. The added benefit of this system is that it can be run on workstations to control the ground station mission operations. The system is designed to be portable to a wide variety of workstation-class computers and drives third party graphics products to provide a visualization interface. The SCL system is normally used as the integrating factor for ground stations. The SCL system is used to sequence operations and control other software packages, both custom and Commercial Off The Shelf (COTS).

The SCL concept is based on the unification of ground and space with the same control system. The SCL system provides a flight control system with an on-board database allowing scripts and rules to have visibility into on-board data samples. The workstation version of the SCL system can be applied from board level checkout up through mission operations. The SCL system not only allows re-use of the underlying control system throughout the phases of satellite development, but also allows re-use of the scripts and rules which are developed to configure the spacecraft. The SCL scripts and rules can be developed and tested in early phases of I&T and stored in a repository for use throughout the life of the spacecraft. This aids in the Configuration Management of "trusted" sequences for spacecraft configuration. These trusted sequences can be managed with a software configuration management tool and referenced throughout the development cycle of the spacecraft program. High-level mission tasking sequences can build upon underlying configuration scripts and rules.

Domain experts can be interviewed and knowledge of system operation can be captured in the form of rules. These rules can be used to build up a simulation of the system and develop Fault Detection Isolation and Recovery (FDIR)

scenarios. The SCL toolkit includes an Integrated Development Environment (IDE) which can be used on a desktop computer to prototype and test control algorithms. The event-driven nature of the SCL system makes it ideal for FDIR scenarios. Interviewing domain experts early in the project allows knowledge to be captured thereby reducing the effects of the brain-drain when key personnel leave the program.

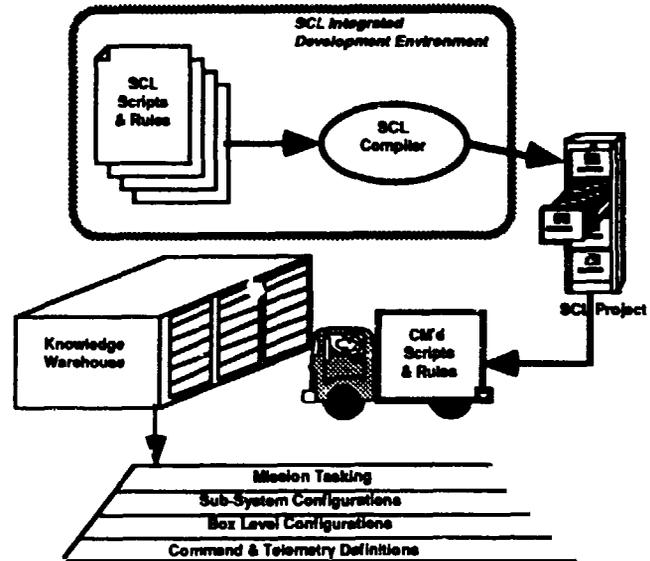


Figure 1. Approach to Knowledge Re-use

The Clementine spacecraft is a system which validates the SCL concept. The SCL software was originally developed for the NCST Advance System Controller (ASC) program. The SCL software and the flight controller and memory cards were re-used for the Clementine mission. Control loops for the attitude control system were analyzed and found to be appropriate for development in a native language rather than SCL.

The Clementine Guidance, Navigation and Control (GNC) software was developed in C and integrated as another task in the real-time operating system. The SCL system was tailored to allow bi-directional communication between the SCL and GNC software. This capability was eventually used to perform automated mapping of the moon. Information from the GNC system was used to sequence the commands required to configure the payload and collect image data.

Another program has recently benefited from the SCL software. The Environmental Research

Institute of Michigan (ERIM) has developed two systems using the SCL system. The Autonomous Rendezvous and Docking (ARD) mission was to explore the feasibility of spacecraft performing autonomous docking maneuvers and fluid transfer experiments. The ARD system was developed using SCL running on a single board computer interfacing to a series of I/O cards. The ARD payload passed acceptance tests and was integrated with the satellite bus, docking subsystem and fluid transfer systems. However, ERIM has removed the ARD payload from the Commercial Experiment Transporter (COMET) and Conestoga launch vehicle manifest. ERIM is currently searching for alternative launch vehicles.

ERIM reused the flight controller hardware and software design from ARD for the Robotic Material Processing in Space (ROMPS) experiment. ROMPS is a Space Shuttle Getaway Special (GAS) experiment which is manifested for flight on STS-64 on September 10, 1994. The ROMPS flight system was largely based on the ARD system with modifications to the low-level software. The ROMPS mission is to perform semiconductor annealing experiments in a microgravity environment. The system will control a robot and an annealing oven. This program has used SCL as part of a low-cost ground station. SCL is used in conjunction with National Instruments LabViews on Macintosh systems. LabViews provides a graphics engine used for visualization of data by the SCL system.

In recent years, SCL has gained a great deal of attention due to the desire for standardization of spacecraft control systems. The SCL system is portable to most embedded microprocessor platforms and operating systems. The underlying messaging system used for the uplink and downlink protocol is isolated from the SCL system. The Clementine system used the CCSDS communications standard although most missions have unique protocols. The AIAA, JPL, the Air Force and NASA have been looking closely at SCL as a basis for a standard on-board system architecture. The fact that SCL can be used on the ground also has added benefits in a "better, cheaper, faster" environment.

The answer to better, cheaper, faster lies beyond the Clementine mission. The Clementine mission was a high-risk, fast-track mission which went from vapor-ware to hardware in roughly two years. A new management approach and many

innovative steps were taken along the way. Traditional or "old guard" methods were sidestepped to meet the aggressive schedule and budget.

Its a money thing

Support for a standard operational approach to spacecraft control is spawned by shrinking budgets. Today's tight budget situations don't allow for fresh starts; millions of dollars can be spent replicating existing technology. Systems such as SCL allow for multi-mission application of the same control system. This standardization reduces software development, training and maintenance costs. Ground stations can be retrofitted to support existing satellites with a higher-level system which supports advanced automation features. Operator workload can be reduced, and advisory systems can be developed using the Expert System which is incorporated in the SCL system.

The key to developing a new software approach is to invest in technology which embraces an Open System architecture, industry standards, and allows room for growth. New technologies can be merged in as appropriate and existing code can be replaced with COTS products. COTS solutions will reduce the development and maintenance costs since they can be spread across a customer base. New technology can be phased-in by using a value-added approach. Older, high-maintenance code can be retired as confidence grows in the newer system.

Training is important. Getting the day-to-day users of the system up to date on the nuances of the system will improve productivity and allow exploitation of the new capabilities of the system. Experts can provide help in choosing the best alternative for implementation of requirements. The experts need to be brought in at the beginning when the Systems Engineering is being done. Too often, a system is force fit into an existing design when it could have been engineered into a more elegant solution.

Below you will find a description of a system which has taken this approach. The Clementine spacecraft as well as the ROMPS GAS experiment have developed operational concepts around the SCL system.

SCL System Architecture

The SCL system consists of five major components:

- The database describes digital and analog objects that represent spacecraft sensors and actuators. The latest data sample for each item is stored in the database. The database also contains derived items that are artificial telemetry items whose values are derived from physical sensors. Examples of derived items could be: average temperature, power based on current and voltage monitors, subsystem status variables, etc. These database objects include command actuators for commanding the spacecraft systems.
- The development environment is a window based application that includes an integrated editor, the SCL compiler, decompiler, cross-reference system, explanation subsystem, and filing system. The development environment is also used as a front-end to control the SCL RTE. A command window is used to provide a command-line interface to the Real-Time Executive (RTE). Extensive use of pull down menus and dialogs are used to control the system.
- The RTE is the portable multi-tasking command interpreter and inference engine. This segment represents the core of the flight software. This portion of the software is available in both C and Ada to allow ease of porting to a specific hardware platform (ground or space).
- The Telemetry Reduction program is responsible for filtering acquired data, storing significant changes in the database, and presenting the changing data to the Inference Engine. Limit checking and engineering unit conversion can be enabled on a point by point basis.
- The project is the collection of SCL scripts and rules that make up the knowledge base. On ground based systems, the project contains an integrated filing system to manage the knowledge base. In the space environment, the binary knowledge base is uploaded to the spacecraft and stored in memory.

Depending on the needs of the user, all components of SCL can be run on a single system, or may be distributed among systems.

The development environment can be used to directly connect to a local or remote version of the SCL RTE. This direct connect capability is also supported for the space segment to allow interactive commanding of the system.

The Clementine Experience

The Clementine management approach was to have a team of engineers to work on the project from cradle to grave. There would be no transition from one team to another. The Clementine team was a talented group of young, motivated engineers. The team had experience on other satellite programs, but was young enough not to be jaded by many of the large DoD and NASA programs. The team made numerous personal sacrifices for an opportunity to shake up the satellite community.

Clementine Command And Telemetry Software

The Clementine system software introduced several new concepts to spacecraft command and telemetry processing. These concepts supported the rapid development of the Clementine flight software. Most of these innovations are generic in nature and can be applied to other spacecraft. The following paragraphs will briefly describe Clementine's command and telemetry software and will highlight some of the innovative aspects of the software.

Clementine Command Processing

The command processing software performs four functions: (1) synchronize and reassemble incoming command data words into command packets; (2) verify and authenticate the command packets; (3) dispatch complete command packets to destination tasks; (4) execute command processing control functions. Clementine commands and data are formatted as packets with a header that includes a word count, a routing code, and a secondary identifier. The command processing software receives these packets as a stream of 16 bit words. The command software reassembles a packet from the incoming data words and after verifying and authenticating the packet, passes it to the operating system through a function call. The operating system software delivers command packets to queues that are assigned to software tasks. This arrangement

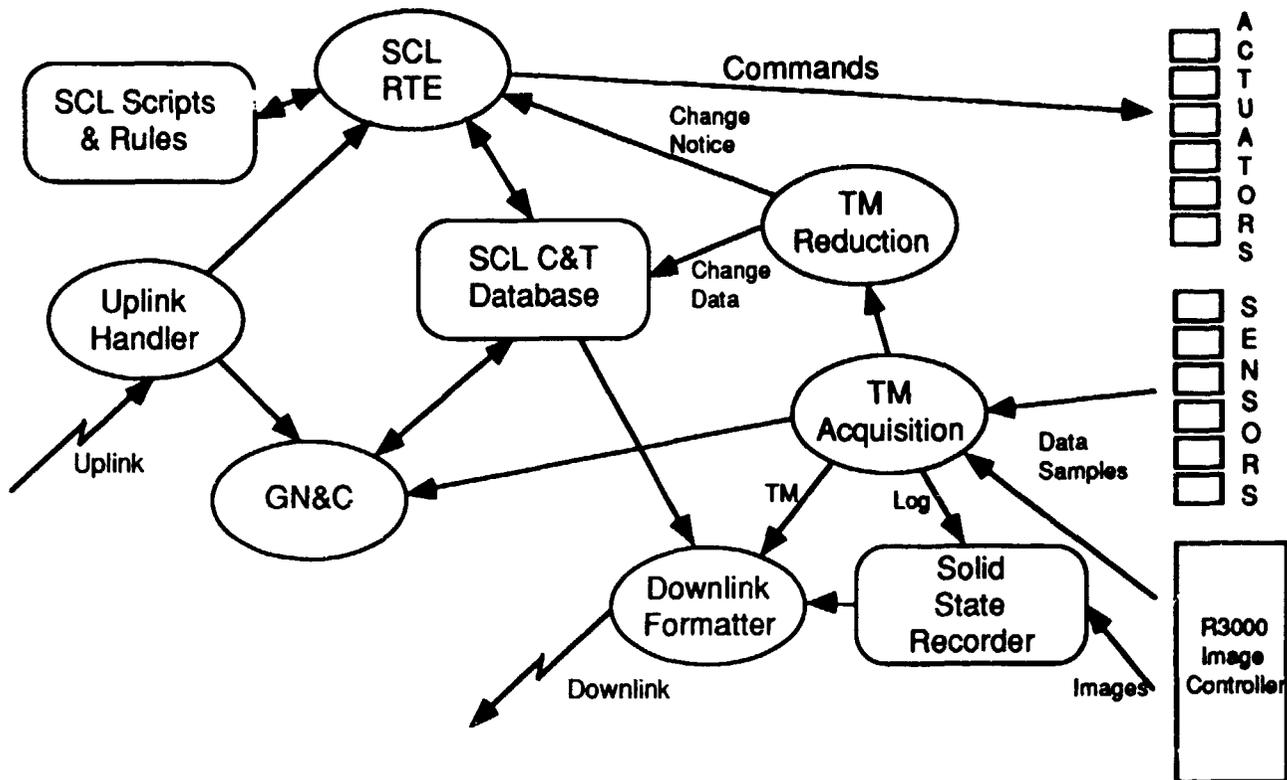


Figure 2. Clementine Flight Software Architecture

distributes the responsibility for command execution among the various Clementine software tasks. Because command execution is distributed to other tasks the Clementine command processing software is simply an input task which forwards messages to other tasks.

The packetized command uplink simplified the design of the command processing task and supported the rapid development and integration of Clementine's entire flight software system because:

1. Only the packet header is fixed, the remainder of the packet is defined by the receiving task. This allowed software designers the freedom to specify command formats that were suited to their requirements.
2. It simplifies the integration of software modules, such as the SCL Real Time Engine, that were reused from other programs. SCL software relied on command formats and interfaces that were defined long before the Clementine program was initiated.
3. New software tasks are added to the system without impacting the command processing

software. A new task only has to create a command queue and then register to receive command packets through the queue. This simplified the incremental build-up of the flight software.

4. Command packets can be rerouted by changing the routing code or by altering the operating system's routing tables. This capability was used operationally to support some of the processor failure modes.
5. New commands can be defined without impacting the command processing software. This supported the incremental build-up of the flight software.

Clementine Telemetry Processing

The Clementine telemetry processing software performs four major functions: telemetry acquisition, telemetry reduction, telemetry distribution, and telemetry logging. All of these functions are implemented by a single software task. The telemetry task operates in one of two modes: bypass or DHU. When in the bypass mode, the telemetry task is responsible for formatting telemetry data into telemetry frames in addition to the four functions listed above. When

it is in the DHU mode the telemetry task transmits its telemetry data to the Data Handling Unit (DHU) which then is responsible for formatting the data into telemetry frames. The process that is responsible for formatting the telemetry frames is responsible for transferring the frame data to the downlink hardware interface.

All of Clementine's telemetry, except for images dumped from the Solid State Data Recorder (SSDR), is organized into packets. Clementine telemetry frames are not filled with commutated data, as is the general case for spacecraft. Instead, Clementine's telemetry frames serve as a transport mechanism for the telemetry packets. The telemetry packets are used to transport the spacecraft's housekeeping, status, and memory data.

Commutated telemetry frames do provide a consistent source of data where housekeeping measurements and status items such as temperatures, voltages and relay indicators are concerned. To fill this need for consistent engineering data, the Clementine telemetry system provides a packet that contains synchronous, commutated data. The content of the Housekeeping telemetry packet, or HK packet, is defined by an uplinkable commutation format. The commutation format specifies the order in which the various spacecraft engineering sensors are sampled. As many as four commutation formats can be stored in the spacecraft memory and any one of the four formats can be active. The Clementine commutation format supports up to 16 variable length minor frames per master frame and subcommutated telemetry items up to 16 deep.

Telemetry Acquisition

The telemetry acquisition function is responsible for processing the commutation formats. The acquisition function processes the format information, building up a minor frame in a temporary buffer. When the minor frame is complete, the acquisition function formats the frame into a single HK packet and then transfers the HK packet to the distribution function. The acquisition function can be commanded to switch to another of the four possible formats at any time and it will begin processing the new format after the current frame is complete.

The acquisition function is also responsible for acquiring packetized telemetry from other software tasks. An example of such a packet is the Attitude packet produced by the Attitude Determination and Control (ADAC) task. This packet contains information that defines the spacecraft's current attitude along with rate and status information. The acquisition function acquires these packets through message queues which it creates and manages. Two queues are created: the critical queue and the normal queue. The critical queue is for status information that is vital to the operation of the spacecraft such as the current vehicle command count, telemetry processing state and the spacecraft time. The normal queue is for all other telemetry packets.

Telemetry Reduction

The telemetry reduction function is responsible for maintaining the current telemetry value database that is provided to support the Spacecraft Command Language interpreter and inference engine. The reduction function receives identified telemetry values from the acquisition function and processes the values before updating the current value database with the telemetry values. Two of the processing options performed by the reduction function are change detection and engineering unit conversion.

The telemetry reduction function also allowed packets of data to be decommutated on-board to allow the SCL script and rules to have visibility into on-board data samples. This was a leap forward from traditional spacecraft software designs.

Telemetry Distribution

The distribution function is responsible for prioritizing and distributing the telemetry packets to the downlink. Packets from the Critical queue are assigned the highest priority and are distributed ahead of all other TM data. The HK telemetry packets are next in priority and packets from the Normal queue are assigned the lowest priority. If the distribution function is operating in the DHU mode, the function transfers the packets in priority order to the DHU through a dual port RAM buffer. The DHU is responsible for inserting the individual packets into the telemetry frame when operating in the DHU

```

-----
-- SEP_DETECT -- Rule to detect separation from Titan II second stage.
-- Schedules separation operations script...this is the
-- initial sequence of events for Clementine
-----

rule      sep_detect
subsystem dspse
category  launch
priority  30
activation yes
continuous yes
  if LVSEPIN1 = SEPARATD and LVSEPIN2 = SEPARATD then
    deactivate sep_detect  --make this rule dormant
    -- establish attitude, take Star Tracker cal. shots
    execute LEO_Sep_OPS in 1 second
  end if
end      sep_detect
-----

-- LOWVDET -- detect low voltage & schedule the safing script
-----

rule      lowVdet
subsystem eps
category  batteries
priority  4
activation yes
continuous yes
  if rawvalue of BATTPMON <= 360 then
    deactivate lowVdef  --make this rule dormant
    execute LEOfsafing in 1 tick
  end if
end lowVdet
-----

-- LEOfsafing -- script which safes the spacecraft
-----

script LEOfsafing
  set DHUSELNO          -- Take no pics
  execute ReactWheelsOff -- RWs off
  set GNC11_ALLSTOP    -- stop all S/C rotations
  set SWCRITE2         -- Image Processor Off
  execute IMUstop      -- IMUs off
  execute TrackersOff  -- STs off
  execute ACSDisable   -- Turn off ACS
  execute CamsOff      -- Cameras off
  -- check if star tracker doors are open...if so close
  if STARAOPN = 1 and STARBOPN = 1 then  -- Close both doors together
    execute ActBothDoors
    execute ACSDisable in 180 seconds
  else
    if STARAOPN = 1 then  -- Close A Only
      execute ACTSTA
    end if
    if STARBOPN = 1 then  -- Close B Only
      execute ACTSTB
    end if
    execute ACSDisable in 180 seconds
  end if
  wait 1 second
  set SWCRITEE          -- Transmitter Off
end LEOfsafing

```

Figure 3. Example of Clementine Scripts & Rules

mode. If the distribution function is operating in the Bypass mode it is responsible for inserting the individual packets into the telemetry frame.

Telemetry Logging

The telemetry logging function is responsible for storing a time history of selected telemetry items in a log file on board the spacecraft. The purpose of the log file is to provide a means of capturing and storing telemetry data on board the spacecraft during periods when the spacecraft is unable to communicate with its ground stations. The log can be dumped by ground command or stored command. When the log is dumped, the log records are formatted into telemetry packets and transferred to the telemetry distribution function.

The log file can reside in either the HKP processor's RAM or on the SSSDR. The log can be maintained in either stop on full format or in a circular format where new telemetry values overwrite the oldest values once the log becomes full. Telemetry items are selected for logging by ground command or by stored command.

The log file is maintained in a change only format, that is, the telemetry items that are selected for logging are first processed by the telemetry reduction function to determine whether the value of the item has changed since the last time it was acquired. If the item did not change it is not stored in the log. If the item did change a record containing a time stamp, the item's identifier, and the item's new value is written to the log.

The logging function is designed to initialize the log with the current value of all items that are selected for logging when the log is created or whenever it is reinitialized. This feature establishes a baseline for the change only values that will subsequently be written to the log.

Telemetry Processing Summary

The Clementine telemetry software introduced several new ideas to spacecraft telemetry systems. These innovations made significant contributions to the rapid development and integration of the Clementine flight software and contributed to the efficient operation of the spacecraft. The innovations include:

1. A packetized telemetry downlink which provides for synchronous, commutated data acquisition.
2. The capability to store multiple telemetry commutation formats on board.
3. The ability to load new HK packet commutation formats from the uplink.
4. An on-board telemetry storage log that is filled with change only telemetry.
5. An on-board telemetry reduction process and current telemetry value database.

Lessons Learned

The SCL system started life as a prototype system which supported only rule-based processing. It became obvious that it would be cumbersome to apply a strictly rule-based system to spacecraft command and control. ICS added the scripting capability to SCL to support procedural, time-based commanding scenarios. The scripting capability was integrated with the rule-based capability so that the system shared a common syntax and command interpreter. The SCL scripting capability is analogous to the Command Storage Memory (CSM) on earlier spacecraft. The SCL scripts and rules share a common Hyperscripting grammar. The system was developed in a manner to allow a core set of directives to be supported, and allow the user to extend the grammar with a mission unique set of directives. The SCL compiler used in the ground development environment allows addition of keywords, and the Real-Time Engine (RTE) can be extended to support the new features at run-time.

The Clementine flight software team was made up of several companies which worked together (around the clock at times) to develop an integrated system. The companies that developed the flight software also developed the ground station software together. This allowed interfaces to be defined more easily and consistently. The relatively small team worked to our advantage since all the players knew each other by name and could interact and make decisions quickly. There were very few managers to interfere with the decision making process. The NRL management "rode herd" over the engineers and coordinated the efforts. The team was able to work together without corporate or political fences.

The engineers that performed the systems engineering also developed the ground and flight systems and flew the spacecraft during mission operations. The cradle to grave philosophy allowed for a consistent interface between engineers. Day to day interaction between companies maximizes progress in the fast-paced development environment. Not having to transition the program from one group to another resulted in a substantial time and cost savings. The engineers who were intimate with the subsystem designs were responsible for the day-to-day tasking of the satellite. This allowed for experts to be available virtually anytime a problem arose.

The development and integration of the software was compressed into a short period of time. If a development testbed for the flight software had been available months sooner, a greater level of testing could have been accomplished. As it was, we had to schedule time at two sites: the testbed, and the flight article. It wasn't unusual to have around the clock and weekend testing, especially towards the end of the schedule. Competition for the testbed was at the point that one company would jump on while another pulled back to correct a software bug. Hardware bugs which took the system down were devastating. Software simulators for testing the Attitude Determination and Control system were refined throughout the life of the program. These simulations, along with the Guidance Navigation and Control (GNC) flight software evolved throughout the mission. New code was uploaded to the spacecraft to handle the current phase of the mission. Code which handled earth orbit was obsolete for handling lunar orbits, etc. This made for an incremental development, test, and operate cycle for the GNC code. This cycle worked out quite well because of the high fidelity of the simulators which were produced.

With the fury of software development activity on a day to day basis, configuration management was a monumental task. At times, 3 shifts a day were modifying and testing code. The ground software was evolving as quickly as the flight software. Two testbeds needed to be kept in vnc. Two accounts were maintained on the testbed minicomputers. One was the operational account, and the other was the development account. Each shift was informed as to which account was the current operational account and

what modifications had been made. Coordination and attention to detail was mandatory.

The tight schedule and late availability of the flight unit and testbed caused a compression of the test schedule. Because the Clementine sponsors were willing to accept some risk, hardware and software testing was limited. The orbital mechanics of the asteroid encounter required that the spacecraft be launched at a given date otherwise the opportunity would be missed. The level of fault tolerant design for both hardware and software are some of the tradeoffs which had to be made. A single string system was acceptable. The software was designed to make up for some of the hardware shortcomings.

Once operational, the team faced new problems. The one glaring problem was burnout of the players. During the first week of operation, many of the team slept on conference room tables, in chairs, and on the floor (in the winter in Alexandria, Virginia). Several people were told to go home or to their hotels to get some sleep. The dedication of the team members was exemplary. Many hadn't seen their families in many weeks, even then it was only for few days.

On-board operations proved to be tricky at times. Many members of the team had developed satellites in the past, but had not dealt with the fact that they must monitor and command the vehicle around the clock. The team tried to keep a two day cushion on the mission tasking. This would allow for light duty on weekends and allow for problems to be investigated without the threat of a missed pass looming over their head.

The set of software tools were limited. No mission planning system existed. Many of the passes followed a standard script: configure the payload, slew the spacecraft, collect images to the solid state recorder, turn off the payload, wait until the earth is in view, slew the spacecraft and dump the solid state recorder. The commands for these scenarios were entered into Microsoft Excel with formulas for the times. The orbit analysts would determine the start time of the scenario and a time could be filled in on the spreadsheet and all other times calculated as an offset from the start time. The command sequences were moved over to the microVAX computers and translated into an upload sequence by the SCL software.

The configuration management of the spacecraft flight software and mission tasking sequences

was limited to a CM tool and a log book. The spacecraft would occasionally have a system reset which would require that all code patches be uploaded and a new set of tasking sequences be uploaded. This had to be managed by hand and was subject to human errors. It would have made a great deal of sense to have a software tool to automate the management of the flight software and mission tasking.

The lesson that Clementine points out is that code re-use is a key factor in the success of a fast track program. Having a flexible architecture which can be applied from one program to another allows for substantial time and cost savings. The Clementine flight software and ground system software relied heavily on software which had been developed for other Naval Center for Space Technology programs. These systems along with other software originally developed for NASA were merged along with some new concepts to develop a flexible command and telemetry system which itself can be used on other programs.

Conclusions

The SCL system has proven that a re-usable system can be successfully used for spacecraft command and control. All of Clementine's requirements were met with the exception of the asteroid flyby. The SCL system also helps promote a standard interface for the many facets of ground and space. The ROMPS Space Shuttle GAS experiment is due to fly in September of 1994. The ROMPS flight will lend further evidence that a multi-mission control system can be deployed. Although ARD and ROMPS are dissimilar missions, they are using the same hardware and software design. There is already talk of another flight for the ROMPS experiment.

The Clementine flight control software is general purpose in nature and can easily be adapted to other programs. The SCL system itself is being commercially marketed for workstations and embedded systems. The SCL system also has applicability to industrial control systems, Intelligent Vehicle Highway System (IVHS), medical, petrochemical and power system industries.

JPL, the AIAA and the Air Force have used the SCL system as a basis for standardization efforts. ICS is presently on several technology steering committees for DoD and Civil space. The SCL system sets a benchmark against which other

systems will be compared. The flexibility of the architecture and the open systems aspect of the SCL software gives the system broad appeal.

The system does introduce information management problems that are overcome by software tools and a disciplined approach to configuration management. This approach must also extend to the distribution of flight and ground databases and knowledge bases. The system is several years into its development, has been the subject of numerous proofs-of-concept, and is in use at several sites. The SCL system provides a low-cost, low-risk solution for many of today's command and control environments.

The success of the Clementine mission lies with the dedication of the team members and their talent in the development of a spacecraft in a surprisingly short period of time. They were able to draw on experience and re-use software and hardware designs from other programs to develop a system which is flexible and pushes the state of the art in software technology for spacecraft.

ORIGINAL PAGE IS
OF POOR QUALITY

MISSION OPERATIONS DATA ANALYSIS TOOLS FOR MARS OBSERVER GUIDANCE AND CONTROL

Edwin P. Kan

Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, Ca 91109

ABSTRACT

Mission operations for the Mars Observer (MO) Project at the Jet Propulsion Laboratory were supported by a variety of ground data processing software and analysis tools. Some of these tools were generic to multi-mission spacecraft mission operations, some were specific to the MO spacecraft, and others were custom tailored to the operation and control of the Attitude and Articulation Control Subsystem (AACS). The focus of this paper is on the data analysis tools for the AACS. Four different categories of analysis tools are presented; with details offered for specific tools. Valuable experience was gained from the use of these tools and through their development. These tools formed the backbone and enhanced the efficiency of the AACS Unit in the Mission Operations Spacecraft Team. These same tools, and extensions thereof, have been adopted by the Galileo mission operations, and are being designed into Cassini and other future spacecraft mission operations.

INTRODUCTION

From launch on Sept. 25, 1992 to Aug. 21, 1993, three days to Mars Orbit Insertion, the Mars Observer (MO) Project was supported by an efficient mission operations Spacecraft Team. This team was made up of units of engineers / analysts with cognizance over the different functions/ subsystems of the spacecraft, including the Attitude and Articulation Control Unit (AACS), Information Unit, Flight Software Unit, Telecommunication Unit, Power Unit, Thermal Unit, Propulsion Unit and Systems Unit.

The Spacecraft Team was responsible for the (i) monitoring, (ii) analysis, and (iii) command & control of the spacecraft. Monitoring included real-time watching of spacecraft telemetry; depending on the criticality of the activities, 3-shift 24-hour operations were often required. Off-shift (and also prime-shift) monitoring and command radiation was always done by another around-the-clock team in the Flight Operations Office.

Analysis consisted of real-time, near real-time, and off-line analysis. This included routine data analysis, spacecraft characterization, health/welfare tracking of the spacecraft and spacecraft hardware, incident/surprise/anomaly data analysis, contingency planning, update/design/development/testing of ground software as well as of flight software.

Command & control was equated with the on-ground planning and execution of preplanned sequences of discrete commands, that were uplinked ahead of time, or uplinked in real-time. These commands and controls were mostly high-level, even though there were times when discrete single-event commands were uplinked, such as hardware turn-on, turn-off and mode-transitions. For an advanced, autonomous spacecraft like MO, the real-time sub-second and sub-milli-second control was naturally executed on-board under the flight software control. In the present context, the sequence design and analysis activities leading to the single and/or sequence command, were considered to be "command & control" activities.

In AACS mission operation, monitoring, analysis, and control/command activities were dedicated to the following major task areas:

- periodic parameter/catalog updates
- routine health and state monitoring, tracking and trending
- maneuver (ΔV) design, and post-maneuver analysis
- hardware calibration design, and post-calibration analysis
- science experiment/sequence design, and post-sequence analysis
- spacecraft event/sequence design, and post-sequence analysis
- real-time operation and support (including maneuvers, calibration sequences, science sequences, command & sequence uploads, flight software uploads, subsystem/spacecraft activities)
- flight control software updates
- testing and verification
- anomaly investigation
- nominal inter-subsystem coordination, planning and interface
- project level coordination, interface and reporting.

Automated tools were indispensable for the monitoring, analysis, command & control of AACS. Tools included displays, list pages, graphical plots, statistics charts, computer programs, data retrieval software, data formatting software, data packaging software, data generation software, data analysis software, mathematics libraries, special graphics analysis tools, command and sequence generation programs, controls simulation software, and spacecraft system simulation/verification test hardware/software (laboratories).

Due to human resource limits, the need for quick turn-around time, and more importantly, the requirement for consistency and correctness of the products, numerous analysis tools were developed. Some tools were general-purpose, and some were custom-tailored to AACS. This paper will categorize and describe these analysis tools.

JPL's MGDS (Multi-mission Ground DATA System) - AMMOS

At the Jet Propulsion Laboratory (JPL), MGDS refers to the hardware/software that supports multi-mission telemetry processing and spacecraft commanding. MGDS consists of a network of workstation-class, multi-tasking computers using standard operating systems, software applications and tools. The overall operations, i.e. MGDS plus workforce, processes, procedures and facilities, constitute the Advanced Multi-Mission Operations Systems (AMMOS).

The MO AMMOS hardware and software system provided the integrated telemetry data retrieval, front-end processing, and archiving functions. While not attempting to describe the AMMOS capabilities which are described in detail in (ref. 1), this paper will highlight the customization of the AMMOS real-time on-line telemetry analysis tools for MO AACS mission operations.

AMMOS is an extension, improvement and modernization of the earlier JPL Space Flight Operations Center (SFOC) for space exploration missions including Voyager and Viking. The 1989 Magellan mission to Venus was the first JPL project to utilize AMMOS in its mission operations. MO, the 1992 mission to Mars, was the second one. Recently, the Galileo mission to Jupiter (launched in 1989) has been converted to AMMOS. Mars Global Surveyor to Mars 1996 and Cassini to Saturn 1997 will also utilize AMMOS.

MO DATA ANALYSIS TOOLS

To perform the activities of monitoring, analysis, and command & control for the major AACS functional tasks discussed above, four (4) categories¹ of data analysis tools were used for MO mission operations:

Cat A. Real-Time On-line Telemetry Data Analysis Tool

¹ The terms Cat A, Cat B, etc. should not be confused with the JPL formal terms of Class A, Class B, etc., referring to the space-flight qualification level of software.

- Cat B. Non-Real-Time Telemetry Data Analysis Tools
- Cat C. Non-Real-Time Data Analysis and Performance Evaluation Tools
- Cat D. Non-Real-Time Data Generation & Viewing Tools

Cat A refers to the processing of "live" real-time telemetry that was broadcasted by AMMOS for real-time monitoring.

Cat B refers to the "near-real-time" and "off-real-time" analysis. "Near-real-time" typically involved the retrieval of telemetry data as recent as a few minutes old. "Off-real-time" referred to the retrieval of "archived data" which are day-old, week-old or older.

Cat C refers to the design and analysis tools, mostly for the generation of design parameters, files, catalogs etc., which form part of the input files and parameter updates to be included in uplink commands and sequences.

Cat D refers to miscellaneous AMMOS processes for the data retrieval, extraction, packaging, reformatting, viewing, file generation, sequence generation with constraints checking, and miscellaneous workstation utilities.

(No attempt is made in this paper to discuss the spacecraft system and subsystem verification/test laboratory.)

Cat A Tools

Real-Time On-line Telemetry Data Analysis Tools were supplied by AMMOS on AMMOS workstations. Major tools were: DMD (Data Monitor and Display), MO_Browser, CV_ (Command and Verification)_Monitor.

DMD provides standard and customizable displays for users to view channelized engineering telemetry and other mission data. A number of display formats are available, including list pages, "printer pages", channel-vs-time plots, channel-vs-channel plots. A whole set of software modules caters to customized data unit expansion, alarm-alerting and display setup.

MO_Browser provides individual stream data viewing down to the bit level. This tool is designed and used by MGDS data analysts more often than by spacecraft mission operations engineers.

CV_Monitor provides real-time monitoring of real-time or sequenced commands. The MO flight software is designed to return verification messages upon the receipt of commands.

Cat B Tools

Most of the non-Real-Time Telemetry Data Analysis Tools were supplied by AMMOS, and the rest customized by AACS engineers.

AMMOS tools facilitate data retrieval, reformatting, plotting, statistical summary, and archiving. These tools include query2plot, ecsv2plt, ecsv2ctab, ecsv2drf, drf2ecsv, ecsv2sum, ecsvmerge, oplot, xvmpplot, ecsvview etc. "ecsv", "ctab", "drf" refers to the data format of comma-separated-value, column-table, and data-row-file files.

Typically, the above tools are PERL scripts. (PERL is an interpreted language with features very similar to UNIX C-shell commands.) A prespecified set of channels of data can be queried from the AMMOS Central Data Base or from UNIX files, after which the channelized data is run through DMD and then output to a file. The three ecsv, ctab, drf formatted files may be the end-products or may be further processed.

A library of mathematical functions is very handy for the post-processing of the drf files. Examples are normalization, quaternion manipulation, trigonometric functions, and coordinate transformation functions. Statistics can also be computed, merged with files of earlier dates, and archived for trending purposes.

Plotting routines includes oplot, xvmpplot and others. Multiple channels versus time, or channel vs channel can be plotted. A special tool is available to view the ecsv files in tabulated forms; this ecsvview tool also has editing and filtering capabilities.

A custom graphics program, the MOBALL², is a geometrical representation tool. MOBALL draws the celestial sphere with the view from outside the sphere looking in, where the MO spacecraft lies at the center of the sphere. J2000 coordinates are used to draw the latitudes and longitudes. Hitting the left, right, up, down arrow keys correspondingly change the view point.

MOBALL was frequently used for viewing the geometry of the MO spacecraft relative to celestial bodies, i.e. Mars, Jupiter, Earth, Sun and stars. A simple wire-frame model of MO at the center of the sphere offered good insight for spacecraft pointing design, instrument pointing occultation analysis, thermal protection pointing, celestial body motion analysis, and star field analysis.

Analyzing star fields was done weekly and sometimes daily, using MOBALL. AACS was designed with a Celestial Star Assembly to collect star crossings for the estimation of spacecraft attitude; star crossings repeated every 100 minutes, MO's spin period. Two star catalogs, one called "ANS"-pointing, and one called sun-pointing, were uploaded to the spacecraft every week. Star field plots on the MOBALL were indispensable tools for star tracking, particularly for the evaluation of loss-of-attitude anomalies.

Another set of custom data analysis tools is embedded with AMMOS DMD. "Derived channels" can be computed in real-time and triggered by their "parent channel". Examples of derived channels are "bit decomposition" of "digital state" channels; "mnemonics assignment" child-channels for numeric-state parent-channels; unit scaling upon trigger-state; coordinate transformation channel (from spacecraft body coordinates to J2000 coordinates); performance-index evaluation from a set of parent-channels. One very informative channel of the last type was: "quaternion_delta of SCP_in_control vs SCP_not_in_control". (The Standard Control Processor, SCP, was MO's flight computer.)

² MOBALL is a C-program written by S. Collins, a MO AACS engineer.

Cat C Tools

Non-Real-Time Data Analysis and Performance Evaluation Tools were analysis tools to generate design parameters, files, catalogs etc., usually included in uplink commands and sequences.

A Performance Analysis System (PAS) was developed by General Electric Astro-Space Division (GE-ASD), the MO spacecraft contractor for JPL. PAS programs include ephemeris generation, star catalog generation, momentum unloading prediction, roll angle optimization for solar panel pointing (e.g. during a maneuver with or without pitching), spacecraft mass change estimates caused by maneuvers, propellant consumption, thruster characterization, etc. These programs were designed to input data files in predefined formats and output data files in predefined formats, according to MO Project specifications. The intent was to combine the analysis and file generation into one "flight-certified Class A software" process.

PAS software runs on AMMOS workstations (UNIX platforms), with X window and Motif graphics package (for the Graphical User Interface), and interfaces with C and Fortran 77 modules.

GE-ASD also provided MOSIM, a controls dynamics and simulation software package. MOSIM has better dynamics simulation, but has slightly different fidelity as the spacecraft verification/test laboratory (VTL) simulation; in VTL, flight computer hardware and software are duplicated, with flight-like interfaces to spacecraft sensors and actuators. MOSIM is written in Fortran, and runs on a Macintosh computer; it runs faster than the real-time rate at which the VTL simulation runs. (VTL was also developed by GE-ASD.)

Customized database spreadsheet programs are part of the Cat C analysis tools. For MO maneuver analysis, a large EXCEL™ workbook was devised with five spreadsheets linked together. Spacecraft parameters such as thruster moment arm, engine I_{sp}, spacecraft mass and inertia properties, controller gains, desired delta_V, burn

times, etc. are strategically designed into "static" data blocks, input data entry cells, and output data cells. This process was to standardize and automate the frequent maneuver analysis (Mars Orbital Ops require maneuvers at 2-3 week intervals.)

Cat D Tools

Non-Real-Time Data Generation & Viewing Tools for data retrieval, extraction, packaging, reformatting, viewing, file generation include AMMOS programs: TOT (Telemetry Output Tool to query data), CDF_WOTU (Central Database Window-On-The-Universe: file retrieval and deposit), MO_GAP_VIEW (data dropout/ gap review), SOE_VIEW (Sequence-of-events viewing and editing), SCLK-to-SCET (spacecraft time conversion, etc. User friendly GUI's accompany these programs.

Sequence and command design tools include SEQGEN (sequence generation), MOCHECK (MO constraints and flight rules checking), SASFGEN (Spacecraft Activity Sequence File Generation), INCON and FINCON (incoming and outgoing spacecraft configuration listing, i.e. before and after a sequence), etc.

Tools & Analysis for Special Mention

Among the above data analysis tools, a few are worth special mention and illustration. Figure 1 shows the DMD "20-plots" page. Some seventy channels are grouped and color coded in this page of nineteen plots. With a time scale over the period of 100 minutes (the spin period of MO), a nominal signature on this 20-plot display was readily associated with a nominally behaving spacecraft. In fact, this was a daily monitoring and reporting tool!

One major feature that makes DMD such a powerful real-time and off-real-time tool is its capability to derive child-channels from parent-channel(s) in real-time. For AACS, 106 child-channels were derived from some 530 parent-channels³. Detailed designs are documented in the AACS Telemetry Dictionary (ref. 2).

³ These numbers apply to the "SCP-in-control"; similar numbers apply to the "SCP-not-in-control". There are extra telemetry for non-SCP data.

User friendly displays are indispensable particularly for real-time monitoring of a spacecraft as complex as MO, where multiple (hundreds of) hardware and software parameters had to be monitored. Man-machine interface techniques and human engineering skills used in display layouts, telemetry channel numbering, mnemonics design, and above all, channel grouping by functional groups and display "rooms" were the key to success in MO. The development of AACS Telemetry Dictionary (ref. 2) was instrumental to this design; a similar development (ref. 3) for the Cassini spacecraft also illustrates the methodology. Table 1 is an extract from the AACS Telemetry Dictionary.

Figure 2 illustrates MOBALL in a sequence design of the Thermal Emission Spectrometer (TES) instrument on MO. The spacecraft wire-frame model provides, among other analysis features, a visual representation of the spacecraft and the TES instrument pointing relative to the Sun and Mars. The sequence was designed to calibrate TES using Mars in its field-of-view, and was successfully executed on Aug. 1, 1993. Details of the sequence design and the use of MOBALL can be found in (ref. 4).

Star catalog generation (weekly) and analysis (weekly; real-time by-demand) were facilitated by star field maps and planet trajectory maps, drawn with MOBALL. Figure 3 shows such a star field map. The methodology in star field analysis can be found in (ref. 5). Also, during the last ten weeks of MO, after a flight software change providing the downlink of the identified stars in the Celestial Star Assembly star identification software, star sightings and identifications were analyzed and tabulated. The latter was meant to calibrate the Standard Star Catalog provided by Honeywell after all the 1801 stars in the catalog were all sighted

While due mention should be made to the Performance Evaluation System (PAS) and the spreadsheet rendition of the maneuver analysis tool, page limitation does not permit further discussion in this paper. More details could be found in (ref. 6).

CLOSING REMARKS

The Mars Observer AACS mission operation was greatly streamlined with the help of the analysis tools, and above all the methodology, discussed in this paper. Some of these tools were generic to JPL's AMMOS (Advanced Multi-Mission Operations System) spacecraft mission operations, and some were specifically tailored to the Mars Observer AACS. These generic tools, and extensions of the custom-tailored tools have been infused into, and are operational in the on-going Galileo mission. They are also being designed into Cassini and other future spacecraft missions.

Acknowledgment

This work was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under contract to the National Aeronautics and Space Administration.

REFERENCES

1. Anon., "Multimission Ground Data System User's Guide, Vol. 1 - 4," JPL Document D-6060, Jet Propulsion Laboratory, Pasadena, Ca., USA, Jan. 1994.
2. Kan, E. P., "Mars Observer - AACS Telemetry Dictionary," IOM# MO-SCT-93-077, Jet Propulsion Laboratory, Pasadena, Ca., USA, 2/4/93.
3. Kan., E. P., "Process and Methodology of Developing Cassini G&C Telemetry Dictionary," Proc. SPACEOPS 1994, 3rd Int. Symp. on Space Mission Operations and Ground Data Systems, held at Goddard Space Flight Center, Greenbelt, Md., USA, Nov. 14-18, 1994.
4. Kan, E. P., "Mars Observer - Thermal Emission Spectrometer Calibration (TES CAL) Analysis Methodology," IOM# MO-SCT-93-203, Jet Propulsion Laboratory, Pasadena, Ca., USA, 3/24/93.
5. Kan, E. P., "Mars Observer - Analysis Behind the Generation of Weekly Star Catalogs" IOM# MO-SCT-93-136, Jet Propulsion Laboratory, Pasadena, Ca., USA, 2/26/93.
6. Kan, E. P., "Mars Observer - Mission Operations Data Analysis Tools," presentation on 8/19/93, also IOM# MO-SCT-93-563, Jet Propulsion Laboratory, Pasadena, Ca., USA, 8/25/93.

Table 1. AACS Telemetry Dictionary - sorted by Functional Groups (excerpt)

S/S	Ch#	Related Channel (or Data Type)	NAME	Category	Row	List	vd1	Sats	Emcy	Mission	Engr	Children Channel #	Parent # (MuxAddr)	red1	red2	red3	red4
F	22	22	S/C_State	SC_RDN	1	20	20				m(FN)						
F	37	37	CIU_flag	SC_RDN	2	20	20	(F)	(F)	(FN)	(FN)	3119/20/27/34/etc					
F	3134	3134	Contingency_mode_SCP1	SC_RDN	3	20	20	(FP)	(FP)	(FN)P	(FN)P		37				
C	2	2	CIU_SCP_n_control	SC_RDN	4	20	20		112	4	0 5						
H	5	5	Telemetry_bit_rate	SC_RDN	5	20	20		X	X	X						
H	25	25	Telemetry_mode	SC_RDN	6	20	20		X	X	X						
V	52	52/53	Mission_phase	SC_RDN	7	20	20		112Pbu	18P	2P		C-53/5/7				
F	3113	3113	REDMAN_switch_status_cont	SC_RDN	8	20	20	(F)	(F)	m(F)*(N)	m(FN)	1103/04/10/13/etc					
F	3	3	ATT_State	AACSdm	1	17	17		(F)	m(F)*(N)	m(FN)						
F	3102	3102	Packed_Attitude	AACSdm	2	17	17		(F)	m(F)*(N)	m(FN)	81/84/75/85/99,4007					
F	4007	4007	New_Att_control_mode_entry	AACSdm	3	17	17		(FP)	m(F)P	m(FN)P		3102 b10				
F	1140	1140	STAREX_inertial_ref_estab'ed	AACSdm	4	17	17		(FP)	m(F)P	m(FN)P		3102 b 0				
F	427	427/8/9	CMD'd_rate_X	AACSdm	28	17	17				m(F)	m(FN)					
F	4093	4093	Sun_SUBM	AACSdm	29	17	17				m(FN)						
F	41	41	SS_State	AACSdm	30	17	17	(F)	(F)	m(F)	m(FN)						
F	3109	3109	STAREX_status_word	AACSdm	31	17	17				m(F)	m(FN)	V-700/ 7711				
F	843	843/ 76	Filter_Body_Quat_1	AACSdm	32	17	17		(F)	(FN)	(FN)	V520 33 500 94 760					
F	851	851/2/3	Filter_Body_Rate_1	AACSdm	36	17	17		(F)	m(F)*(N)	m(FN)						
F	875	875/6/7	Pos_Error_Angle_X	AACSdm	39	17	17	(F)	(F)	m(F)*(N)	m(FN)						
F	411	411/2/3	SYST_MOM_X	AACSdm	42	17	17	(F)		m(F)	m(FN)	V-535/6		12	0 0	0 0	-12
F	412	411/2/3	SYST_MOM_Y	AACSdm	43	17	17	(F)		m(F)	m(FN)	V-535		12	0 0	-0 0	-12
F	413	411/2/3	SYST_MOM_Z	AACSdm	44	17	17	(F)		m(F)	m(FN)	V-535/6		12	0 0	-0 0	-12
V	535	535	SYST_MOM_RSS	AACSdm	45	17	17	(FP)		m(F)P	m(FN)P		411/ 73		0 0	-0 0	
V	536	536	X+Z_MOM_RSS	AACSdm	46	17	17	(FP)		m(F)P	m(FN)P		411/73		0 0	0 0	
V	760	760	SCP1/2_deltaQUAT	AACSdm	47	17	17			(FP)m(N)F	(FN)P		FN-843 5	0.5	0.1	-0.1	-0.5
M	349	349	AGC downlink	AACSdm	\	\	\										
F	3064	3064	Maneuver_n_progress	deltaV	1	4,14	14		(FP)	m(F)P	m(FN)P		3102				
F	2047	2047	Maneuver_flag	deltaV	2	14	14			m(F)	m(FN)						
F	2048	2048	Maneuver_status	deltaV	3	14	14			m(F)	m(FN)						
F	3924	3924	Thrustor_Configuration	deltaV	4	14	14			m(F)	m(FN)						
F	430	430/1/2	DeltaV_X	deltaV	5	4,14	14			m(F)	m(FN)						
F	431	430/1/2	DeltaV_Y	deltaV	6	4,14	14			m(F)	m(FN)						
F	432	430/1/2	DeltaV_Z	deltaV	7	4,14	14			m(F)	m(FN)						
F	3177	3177/82/87/92	RWA_power_on_X	RWA	1	1,11	11	(FP)	(FP)	m(F)P	m(FN)P		1841/ 74				
F	3182	3177/82/87/92	RWA_power_on_Y	RWA	2	1,11	11	(FP)	(FP)	m(F)P	m(FN)P		1841/ 74				
F	4069	4069	RWA_configuration	RWA	5	11	11										
F	3178	3178/83/88/93	RWA_power_limited_X	RWA	6	11	11	(FP)	(FP)	m(F)P	m(FN)P		1841/ 74				
etc	etc	etc	etc	etc	etc	etc	etc	etc	etc	etc	etc	etc	etc	etc	etc	etc	etc

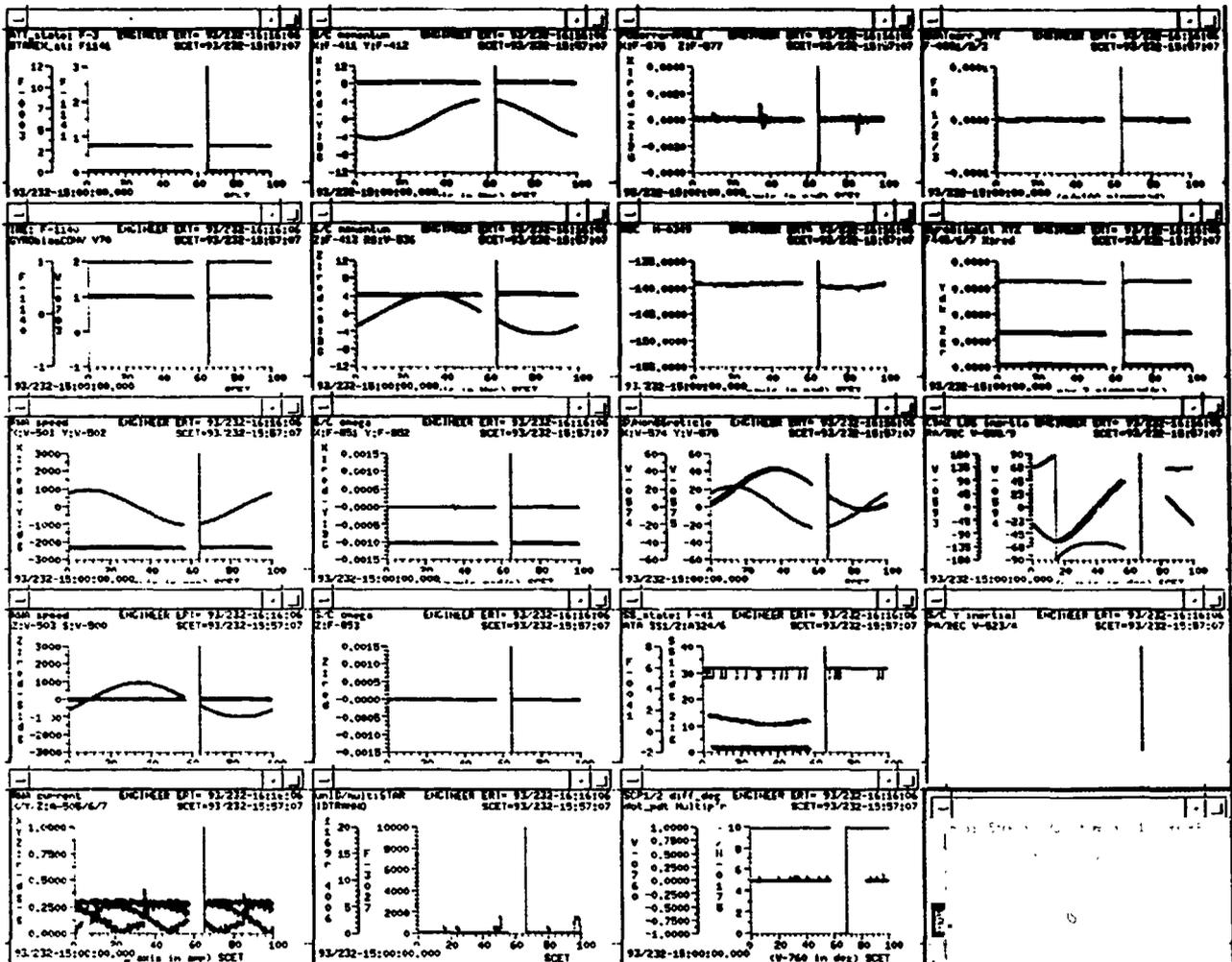


Figure 1. "20-plots" Composite Page, Plotting Major MO AACs States
 (Plot #1 on left-hand upper corner, down the column to Plot#5,
 etc., Plot #16 through Plot#19 on the right most column)
 (Original display in color, representing a maximum of 4 channels per plot.)
 (Horizontal scale in SCET, Spacecraft Event Time; period of 100 minutes = MO spin period)

- | | | | |
|-------------------------------------------------|-------------------------------------------|------------------------------------------------|----------------------------------------------|
| P1. Attitude_State
STAREX_State | P6. S/C_Momentum: X; Y | P11. (Attitude)_POS_error:
X; Y; Z | P16. Quaternion_Correct'n:
X; Y; Z |
| P2. Inertial_Ref_Acquired
GyroBias_Converged | P7. S/C_Momentum: Z;
S/C_Mom: RSS(X+Z) | P12. AGC (db gain) | P17. GyroBias_Estimate:
X; Y; Z |
| P3. RWA_Speed: X; Y | P8. S/C_Rate: X; Y | P13. Sun Sensor Reticle
Reading: X; Y | P18. Celestial Sensor Ass.
J-2000 RA; DEC |
| P4. RWA_Speed: Z; S | P9. S/C_Rate: Z | P14. Sun Sensor State;
ATA: SS1; SS2 | P19. S/C Body :
J-2000 RA; DEC |
| P5. RWA_Current: X; Y; Z | P10. UNID_Star; Multi_Star;
IDTRANNO | P15. SCPI/2 diff_degree;
dot_pdt_multiplier | P20. "xterm" window |

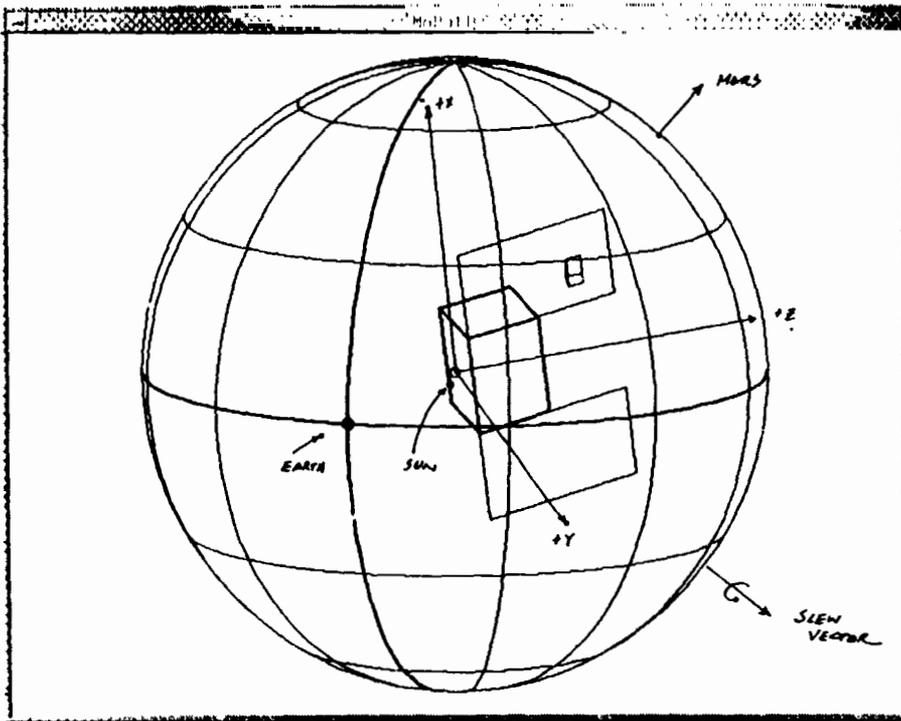


Figure 2. MOBALL Pointing Analysis - TES Instrument Calibration

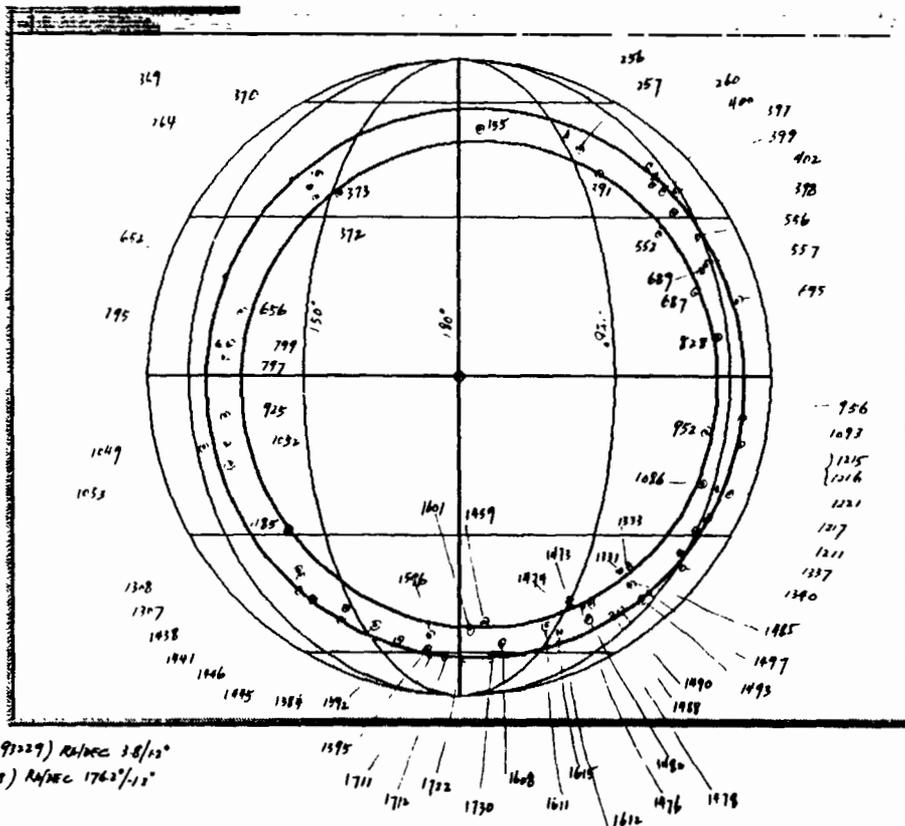


Figure 3. MOBALL Star Field Analysis - 93229 ANS Star Catalog

Generic Trending and Analysis System

Lori Keehan
Code 511
NASA/GSFC
Greenbelt, MD 20771

Jay Reese
Loral Aerosys
7375 Executive Place
Lanham, MD 20706

ABSTRACT

The Generic Trending and Analysis System (GTAS) is a generic spacecraft performance monitoring tool developed by NASA Code 511 and Loral Aerosys. It is designed to facilitate quick anomaly resolution and trend analysis. Traditionally, the job of off-line analysis has been performed using hardware and software systems developed for real-time spacecraft contacts; then, the systems were supplemented with a collection of tools developed by Flight Operations Team(FOT) members. Since the number of upcoming missions is increasing, NASA can no longer afford to operate in this manner. GTAS improves control center productivity and effectiveness because it provides a generic solution across multiple missions. Thus, GTAS eliminates the need for each individual mission to develop duplicate capabilities. It also allows for more sophisticated tools to be developed because it draws resources from several projects. In addition, the GTAS software system incorporates Commercial Off-The-Shelf Tools Software(COTS) packages and reuses components of other NASA-developed systems wherever possible.

GTAS has incorporated lessons learned from previous missions by involving

the users early in the development process. GTAS users took a proactive role in requirements analysis, design, development, and testing. Because of user involvement, several special tools were designed and are now being developed. GTAS users expressed considerable interest in facilitating data collection for long term trending and analysis. As a result, GTAS provides easy access to large volumes of processed telemetry data directly in the control center. The GTAS archival and retrieval capabilities are supported by the integration of optical disk technology and a COTS relational database management system.

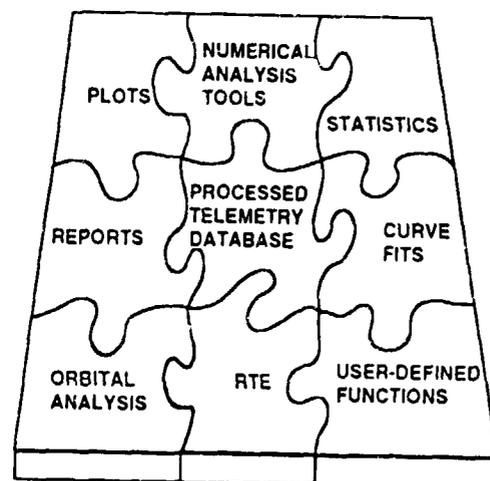


Figure 1: GTAS Pieces

BACKGROUND

Until now, off-line analysis has been performed using collections of tools developed by satellite Flight Operations Team(FOT) members. Separate toolsets have been developed within each project and often by several project members. Collectively, the capabilities of the tools have met the needs of the FOT, but the replication and variance between projects and between FOT members has several drawbacks. Capabilities have been lost when an individual leaves a team or when the FOT contractor is replaced. Similar capabilities were often developed many times, adding to the cost of operations. Since the tools were developed within the constraints of the FOT resources, more sophisticated and efficient tools were not considered. In addition, the task of offline analysis was made more cumbersome because the analysts did not have direct access to processed data. GTAS is being developed to improve this situation.

GTAS METHODS & POLICIES

Constrained budgets and an increasing number of missions force GSFC to evaluate methods for developing and operating systems at a lower cost. GTAS is an example of this process improvement. First, the GTAS project is being developed to meet the needs of multiple current and future missions. It draws cross project support, promotes the sharing of technology, and attempts to eliminate the development of duplicate capabilities. It uses lessons learned from previous mission experience and it generates a forum for cross mission contact.

Second, it takes the task of generating off-line analysis tools away from each individual FOT member and gives it to

individuals who are trained in control center development. Thus, primary off-line analysis tools are no longer expected to be developed separately by each end user in his free time; rather, they are an inherent part of the control center system.

Third, GTAS attempts to take advantage of Commercial Off-The-Shelf Tools (COTS) software packages and reuses components of other NASA-developed systems. COTS products are extremely powerful and provide a cost-effective method for meeting many missions requirements. Currently, GTAS uses a plotting and analytical software package called PVWAVE, a optical jukebox file management software package called AMASS, and the ORACLE relational database system. In addition, GTAS integrated TOSA, an existing NASA developed project, into its delivery. This product provides the end-user the ability to monitor time-varying parameters based on signature analysis and orbital events.

ENVIRONMENT

GTAS is developed within the Transportable Payload Operations Control Center (TPOCC) environment. GTAS is being used or is planned to be used by the following projects: Fast Auroral Snapshot Explorer(FAST), Submillimeter Wave Astronomy Satellite (SWAS), WIND, Polar Plasma Laboratory (POLAR), Solar and Heliospheric Observatory(SOHO), X-Ray Timing Explorer (XTE), Advance Composition Explorer(ACE), Tropical Rainfall Measuring Mission (TRMM), and Far Ultraviolet Spectroscopic Explorer (FUSE).

The TPOCC systems utilize UNIX workstations, X-terminals, and VME-bus systems connected via a local area network.

The TPOCC Front End Processor(FEP) processes data in real-time. Its sources of input are mission dependent, but they commonly include NASCOM blocks, CCSDS frames, Data Capture Facility's Packet Processor (PACOR)files, and telemetry history files. Each Mission Operations Center(MOC) development task is responsible for developing a Real-time processing capability. Real-time processing and replay are functionally identical. Therefore GTAS uses the output to the MOC processing as the source of its input. Thereby, eliminating the development of duplicate functionality.

IMPLEMENTATION

Resolving some spacecraft anomalies in the control center is like looking for the proverbial needle in a haystack for the spacecraft analysts. They must survey large volumes of data to find one byte of information that caused the anomalous situation. GTAS provides the following features to assist the spacecraft analysts in their search: statistics, relational telemetry expressions, plotting and mathematical tools, and an archival and retrieval system.

Statistics

GTAS routinely ingests subsets of

telemetry data and creates statistical data sets. Statistics are stored with both long and short term granularities; users may select to generate statistics in millisecond, second, hourly, daily, orbital, or user-defined intervals. For analysis using plots, statistical reduction has the advantage of tremendous efficiency savings without the drawbacks associated with data thinning or interval sampling. For generating a long period assessment of the spacecraft's performance, a plot of the parameter's statistics provides a quicker and more readable end product. For example, a full 24-hour period contains 1440 one-minute statistical data points versus over one million points without statistical reduction. Figure 2 displays the graphical user interface used to select a parameter's statistical intervals.

Relational Telemetry Expression

Previous mission analysts have expressed a need to facilitate data collection to support anomaly detection and trend analysis. GTAS provides an event-driven data capture tool called a Relational Telemetry Expression (RTE). It will capture subsets of processed data to support analysis. A RTE is a triad consisting of the set of user-defined telemetry conditions, an evaluation criteria, and a list of mnemonics to output when the conditions are met. The

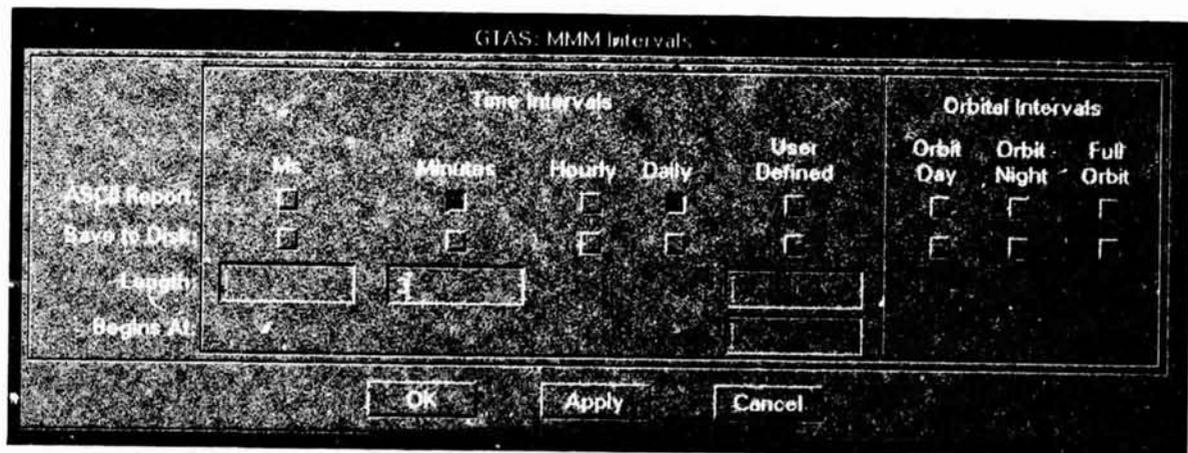


Figure 2: GUI to Select Statistical Intervals

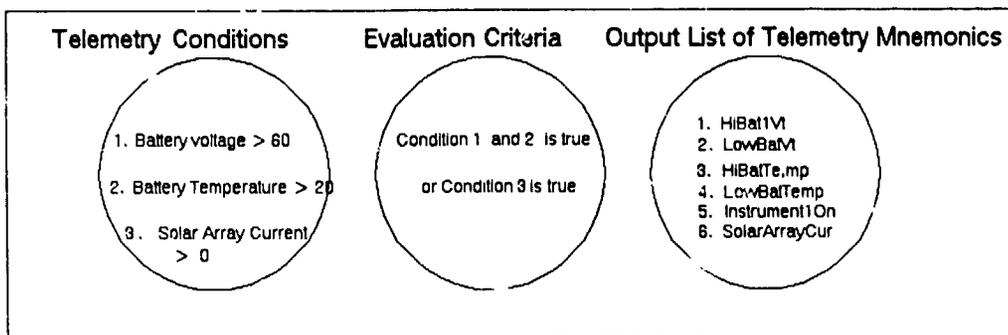


Figure 3: Sample Content of an RTE Expression

RTE task evaluates the user-defined conditions based on the evaluation criteria, then outputs the list of related mnemonic values. True or false values of RTEs are written to an output file, and upon request, ingested into the trending database. The results of this RTE are used to evaluate broader boolean equations then are used to evaluate a particular spacecraft state by the real-time system. Figure 3 shows the contents of a sample RTE expression.

Plotting and Numerical Analysis

GTAS plots provides users with a multi-dimensional visualization tool. It uses advanced graphical techniques to accelerate the search for patterns and trends in large technical datasets. Several types of common plot output are telemetry vs. time, telemetry vs. telemetry, statistics vs. telemetry, statistics vs. time, statistics vs. statistics, RTE vs time, RTE vs. RTE, and RTE vs statistics. The key to the GTAS plotting software is its flexibility. Users may plot in portrait or landscape mode, display multiple plots per page, choose grid options, choose axis lengths, select the number of tickmarks, specify the length of the tickmarks, annotate text directly on the plot, zoom in or out of the plot, etc. A sample plot is pictured in Figure 4.

In addition to the plotting tools, GTAS gives the user access to hundreds of numerical library functions such as fast fourier transforms and curve fitting routines. It also provides convenient access to these numerical tools directly from the plots. Reference 6 contains a complete listing of GTAS plotting and numerical analysis capabilities.

Archival & retrieval

Traditionally, the task of offline analysis was extremely cumbersome because the end users did not have direct access to processed historical data. They were forced to rely on outside individuals to retrieve raw historical data. These data retrievals could take anywhere from several days to several weeks; some older data was virtually impossible to retrieve at all. Once the data was retrieved, it needed to be processed. This task could also be very time consuming, especially if realtime resources could only be scheduled during non-contact periods.

GTAS, however, archives processed data directly in the control center for easier access to the end user. To do this, GTAS integrated a optical disk mass storage system with its trending database. The system is capable of storing over 40 GB online. This

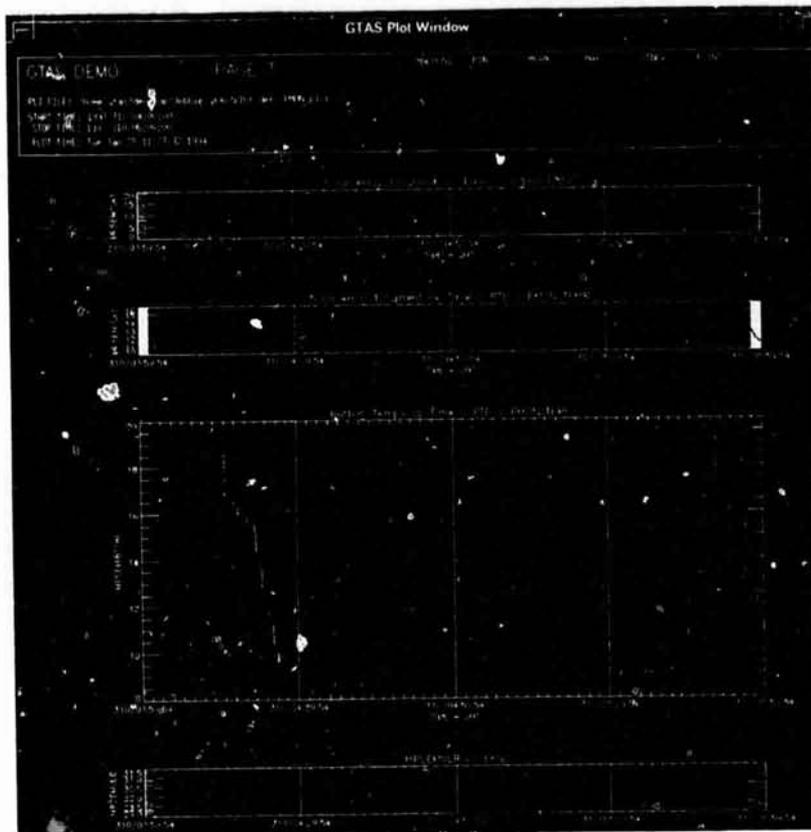


Figure 4: Sample Plot

provides a library of approximately 30 days of subsetting telemetry directly in the control center. Plus, data retrievals no longer require outside intervention and retrieval times are reduced from weeks to minutes.

The GTAS archival and retrieval system also provides a data editor capability. This will allow the users to do "what-if" analysis while also maintaining data integrity. Only privileged users will be allowed to make permanent changes to the data in the database. All other users will have direct access to the data and may manipulate and categorize the data as they see fit without effect to the mission's trending database.

REFERENCES

- 1). Computer Sciences Corporation and Integral System, Inc., *Transportable Payload Operations Control Center (TPOCC) Generic User's Guide*, (Revision 2), September, 1991.
- 2). NASA, *Generic Trending and Analysis Workstation Design Concepts Document*, Version 2.0, January 1993
- 3). Loral Aerosys, *Generic Trending and Analysis System(GTAS) System Requirements Document(SRD)*, June 1993
- 4). Loral Aerosys, *Generic Trending and Analysis System(GTAS) Detailed Design*

Specification (DDS), April 1994

5). Loral Aerosys, *Generic Trending and Analysis System (GTAS) Release 1 User's*

6). Visual Numerics Inc, *Command Language Reference Volume 1 & 2*, 1993

A GRAPHIC SYSTEM FOR TELEMETRY MONITORING AND PROCEDURE PERFORMING AT THE TELECOM S.C.C.

Author: Jean Philippe LOUBEYRE

Address: CNES - CT/EO/ER/TC
18 avenue Edouard Belin
31055 TOULOUSE CEDEX
FRANCE

Abstract

The increasing amount of telemetry parameters and the increasing complexity of procedures used for the in-orbit satellite follow-up has led to develop new tools for telemetry monitoring and procedures performing.

The name of the system presented here is GRAPHIC SERVER.

It provides an advanced graphic representation of the satellite subsystems, including real-time telemetry and alarm displaying, and a powerful help for decision making with on line contingency procedures.

Used for 2.5 years at the TELECOM S.C.C. for procedure performing, it has become an essential part of the S.C.C.

Key words: Satellite telemetry displaying, on-line procedures, functional graphic mimics.

Introduction

The TELECOM S.C.C. is in charge of the control and the follow-up of the French TELECOM satellites. Three satellites are in orbit today: TELECOM 1C the last model of the TELECOM 1 satellites, TELECOM 2A and 2B the two first models of the TELECOM 2 family.

The main task of the S.C.C. is to perform all operations required for station-keeping and satellite subsystems management.

The increasing complexity of spacecraft

subsystems and procedures, and the increasing amount of telemetry (TM) parameters led to develop a new tool called "Graphic Server" providing a friendly man-machine interface to monitor and display all TM parameters, both in routine phase and during procedure performing.

Nowadays, this tool has been used at the TELECOM S.C.C. for 2.5 years.

This paper will first give a brief summary of the architecture and the facilities of the Graphic Server, then present the result of its operational use.

General description

The Graphic Server is a system displaying real-time telemetry on up to five simultaneous graphic displays. It is connected to the S.C.C. telemetry acquisition and monitoring system from which it receives TM parameter values and alarm codes for the selected satellites in order to update all the graphic items on the displayed mimics.

It also manages a graphic interface used by the operator to choose the appropriate mimics among the available ones in the mimic base, and to request the involved TM parameters to the S.C.C. computer.

All the available mimics are previously designed using the off-line application which provides a graphic editor, consistency tests, a simulation tool and storage facilities.

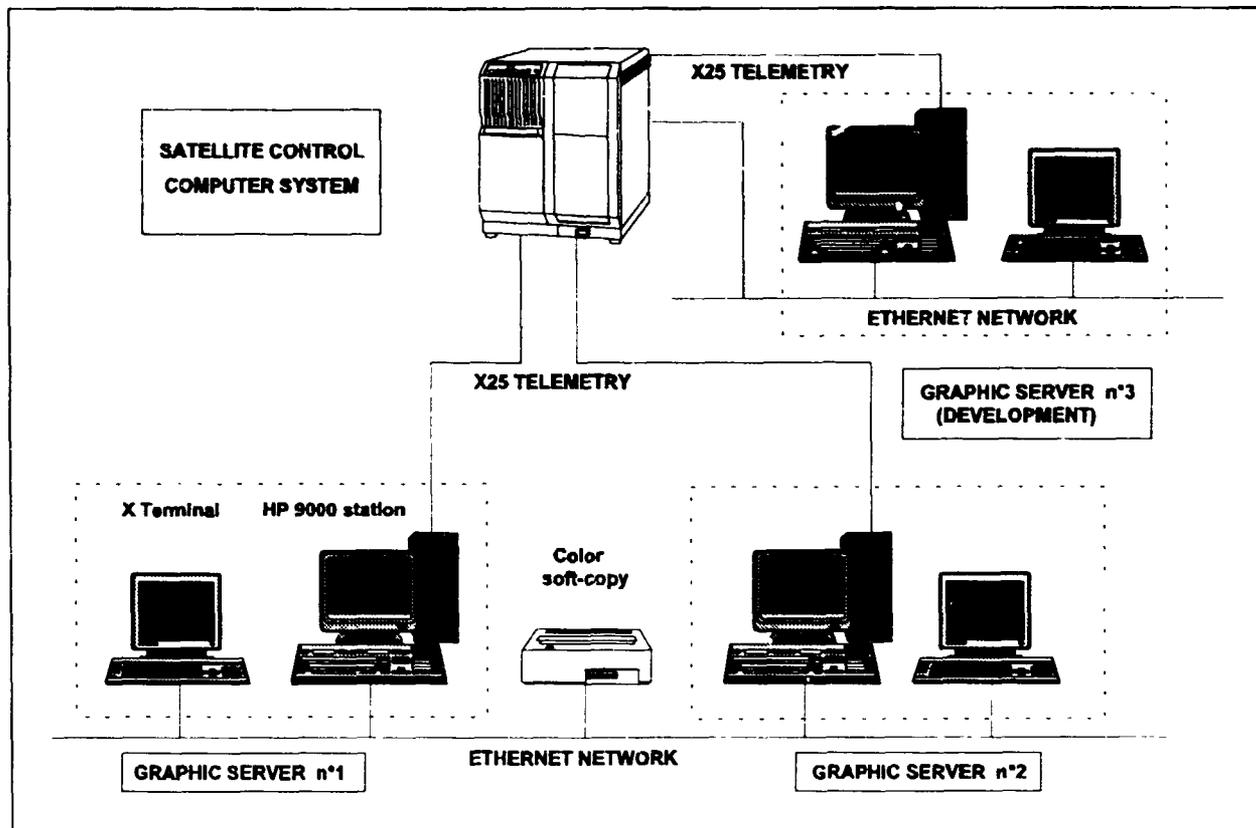


Figure n°1 : Hardware description

Hardware environment

The Graphic Server is supported by a Hewlett Packard configuration (see figure n°1)

- a HP 9000 computer from the 800 series under HP UX with a 335 Mb disk, 24 Mb of RAM memory and a 16 tracks streamer.
- a high resolution 19 inches bit-map display as the master display.
- an X-terminal with a 19 inches display as the slave display.
- an alphanumeric display as the system supervisor terminal.

Communications between the Graphic Server and the S.C.C. telemetry acquisition system are supported by:

- an X25 link for real-time telemetry .
- an ETHERNET link for the development Graphic Server work station

Internal communication between the HP work station and the X-terminal is supported by a local ETHERNET link.

Software environment

The software configuration implies the following items:

- HP UX operating system
- C compiler
- HPGKS, STARBASE, X25, FORTRAN, and X11 environment.
- ANIMATOR, a graphic software package developed and distributed by the SYSECA company
- the Graphic Server application.

Application description

The Graphic Server application provides the off-line mode including all tools for creating, checking, and storing the mimics, and the real-time mode used for TM data acquisition,

mimics updating and operator's requests handling (see figure n°2).

The real-time mode

The real-time application performs the TM parameters acquisition, and using the "graphic real-time environment" developed with the off-line mode, actuates the animation of the

telemetry data flows, only data involved in the displayed mimics are sent to the Graphic Server. Each new mimic request will first trigger a telemetry data request before displaying and updating this new mimic. These data are already processed by the S.C.C. telemetry acquisition and monitoring system, and sent to the Graphic Server under a label, a

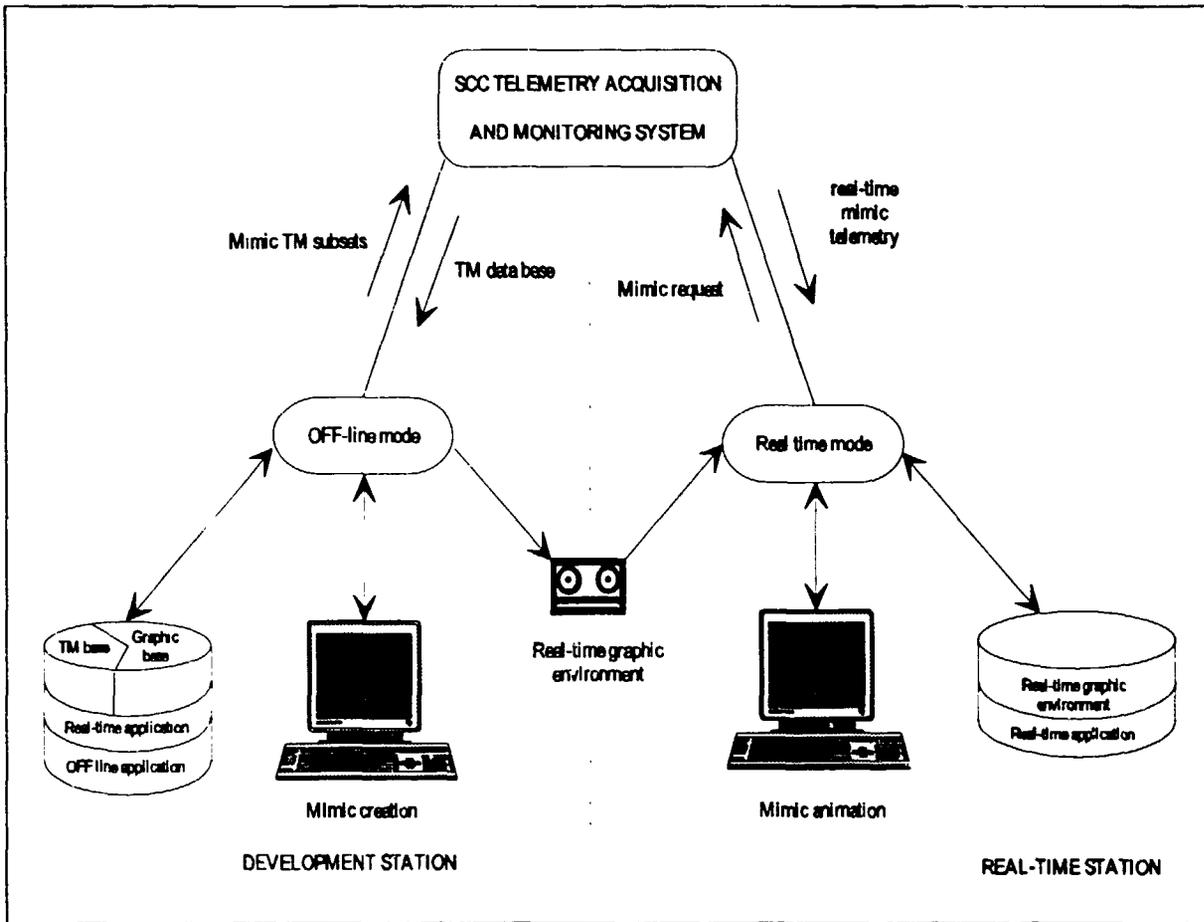


Figure n°2 : Real-time and off-line modes

graphic items used in the displayed mimics. It enables to display and update up to five mimics, one full screen on the bit map display, and one full screen, (or four quartered screen) on the X-terminal display according to its software configuration.

Telemetry data can be real-time or replayed from one or several satellites, or simulated data from a satellite simulator. In order to minimize

raw value, an engineered value, and an alarm code. Telemetry data values are used to update all graphic representations, and alarm codes to update their color.

A specific default representation is also used for TM parameters which have never been received.

The off-line mode

The off-line mode, only available on the development work station, is used to create, check, test and store the mimics.

Each mimic can include the following kinds of graphic items: static background drawings, dynamic graphic items (graphic symbols, numeric or alphanumeric values, auto-scrolling curves) used to display the TM parameter values and alarm codes, and clickable areas used to control the displayed mimics.

All these items are created by the system manager and stored into specific libraries, so they can be used again when creating new mimics.

The first action to create a mimic is to define the background drawing with the graphic editor, then to pick up (or create) dynamic items from the libraries, according to the way you want to display each TM parameter (several simultaneous representations are allowed for the same TM parameter).

The second step is to associate those graphic items with the TM parameter labels.

After compilation, consistency tests are performed using the telemetry data base exported from the S.C.C. telemetry acquisition system.

The third step is to export to the S.C.C. telemetry acquisition system, the "mimics TM parameter subsets". These subsets will be used by the system to send to the Graphic Server the involved TM parameters after a mimic request. As a final step, you have to generate and store the "graphic real-time environment" which will be used by the real-time mode.

A simulation tool provides the ability to test created mimics before using them with real time telemetry.

Using the system

The Graphic Server tool was implemented in December 1991 as an additional mean for telemetry displaying in the TELECOM S.C.C. and in the TELECOM 2 payload centers.

The graphic environment has been developed for two years by the TELECOM 2 spacecraft analysts according to the operational needs. More than 250 mimics were created, using about 1000 graphic items, enabling to display more than 2000 TM parameters.

Mimic ergonomics definition

Considering the amount of TM parameters and so the number of mimics to create, the first job was to define graphic ergonomics rules for the development of the mimics in order to provide a friendly access to TM parameters and an easy understanding of the satellite subsystems.

Color codes: a specific color code is used to identify each TM parameter alarm status (not received, normal, first level or second level of alarm,), telecommand labels, telemetry labels, static items (without telemetry), wires or links, ON or OFF equipment, clickable areas.

Symbol codes: generic patterns used in numerous mimics were created with standard graphic items (telecommand cartridges, telemetry cartridges, automatic reconfiguration orders, warnings, TOPs, switches...etc.) enabling an easy perception through a whole mimic.

Mimics organisation

Several kinds of mimics were created according to the operational uses:

Alphabetic alphanumeric TM parameter lists: These mimics displaying both engineered and raw values of TM parameters allow to reach immediately any TM parameter using its label, without knowing the other kinds of mimic where they are involved in. They can be used to check calibration functions of TM parameters with both engineer and raw values.

Thematic alphanumeric mimics: These mimics display groups of TM parameters, sorted according to a satellite subsystem, function, equipment or procedure only in engineered values.

Synoptic mimics:

These kind of mimic can be functional synoptics of satellite subsystems, control panels, monitoring mimics, or on line

mainly used to perform complex procedures (see figure n°3).

Control panels are subsystem (or whole spacecraft) syntheses and are used to check

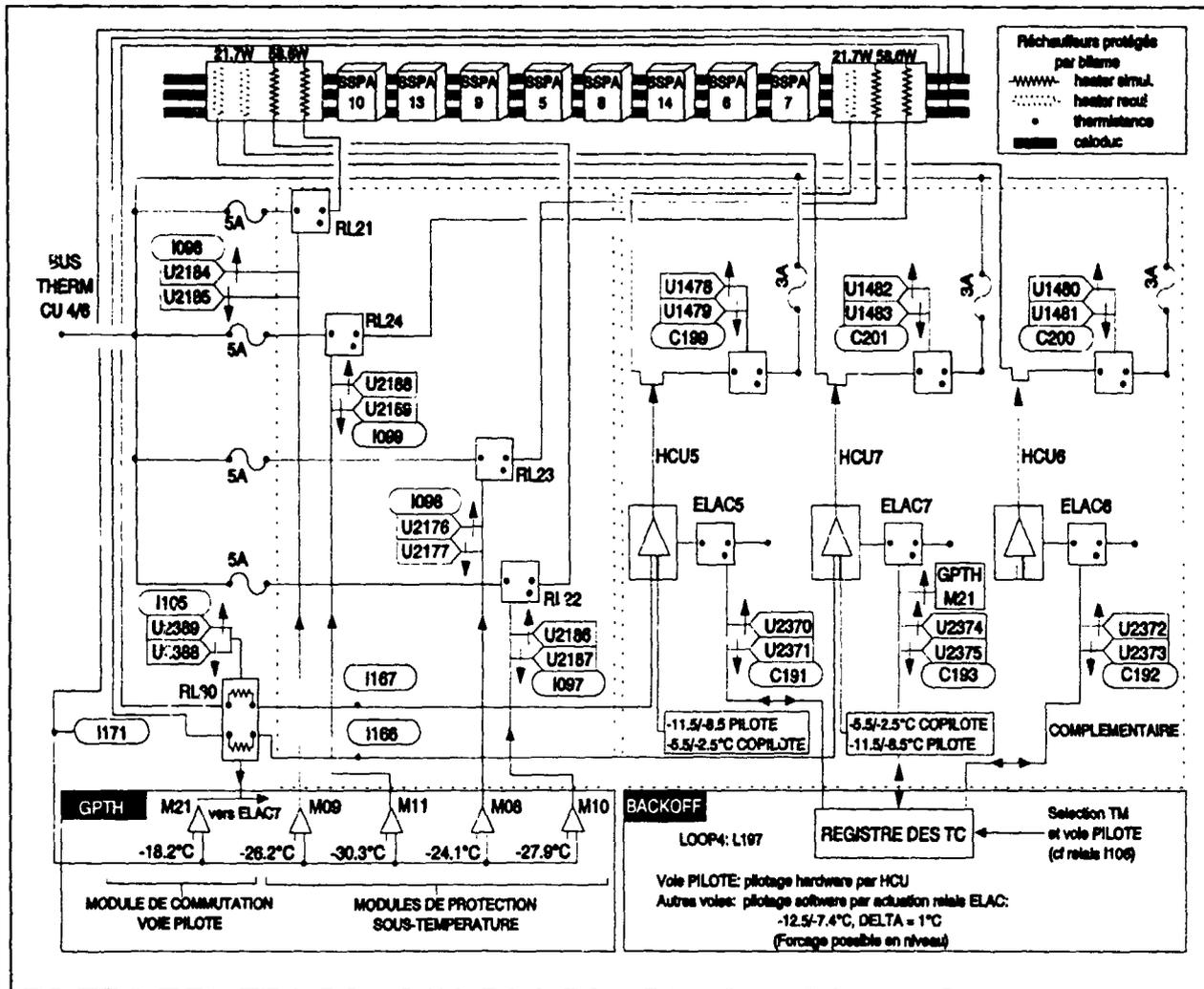


Figure n°3 : Functional synoptic mimic

procedure. Functional synoptics are organised in a hierarchic way with clickable areas to move through the functional tree from high level synoptics to fully detailed ones. Using all kind of graphic items created by the system user, they display TM parameter values and labels, telecommand labels and expected effects, automatic reconfiguration orders and functional schemes. As they provide an easy understanding of satellites subsystems, they are

satellite configuration (see figure n4). Monitoring mimics are developed as a guideline for some contingency procedures which require short time reaction. They display the involved TM parameters, decision trees with clickable areas allowing to display on line procedures (see figure n°5).

Curves mimics

These mimics display auto-scrolling curves of TM parameters, and are mainly used to monitor

some specific operations such as manoeuvres or eclipses and as routine displays.

Procedures mimics

These mimics include procedure schemes, explanations, and involved TM parameters. They are designed to minimise the operator's response time for the procedure application (see figure n°6).

Real-time man-machine interface

The MMI is used to display any mimic, on any of the five screens using any satellite real-time (or replayed) telemetry data flow. This dialog is enabled by several kinds of clickable areas (identified by their color) on the mimic displayed on the master screen.

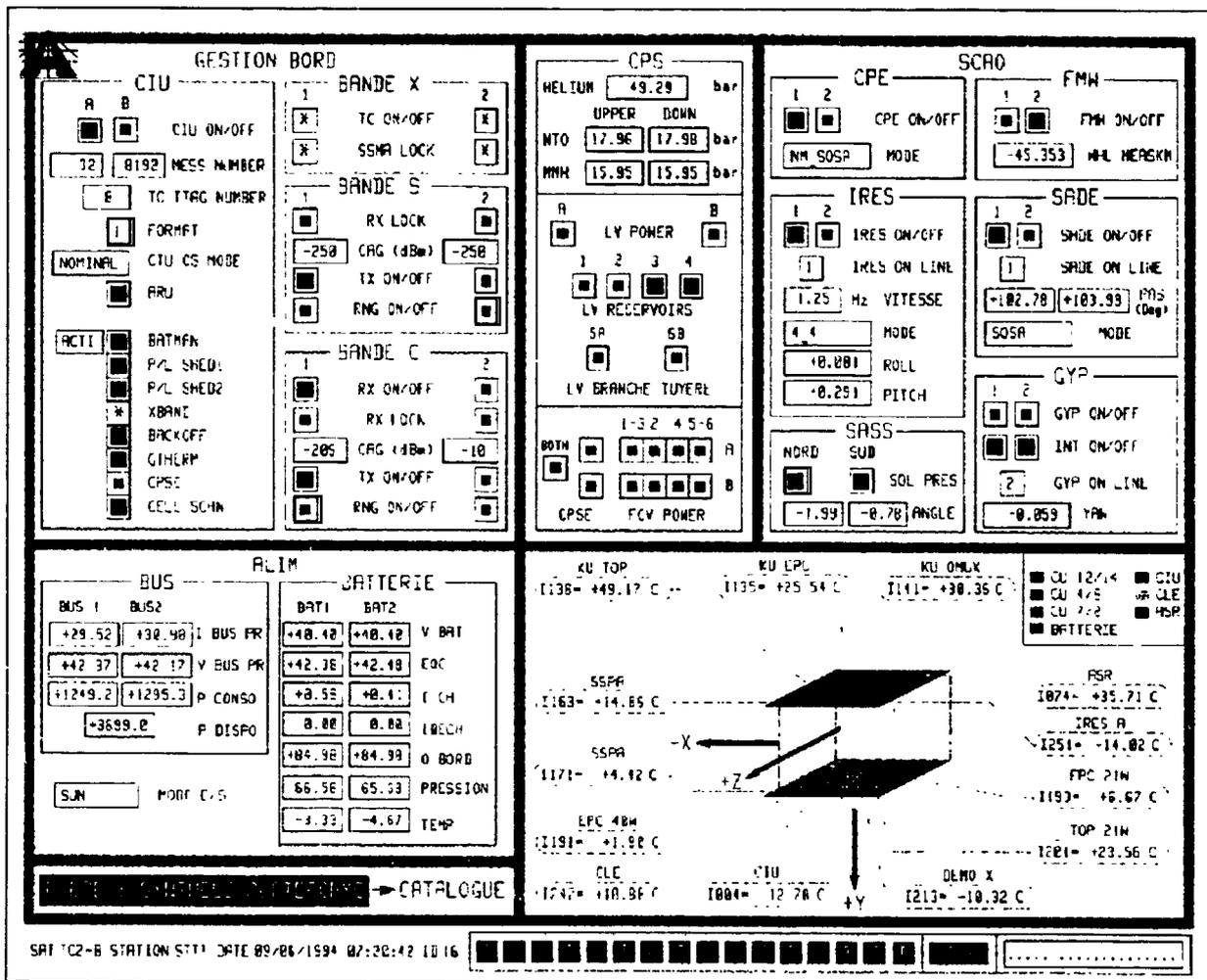


Figure n°4 : Control panel mimic

Directory mimics

This type of mimic is used to display directories of each kind of mimic. It displays the mimics titles and labels and provides an immediate display of the requested mimic clicking on its label.

Keyboard requests through a dialog box

This is the generic mean to create a request. The operator has to define the following items : (mimic label / satellite / real-time or replayed telemetry / screen number) with the keyboard using first the clickable dialog box available in any mimic. It requires to know the mimic label.

Mimic request through clickable graphic items
 These items (created by the system user), identified by their color, are included in the

using the keyboard.
 As these buttons are displayed at the bottom of any mimic, they provide the ability to create temporary links between mimics used for a

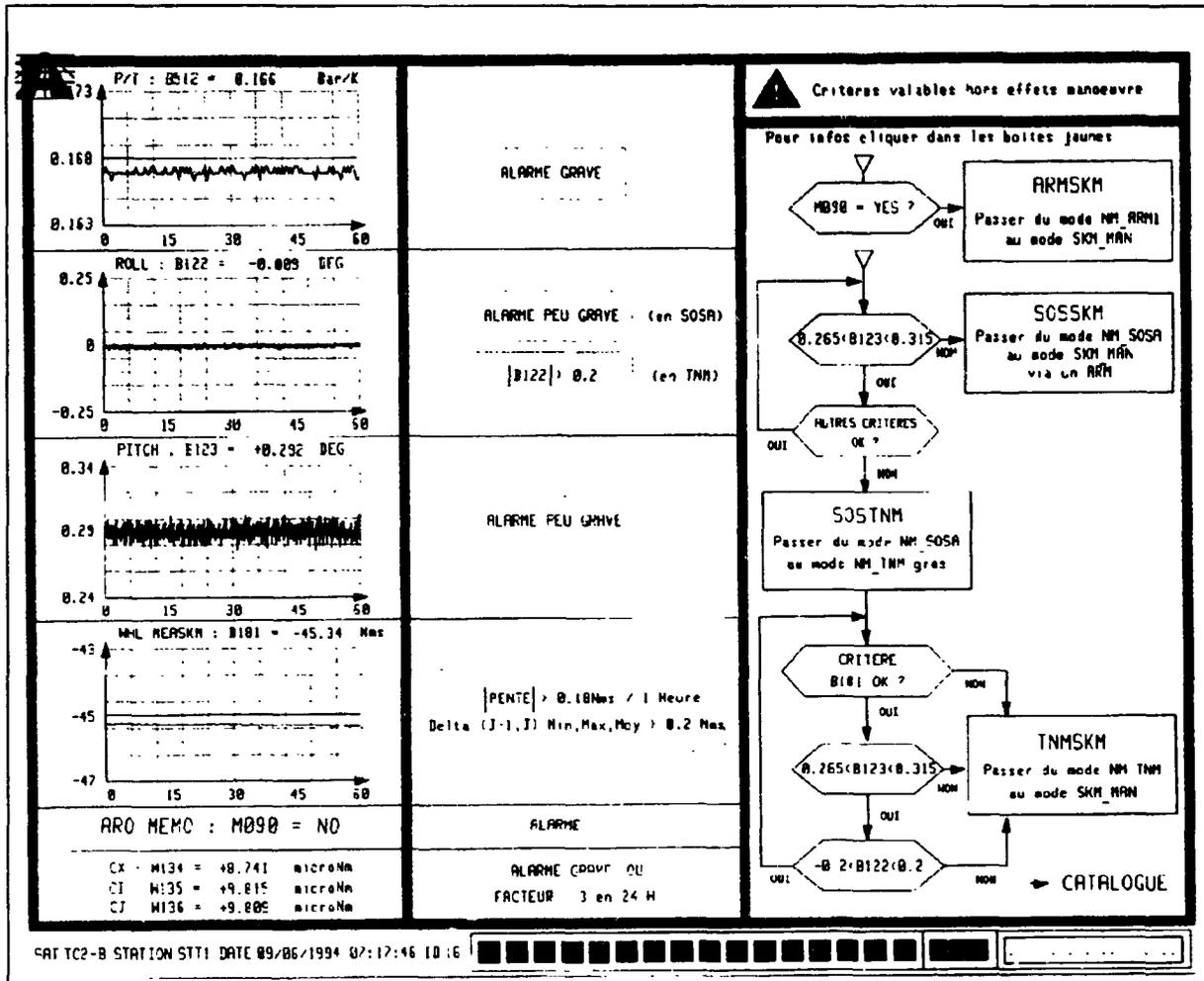


Figure n°5 : Monitoring mimic

mimics according to the logical links between them. They are mainly used to move through functional synoptics and to reach any mimic from specific displays providing mimic directories. They allow to display a new mimic only on the master screen.

Dynamic buttons

A graphic interface enables to program 16 buttons choosing a combination of the following items for each of them (mimic label / satellite / real-time or replayed / screen number). These buttons are the only way to request for a mimic on the slave screen without

particular procedure. To improve this selection, the system allows to store 15 programs of the 16 buttons. These programs are defined, named stored and selected by the operator according to particular procedures or phases (example : "Manoeuvre" or "Eclipse" program). By this way, the operator has the ability to display immediately any mimic of the involved ones, (without knowing its name) when he performs a procedure.


→ CATALOGUE

OBJECTIF

Configurer le satellite du mode a voie solaire (NM_SOSA) au mode 3 axes a tuyeres (SKM_MAN) dont la capacite de controle est la plus robuste.

Le principe est de declencher un ARM et d'effectuer une manoeuvre de duree nulle utilisant la branche B, la premiere tranquillisation ayant ete prealablement fixee a une duree de 2 heures.

MOYEN : Plan de telecommande : 505SKMH

Ce plan correspond a une utilisation de 1 IPES2 en mode 2/4_HIVER (trace SUD non utilise)

- 1- Ce plan est deja prepare en avance a chaque relance TR.
- 2- Afin d'optimiser le temps de reaction le controle a posteriori des TC a ete supprime. Le controle TM sera fait a certaines etapes :
 - o en au plan

1- Executer le plan jusqu'au bloc BESRINH inclus puis verifier la TM

REPERE	TM	REFERENCE	LIBELLE
M600 =	NM_SOSA	SK_ARM2 ou (SK_ARM3)	MODE_SUBMODE
B001/02 =	ON OFF	OFF ON	AOCS CVA /B
B003/04 =	ON_CVA ON_CVA	ON_CVA ON_CVB	KERNEL A/B
B005/06 =	ON_CVA ON_CVA	ON_CVB ON_CVB	PERIPH 1/2
M001/02 =	ON OFF	OFF ON	CPE A/B
B007/08 =	ON OFF	OFF ON	TTE A/B
B009/10 =	TC_INH TC_INH	TC_ENA TC_ENA	TTE A CD RL
B026/?? =	TC_INH TC_INH	TC_INH TC_ENA	SNE A CD RLY /B
B028/29 =	ON OFF	OFF ON	MACS A/B
B030/31 =	ON OFF	OFF ON	SRE A/B
B090/99 =	ON OFF	OFF ON	IRES 1/2
B105 =	LENTE	RAPIDE	IRES ONL RATE
B506 =	4.4	2.4_HIV	MODE IRES
B201/202 =	ON OFF	OFF ON	SAGE 1/2
B130/131 =	OFF OFF	OFF ON	CPSE 1/2
B171/172 =	FERMEE FERMEE	FERMEE OUVERTE	LV5 / LV6
B135/39 =	OFF OFF	OFF ON	FCV1-3 A PHR /B
B135/40 =	OFF OFF	OFF ON	FCV2 A PHR /B
B138/142 =	OFF OFF	OFF ON	FCV5-6 A PHR /B
B079/08 =	OFF OFF	OFF ON	GYP 1/2
B083/82 =	ON ON	OFF OFF	GYP INT ST 1 /2
B084 =	ON	ON	GYP (1,2) TEMP
B086 =	9.06	> 60 C	GYP 2 TEMP
M034 =	0001	XX3F	LAST CXCD 1
M035 =	4123	0000	LAST CXCD 2
M020 =	OFF	OFF	ESR TIMER
M120 =	IRES_1	IRES_2	IRES MONITG
M095 =	ENA	ENA	ARE EP CRIT
M090 =	NO	NO	ARO MEMO
B125 =	PRESENTE	PRESENTE	IRES CL EP4

2- Terminer l'execution du plan TC et verifier la TM

B083/82 =	ON ON	ON ON	GYP INT ST 1 /2
M600 =	NM_SOSA	SK_MAN	MODE_SUBMODE

3- Verifier que l'attitude ne diverge plus

SAT TC2-B STATION STT1 DATE 09.05.1994 07:35:05 ID 0

Figure n°6 : Contingency procedure mimic

Conclusion

Initially designed to display telemetry mimics in the Payload Control Centers, the Graphic Server tool has become a powerful tool to perform procedures.

Its great flexibility, the numerous graphic facilities provided, and its friendly man-machine interface have allowed the users themselves to develop a fully detailed representation of the satellites subsystems, as well as on line contingency procedures, in order to improve operations safety.

Designed with very few TELECOM 2 specific software modules, it could be easily adapted for any Satellite Control Center and more generally speaking to any monitoring system

with the development of a new interface between the Graphic Server application and data sources.

SMALL SCALE SEQUENCE AUTOMATION PAYS BIG DIVIDENDS

Bill Nelson
 Jet Propulsion Laboratory
 California Institute of Technology

ABSTRACT

Galileo sequence design and integration are supported by a suite of formal software tools. Sequence review, however, is largely a manual process with reviewers scanning hundreds of pages of cryptic computer printouts to verify sequence correctness. Beginning in 1990, a series of small, PC-based sequence review tools evolved. Each tool performs a specific task but all have a common "look and feel." The narrow focus of each tool means simpler operation, and easier creation, testing and maintenance. Benefits from these tools are (1) decreased review time by factors of 5 to 20 or more with a concomitant reduction in staffing, (2) increased review accuracy, and (3) excellent returns on time invested.

Key Words: Sequence review, sequence automation

THE GALILEO SEQUENCING PROCESS

The Galileo sequencing process is a "top down" process that consists of two overlapping functions: the design and integration function and the review function. Both are iterative processes with a considerable amount of manual interaction. "Top down" means that development proceeds from the general to the specific. The major steps along the way are:

- A Planning phase which specifies the timing of mission phases and major

activities. It covers one or more years and is the general guide for later, more detailed sequencing.

- A Design and Integration phase where the timing and placement of the major activities is finalized and where minor and supporting activities are added, all subject to timing and other resource constraints
- A Specification phase, where details are added, parameters are specified and commands "expanded" from predefined routines. The end product of the specification stage is the final command level sequence.

The design and integration stage in particular benefits from prepackaged and pretested activities called Profile Activities or PAs. A Profile Activity is a sort of sequencing subroutine that encapsulates the commands making up its activity. Each PA has a name, a unique ID, a starting time, and a duration. Most also have further parameters that will later control the composition and timing of the encapsulated commands. PAs are an abstraction tool that frees the sequence designer from concern for the details of an activity. In the earlier stages, the designer need only consider the PA function, its start time and duration in integrating the activities into a composite whole. Unique activities are specified by a general purpose PA called the UTILITY PA. It has a start time and duration but no parameters. Its commands are added manually later in the expansion process.

After a sequence is integrated for the first time, it goes through an iterative development cycle of integration, review, correction and addition, and re-integration. As the cycle progresses, the sequence becomes more detailed and specific. General activities have more parameters specified. Supporting activities are added and made more specific, and resource predictions are updated. This "fleshing out" takes a big leap forward with the expansion step which results in a listing of all the specific spacecraft commands.

Once the development cycle begins, each iteration is reviewed by anywhere from half a dozen to nearly two dozen people. Reviewers represent various science instruments or engineering subsystems, ground station operations, and general spacecraft and sequencing perspectives. At earlier stages, the checks are fewer and more general while at later stages they, like the sequence itself, are more detailed and specific. Each reviewer uses checklists specific to these various development stages.

Because it is an obviously difficult job to integrate hundreds of PAs into a limited time span under numerous constraints, sequence design and integration tools have received considerable attention. The process is far from automatic but at least there are support tools to manipulate activities, to design experiments, to manage resources and to present activities graphically. Further, software development continues to stress sequence design and integration tools

SEQUENCE REVIEW SUPPORT

The review part of the cycle has received considerably less support. Most reviewers still go through hundreds of pages of cryptic computer printouts, manually highlighting

items, checking for problems and marking their checklists. Only two mainframe based tools, the CHECKER module of SEQGEN and the STRIPPER program provided any sequence review support.

CHECKER is a hard coded constraint checker. While it can compare actual states against predicted or required states, and can check timing, those abilities are hard coded and limited to (usually) the simpler flight rules. CHECKER is also often out of date. With limited programming resources, it is simply not important enough to keep current. Spurious warnings are common and each must be checked and resolved by hand.

STRIPPER is a data extractor driven by a fixed, change controlled database. It was designed specifically to extract commands and it depends on the rigid sequencing format for proper operation. It cannot extract arbitrary text or scan arbitrary locations on a line. Because by policy, there is only one strip per subsystem, multiple or custom strips are impossible. Generally, STRIPPER is used to create a subset of the main sequence product containing only the commands specific to a given instrument or subsystem. Most science instruments and some engineering subsystems benefit from STRIPPER but those requiring a more global view such as Fault Protection, Power or Telecom do not. STRIPPER may reduce the product from several hundred pages to less than one hundred but those pages must still be reviewed by hand.

Beginning in 1990, a series of small, PC-based sequence review tools evolved. These were created by reviewers in their spare time and in response to their own needs. They were without official support and were unburdened with the paperwork and change control of more formal tools.

SKIMX, A DATA EXTRACTOR

One of the first of these tools was SKIMX, a data extractor so named because it could "skim" any arbitrary text, "x," from a file. SKIMX accepted "match strings" from user prompts or from a file and extracted all lines containing any "match string" text. This gave sequence reviewers a means of creating custom strips. If a check required comparing two commands, for example, SKIMX would find all occurrences of the two commands - and only those commands. Comparison was then straightforward. In effect, the sequence could be separately interrogated for each of the different checklist checks. This simple tool alone cut review times by factors of 2-4. It also represented an excellent return on time invested.

SKIMX has several features that adapted it particularly well to sequence review. It could save the matched lines to a file for later use or for pasting into the reviewer's comments. It accepted frequently used sets of "match strings" from pre-defined

datafiles. It counted the number of matches or reported "No match found" which simplified checking for forbidden commands. This feature was sometimes used simply to quickly count the number of occurrences of events. SKIMX could report matches in either physical or logical lines. PAs are built as a single, comma delimited logical line with the end of the logical line denoted by a semicolon. A long logical line may take several physical lines, each intermediate physical line ending in a comma. Sometimes matching only the physical line is sufficient, sometimes the full PA, the logical line, is required.

The original SKIMX was created in a single day and when printed took all of four pages. Code for the actual "skim" occupied only half a page with the rest being help screen text, user prompting, and commenting. Within six months, SKIMX was regularly used by about a half a dozen people who reported anywhere from two to eight hours saved per review.

```
SKIMX V1.6 9/30/93

SKIMX finds all lines containing any specified string or strings. SKIMX
ignores upper/lower case. Matches may be saved to an Output File.
USAGE: SKIMX [/x]...[/x] [Input FileSpec [,Output FileSpec]] where
/x represents any of these options:
/B for BLACK AND WHITE (monochrome) monitors.
/C to force upper/lower CASE SENSITIVE matching.
/FMatchFILESpec to read MatchStrings from plain ASCII MatchFILE.
/H for HELP screen (this screen).
/Kword to enter a single KEYWORD MatchString from the command line.
No blanks, slashes, commas or '<,>,' characters allowed.
/M[m][n] for MULTIPLE lines per item (like ORPRO files). Omit 'm'
for special handling of $ and * header lines, use 'n' = decimal
ASCII value to change terminator.
/Q for QUICK output - no output to screen while working.
/R to REVERSE the sense of the match. This option OMITs matching
lines and only lines WITHOUT any matching strings are saved.
Input FileSpec is the file of data to skim,
Output FileSpec is the file where skimmed output is put. If omitted,
output is to screen only. NOTE: comma must separate FileSpecs.
Hit any key to continue
```

Figure 1 - SKIMX help screen

SAFPRINT reformats the Station Allocation File,

```
PA2, STALOC, 362A, PRI, 94-192/21:52:03.010, 07:15:00, +07:15:00, OMT, GLL GOE,  
CFG 608I, T/P DMSCOND, 94-192/22:25:00.000, 14, 1733, 94-92/23:40:00.000,,  
S, 3,, N,, 94-193/05:25:00.000, 94-193/05:40:00.000;  
PA2, STVUPD, 360A, PRI, 94-192/22:32:28.772, 09:06:21, +09:06:21, OMT, GLL GOE,  
DSN VIEW, FJ7-5 ME 36.4 1994-193T03:38:37, 14, 1733, 94-192/23:05:26.000,  
94-192/23:05:26.000, 94-192/23:29:15.000, 94-193/07:47:58.000,  
94-193/08:11:47.000, 94-193/08:11:47.000;  
PA2, STHAND, 366A, PRI, 94-193/00:22:57.546, 00:15:00, +00:15:00, OMT, GLL GOE,  
DSN U/L, ACQUIRE UPLINK, 94-192/23:50:00.000,,  
14,, +00:05:00,, 100.0, HIGH, 4.0,,,,, S;  
PA2, STHAND, 366B, PRI, 94-193/05:57:59.529, 00:15:00, +00:15:00, OMT, GLL GOE,  
DSN U/L, TXR OFF, 94-193/05:25:00.000, 14,, -00:05:00,,,,,, S;
```

into a more readable format:

```
14 RISE: 94-192/23:05:26 SET: 94-193/08:11:47 MaxE1: 36.4 at 03:38:37  
1733 BOT: 94-192/23:40:00 EOT: 94-193/05:25:00 ESC: T/P DMSCOND  
HI XON: 94-192/23:50:00 XOFF: 94-193/05:25:00 CFG: 608I, DUR: 05:45;
```

Figure 2 - SAFPRINT input and output

Now, some four year later, over two dozen people use SKIMX and the time saved to date is well over 1000 hours. (Since copies of SKIMX are kept on several Galileo servers, total usage is unknown). SKIMX itself has grown to seven pages but still represents a return on time invested of well over 1200 per cent.

DATA REFORMATTERS

Another early tool was SAFPRINT. This utility cast the Station Allocation File into a more readable format and in the process made some simple constraint checks. During its creation, SAFPRINT found errors in five consecutive Station Allocation Files. In response to this, SAFPRINT's constraint checking was expanded and a companion program, SAFCHECK, was created to check for timing errors. SAFPRINT and SAFCHECK were so successful that the Mission Control Team, the group responsible for creating and maintaining the Station Allocation File, adopted them as part of their standard internal checking procedures. There have been no timing or logical errors in any

Station Allocation File pre-checks with the SAFPRINT suite of tools.

SAFPRINT is also used in sequence development. Here, however, its ability to convert allocations from their ground timeframe to spacecraft time is as valuable as the better format. Furthermore, SKIMX can be used to interrogate the reformatted file to locate allocations by day or by scheduled activity. Success with SAFPRINT demonstrated that just casting data into a more convenient format is sometimes sufficient to gain significant savings of time and effort.

OPEVENT followed in the reformatting tradition by reformatting the unexpanded products. It gave the user the ability to select which PAs to reformat and which to ignore. Of the PAs being reformatted, the user could select which parameters to display, the display order and the titles to assign. One other unique feature was the ability to do time arithmetic on parameter fields. This made it possible to turn a start time and duration into an end time, or to make some limited ground to spacecraft time

conversions. The result was a sequence summary that, like SAFPRINT, could be further interrogated with SKIMX. Custom reformats with OPEVENT provide one of the few tools for assisting reviewers in checking the unexpanded products. Since the PA description fields are not passed through into the expanded products, the summary is also the only easy way to spot significant activities -- the other means, the timeline, is primarily used as an early planning tool and is not kept updated.

The reformatting capabilities of OPEVENT have also been used to provide management with summaries of sequence activities and to provide alternative re-formats of the Station Allocations File.

TELECOM SUBSYSTEM CONSTRAINT CHECKERS

Finding and organizing or reformatting data simply did not address some review problems. Constraints with complex rules, those depending on current spacecraft state, those requiring time calculations and those without an easily identified trigger generally exceeded the abilities of SKIMX and OPEVENT.

One such difficult constraint was the Telecom check that no spacecraft events occurred during a data outage. Data outages were triggered by three types of events: (1) data rate changes, (2) switching between coherent and non-coherent mode, a function of both a commanded spacecraft state and

An OPEVENT reformat of the Station Allocations File

```
*CREATION 94-222/18:37:05.000
*BEGIN 95-268/19:10:08.439
*CUTOFF 96-014/17:13:26.530
*TITLE STATION ALLOCATIONS FILE FOR JAJOE-5
95-268/22:28:50 STALOC,362A DSS 14 BOT: 95-268/23:13:50 EOT: 95-269/03:13:50 CFG 008S
95-269/18:53:43 STALOC,362B DSS 14 BOT: 95-269/19:53:43 EOT: 95-270/02:58:43 CFG 608I
95-271/18:53:28 STALOC,362C DSS 14 BOT: 95-271/19:53:28 EOT: 95-272/02:58:28 CFG 608I
95-272/18:23:21 STALOC,362D DSS 14 BOT: 95-272/19:23:21 EOT: 95-273/02:58:21 CFG 608I
95-274/18:23:06 STALOC,362E DSS 14 BOT: 95-274/19:23:06 EOT: 95-275/02:58:06 CFG 608I
```

An OPEVENT reformat of the Comet Shoemaker-Levy observation sequence

```
94-198/02:56:16 DLKCAP,364J S-Band Sup Bit Rate: 10
94-198/03:46:00 CMDNRO,480LC Dur: 08:36:00 Rate: 10 Desc: EVENT B BUFFER MRO PT 1
94-198/05:29:27 CMDRS,157JB Dur: +CDS 02:00:0 Desc: NIMS FRAGMENT C OBSERVATION
94-198/05:31:16 UTILITY,20JB Dur: CDS 96:00:0 Desc: NIMS RECORD FRAG C
94-198/05:31:28 SCITLM,176JB ELSMPW CHG: NO S_HI_LO: NONE Desc: FRAGMENT C OBSERVATION
94-198/05:31:28 TARGET,165JB Dur: +00:04:04 Body: JUPITER Desc: FRAGMENT C OBSERVATION
94-198/05:31:28 CSNOS,117JB Dur: +CDS 96:00:0 Desc: NIMS FRAGMENT C OBSERVATION
94-198/07:08:32 CMDRS,157JZ Dur: +CDS 02:00:0 Desc: NIMS FRAGMENT C OBSERVATION
94-198/07:24:40 UTILITY,20VM Dur: :05: Desc: SAFE S/P FOR SAS MAINT
94-198/08:11:14 DLKCAP,364K S-Band Sup Bit Rate: 40
94-198/09:41:13 SCITLM,61D176KD ELSLRS CHG: NO S_HI_LO: NONE Desc: UVS F-F ON JUPITER
94-198/09:49:18 SCITLM,611D176ID MCGIM4 CHG: NO S_HI_LO: NONE Desc: SSI_SL-9_IMPACT_D
94-198/09:41:02 UTILITY,20EA Dur: 2:15:00 Desc: SSI/UVS RECORD FRAG D
94-198/09:41:13 TARGET,165ID Dur: +02:02:22 Body: JUPITER Desc: SSI_SL-9_IMPACT_D
94-198/09:41:13 SMOS,118ID Dur: +CDS 116:78:0 Desc: SSI_SL-9_IMPACT_D
94-198/09:41:13 INITRS,129ID Dur: +CDS 01:01:0 Desc: SSI_SL-9_IMPACT_D
94-198/09:56:14 DLKCAP,364L S-Band Sup Bit Rate: 10
```

Figure 3 - OPEVENT output examples

the presence of an uplink to the spacecraft, and (3) station Begin-Of-Track. Outage duration depended on the data rate and was expressed as a probability of successful lockup. The faster and less restrictive lockup time applied only to certain events and only during certain mission phases. Station Begin-Of-Track did not have a separate and unique line in the review product.

OUTCHK for "Outage Check" was the program written to perform this task. It had to do all the following:

- track the spacecraft data rate and coherency mode,
- determine when station Begin-Of-Track occurred and "trigger" an outage for it,
- resolve overlapping station coverage,
- resolve overlapping data outages,
- identify all data outages and compute their durations, and
- identify any spacecraft activity in any outages discovered.

As a test, the time to hand check a particular sequence for data outages was recorded. It took the analyst 14 hours to complete. OUTCHK was then run on the same sequence. Its elapsed time, including the time to print its report, was 12 minutes. A comparison of the two checks showed that OUTCHK had correctly identified all data outages found by the analyst, had correctly timed all data outages including several the analyst had not, and had found three more outages that had been missed in the hand check. This represents a seventyfold decrease in checking time with increased accuracy as well.

OUTCHK was written part time in about three weeks with fewer than 80 hours invested. Even with updates, it still has fewer than 120 hours invested while the estimated time savings run well over 1000

hours. This represents over an 800% return on time invested.

Two other related tools are also used for difficult telecom constraint checking, one to verify events have ground station coverage and the other to verify the data rate is supportable. Combined with OUTCHK and SKIMX, these tools have cut average Telecom review time by a factor of about twelve: what once took a week is now done in an afternoon.

By launching the checking programs from a batch file, still more of the user's time can be saved. Typical sequences take from five to fifteen minutes to process through the Telecom sequence checking batch file. During this time, the user is free for other duties.

UTILITY PROGRAMS

The sequence review effort has also been aided by several small utility programs. The first of these, DAYS, converted calendar dates to and from day-of-year and computed the day-of-the-week. DAYS covers the years 1583 (the beginning of the Gregorian calendar) through 9999. Two digit years are assumed to lie between 1980 and 2079. Typing "DAYS TODAY" returns the current date in both calendar and day-of-year formats (or an error message if the computer's clock isn't current).

TIMECALC adds and subtracts times in hours:minutes:seconds format. It has a memory store and recall function that is ideal for adding or subtracting a one-way light time from a series of number.

PA_RENUM was originally written to change the PA identification suffixes after a file had been created or edited by cutting and

pastings PAs. At the request of several users, it was expanded to also renumber sub-PAs and commands. PA_RENUM isn't often needed but when PAs must be renumbered, the only alternative is change each suffix manually.

COMMON CHARACTERISTICS

Shortly after the creation of SKIMX, it was apparent that there was no easy means for users to verify they had the latest version. This led to the definition of a common user interface, the general format being shown in Figure 1, the SKIMX help screen. All programs show date and version, all accept options before filenames, all use the forward slash as an option switch character, and all respond to "/H" with a standard help screen.

To facilitate batch file operation, all programs accept command line input. If required information is missing, the user will be prompted to supply it. Programs verify that the specified files exist and will re-prompt if necessary.

Programs benefit from a "toolkit" of utility and support routines, about half written in assembler, that provide services such as time addition and subtraction, parsing the

command line, tokenizing a logical line, verifying file existence, setting up help screens and screen colors, and modeling various spacecraft and ground resources. The toolkit both enables and enforces much of the commonality among the programs.

Most of the programs also have accompanying "DOC" files that expand on what each program does, how it does it, what its options are, and often includes review tips or other usage information.

OBSERVATIONS AND CONCLUSIONS

This suite of programs shows worthwhile savings of time and effort can be achieved with a relatively small programming effort. The problems and programs may be relatively small but that doesn't mean insignificant: for example, the Telecom unit will use these programs instead of hiring two additional analysts during the intensive Orbital Operations phase of the mission.

By finding and organizing data, by presenting it in more easily understood ways, and by performing rote logical tests and checks, these small scale sequencing review tools have dramatically reduced the time and effort required of this formerly all manual process.

SAFETY ASPECTS OF SPACECRAFT COMMANDING

Mr. N. Peccia
European Space Operations Centre (ESOC),
Robert-Bosch-Str. 5, 64293 Darmstadt, Germany

ABSTRACT

The commanding of spacecraft is a potentially hazardous activity for the safety of the spacecraft. Present day control systems contain safety features in their commanding subsystem and in addition, strict procedures are also followed by operations staff.

However, problems have occurred on a number of missions as a result of erroneous commanding leading in some cases to spacecraft contingencies and even to near loss of the spacecraft. The problems of checking commands in advance are increased by the tendency in modern spacecraft to use blocked/time-tagged commands and the increased usage of on-board computers, for which commands changing on-board software tables can radically change spacecraft or subsystem behaviour.

This paper reports on an on-going study. The study aims to improve the approach to safety of spacecraft commanding. It will show how ensuring "safe" commanding can be carried out more efficiently, and with greater reliability, with the help of knowledge based systems and/or fast simulators.

The whole concept will be developed based on the Object-Oriented approach.

Keywords: Telecommanding, Safety, Predictive Knowledge, Object Oriented

1. INTRODUCTION

This paper gives an interim report on a study of the safety aspects of spacecraft commanding. The overall aim of the study is to demonstrate the feasibility of model-based command checking.

The study examines user requirements for such a system. Based upon these requirements the functional requirements and the architectural design is being produced. Finally a prototype of at least the basic mechanisms of the design will be developed and demonstrated.

The whole concept will be developed based on the Object-Oriented approach. The common environment must provide the different spacecraft users with the same kind of user interface facilities in order to offer a consistent operational environment.

The ESA SCOS II system (under development) is being taken as the reference system to be interfaced. SCOS II will operate in a hardware and basic software environment that is vendor-independent.

The function of a SCOS II (Spacecraft Control Operations System) system are seen as a collection of independent models of various parts of the spacecraft and the ground segment. SCOS II will therefore provide a library of 'building blocks', which can be combined in various ways to produce the overall model. To allow this to be done easily,

object-oriented software engineering technology has been updated for analysis and implementation of SCOS II. Specifically the Coad/Yourdon method and the C++ programming language have been chosen.

Not all missions are the same, which led to make modifications to the library building blocks to be used in a specific mission. Using an object-oriented technique known as 'inheritance', it will be possible to provide a customised building block for a given mission, whilst maintaining the same interface.

The SCOS II system will be hosted on a Local Area Network (LAN) of distributed UNIX workstations. Some centralised services of the system will be provided by server processors (client-server concept). The use of a distributed system also offers advantages in terms of system availability and failure tolerance.

An initial delivery of the SCOS II system is foreseen for end 1994. It will contain basic functions of the system. The Huyghens-Cassini, Envisat and XMM spacecrafts will make use of the SCOS II infrastructure software.

2. BACKGROUND

2.1 CURRENT STATUS

It is useful to describe first the general ESOC approach to handling of commands by the Mission Control System (MCS) for currently supported missions, which however can be significantly modified for specific missions.

- Command Preparation Checking

In the command database to determine allowable ranges of parameters, etc.

Automatic checks on "manual or automatic stacks of commands" at time of entry of command parameters.

- Pre-Transmission Validation (PTV) of commands

The normal route for all commands involves a pretransmission validation (PTV) before the command is passed to the ground station for uplink. PTVs are defined in the command database.

Checks normally performed in PTV are:

TC configuration (e.g. check that the TC subsystem has not been disabled)

Spacecraft and subsystem status, as computed from incoming telemetry parameters. The TM parameters and the mode computation are specified in the command database. PTV can be disabled by the operator and by the command source. PTV does not provide for limit checking or other checks of individual command parameters or of parameters sets.

- Checking of command contents

This is not a standard facility on the ESOC Mission control system; it varies from one mission to the other. Any such checks performed are limited since :

They are only static limit checks (e.g. lower and upper limits) on individual parameters.

Many commands cannot be checked against fixed limit checks alone because of interdependence between

parameters.

The correctness of multiple command activities cannot correctly be checked.

Command parameters are obviously important parts of a command and for some commands the value of the parameters can be vital for the spacecraft safety.

No on-line checking of combination of commands and command parameters nor pre-execution validation of commands against predicted spacecraft status is carried out or envisaged for current "in flight" or near future missions (ERS-2, ISO, CLUSTER)

2.2 PLANNED DEVELOPMENTS

Future missions to be supported by the ESA SCOS II (under development) will be controlled using approaches to commanding which are likely to differ significantly from the current one. Special services should be provided to increase the safety of commanding. Two additional types of conditions will be used in making these safety checks :

- a predicted set of conditions in the on-board status applicable at the (future) time of execution (and not necessarily at the time of release)
- a set of "operational constraint" rules to be obeyed following command execution.

These checks are carried out based on a prediction of the on-board status at the planned execution time (Predictive Knowledge). Thus a capability to propagate the on-board status needs to be available for all the potential

sources of commands (Manual Command, on-board Master Schedule and ground automatic command files).

Predictive Knowledge allows the prediction of future states of the system under control (satellite modes, measurements, etc) from a "known initial state" and taking into account planned commanding activities and predicted mission events.

This Predictive Knowledge can be produced in two ways :

- Evolution of the system in the absence of any commanding activity (Evolution Predictive Knowledge)
- Evolution of the system under the influence of Telecommanding (commanding Predictive Knowledge).

In addition , detected or predicted on-board autonomous actions can be treated in an analogous manner to telecommand actions. Specific attention shall be given to the handling of asynchronous on-board actions (these are often the result of failures and related on-board corrective actions).

This knowledge may be in the form of algorithmic, heuristic or mathematical models. The predictions will be required both over a short term (e.g. for satellite health monitoring) and over a long term (e.g. to validate a plan spanning several days).

3. OVERALL APPROACH

The study has the following steps :

- Problem, methodology analysis and evaluation of the ESOC requirements

- Software Requirements Phase
- Architectural design of the system
- Prototyping and demonstration of the basic design

4. BASIC REQUIREMENTS

The central idea is the use of a Model of the satellite. The definition of this Model of the spacecraft is the most critical part of the study. It is of course of major importance that the real system is modelled as close as possible. The Model has to run quickly to allow predictions for some time in the future (typically 48 hours for EURECA) in case of on-board time-tagged commands checking.

During operations this Model must be capable to be connected to (or be a part of) the spacecraft control system whereas during the validation phase to the Expert Tool system for FOP (Flight Operation Procedures) production. The following scenarios are considered :

- "On-line" : The Model is part of the mission Control System (SCOS II), and each command is checked (.e.g for consistency with the modelled "image" of the spacecraft) before being released for uplinking to the spacecraft.
- "Near Realtime" : The set of commands to be sent to the spacecraft (either from Manual Command or Automatic Schedule) are previously uplinked (or could be done "directly" by the system) to the Model respecting the "timelining" (timing and ordering of activities). This should allow the

user to view the changing state of the Model while it is being "operated" and will also perform concurrent safety checking and validation of the operations in each scenario exercised.

The command validation function (in the Model) should use the Predictive Knowledge of the impact of the command (together with any other planned or predictable actions) to cause the rejection of a TC based on predicted effects which violate any health criteria. This information will be passed to SCOS II, which will inhibit the uplink of the command.

During Planning validation (sequence of commands as output of the mission planning) it will normally be necessary to propagate the mission state during the planning interval in order to :

- establish that pre- and post-conditions for activities are fulfilled
 - to confirm that health criteria are continuously satisfied during the planning interval
- "Off-line": User selected Flight Control Procedures (FCP), Contingency Recovery Procedures (CRP) or timelines shall be applied to the Model in order to validate the operations (Procedure Validation).

The following Model operating scenario could be envisaged :

- The Model is initialised with the available TM in order to synchronize the its internal state with the real state of the spacecraft.

- As a second step the Model is let to evolve by means of a prediction generation function, taking into account the planned on-board mission events and / or commanding activities.

The Model could also be used as follows:

- Verification of commands executed in the past (e.g. comparison of playback telemetry and predicted mission status)
- Monitoring functions including the display of predicted telemetry parameters during "non visibility" periods.
- Diagnosis : The deviations of predicted values from the expected ones could be detected and analysed. To this aim a knowledge not completely contained in the Model is required (e.g. diagnosis charts and fault trees contained in the spacecraft Operations Requirement Handbook)

The Model is a central concept on this study. It predicts mission states related to future mission times. The selected approach is based on two types of model :

- A complete Model for near real time and off-line scenarios

Detailed spacecraft subsystems models are developed at ESOC for each mission, as part of spacecraft dynamic simulators used for validation of control system software and Flight Control Procedures as well as for staff training. This type of simulators run 30 times faster than real time when running on an ALPHA VAX platform.

The Model is extracted from an

existing spacecraft simulator. It shows the best precision in the states prediction in spite of a lower speed. For this reason it will be used when greater accuracy is required.

- A simplified Model using knowledge based techniques for real time scenarios

High speed performances are met but a lower accuracy in the computation of predicted states is shown. The Model is build up extracting the mission information from a selected repository (e.g. the Mission base in SCOS II) and adding manually the missing information.

This two Model approach should be used for model validation. In order to trust such a system strong emphasis should be put into the verification and validation of the models.

5. SOFTWARE REQUIREMENTS

The Software Requirements Document defines the functional requirements of the system according to the SCOS II Development Standards.

The document covers the system functionality, outlines standards for input and output data which they should handle, and shows how they should interface to the wider operational environment in the future.

The whole concept is being developed based on the Object Oriented approach. The expected benefits of OO for the Model of the spacecraft are :

- natural modelling of the architecture of the spacecraft

- flexibility (via properties of inheritance and polymorphism)
- different levels of abstraction, permitting viewing of the Model at different levels of complexity
- potential of reusability

The design and implementation of the system should support the Object Oriented Paradigm. The system should interface with SCOS II and should be based on "open architecture" so as to allow for additionally functionalities via added modules.

The system has to be based on UNIX, and developed and maintained on SUN platforms. However it will be capable to run on any of the main line of available UNIX platforms (e.g. SUN, HP, IBM and Digital).

The main constraints are the following:

- The system should access the SCOS II Mission Information Base to derive the Predictive Knowledge, the operational constraints and the execution verification criteria. The user should not insert significant additional information.
- The system should not cause detectable performance degradation on SCOS II real operations.
- The system should have the capability of synchronizing its internal Model status with the real spacecraft data and status.

After an Object Oriented Analysis of the system the following OO diagrams were produced :

- Model OO Diagram

It focuses both on the Model related abstraction level and on the high level internal decomposition of the system. The two Model approach is introduced as a keypoint in the whole system organization. A "complete" Model cooperates with a "simplified" one to obtain the best performances in terms of accuracy and computation speed.

- Database level OO Diagram

It shows the database internal organization focusing on the elements needed to build the Model (e.g. system element, activity, application criteria of system elements, verification and validation criteria of activities)

- Operational Context Diagram

It describes the different operational scenarios, particularly the real time case which is the most complex one

The following interfaces are envisaged:

- SCOS II command stacks (e.g. manual and automatic stacks)
- SCOS II Mission Implementation Base
- Display of system outputs on SCOS II Man Machine Interface
- Telemetry acquisition from SCOS II telemetry Processor
- Flight Operations Procedures Set Tool to read and process Flight Operation Procedures in the off-line case
- Model of an existing spacecraft

simulator to be used as the "complete" Model

6. CONCLUSIONS

At the time of writing this paper (July 1994) the Architectural Design Phase is in progress. This phase defines the architectural concept, considering all functions and also how the system should support future expansion and modification of functionality. The Architectural Design Document should include detailed descriptions of all critical design elements, such as data storage architecture and access methods, control data structures, knowledge representation and all external data interfaces.

During a second phase the study should produce the following :

- A detailed Design and implementation of a prototype. A spacecraft subsystem should be identified to develop such a prototype (a partial Model). It will be integrated with the SCOS II system at ESOC
- A Detailed Design Document (DDD) of the prototype
- A Software User Manual (SUM) of the prototype

This study aims to produce a prototype to improve the approach to safety of spacecraft commanding by using model-based command checking systems. This philosophy can then be used for upcoming ESA missions such as those of XMM and Integral.

7. REFERENCE DOCUMENTS

1. ESA Software Engineering Standards, ESA-PSS-05-0 Issue 2, February 1991
2. SCOS II Development Standards, SCOS II-CON-001, Issue 1, January 1993
3. SCOS II User Requirement Document, Issue 1, March 1993
4. SCOS II Software Requirement Document, Draft 2, May 1993
5. SIMSAT Simulator Designers Manual, issue 1, May 1993
6. Study "Safety Aspects of Spacecraft Commanding" Problem Analysis and Concept Definition, Dataspazio, March 1994
7. Study "Safety Aspects of Spacecraft Commanding" Software Requirements Document, Dataspazio, June 1994
8. "Object oriented analysis", Peter Coad & Ed Yourdon, 2nd edition, Prentice Hall, 1991

The Development and Validation of Command Schedules for SeaWiFS

Robert H. Woodward
General Sciences Corporation
Laurel, Maryland

Watson W. Gregg
NASA/Goddard Space Flight Center
Greenbelt, MD

Frederick S. Patt
General Sciences Corporation
Laurel, Maryland

ABSTRACT

An automated method for developing and assessing spacecraft and instrument command schedules is presented for the Sea-viewing Wide Field-of-view Sensor (SeaWiFS) project. SeaWiFS is to be carried on the polar-orbiting SeaStar satellite in 1995. The primary goal of the SeaWiFS mission is to provide global ocean chlorophyll concentrations every four days by employing onboard recorders and a twice-a-day data downlink schedule. Global Area Coverage (GAC) data with about 4.5 km resolution will be used to produce the global coverage. Higher resolution (1.1 km resolution) Local Area Coverage (LAC) data will also be recorded to calibrate the sensor. In addition, LAC will be continuously transmitted from the satellite and received by High Resolution Picture Transmission (HRPT) stations. The methods used to generate commands for SeaWiFS employ numerous hierarchical checks as a means of maximizing coverage of the Earth's surface and fulfilling the LAC data requirements. The software code is modularized and written in Fortran with constructs to mirror the pre-defined mission rules. The overall method is specifically developed for low orbit Earth-observing satellites with finite onboard recording capabilities and regularly scheduled data downlinks. Two software packages using the Interactive Data Language (IDL) for graphically displaying and verifying the resultant command decisions are presented. Displays can be generated which show portions of the Earth

viewed by the sensor and spacecraft sub-orbital locations during onboard calibration activities. An IDL-based interactive method of selecting and testing LAC targets and calibration activities for command generation is also discussed.

INTRODUCTION

The Sea-viewing Wide Field-of-view Sensor (SeaWiFS) is scheduled to be launched aboard the SeaStar satellite in 1995 as one of the Earth Probes projects in Mission to Planet Earth. The principal goal of the SeaWiFS mission is to provide a global set of ocean chlorophyll concentration (ocean color) every four days. To achieve this goal, SeaStar will be launched into a nearly circular, sun-synchronous orbit at 705 km. The sensor will be mounted on a tilting platform which can be pointed 20 degrees fore or aft of nadir as a means of avoiding sun glint. Table 1 summarizes some of the key SeaStar/SeaWiFS specifications. Two sets of data will be recorded onboard and subsequently downlinked at the Wallops Flight Facility using the S-band frequency: Local Area Coverage (LAC) which has 1.1 km nadir resolution and a 2800 km swath width, and Global Area Coverage (GAC) which is LAC subsampled for every fourth pixel and every fourth line over a 1500 km swath. Recorded LAC data is used primarily for sensor calibrations. In addition, LAC data will be continuously broadcast using the L-band frequency to High Resolution Picture Transmission (HRPT) stations.

Table 1. SeaStar/SeaWiFS specifications.

Orbit characteristics:	
sun synchronous	
descending noon equatorial crossing	
98.2 degree inclination	
98.9 minute orbital period	
0.02 eccentricity	
Instrument characteristics:	
20 degree fore and aft sensor tilt	
116.6 degree scan width (LAC)	
8 bands (visible and near infrared)	
10 bit digitization	
6 scans/second	

In a unique agreement between the private sector and NASA, Orbital Sciences Corporation (OSC) assumes responsibility for building, launching, and operating the instrument (SeaWiFS) and the spacecraft (SeaStar). NASA will then obtain data from SeaWiFS by means of a data purchase from OSC. This novel agreement was designed to deliver the spacecraft at a reduced cost and over a tighter schedule. To assist in meeting this goal, OSC has subcontracted Hughes/Santa Barbara Research Center (SBRC) to build the radiometric instrument. NASA/Goddard Space Flight Center (GSFC) is responsible for developing sensor and spacecraft command sequences to maximize the scientific usefulness of the data. The primary link to OSC is through SeaWiFS Mission Operations (MO) at NASA/GSFC which, among other tasks, is charged with the responsibility of ensuring the collection of GAC, LAC, and calibration data through the submission of weekly and daily command schedules to OSC.

Because of a stringent set of cascading directives developed for SeaStar/SeaWiFS operations, the problem of developing command schedules lends itself to a hierarchical set of algorithms. This in conjunction with an accurate orbit model and other operational inputs such as downlink times and instrument tilt times permit the development of modular software to generate complex command schedules. The command scheduler is similar

in nature to the more generic rule-based expert system discussed in Hughes *et. al.* (1993). Figure 1 shows a generalized flow chart illustrating some of the logic used by the command scheduler. The scheduler is propagated in one second time increments to reflect the minimum command update frequency of the SeaStar system. This update frequency is also used in orbit propagations which are read by the command scheduler.

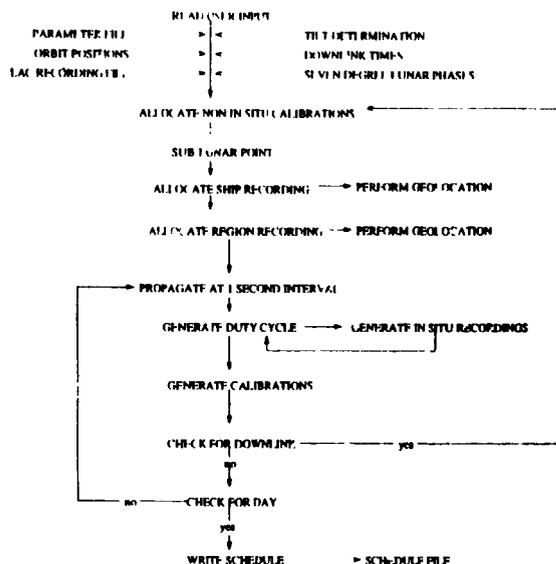


Figure 1. Flowchart illustrating the general processing stream of the command scheduler.

Description of scheduling rules

The primary goal of this mission is to obtain global coverage of ocean chlorophyll every four days. This is followed in order of importance by the acquisition of the recorded LAC which will be used primarily for instrument calibration and characterization. A summary of the SeaWiFS mission goals is listed below in descending order:

1. Record a global set of GAC data onboard
2. Transmit and acquire all recorded GAC data on the ground
3. Record LAC data of calibration targets with the following priorities:
 - lunar calibration
 - calibration using irradiances reflected off diffuser plate
 - detector performance
 - interchannel gain measurements
 - pre-selected ship/buoy and region targets
4. Transmit and acquire recorded LAC data on the ground
5. Broadcast real-time LAC data

AUTOMATED PRODUCTION OF COMMAND SCHEDULES

The command scheduler is a modularized FORTRAN program which reads previously generated orbit position and creates time-ordered command schedules that meet the mission goals. FORTRAN was chosen as the software language for the production code to maintain consistency with existing orbit propagation models which are coded in this language (Patt *et al.*, 1993). Orbit positions are read by the scheduler and are highly integrated into the functionality of the program. Orbit positions and instrument timing times are produced by separate stand alone modules which are executed prior to initiating a scheduler run. These stand alone programs allow flexibility in creating scheduler inputs.

Outputs from the command scheduler are produced as daily and weekly ASCII files consisting of a time-ordered list of spacecraft and instrument commands. In addition, LAC and GAC recording log and error log files are also written by the scheduler. Commands are abbreviated to 16 byte strings to permit portability with PC programs which are currently under development. Table 2 lists all the commands produced by the scheduler. The American Standard Code for Information Interchange (ASCII) was chosen as the output format in part to allow quick visual verification.

Table 2 SeaWiFS commands

COMMANDS	EXPLANATION
ACS Opn Loop On	Initiate lunar cal maneuver
ACS Opn Loop Off	Stop lunar cal maneuver
Chg Gain Band	Change gain
Chg TDI Band	Change detector configuration
Chg Tilt Aft	Point sensor aft
Chg Tilt Forward	Point sensor forward
Chg Tilt Nadir	Point sensor nadir
Recorder Dmp On	Initiate downlink
Recorder Dmp Off	Finish downlink
Elet Lun Cal On	Initiate lunar cal
Elet Lun Cal Off	Finish lunar cal
Elet Sol Cal On	Initiate solar cal
Elet Sol Cal Off	Finish solar cal
Elet TDI Cal On	Initiate detector cal
Elet TDI Cal Off	Finish detector cal
Elet Turn On	System electronics on
Elet Turn Off	System electronics off
GAC Partition	Partition recorder
GAC Recorder On	Initiate GAC recording
GAC Recorder Off	Finish GAC recording
LAC Recorder On	Initiate LAC recording
LAC Recorder Off	Finish LAC recording
LAC Xmitting On	Initiate LAC transmission
LAC Xmitting Off	Finish LAC transmission
Lun Cal Pitch Rt	Set lunar cal pitch rate
L-Bd Xmitter On	Turn on L-band transmitter
L-Bd Xmitter Off	Turn off L-band transmitter
Rst Nml Pitch Rt	Reset pitch rate
Rst Tilt Aft	Reset tilt to aft
Earth Mode On	Initiate Earth viewing mode
Sol Cal Mode On	Prepare for solar cal
S-Bd Xmit On	Turn on S-band transmitter
S-Bd Xmit Off	Turn off S-band transmitter

Table 3 Command sequences illustrating a typical duty cycle.

L-Bd Xmitter On	27	1	1994	84	604	2999	72.32	72.68
Elet Turn On	17	1	1994	84	604	2999	72.32	72.68
Earth Mode On	30	1	1994	84	639	2999	72.32	72.68
Rst Tilt Aft	29	20	1994	84	654	2999	72.32	72.68
Chg Gain Band 1	3	1	1994	84	659	2999	72.32	72.68
Chg Gain Band 2	3	1	1994	84	659	2999	72.32	72.68
Chg Gain Band 3	3	1	1994	84	659	2999	72.32	72.68
Chg Gain Band 4	3	1	1994	84	659	2999	72.32	72.68
Chg Gain Band 5	3	1	1994	84	659	2999	72.32	72.68
Chg Gain Band 6	3	1	1994	84	659	2999	72.32	72.68
Chg Gain Band 7	3	1	1994	84	659	2999	72.32	72.68
Chg Gain Band 8	3	1	1994	84	659	2999	72.32	72.68
Chg TDI Band 1	4	0	1994	84	660	2999	72.32	72.68
Chg TDI Band 2	4	164	1994	84	660	2999	72.32	72.68
Chg TDI Band 3	4	0	1994	84	660	2999	72.32	72.68
Chg TDI Band 4	4	26	1994	84	660	2999	72.32	72.68
Chg TDI Band 5	4	0	1994	84	660	2999	72.32	72.68
Chg TDI Band 6	4	74	1994	84	660	2999	72.32	72.68
Chg TDI Band 7	4	0	1994	84	660	2999	72.32	72.68
Chg TDI Band 8	4	161	1994	84	660	2999	72.32	72.68
GAC Recorder On	20	1	1994	84	660	2999	72.32	72.74
LAC Xmitting On	24	1	1994	84	664	2999	72.32	72.68
LAC Recorder On	22	1	1994	84	868	2999	60.88	60.36
LAC Recorder Off	21	0	1994	84	78	2999	59.14	58.55
Chg Tilt Forward	6	-20	1994	84	180	2999	2.57	1.07
GAC Recorder Off	19	0	1994	84	3063	2999	-69.47	72.72
LAC Xmitting Off	23	0	1994	84	3064	2999	-69.52	72.78
L-Bd Xmitter Off	26	0	1994	84	3066	2999	-69.52	72.78
Elet Turn Off	16	0	1994	84	3067	2999	-69.52	72.78

Table 3 shows a command schedule segment for a typical duty cycle. On each line, the abbreviated commands appear on the left followed by a command code, configuration code, year, day of year, second of day, sub-orbital latitude, and sub-orbital solar zenith angle. Dummy values are used in this example for the command codes. The orbital duty cycle commences on each orbit when the solar zenith angle of the sub-orbital point exceeds a

threshold value which is currently set to 72.7 degrees for a nominal SeaWiFS orbit. This provides balanced solar zenith angle coverage for a required 40 minute duty cycle per orbit.

Sun glint from the ocean surface can significantly contaminate radiances observed by remote sensors. SeaWiFS has the capacity to tilt 20 degrees fore or aft (toward the North Pole on the descending node) of nadir as a means of minimizing glint. On the descending orbit the instrument will be tilted 20 degrees aft as the duty cycle commences. Near the solar declination, the instrument will be tilted 20 degrees fore. Several tilting algorithms have been developed. The program TLTMNGLT minimizes sun glint by checking orbit position and sun angles to determine times of maximum sun glint. The instrument tilting times are then computed on an orbit-by-orbit basis. The program TLTMNFST provides a faster, less accurate determination of tilting time by using the same algorithm as TLTMNGLT to compute the orbital tilt time for the orbit closest to the midpoint of a day. The program then steps forward and backward in time using increments equal to the orbital period to determine other tilting times for an entire day. The current operational plan is to use the staggered tilting algorithm in the program STAGTILT which seeks to minimize sun glint and maximize Earth coverage using a four day cycle of shifting the tilt above the glint for two days and below the glint for two days (Gregg and Patt, 1994).

At the start of execution, the command scheduler prompts the operator for year, day of year, and number of days of the run. As an alternative, an operator can create a 'date.dat' file with the Unix command "date>date.dat". The scheduler checks if "date.dat" is present and contains the current date. If these conditions are satisfied, the scheduler extracts the date information and only prompts the operator for the duration of the run. In addition to these inputs, the scheduler is manipulated in part by inputs from a parameter file and daily LAC recorder files which are read by the scheduler. The former file contains values on scheduler operation specifications which change

infrequently; the latter file contains information on ship/buoy and region targets and calibration frequencies used in allocating the flight recorder. Ships and buoys are handled identically by the scheduler and will be referred to simply as ships from this point on.

The most challenging aspect of command scheduling logic involves the allocation of the LAC flight recorder partition. The overall recording priorities used in the recorder allocations are listed under item 3 of Table 2. Lunar calibrations have top priority followed by solar calibrations, detector performance assessment, and interchannel gains performance assessment. Earth targets (ships, buoys, and regions) have lowest priority with ships having priority over regions. Detailed descriptions of calibrations are found in Woodward *et. al.* (1993).

The daily LAC Recorder File (Table 4) is read by the scheduler during the processing when a nocturnal downlink is encountered for an ascending pass (local midnight downlink). The timing is done so as not to interfere with potential LAC recording events. Each ship in the file has a corresponding longitude, latitude, priority, and recording duration in seconds. Each region has corresponding starting and ending longitude and latitude (defining a rectangular box) and a priority. The weekly frequency of solar, lunar, interchannel gain, and detector calibrations are also specified. The scheduler uses this information for allocating LAC recording space for each of the next two downlink recording periods. Ships and regions are each assigned priorities; the lower the value, the more likely a target will be recorded. All viewed ships are allocated before any region is allocated. In other words, the target with the lowest priority number has recorder space allocated first, followed by the target with the next lowest number, and so on. This means that the scheduler looks over the entire recording period and allocates recorder space on the basis of target priority rather than on the basis of target view time.

Table 4 Example of a typical LAC recording file "In situ" precedes the number of ship targets which is followed by the ships with corresponding longitude, latitude, priority, and, recording duration (s) "Regions" precedes the number of region targets which is followed by the regions with corresponding longitudinal limits, latitudinal limits, and priorities. The weekly frequencies of solar, lunar, interchannel gain, detector calibrations are last

Calibration Targets				
1994 84				
In-situ				
11				
Clark's Buoy	-156.3400	18.6700	1	30
Bermuda Buoy	-71.9000	32.1200	6	30
JGOFS	63.2500	19.4000	3	30
NOAA S Atlantic Bight	-77.5200	32.0300	4	30
NOAA Gulf of Cal	-107.2600	22.1100	5	30
S.Africa	10.2300	-32.8700	6	30
Galapagos	-92.6200	-3.2500	8	30
Gulf of Mexico	-86.8600	24.7900	3	30
Oregon St	-131.1400	45.7700	4	30
Navy Bering Sea	-175.5800	63.4200	10	30
Pacific	175.0000	1.0000	11	30
Regions				
4				
Sargasso Sea	-70.0000	-45.0000	20.0000	30.0000
Gulf of Mexico	-110.0000	-80.0000	17.0000	31.0000
Galapagos Area	-105.0000	-75.0000	-15.0000	0.0000
Micronesia	135.0000	180.0000	0.0000	15.0000
Solar Calibration				
14				
Lunar Calibration				
1				
Intergain Calibration				
14				
TDI Check				
14				

Recorder partitioning

The onboard flight recorder has a storage capacity of 119.2 mb. A daily determination of GAC recording requirements is made by the scheduler during the processing of each local midnight downlink. This involves summing the total and partial duty cycles for the two subsequent recording periods. Using the maximum of these values, a section on the recorder is reserved for GAC and the remainder is reserved for LAC. Since this partitioning is performed once a day, the recorder is not fully utilized for the downlink with the shorter GAC recording period.

Lunar Calibrations

Current plans for onboard calibrations include a backorbit maneuver to scan the lunar surface near a full Moon event (using the closest orbit to a seven degree lunar phase angle). The seven degree phase angle was chosen as a means of enhancing the calibration consistency. A full Moon is defined at the point of the Moon's closest approach to the anti-solar point. The Moon was chosen as a calibration source due to its reflective stability compared to onboard calibration sources which can be expected to degrade with time. During lunar

calibrations the spacecraft will pitch 360 degrees on the backorbit spanning a 40 minute period thus allowing the Moon to come into view of the sensor. This operation can, at best, be performed twice a month when the Moon is coming into and out of full phase.

Solar Calibrations

Unlike lunar calibrations which are restricted to particular orbits, solar calibrations can, in principle, be performed on any orbit. However to maintain consistency, solar calibrations are constrained to the first orbit of the GMT day and the orbit midway between the local midnight and local noon downlink. Solar calibrations are scheduled to occur as the spacecraft sub-orbital point makes its closest approach to the South Pole. For this operation the instrument is commanded to tilt aft 20 degrees and LAC data is collected along the back scan where the sensor views a solar diffuser plate. It is expected that these calibrations will provide high frequency instrument calibrations anchored by the more stable lunar calibrations.

Detector and interchannel gain checks

In general, these calibrations are identical to solar calibrations in terms of spacecraft location and sensor tilt configuration. The detector check will involve sampling each of the four detectors for each band as well as a combination of all four while scanning the solar diffuser plate. Interchannel gains will be checked by applying an electronic calibration pulse to each detector following the diffuser scan.

In situ calibrations

Recording of in situ targets for instrument calibrations involve the most complicated logic in the scheduler. The basic concept is to record data over a target coincident with the recording of data on a ship. Accurate geolocation algorithms are essential for the task of precisely recording specified coordinates on the Earth's surface. Geolocation algorithms which assume

an ellipsoidal Earth and employ vector and matrix computation to enhance efficiently are used by the scheduler (Patt and Gregg, 1994). These algorithms were implemented and tested in the AVHRR/Pathfinder project.

Among the complexities with in situ recordings are Earth targets with overlapping recording periods, differing tilt configurations, variable record times and target priorities, and conflicts with HRPT visibility masks. LAC recording is blocked when an HRPT station is in view of the satellite since these data can be obtained through agreements with the HRPT facilities. In addition instrument tilts are deferred if a conflict occurs with a ship target. All these factors play a role in the allocation algorithms. All ship targets in view of the sensor scan are recorded as long as recorder space is available. The duration of each ship recording is specified in the LAC recording file. Any remaining recording space is then used for recording scans of region targets. A region is recorded as long as the central pixel of the scan is within the rectangular region area. Default regions are specified in the parameter file to insure complete usage of the LAC partition in the flight recorder. The size and location of the default regions are chosen by the Project Scientist by considering downlink orbits and viewing geometries.

SCHEDULE VERIFICATION AND DISPLAY

The Interactive Data Language (IDL) was used to produce software tools for the graphical display command schedule performance. IDL was chosen in part since this package provides tools for relatively easy development of graphical user interfaces (GUI's). These interfaces allow quick and mostly error-free updates of inputs to the verification programs.

Rapid Verification of the Recording of LAC Targets

An IDL package named PLOTDOWN (plot LAC recording for downlinks) was created to acquire a quick-look at the budgeting of LAC

recorder space. Figure 2 shows the GUI for PLOTDOWN. In general, an operator selects the input files which specify the schedule, orbit propagation, and downlink times, and chooses one of the following types of plots:

PLOT ORBITS - plots only orbit tracks

PLOT ALL LAC SCANS - plot all LAC recording

PLOT IN SITU SCANS - plot ship and region recordings

PLOT SOLAR SCANS - plot spacecraft position for solar calibrations

PLOT LUNAR SCANS - plot spacecraft location for lunar calibrations

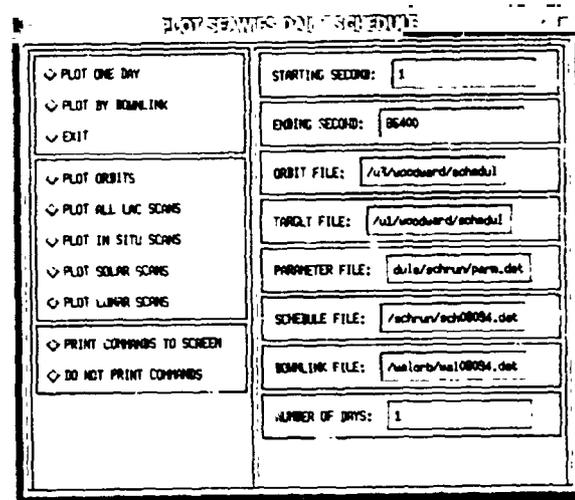


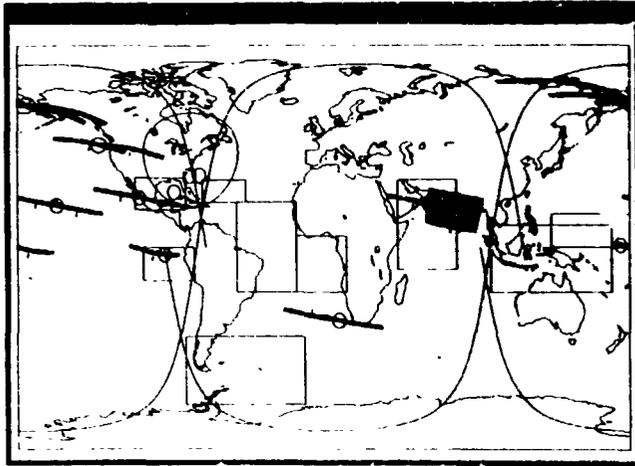
Figure 2. GUI for the program PLOTDOWN. An operator selects input files and plotting options to create plots of LAC scans.

A separate window is then created with an equi-rectangular projection of the Earth's continents and the specified type of plot is produced on this projection (Figure 3). This makes it possible for an operator to visually inspect the performance of the LAC partition in the onboard recorder.

Figure 3 illustrates two examples which illustrate the effects of some rules used in constructing command schedules with regard to in situ targets. Figure 3a shows that all the ships are recorded except those within the GSFC visibility mask. Figure 3b shows an unrecorded ship near the west coast of South America by the Galapagos Islands. This

occurred as a result of lunar, solar, detector, and interchannel gain calibrations which supersede the ship during this recording period. In addition, the Galapagos ship was given a lower priority than the other ships that are viewed and recorded. The figures also illustrate another consideration for scheduling in situ recordings: due to the nature orbit tracks for polar orbiting satellites, ships and regions at higher latitudes have a higher recording frequency.

a)



b)

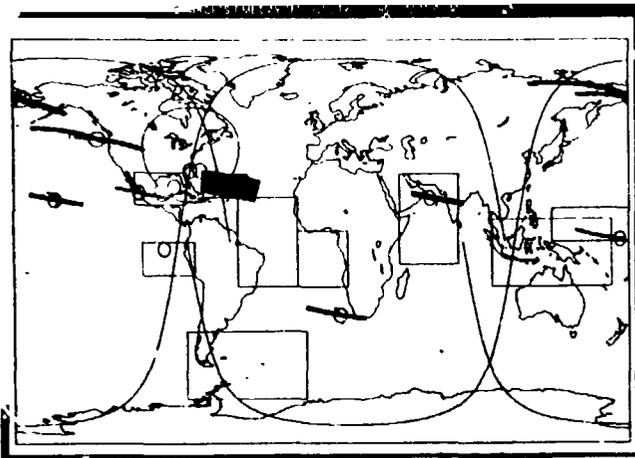


Figure 3. Two plots produced by PLOTDOWN illustrating LAC scans of the Earth's surface. Ships appear as small circles, regions as rectangles, HRPT visibility mask as a large circle. The orbit tracks for the two downlink orbits are also plotted.

Detailed Verification of Duty Cycle

A comprehensive examination of scheduling activities is essential to assure that the spacecraft/sensor systems are functioning properly. To assist in evaluating the command schedule an IDL package named COLOR_IT (create color-coded plots) was created. This utility can be used to produce a color-coded plot of the daily spacecraft and sensor operations. This allows for a visual inspection of the activities impacting the recorder including all GAC and LAC recordings. In addition, other aspects of the scheduling such as duty cycle initiation, Earth coverage, and tilt times can be visually verified. Figure 4 shows the GUI for COLOR_IT. An operator can first create the color palette to be used for differentiating scheduled activities. Input files can then be selected and a plot created.

PROT SEAVES DATE SCHEDULE

STARTING SECOND: 1	LAND COLOR: 103
ENDING SECOND: 06400	OCEAN COLOR: 102
ORBIT FILE: /u3/woodward/schedu1	GAC OFF LAND COLOR: 122
TARGET FILE: /u1/woodward/schedu1	GAC OFF SEA COLOR: 47
SCHEDULE FILE: /u1/woodward/schedu1	TERMINATOR LAND COLOR: 150
PARAMETER FILE: /u1/woodward/schedu1	TERMINATOR SEA COLOR: 1
<input type="checkbox"/> SELECT COLOR TABLE <input type="checkbox"/> START <input type="checkbox"/> EXIT	TARGET COLOR: 220
	SOLAR CAL COLOR: 214
	LUNAR CAL COLOR: 215
	GAC COLOR: 218
	LAC COLOR: 196

Figure 4. GUI for the program COLOR_IT. An operator selects input files and creates a color table.

INTERACTIVE UTILITY FOR MANAGING ONBOARD RECORDER

The IDL-based utility Calibration and validation Tool of Local Area Coverage (CATLAC) was developed by MO to assist the Calibration and Validation element of SeaWiFS in assigning LAC Earth targets and calibration frequencies (Woodward *et. al.*, 1994). In general, CATLAC permits a user to allocate and verify onboard LAC recorder space. This is done through an interactive display located in the GUI which allows an operator to graphically create ship and region targets and verify recording scenarios. Other calibration frequencies can also be specified and tested by spawning a command scheduler run and plotting the subsequent LAC recorder activity.

CONCLUSIONS

The utilities presented in this paper present some mechanisms for dealing with problems often encountered in the scheduling of activities with Earth-orbiting spacecraft. Many of the solutions are tailored specifically for SeaWiFS, but general applicability to other Earth orbiting systems is possible with minor modifications. Most of the IDL-based graphical utilities are in the process of being ported to separate graphics libraries on a Unix workstation and a PC.

REFERENCES

- Gregg, W.W. & Patt, F.S. Patt (1994). Assessment of tilt capability for spaceborne global ocean color sensors. *IEEE Transaction on Geoscience and Remote Sensing* (in press).
- Hughes, P.M., Shirah, G.W. Shirah, & Luczak, E.C. Luczak (October 1993). Advancing satellite operations with intelligent graphical monitoring systems. *The American Institute of Aeronautics and Astronautic Computing in Aerospace IX Conference*, San Diego, CA.
- Patt, F.S, Hoisington, C.M., Gregg, W.W., & Coranado, P.L. (1993). Analysis of selected orbit propagation modes for the SeaWiFS mission. *NASA Technical Memorandum 104566, Volume 11*, 16 pp.
- Patt, F.S. & Gregg, W.W. (1994). Exact closed-form geolocation algorithm for Earth survey sensor. *International Journal of Remote Sensing* (in press).
- Woodward, R.H., Barnes, R.A., McClain, C.R. McClain, Esaias, W.E., Barnes, W.L., & Mecherikunnel, A.T. (1993). Modeling of the SeaWiFS solar and lunar observations. *NASA Technical Memorandum 104566, Volume 10*, 26 pp.
- Woodward, R.H., Gregg, W.W., & Patt, F.S. (1994). CATLAC - calibration and validation analysis tool of local area coverage for the SeaWiFS mission. *NASA Technical Memorandum 104603*, 20 pp.

Operations

3. Control Centers		Page 615
OP.3.a	Towards Cheaper Control Centers <i>Lionel Baize</i>	617-624
OP.3.b *	The SILEX Experiment System Operations <i>B. Demellenne</i>	625-631
OP.3.c	EURECA Mission Control Experience and Messages for the Future <i>H. Hübner, P. Ferri, W. Wimmer</i>	633-640
OP.3.d	SCOS II: ESA's New Generation of Mission Control System <i>M. Jones, N. C. Head, K. Keyte, P. Howard, S. Lynenskjold</i>	641-647
OP.3.e	CCS-MIP: Low Cost Customizable Control Centre <i>Christian Labezin, Pierre Vielcanet</i>	649-654
OP.3.f	The IUE Science Operations Ground System <i>Ronald E. Pitts, Richard Arquilla</i>	655-662
OP.3.g	Efficient Mission Control for the 48-Satellite Globalstar Constellation <i>Dan Smith</i>	663-670
OP.3.h	Mini All-Purpose Satellite Control Center (MASCC) <i>Gérard Zaouche</i>	671-678

* Presented in Poster Session

Towards cheaper control centers

Lionel BAIZE

C.N.E.S - CT/TI/PS/SI BPI 1501, 18 av. E. Belin TOULOUSE Cedex - FRANCE

ABSTRACT : Today, any approach to the design of new space systems must take into consideration an important constraint, namely costs. This approach is our guideline for new missions and also applies to the ground segment, and particularly to the control centre. CNES has carried out a study on a recent control centre for application satellites in order to take advantage of the experience gained. This analysis, the purpose of which is to determine, *a posteriori*, the costs of architecture needs and choices, takes hardware and software costs into account and makes a number of recommendations.

PREAMBLE

The Télécom2 satellite control computer system (SICS : *Système Informatique de Contrôle des Satellites*) is the continuation of the SICS-P (provisional system), which was used for the positioning and station keeping of the Télécom2A and Télécom2B satellites until the switchover from SICS-P to SICS (October 20, 93). Since this date, the SICS has been controlling 2 station keeping satellites.

This system was developed by the Information Processing sub-directorate of CNES, the prime contractors being the Matra Marconi Space France industrial group and Syseca.

1 - THE TELECOM2 SYSTEM

Designed as the continuation of the Télécom1 programme, the Télécom2 programme consists, in operational mode, of 3 satellites placed in geostationary orbits at -8°, -5° and 3° EAST. Two are in operation and 1 is on standby. Each satellite is made up of a stabilized 3-axis platform of EUROSTAR type, with 3 payloads and associated antennae for the following 3 missions:

- 12/14 GHz for new communication services in metropolitan France,
- 7/8 GHz for links specific to the Ministry of Defence,
- 4/6 GHz for classical links with the Overseas Territories,

2 - GROUND SEGMENT FOR TELECOM2 POSITIONING AND STATION KEEPING

The ground segment for Télécom2 positioning and station keeping comprises specific facilities and is also supported by CNES multimission facilities used for emergency purposes and in the launch and early orbit phase.

These specific facilities are :

- 2 Specialized Control Centres (SCC) with identical functions and capable of providing TTC control of 3 satellites 24 hours a day. Two satellites may be kept in a geostationary orbit while the third one is being positioned. The SCC have facilities for telemonitoring and remote activation of the station keeping control centres, in particular in order to initiate tracking measurements,
- 4 Network Control Centres for the 3 payloads. Each of these centres receives from the operational SCC, the telemetry data required to monitor the payloads,
- 4 4/6 GHz stations dedicated to TTL control of the 3 satellites,
- 3 7/8 GHz stations for traffic control, capable of providing TTL support simultaneously to 2 satellites,
- 3 4/6 GHz stations, Overseas (Réunion, Guiana), operating as "repeaters" to perform tracking measurements by turn-around with the previous 4 stations. The SCC provides simplified telemonitoring of these stations,
- 1 X25 network for specific Télécom2 data transmission (Réseau de Transmission de Données-Télécom2), which links the SCC to the stations and the NCCs.

3 - ENTITIES OF THE TELECOM2 CONTROL CENTRE

To carry out its mission, the Specialized Control Centre consists of several entities:

- the Satellite Control Computer System (SICS), responsible for real time tracking, telemetry and command functions and for their distribution to other entities, as well as for deferred batch processing functions.
- the orbitography computer (SUN hardware), using tracking measurements preprocessed by the SICS for orbit determination, prediction and computation of operations,
- Complementary Computer Facilities (CCF), which are micro-computers (PC), responsible for real time and deferred data processing,
- the cyphering bay,
- the Expert System (SUN hardware), which performs deferred analysis of data from the SICS,
- the Connection Unit to the RTD, which is the network entry point,
- the dynamic simulator (Digital hardware), used for practising or exercise purposes,
- the GASCON system (Hewlett-Packard hardware), for the telemonitoring and remote activation of stations and for initiating tracking measurements and station reconfigurations,
- the Technical Memory Management System (TMMS - SUN hardware), which performs deferred formatting of data from the SICS.

4 - SICS

The SICS real time functions are as follows :

- . management of data received and to be transmitted to the 2 communication networks (dedicated and mulimissions networks),
- . permanent processing of 4 TM data flows for automatic or visual monitoring and for data exportation. The 4th TM flow comes either from the simulator or from a redundant station,
- . preparation and sending of necessary commands, whether or not cyphered, possibly for 4 entities,
- . preprocessing of tracking measurements used by the operator to assess results and decide upon action to be taken, and compression of these measurements

- . preprocessing of the station calibration measurements,
- . storage of data received and of part of the processed data, for later analysis,
- . real time and deferred distribution of information to the other user entities of the SCC and to the NCCs.

The SICS deferred functions are :

- . supply of data for orbitography and operation processing,
- . all telemetry data classifying (trend analysis and replay), necessary for the analysis of changes in the satellites and for investigations in case of anomaly.
- . selective or statistical analysis of events recorded in the various logbooks,
- . management of real time block diagrams used for telemetry viewing at the SCC and the NCCs,
- . management of the so-called operational data, i.e. telemetry, commands, monitoring files, telemetry pages, etc.,
- . data backup for later use.

All these functions are implemented with a level of performance matching mission requirements and with ergonomics adapted to non computer experts for operations related to the basic functions (real time and TM data classification).

4.1 - General architecture of the SICS

This architecture is made up of 4 main entities, 3 of which operate on Digital hardware interconnected via an Ethernet network :

- a DEC MIRA "FRONT-END" computer, mainly responsible for real time processing,
- a DEC MIRA "BACK-END/DATA SERVER" computer, in charge of deferred processing, data storage and archival, data exportation to local or remote subscribers, and importation of graphic pages (block diagrams) from internal graphic servers,
- five dual-screen operator workstations, responsible for viewing real time or deferred telemetry, preparing commands and managing dialogue with the operator as well as feeding the video distribution system for the command and dwell page,
- the Graphic Server entity (3 HP computers), in charge of creating and viewing the graphic block diagrams, converting block diagrams dedicated to the NCCs and generating video TM pages.

The FRONT-END processor is connected via the RTD to the TTC Télécom2 stations and, through synchronous serial links to the 2 GHz stations and the satellite simulator on the one hand, and to the command cyphering bay, on the other hand.

The MIRA computers consist of two redundant microvax processors and a line switch. Each processor has its own input/output lines and other I/O lines connected to the switch. Only the nominal processor is connected via the switch to the external I/O lines, whereas the redundant processor is separated from these lines. An automatic system is used for failure detection and switching external lines from the nominal processor to the redundant one.

The MIRA computer manages the automatic switching of external links and allows free selection of the role of the processors, which may be :

- hot redundancy: applications only using inputs/outputs through a specific line may be run in parallel on each processor,
- active/standby redundancy : applications using inputs/outputs through a switchable line are active on the nominal processor and on standby on the redundant processor. The switching system activates them when changing from the nominal processor to the redundant processor,
- dedicated processor : a processor runs applications, without redundancy with the other.

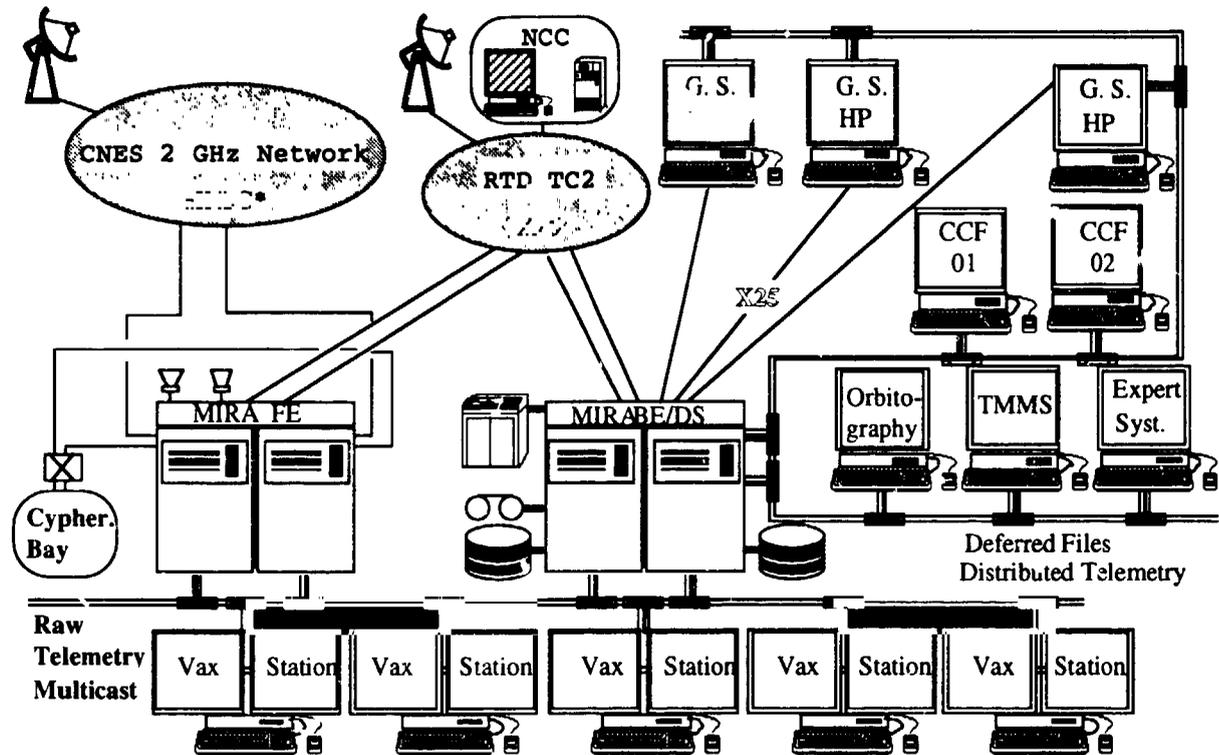


Fig. 1: Architecture of the SICS within the Télécom2 ground segment

4.2 - Functional description of the SICS

The system architecture has the main following features :

- reception and processing of raw telemetry lines by the MIRA front-end processor.
- multicast, on the local network, of raw telemetry data, derived parameters and results by the MIRA front-end processor following processing,
- processing of data distributed by the MIRA back-end processor/data server and transmission to local and distant subscribers,
- processing of data distributed to the operator workstations for viewing purposes.

The nominal processor of the front-end computer receives telemetry data from the TTC stations and/or the dynamic simulator, processes this data in real time, line by line, every 1.2 seconds (line acquisition, parameter calibration and control), and multicasts the raw telemetry line, derived parameters and control results to the other elements ;
the nominal processor and the redundant processor of the back-end computer/data server simultaneously archive telemetry data received ;
the operator workstations process the TM blocks distributed to the network by the front-end computer to view telemetry and control alarms ;
the nominal processor of the back-end computer/data server processes TM blocks distributed to the network by the front-end computer for telemetry exportation to subscribers.

Telemetry replay is performed by reading the archived data on the back-end processor/data server and transmitting them to the front-end computer for processing;
the trend analysis is defined by the operator from his workstation. The nominal processor of the back-end computer/data server retrieves, processes and makes the archived data available for viewing purposes.

Tracking data are received by the front-end computer. They are compressed by the nominal processor of the back-end computer/data server.

Command transmission is performed by the front-end computer.

Synchronization is carried out by the nominal processor of the back-end computer/data server, which cyclically gets the universal time and transmits it to the other computers.

5 - COST ELEMENTS

5.1 - Hardware

An outstanding feature of the Télécom2 control centre and particularly of the SICS is the number of machines used. This may be explained by the following factors:

- the various origins of the different systems and sub-systems, resulting in the fact that each sub-system has its own dedicated machine without any attempt to optimize the use of such a machine (why should a machine not perform several deferred functions, and even real time functions?)
- availability constraints required by the satellite. For the SICS, the constraint imposed was a maximum unavailability of 5 minutes during the critical phases, concerning the telemetry and command functions. This made it necessary to double the number of microvax computers on each site. The availability factor must be globally taken into account and one must consider that an in-flight failure and a ground failure occurring simultaneously represent a double failure of the complete system or one must be aware of costs induced by the operation of the system.
- 24 hours a day operation, which results in constraints upon the workstations. The operators must have workstations which are both dedicated to a definite satellite and user friendly. This accounts for the number of workstations and the presence of double screens.

Costs induced by material maintenance are high (Fig 2) : using a several year old configuration is expensive and the costs increase with time. Its replacement by a more recent configuration or its upgrading using manufacturers' kits may be cost-effective. Such an operation is quickly depreciated if one consider cost saving at the maintenance level. Examples of this approach could be the conversion of SICS computers from micro VAX 3500 to micro VAX 4000/200 or the purchase of HP715 configurations instead HP835. These operations would be depreciated as of the third year of maintenance. Moreover, purchasing the operational configuration should be delayed as long as possible and, if possible, this configuration should not be supplied as of the development phase.

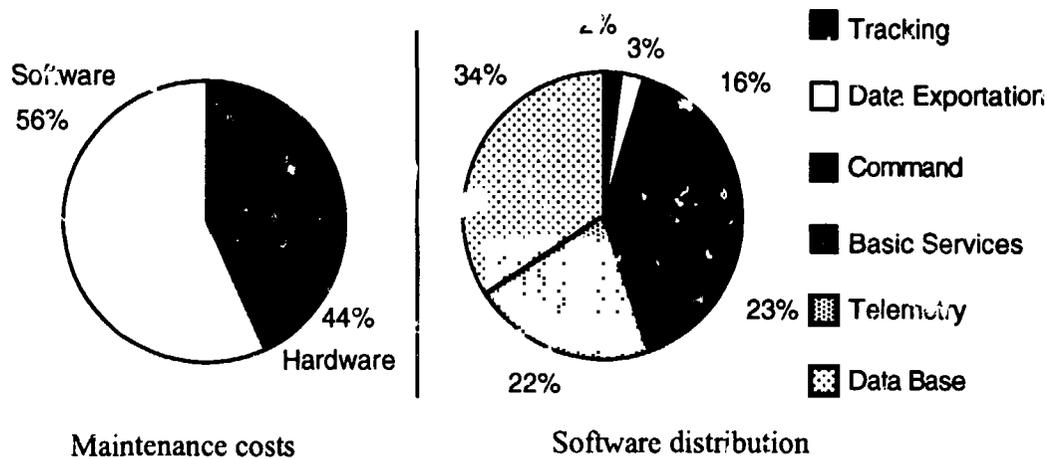


Fig 2: Maintenance cost estimate and size of application software per sub-system

5.2 - SOFTWARE

The SICS software performs many functions, offers great versatility of use and has been developed with high quality and documentation requirements. It thoroughly complies with its specifications and users are entirely perfectly satisfied.

However, one often forgets that any requirement, whether technical or quality related, has to be paid for and developments made on a fixed firm base do not show the price of each requirement. The industrialist must price a work, whereas neither the technical specifications nor the development rules are detailed in writing. Thus, he can only indicate a global amount. As an alternative, the development contracts could be divided into two parts: a specification phase resulting in proposals priced according to the requirements considered and with alternative proposals, then the fixed performance phase, which then could be implemented, knowing the costs and deadline of each requirement. Precise knowledge of the costs is a factor which contributes to their reduction.

The process complexity was determined, considering the number of modules making up the processes, the size, complexity (cyclomatic number) and the number of calls for routines external to each module.

Efforts required to develop sub-systems (see project results in ref. [SICS 3]) are appropriate to their complexity and to the distribution non conformance reports. There is indeed a direct relationship between complexity and costs.

Among the technical facts which may be brought out because of their complexity, let us mention:

- the existence of local data bases on each workstation, which required the development of updating, storage and distribution procedures as well as monitoring their impact on the Ethernet network, in all, near 400 modules with more than 800 calls for other routines,
- it was not easy to develop the capacity to automatically issue commands for a telemetered event, as each operator workstation may issue commands (with exclusion mechanisms) to any satellite, and this function must be exclusive of the interactive command sessions. For information, just managing its inhibition needed 10 modules using 20 routines,
- management of the synchronous protocol specific to the CNES 2 GHz network and to the encoding rack required programming an input/output board, in all 32 modules using 72 routines,
- the Man Machine interface requires many software programmes. It is present in all the functions and carries much weight in the production and presentation of information. If we consider only the processes entirely dedicated to this function, managing the MMI requires 111 modules using 225 routines, i.e. more than 3500 executable instructions.

Although it was necessary to develop delicate mechanisms for disk data recovery, the availability factor mainly relied on MIRA software for the switching of nominal computers to redundant computers and thus induced few developments.

Fig 2 shows the significant share of the basic services that manage external interfaces and coordinate the SICS operation. They are closely related to the hardware architecture. The volume of the telemetry sub-system is explained by the complexity of the deferred function part, processing of the satellite-ground interface and viewing (block diagrams, pages, etc.).

To reduce this complexity and therefore the related costs, the architectures must be simplified as follows :

- dedicate workstations, which corresponds to the operational reality and may simply be done by a "login" procedure,
- simplify the MMI, as there is no need to display all functions by means of windows and menus. It is not necessary to go backwards and risk operational errors, but prohibiting any keyboard entry and displaying all information in graphic form are expensive. Is an interface similar to that of office workstations really necessary for all functions ?
- avoid in-house protocols as far as possible, because they need to be programmed and maintained, whereas manufacturers tend to give them up,
- reduce customer-server links, as the SICS multicast feature is a very good concept and suppresses the transmitter-receiver link and should be extended to the distribution of raw and physical telemetry,
- think to satellite ground interface in terms of exploiting the data it carries,
- use CCSDS standards (software exists or will exist in a short period of time),
- any memory zone of the satellite and command stack especially has to be dumpable.

6 - CONCLUSION

The approach to space projects must be improved. To reduce ground costs, requirements must be adapted to needs. Margins must no longer be included in requests: if the constraint is "N", the supplier must be asked to provide "N" and not "N+X%", and, as each intermediary adds his own margin, constraints are reached, which are hard to fulfil and completely unjustified.

The complexity of satellite-ground interfaces must also be carefully assessed, CCSDS recommendations have to be taken into account,

Critical phases, with their need for availability and specific interfaces, must not affect the whole system life. Ideally, only the positioning should be critical and should be performed in a dedicated configuration.

Customizing the costs of each requirement is unquestionably a savings factor, whether this requirement is technical or methodological, and is achieved by custom designing the specification phase under a specific contract.

Multiplication of hardware configuration must be limited, whether by using common input-output services, sharing sub-system configurations or accepting the deterioration of some functions in case of failure, such as for example reconfiguring a deferred processing machine into a real time machine.

Fortunate as we are to have a product which offers many functions, we should take advantage of it by offering it, within a line of products, to other projects, which therefore will have a low cost basis (as development has already been done) for a precise assessment of the adaptations needed.

REFERENCES

- [SICS 1] H. Pasquier: "Data exportation in Telecom2 control system", *First International Symposium on Spacecraft Ground Control and Flight Dynamics*, São José dos Campos, Brazil, February 94.
- [SICS 2] "Spécification technique de besoin du SICS-D TÉLÉCOM2", *Centre National d'Etudes Spatiales*, Toulouse, France.
- [SICS 3] Matra Marconi Space France and Syseca SICS project team: "Bilan Projet du SICS-D TÉLÉCOM2" (version provisoire), *Centre National d'Etudes Spatiales*, Toulouse, France.
- [SICS 4] B. Fébrier Matra Marconi Space France and Syseca: "Dossier d'architecture matérielle et logicielle du SICS-D TÉLÉCOM2", *Centre National d'Etudes Spatiales*, Toulouse, France.

The SILEX experiment system operations

B. DEMELENNE - EUROPEAN SPACE AGENCY - B6890 REDU - BELGIUM

ABSTRACT

The European Space Agency is going to conduct an inter orbit link experiment which will connect a low Earth orbiting satellite and a Geostationary satellite via optical terminals. This experiment has been called SILEX (Semiconductor Inter satellite Link EXperiment). Two payloads will be built. One called PASTEL (PASSager de TELecommunication) will be embarked on the French Earth observation satellite SPOT4. The future European experimental data relay satellite ARTEMIS (Advanced Relay and TEchnology MISsion) will carry the OPALE terminal (Optical PAYload Experiment).

The principal characteristic of the mission is a 50 Megabits flow of data transmitted via the optical satellite link. The relay satellite will route the data via its feeder link thus permitting a real time reception in the European region of images taken by the observation satellite. The PASTEL terminal has been designed to cover up to 9 communication sessions per day with an average of 5. The number of daily contact opportunities with the low earth orbiting satellite will be increased and the duration will be much longer than the traditional passes over a ground station. The terminals have an autonomy of 24 hours with respect to ground control. Each terminal will contain its own orbit model and that of its counter terminal for orbit prediction and for precise computation of pointing direction. Due to the very narrow field of view of the communication laser beam, the orbit propagation calculation needs to be done with a very high accuracy.

The European Space Agency is responsible for the operation of both terminals. A PASTEL Mission Control System (PMCS) is being developed to control the PASTEL terminal on board SPOT4. The PMCS will interface with the SPOT4 Control Centre for the execution of the PASTEL operations. The PMCS will also interface with the ARTEMIS Mission Control System for the planning and the coordination of the operation. It is the first time that laser technology will be used to support inter-satellite links in Europe. Due to the complexity and experimental character of this new optical technology, the SILEX experiment control facilities will be designed to allow as much operational flexibility as possible.

INTRODUCTION

The European Space Agency (ESA) has initiated the SILEX experiment to test the optical technology for intersatellite communications. This experiment will provide a high data rate Inter-Orbit Link (IOL) between the low earth orbiting terminal (called PASTEL) mounted on the French SPOT4 earth observation satellite and the geostationary terminal embarked on the ARTEMIS data relay satellite. The launches of these two satellites are currently foreseen in 1997. ESA in collaboration with CNES (the French space agency) has setup a ground segment to control the experiment.

SILEX MISSION CHARACTERISTICS

The optical terminals involved in the SILEX experiment have been designed such that the following system specifications can be respected. The link capacity can transmit 50 Mbps of useful data from the SPOT4 satellite to be relayed by ARTEMIS via its feeder link to the ground image receive station. The wavelength range is 800 to 850 nm. The optical power shall not exceed 60 mWatts during the communication period and 500 mWatts for the beacon required on the LEO terminal during the acquisition and the link establishment. The routine link acquisition shall not require more than 150 seconds with a success probability of 95 %. The terminal shall have an autonomy of 24 hours with respect to the ground control. The terminal has its own computer and software to provide on board monitoring and control of its equipment such that it will be able to reconfigure itself in a safe mode in case of anomaly in order not to interfere with the SPOT4 satellite. The PASTEL terminal is located on the non-earth facing panel of SPOT4 and the OPALE terminal is located on the earth facing panel of ARTEMIS. The current design of the optical link foresees up to 9 communication sessions per day between the two satellites with an average of 5 per day. Figure 1 shows the available visibility area which allows optical terminal communications between the SPOT4 satellite and the ARTEMIS satellite located at 16.4 deg. East. Some constraints need to be taken into account for the definition of the visibility area such as the mounting of the PASTEL terminal on SPOT 4 and its limitation in angular speed. Figure 2 gives a sample for one day of possible contact between the two optical terminals during the SPOT4 satellite day time.

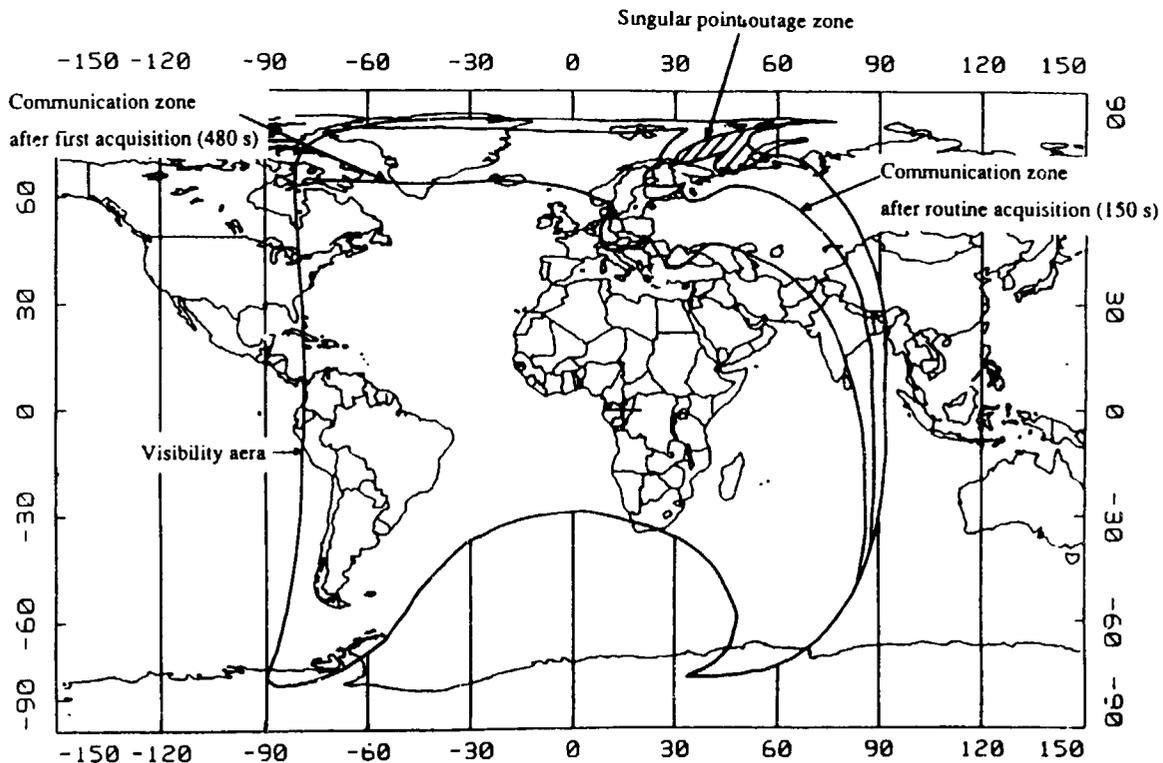


Figure 1: Optical visibility between SPOT4 and ARTEMIS

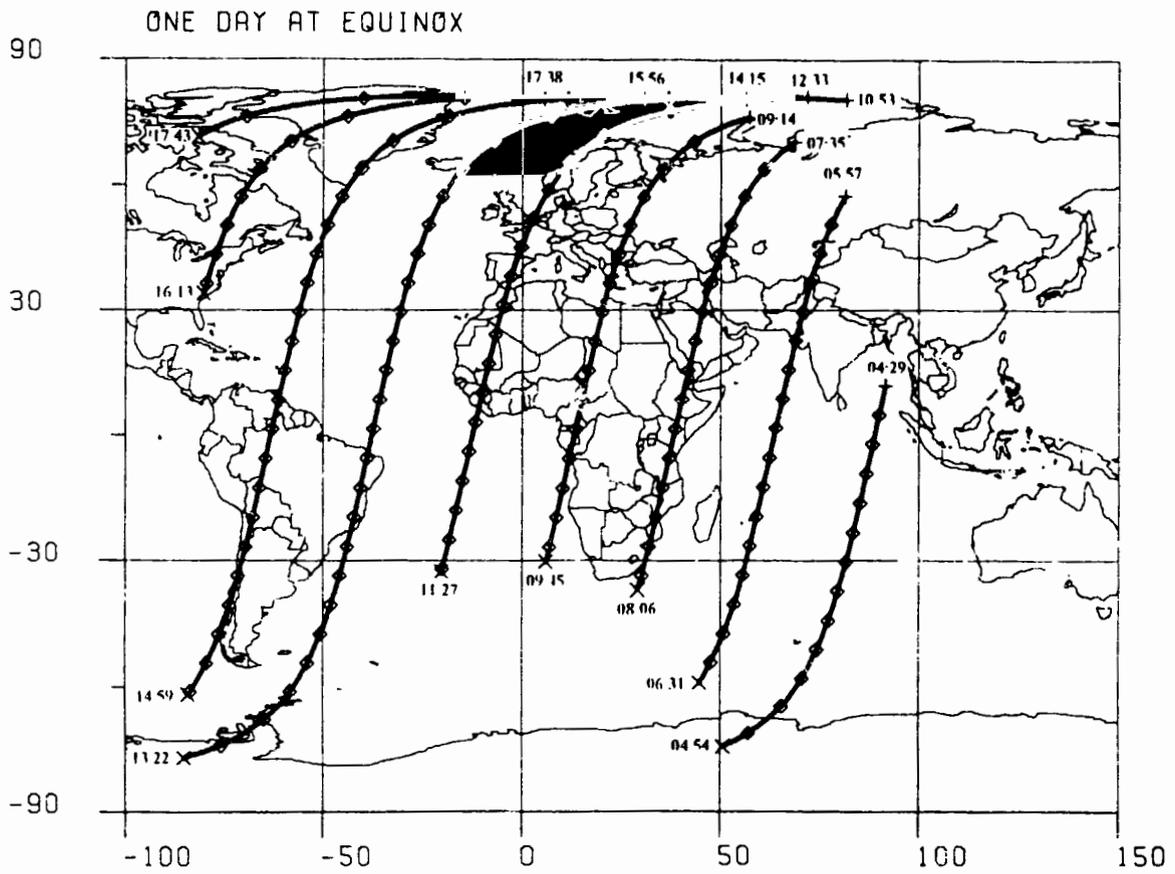


Figure 2 : Visibility between PASTEL and OPALE at Equinox during SPOT4 day time.

OVERALL GROUND AND SPACE SEGMENT

Figure 3 is an overview of the ground segment involved in the SILEX experiment. One part of the ground segment is under the control of ESA and the second is the responsibility of CNES. ESA is responsible for the operations of the PASTEL mission control system located in Redu (Belgium), the ARTEMIS mission control system located in Italy which controls the ARTEMIS spacecraft via the Tracking, Telemetry and Command (TTC) antenna and the Payload test facilities to monitor and check the ARTEMIS payload (PTL/IOT). CNES is responsible for the control and monitoring of the SPOT4 satellite from its control centre (CMP) located in Toulouse (France) connected to its S-band control station network and for the reception of the images taken by the SPOT4 satellite via either the ARTEMIS feeder link on the SRIP station or directly from the SPOT4 satellite when in visibility of the SRIP station. CNES is also coordinating the SPOT4 mission with its commercial operator for the scheduling of the image recordings. ESA is responsible for the operations of the two optical terminals PASTEL, on board SPOT4, and OPALE on board ARTEMIS. For the control and monitoring of the PASTEL terminal, ESA and CNES have setup an interface to exchange all the data needed for the terminal control and scheduling of PASTEL usage.

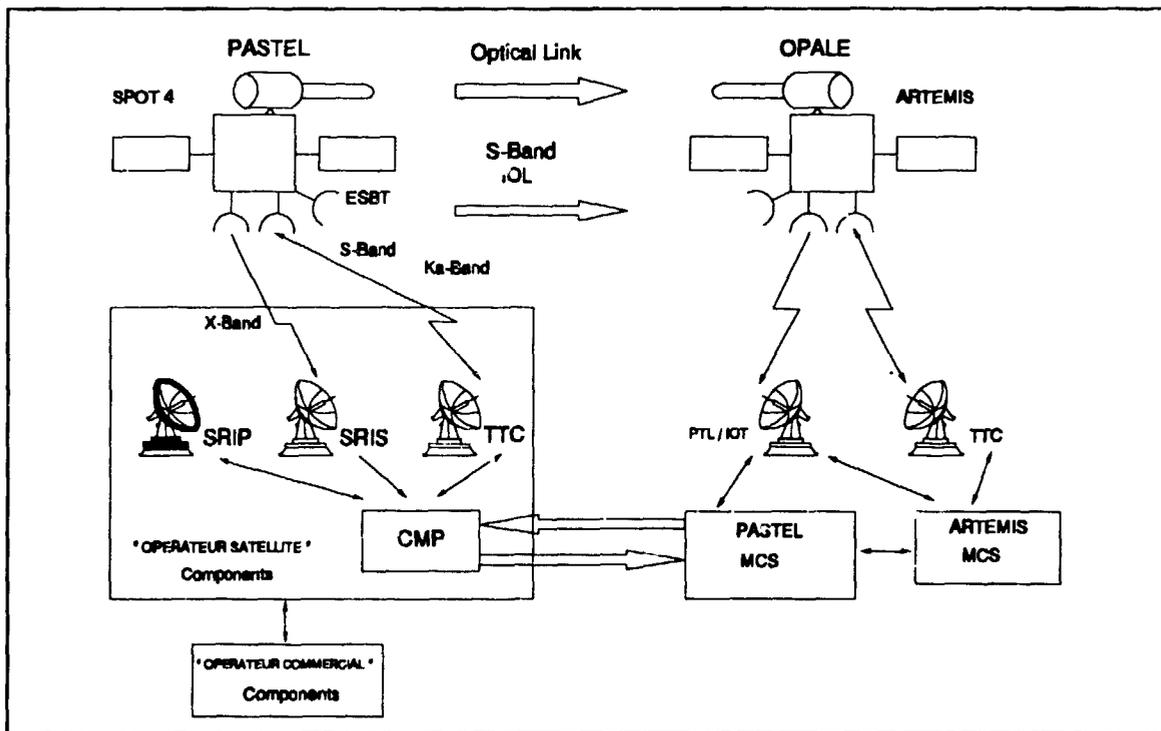


Figure 3: SILEX Experiment Ground Segment

OPTICAL TERMINAL OPERATIONS CONCEPT

The OPALE terminal is controlled from the ARTEMIS mission control system in Italy and the PASTEL terminal is controlled from the PASTEL mission control system in Belgium via the SPOT4 control centre (CMP) located in France. This constraint has led to the following operations concept for PASTEL:

- * The PASTEL Telemetry and Telecommand interface function is accomplished by the SPOT4 TTC subsystem via CNES ground stations and SPOT4 control centre.
- * The PASTEL routine operations are conducted from the PASTEL mission control system in an off-line manner.
- * For PASTEL monitoring, the full SPOT4 raw telemetry is provided to the PMCS about 30 minutes after each pass of SPOT4 over one of its ground stations.
- * For PASTEL commanding, telecommand files are generated by the PMCS and sent to the SPOT4 control centre by 1800 hours every day to be ready for an uplink on an evening pass of SPOT4 over the Aussaguel station located near Toulouse.
- * Two categories of telecommand are foreseen. The first TC type is that executed directly by the SPOT4 on board computer to activate the PASTEL terminal. The second TC type is that transferred by the SPOT4 on board computer to the PASTEL on board computer which will execute it to control the terminal. The first category of TC is not directly coded into a binary format by the PMCS. These TCs are translated using a pseudo language for security reason and sent within TC files from the PMCS to the CMP which manually inserts them in their next TC uplink plan. The second category of TC is directly coded in binary format by the PMCS and sent to the SPOT control centre which will encapsulate them in the TC uplink format of SPOT4 after checking that they are addressed to PASTEL and not to another payload of SPOT4.
- * In addition to the TM/TC files, scheduling information for the planning of PASTEL and OPALE usage will be exchanged on a well defined scenario between the SPOT4 CMP, the ARTEMIS mission control system and the PASTEL mission control system
- * On top of the routine operations foreseen for PASTEL and OPALE, contingency scenarios have been defined between the two control centres such that the outage of the optical link between the two satellites can be minimised.

PASTEL MISSION CONTROL SYSTEM

The PASTEL mission control system is fully responsible for the operations of the PASTEL terminal and the scheduling of the OPALE terminal. This centre is interfacing with the ARTEMIS control centre for the operations of the OPALE payload and with CNES for the operations of the PASTEL terminal on board SPOT4. As the optical terminal operations require an orbit determination accuracy such that both terminals know the position of the other to be able to establish the optical link, the interface between the SPOT4 CMP and the PASTEL mission control has been designed to ensure that the daily flow of information needed for the optical link operations is exchanged in a minimum of time. Figure 4 gives an overall view of the PMCS components.

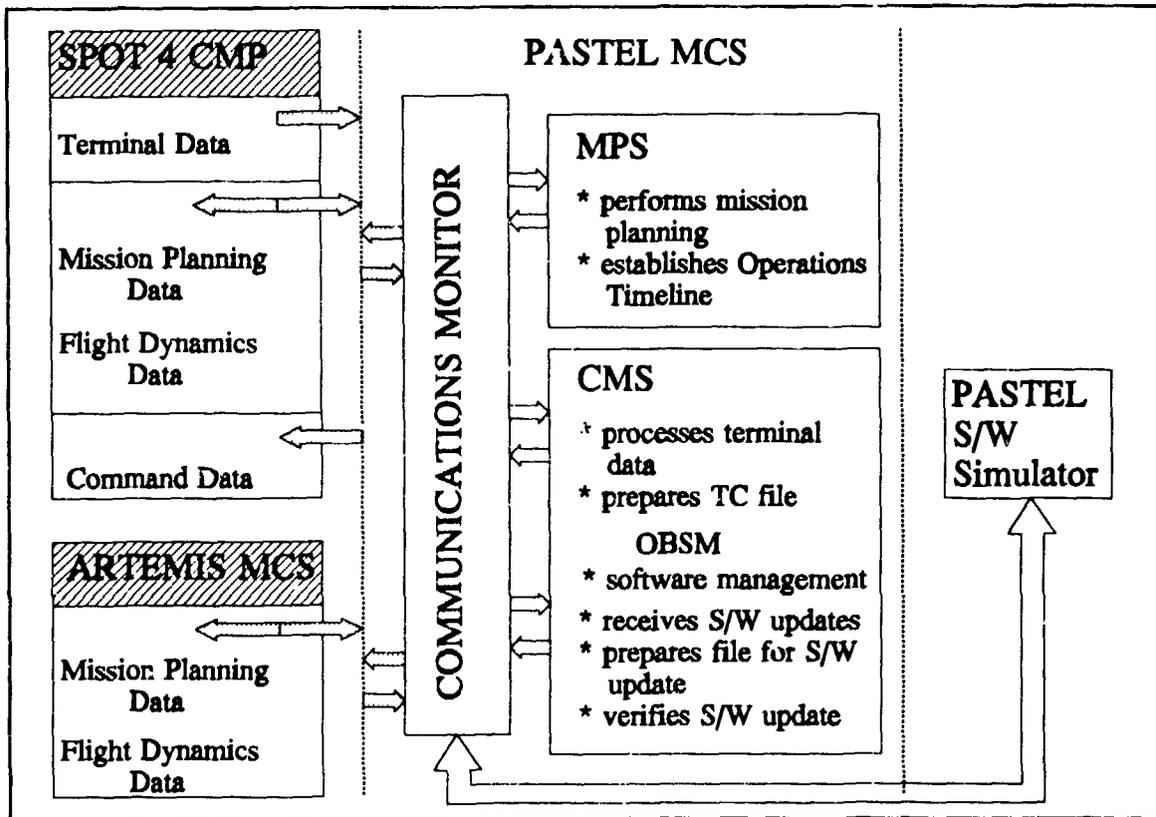


Figure 4: PASTEL MCS Configuration

The PASTEL mission control system includes the following elements :

- The Communication Monitor which controls and monitors all the files exchanged between the SPOT4 control centre, the ARTEMIS mission control system and the PASTEL mission control system. This Communication Monitor normally works automatically but for special situations such as communication network degradation or a switch to the back up PMCS or CMP computers, it can be operated manually.
- The PASTEL Mission Planning System (MPS) will provide all the functions needed to plan the execution of the optical link operation. This system will plan and coordinate with CNES and the ARTEMIS mission control system the usage of the optical link several weeks in advance. One week before the operations, the detailed operations timeline to be executed by the ARTEMIS and PASTEL mission control systems is issued.
- The PASTEL control and monitoring system (CMS) will include all the functions needed for the in orbit operations of the PASTEL terminal on board of SPOT4. The CMS will process the SPOT4 telemetry related to the PASTEL terminal. It will generate the TC request file on a daily basis based on the operations timeline provided by the MPS. The CMS will also include the on board software management system (OBSM) for the management of the software loaded into the PASTEL computer. The OBSM will be able to receive new releases of the on board software from the manufacturer and to generate the appropriate telecommands to update the PASTEL on board software. The OBSM will also receive dumps of the on board software such that correct loading can be verified.
- A PASTEL software simulator has been attached to the PMCS to allow the PMCS to validate new operational procedures for PASTEL or to validate any new PMCS software release without the need of the SPOT4 CMP.

CONCLUSIONS :

The European Space Agency has initiated the development and operation of the first European free space optical communications system. The demonstration of optical technology in space will be proved by the SILEX experiment and the European Space Agency is conducting further research to minimise the weight of optical terminals and to improve their performance. The SILEX experiment is still under development with launch dates foreseen in 1997 for the two satellites (ARTEMIS and SPOT4) with their optical terminals.

EURECA MISSION CONTROL EXPERIENCE AND MESSAGES FOR THE FUTURE

*H.Hübner, P.Ferri and W.Wimmer
European Space Operations Centre,
Robert-Bosch Straße 5, 64293 Darmstadt, Germany*

ABSTRACT

EURECA is a retrievable space platform which can perform multi-disciplinary scientific and technological experiments in a Low Earth Orbit for a typical mission duration of six to twelve months. It is deployed and retrieved by the NASA Space Shuttle and is designed to support up to five flights. The first mission started at the end of July 1992 and was successfully completed with the retrieval in June 1993.

The operations concept and the ground segment for the first EURECA mission are briefly introduced. The experiences in the preparation and the conduction of the mission from the flight control team point of view are described.

Key Words: EURECA, Spacecraft Operations, Fault Management, On-Board Autonomy, Rendezvous Operations.

1. INTRODUCTION

The EURECA mission represented in many aspects a completely new challenge from the mission control point of view. The main features were the extremely limited ground contact, about 5% of the total mission time during the science phase, which demanded a high level of on-board autonomy (Ferri and Wimmer, 1994; Hübner and Wimmer, 1994), the deployment and retrieval by the Shuttle, including the safety aspects related to the manned spaceflight, the rendezvous activities and the complex inter-agency operations involving the Orbiter, two control centres, ground stations and data relay satellites, the concept of packetised telemetry and telecommands, for the first time fully implemented on a European spacecraft, and the large number of possible payload configurations.

After launch and deployment by the Shuttle into a

424 km circular orbit, EURECA was manoeuvred to the operational orbit of 508 km altitude, where the microgravity environment was established. This was followed by a ten month science phase in which the experiments, in the area of microgravity science, space science and space technology were performed under low acceleration conditions. About one month before the predicted time of launch of the retrieval Shuttle a series of orbital manoeuvres to lower the orbit and to match the retrieval orbital requirements commenced. Shortly before the launch of the retrieval Shuttle EURECA concluded all orbital manoeuvres, and the Orbiter reached it after a three days flight, capturing the spacecraft, safely storing it in the cargo-bay and returning it to Earth after 11 months of flight and more than 5000 revolutions around our planet. For a detailed summary of the EURECA mission, see Wimmer and Ferri (1994).

The EURECA ground segment was designed around the main mission characteristics (Ferri and Kellock, 1992). During the science mission phase contact with the spacecraft was achieved via two ESA ground stations at Maspalomas and Kourou, and a control centre located in Darmstadt, which could also make use of a third station in Perth as a back-up. These ground stations provided in total a daily sequence of about eight contact periods of 5 to 10 minutes, spaced by 90 minutes. A long non-coverage period of about 9 hours occurred between two consecutive sequences of station passes.

During the critical mission phases i.e. during deployment, orbit manoeuvres and retrieval operations, the third ground station at Perth was added, to increase the contact time.

In the phases when EURECA was attached to or in proximity of the Shuttle, contact with the spacecraft was established via the NASA Communications Network (NASCOM), the

NASA Tracking and Data Relay Satellite (TDRS) system and the Orbiter, which guaranteed a practically continuous coverage.

2. MISSION CONTROL CONCEPT AND EXPERIENCES

The manifold characteristics of EURECA's mission profile and the high degree of autonomy and new techniques integrated on-board, required a ground control system and a control concept (Van Casteren and Ferri, 1989) capable of supporting both, a traditional on-line and an advanced off-line operations approach. The traditional approach was characterised by manual uplink of individual telecommands, housekeeping telemetry monitoring and alarm processing. The advanced approach involved typically the activation of on-board application programs for implementing the required operations. The corresponding commands were prepared while the spacecraft was not in contact with the control centre. During ground coverage periods the accumulated spacecraft telemetry was dumped to ground via a high speed link and the prepared series of time tagged commands were uplinked. The execution verification of the on-board activities was based on event messages from the application programs and took place when the spacecraft was not visible anymore to the groundstation.

In this section the characteristics of the operation concept related to the different mission phases are briefly described, followed by a discussion on the most important experiences and lessons learned.

Mission Preparation

The main purpose of the mission preparation activities of the Flight Control Team was to specify the requirements for all the EURECA dedicated ground segment facilities, to customise the mission control software via the preparation of an operational database driving the telemetry processing and the telecommand generation functions, and finally to prepare and validate, based on the spacecraft users manual and other design documentation, the Flight Operation Plan (FOP). This document contains the detailed timelines of all phases of the mission and all the nominal and contingency procedures foreseen for the conduction of the spacecraft and payload operations.

The two major verification activities before the start of the mission were the System Validation Tests (SVT), to verify all functions of the mission control system and the operational database via a direct link with the real spacecraft, and the simulation campaign, which started about six months before launch and was resumed during the last weeks of the mission to test the new timelines of the retrieval phase. The main purpose of the simulations was to validate the FOP and to train the mission control team in all activities of the mission. The simulation programme for EURECA culminated in the joint simulations with the participation of ESOC, the NASA MCC and the Shuttle crew.

The major problem encountered in this phase was the lack of proper documentation on spacecraft design and functionality. Although this tends to be a common problem of many space projects, in the case of EURECA it was particularly serious due to the complexity of the spacecraft and the large amount of software functions implemented which were not sufficiently described and kept changing during the spacecraft integration process until very late in the programme. This had a severe impact on the workload required for the preparation of the operational database. Frequent changes and corrections were necessary to adapt the database to the new documentation or to the changes in the software. In addition, the lack of previous experience with the packet telemetry concept caused a significant underestimation of the work required for the preparation of the telemetry database.

Another underestimation of the mission preparation effort, also caused by the lack of previous experience, occurred in the area of the interface with NASA. The activities related to the NASA interface involved operational discussion, participation to meetings and telecons, formal review of NASA documentation, preparation and execution of joint integrated simulations. This work had to be carried out by the flight control team in parallel to the other normal mission preparation activities. This problem became evident and highly dangerous during the mission, when similar activities had to be carried out in preparation of the retrieval mission, while the demanding mission control activities of the science phase had to be conducted in parallel by the same people.

The System Validation Tests for the EURECA mission were also anomalous, in comparison to previous projects. The incomplete status of the spacecraft at the time of the first test slot, combined with the frequent announcements of launch delays due to the unstable situation of the Shuttle programme after the Challenger accident extended the test phase to a period of two and a half years, during which more than ten weeks of actual test time with the spacecraft were utilised by ESOC. Although a large part of this time was spent re-testing spacecraft subsystems and functions which did not properly work the first time, the extended test time available for ESOC (for a typical project four to five weeks of test time are reserved for SVT in the last year before launch) allowed the flight control team to integrate the knowledge of spacecraft design and functionality which could not be satisfactorily built on the documentation. The disadvantage of this approach was that a large amount of unforeseen manpower had to be invested in this phase, reducing the quality of the other mission preparation activities.

The preparation of the science operations phase suffered, as a consequence of the above mentioned problems, from the little time dedicated to the definition of nominal procedures for planning and conduction of the routine activities. The software developed for the mission planning tasks was too rigid and restrictive to cope with changing planning requirements and revised payload control concepts. This did not help reducing the overload of the flight control team in the execution of the daily activities, it even required additional manpower for extending the mission planning software and integrating it into the working environment. Other software tools available to the team also created some problems, due to unfriendly user interfaces or insufficient support functions. The characteristics of the EURECA flight control team also caused an uneven distribution of work among the different team members. This situation evolved as a consequence of the fact that the team was based on a small group of engineers who worked on the mission preparation for many years, until only a few months before launch a number of new engineers was added. The result was that the experienced engineers were overloaded and had little time, in the last critical months before launch and during the mission itself, to gradually pass

responsibilities to the new team members. In addition the short duration of the mission, the number of spacecraft failures during the science phase, and the intense activities in preparation of the retrieval, which started only a few months after launch, resulted in never reaching a stable, routine phase of the mission operations, in which procedures and responsibilities could have been effectively consolidated.

The experience of the EURECA mission preparation showed that an earlier team build up is absolutely required for a mission of this level of complexity. A kernel of at least five operations engineers should work on the project, in conjunction with the spacecraft and ground segment developers, for several years before launch. The interface with NASA has to be given more emphasis within and outside the flight control team at ESOC. This implies that the nomination of a Flight Director for a mission involving joint operations with the NASA Shuttle environment should occur at least two years before launch, to allow him to familiarise himself with the mission and to supervise the discussions on operational interfaces. The problems encountered with the database generation and lack of information on the spacecraft design could be solved by allowing a deeper and earlier involvement of the ESOC operations personnel in the spacecraft development and particularly in the related integration and testing activities.

Critical Mission Phases

A detailed description and analysis of the critical deployment and retrieval operations can be found in Ferri *et al.* (1993); this section presents a general overview and the most important experiences.

Twelve hours after launch in the cargo-bay of the Space Shuttle Atlantis on the mission STS-46, the EURECA internal power was initially activated by the Shuttle astronauts via switches located in the crew compartment. The commanding activities started immediately after reception of the first telemetry via the NASCOM network. The spacecraft was lifted by the Shuttle robotic arm out of the cargo-bay, while the spacecraft activation continued, including the deployment of the RF antennae and the solar array wings. After release from the Shuttle, the three-axes stabilised attitude was acquired, and the preparation for the

first orbit manoeuvre continued. Due to a number of unforeseen fine-tuning activities in the software tables driving the attitude control subsystem on-board and a problem in the ground control computers the orbit manoeuvre was delayed by four days.

The orbit manoeuvre phases were critical phases of the mission to be handled only via the ESA ground stations. The deployment manoeuvres were executed nominally after the correction of an interface problem between two ground computers, causing wrong software parameters to be uplinked to the spacecraft, which delayed the start of the phase. The retrieval manoeuvres, however, uncovered deficiencies in the design of the attitude and orbit control subsystems. First of all, the non-negligible orbital effects of the attitude control in some control modes using hydrazine was underestimated in the design and caused significant changes and higher risks in the conduction of the entire retrieval campaign. The effect of attitude mode changes had to be measured and taken into account in the orbit manoeuvre strategies, but this with a low confidence since many of the mode changes were executed autonomously by the spacecraft and were to a certain extent unpredictable. The loss of two gyroscopes during the nominal mission left the spacecraft with no redundancy for the final phases, but also uncovered a problem in the attitude control software which had to be worked around via complicated and dangerous operational procedures. Finally a problem in the on-board software in charge of compensating the gyro drift was detected by chance before the start of the first descent orbit manoeuvre. The manoeuvre strategy and procedures had to be changed, and a number of unsuccessful attempts had to be executed before a stable work-around approach was defined and the retrieval orbit was reached.

After three days of approach, the Shuttle orbiter reached the proximity of EURECA, which in the meantime had stopped all orbit manoeuvres. In the last revolution around the Earth ESOC configured the spacecraft for retrieval in the Shuttle bay, slewing in a predefined attitude, retracting solar array and antennae, and deactivating and safing all the hazardous subsystems like the hydrazine reaction control system. The final approach of the Shuttle proceeded nominally and the spacecraft was first captured with the robotic arm, and later stowed into the cargo-bay and deactivated. A

problem in the final latching of the RF antennae to the body of the spacecraft forced an EVA (Extra-Vehicular Activity) to manually press the antenna booms while ESOC was commanding the latches. This was successfully executed the next day and EURECA returned safely to Earth at the end of the Shuttle mission, a few days later.

The retrieval phase scenario was simpler than the deployment one, and the decision to execute all the time-critical deactivation operations automatically on-board via a time-tagged sequence of commands removed most of the criticality and in particular the dependence from the ground contact which, due to the communications problems experienced in the deployment phase, was not fully trusted. As an additional back-up, NASA offered to add a number of NASA and RTS ground stations for the duration of the critical retrieval phases. The need for an operational contact with EURECA via the additional station did not materialise, but their presence helped in increasing the confidence in the success of the mission. The criticality of the retrieval operations mainly derived from the degradation of the spacecraft performance, in particular in the area of power and in the number of gyroscopes available for attitude control. Fortunately no additional major failures occurred during the final phase of the mission, and every major subsystem performed nominally.

The nature of the deployment and retrieval phases dictated a typical real-time approach to the operational documentation: detailed timelines were produced for the nominal and main contingency cases, which would merge in time order all the activities of all the parties involved. For the Shuttle proximity phases the three timelines of the Orbiter crew (Flight Plan), of the Houston MCC (Ops Support Timeline) and of ESOC (Flight Ops Plan) had to be synchronised. Details of the activities like commands and monitoring parameters were contained in flight control procedures called by the timelines.

The mission control team at ESOC was established according to the standard approach adopted for other projects, with three main groups of controllers responsible for spacecraft operations, ground segment operations and flight dynamics, under the central authority of the Flight Operations Director. Consultancy on all aspects of spacecraft design and functionality was

provided by the project support team, formed by experts of the spacecraft manufacturers and the ESA project team. For both deployment and retrieval phases one of the main critical aspects was the crew safety constraints on the EURECA operations. Due to the very limited visibility of the EURECA status available to the Shuttle crew and to the NASA flight controllers, this was fully delegated to ESOC. When EURECA was in the Shuttle cargo bay or in its proximity the safety status of the spacecraft was continuously monitored at ESOC and reported to the NASA mission controllers. Multiple failure tolerance was implemented in the mission control software to avoid inadvertent uplinking of hazardous commands to the spacecraft at the wrong time. One of the difficult tasks was to continuously derive the safety status of the spacecraft from the available telemetry, which in some cases was not complete and explicit enough for a real-time judgment, in particular in the activation and deactivation phases, when the spacecraft configuration was continuously changing. One of the improvements successfully implemented in the flight control team at ESOC for the retrieval mission was the assignment of a dedicated operations engineer to the safety monitoring, assessment and reporting to the Flight Director.

Concerning the experience gained in the EURECA critical phases it should be stressed that in particular the deployment phase suffered a large number of major anomalies, many of which occurring in parallel, which were kept under control without any impact on the crew safety nor on the mission success, and with only minor delays. From the errors discovered in the on-board attitude control parameters and in the communications between the thermal control and the data handling subsystem important lessons could be learned in the way autonomous functions have to be implemented and operated.

An important experience resulting from the retrieval phase was the preparation and execution of the EVA procedure to latch the antenna booms. The frenetic preparation of a completely new procedure in the night before the EVA became necessary due to a double failure situation, the antenna latching problem and the failure of the power interface via the robotic arm to the EURECA thermal control, which forced the ground controllers to berth the spacecraft with unlatched antennae, to avoid thermal problems. This starting position for the EVA was not

foreseen, and the final success of the activity was a major achievement in the overall NASA-ESA cooperation for this mission.

The traditional approach to the critical mission phase operations proved to be successful in this extremely dramatic scenario; the deficiencies in the timely monitoring of safety items was successfully corrected in the retrieval phase by the introduction of a dedicated controller position.

Science Operations Phase

Eighteen days after launch the spacecraft was successfully configured for the science operations phase, including the activation of the freon cooling loop, the micro-gravity measurement system, and the low-thrust attitude control system. In addition each payload instrument was at least activated once and checked out. The ground segment configuration was characterised by an off-line operations scheme and a close interface with the Project Scientist, who coordinated the input of the more than 30 scientists, representing them in the EURECA Weekly Operations Review Meeting at ESOC. The science community could receive telemetry data electronically in their home institutes via an active Data Disposition System; Principal Investigators were able to request changes to the configuration of their instrument via a Telecommand Request interface (via FAX or E-Mail) in response to their evaluation of spacecraft and payload telemetry.

The mission operations scheme applied during the science operations phase consisted of three main tasks: mission planning, real-time pass operations and spacecraft performance monitoring.

The mission planning task was performed daily in order to prepare all inputs required for both the pass operations and the spacecraft performance monitoring. Based mainly on a version of the Mission Baseline Plan updated every two weeks, on decisions taken in the last Weekly Operations Review Meeting, on the most recent Telecommand Requests, and on the potential feedback from the monitoring task, a file was prepared which contained the commands to be uplinked to the on-board Master Schedule during the next ground contact periods. In addition, detailed instructions for non-standard operations to be carried out by the spacecraft controllers during the next ground station passes were prepared on paper.

Spacecraft Controllers were in charge of preparing and conducting the pass operations. Flight Control Procedures (FCP) detailed the required standard activities such as uplink of telecommands, Master Schedule, monitoring of telemetry, dump of on-board memory and transfer of dumped data from the ground station to the control centre. A short list of basic health checks were part of the standard activities to be performed in every pass. The results of these, together with the alarms raised automatically by the control software in case of out-of-limit conditions in the telemetry, were used to detect severe anomalies of the spacecraft in real-time. Only in very few cases, requiring easy and well defined recovery actions, on-line Contingency Recovery Procedures could be used during the short passes. For all other anomalies, off-line recovery strategies were applied, either by an on-call system engineering support or as part of the performance monitoring task.

Spacecraft Performance Monitoring normally started when all the telemetry generated during the day, downlinked from the spacecraft and temporarily stored in the ground stations, was received and pre-processed at ESOC, and all the post-pass activities were completed. Based on the results of telemetry and telecommand verification checks, automatically performed by the control system, findings during the manual screening of report and exception messages and results from the special checks eventually indicated in the instructions from the mission planner, the overall spacecraft performance could be assessed, and in particular the successful execution of operations verified. If recovery actions were required, these were turned into internal Telecommand Requests and handed over to the engineer in charge of the next planning session. Once per week the activities were reported to and discussed in the Weekly Operations Review Meeting.

The sequence of the science operations was mainly driven by the limited availability of electric power and external events or constraints. Long term experiments, in particular those which could not be interrupted without endangering their mission product were given precedence in planning. Further resources to be considered during planning were the on-board storage capacity, the amount of application programs allowed to be run in parallel and the available

cooling capability.

The science operations could be implemented to a large extent according to the schedule laid down in the baseline plan prepared pre-mission. All science objectives, with few exceptions when severe equipment failures were encountered, could be fulfilled in the first 5 and a half months of the science mission phase. After this time the freon cooling system had to be deactivated, due to power shortages caused by the degradation of the solar panels, excluding operations of actively cooled payloads from that time onwards. However, the rest of the payload could continue operating, resulting in an over-performance of up to 175% w.r.t. the planned science program.

Highlights of the payload operations (for details see Innocenti and Mesland, 1993) were, among others, the first use of an inter-orbit communications link via Olympus satellite for operational purposes (uplink of Master Schedule, Nov. 24, 1993), the direct transmission of payload telemetry to home institutes (Oct. 15, 1993), the parallel operations of solar science instruments with their 'sister instruments' on the ATLAS-2 mission flown on a Space Shuttle, the successful EURECA depointing to support additional WATCH observations of different areas of the X-ray sky.

Most of the payload instruments experienced anomalies during the mission. The most severe cases encountered were the loss of the Radiofrequency Ion Thruster Assembly (RITA) quite early in the mission, the problems of the primary cooling system in the Protein Crystallisation Facility, and the Timeband Capture Cell Experiment foil movement failure. Other payload instruments showed relatively minor problems, often in the area of the communication functions with the data handling subsystem, which could be worked-around operationally and did not seriously reduce their science return.

The functionality and the Man-Machine Interface of several tools in the working environment of the operations team were not appropriate to the tasks they were used for. This had to be overcome by many additional manual steps, which were very time consuming and error-prone (e.g. long sequences of commands with many parameters had to be typed in by hand because no interface existed between the computer which received the

data electronically as part of Telecommand Requests and the control system computer). Many of these shortcomings arose because functions had to be dropped during the implementation stage due to time or budget limitations.

During its eleven months in orbit EURECA experienced a relatively high number of on-board anomalies, which had to be recovered from ground or counteracted by operational work-around solutions. Development, testing and execution of work-around solutions put a significant additional workload on the operations team. In particular in the beginning of the mission, when frequent communications outages between payload or subsystems and the data handling subsystem had to be recovered manually from ground, the pass-operations were seriously affected. Very often the scheduled pass activities, in particular dumping of telemetry from the on-board storage, had to be delayed or spread over several passes.

For the spacecraft performance monitoring the on-board communications problems caused further difficulties because the observability of the spacecraft temperatures was lost completely until recovery, i.e. either corrupted or old data were downlinked during this time. This data had to be manually filtered out if used for further analysis until a specific filter program was developed. A new on-board software was developed in the first weeks of the mission to recover autonomously from the above problem (Domesle *et al.*, 1994). This reduced the observability outages and simplified the pass operations significantly. Unfortunately the new software could not completely solve the problem due to other software design limitations on-board, therefore recovery was still left to the ground about once every fortnight.

Aggressive and unpredictable degradation of solar array performance forced the flight operations team to take additional power margin into account in science operations planning. A more special passive retrieval scenario was developed, in case the power loss would occur to support the retrieval as planned. However, the initial tendency of the solar array performance did not continue and no mission time had to be sacrificed, nor had an emergency recovery scenario to be applied.

However, the impact of this degradation on science operations was limited only due to the fact that a high power consumer instrument, RITA, failed after one month operation, releasing a large amount of power to the rest of the payload.

For many of the anomalies encountered work-around solutions could be found. This process however did not only require to reconfigure on-board items or to uplink new on-board software, but also to update operational documentation like in the User Manual and the FOP. In some cases new software had to be written for special evaluation purposes. Before a decision on a work-around solution could be taken, potential side-effects had to be excluded. This was extremely difficult in those areas where little on-board changes could develop large effects (e.g. in the area of Attitude Control Subsystem fault management software), or complex dependencies between real-time procedures (e.g. power degradation fault management) were not sufficiently documented. In some cases work-around solutions could not be applied since a final assessment of the side effects could not be made with the available simulation tools on ground. In other cases it was found out later that side-effects had been overlooked.

Summarising the experiences from this phase it must be said that the operations concept used for this mission phase was in general well suited to the mission characteristics. Its inherent flexibility allowed to implement the planned mission operations, to isolate and recover almost all observed anomalies and to define all required work-around solutions without introducing significant delays to the mission progress. Critical operations, like On-Board Software Maintenance activities, could be integrated in this approach as well. Weak points have been identified in the functions and the man-machine interface of the tools in the operations environment, which could be improved without major efforts. Problems encountered on the spacecraft seem to imply the need for more robust and flexible functionalities, on-board and on-ground, in order to cope with unforeseen anomalies and to support the implementation of work-around solutions. As a multiple work-around solutions situation becomes extremely difficult to manage, an increased effort should be spent during spacecraft development and test.

Lessons Learned

EURECA provided an excellent opportunity to build up a unique operational expertise in Europe in the following areas: manned spaceflight, including commanding and crew safety responsibilities; rendezvous activities; joint operations with NASA involving Shuttle, data-relay satellites, and complex ground segments; multidisciplinary science missions in low Earth orbit; advanced autonomous space-segments.

In running this mission a wide range of experiences was gained by using the spacecraft and the ground segment and by applying the described operations approach. The main lessons learned in the different areas are summarised in the following.

Spacecraft. Design, development and testing should aim to produce highly robust, flexible, and reliable components in order to avoid failures and malfunctions on one hand and to minimise the impact of unforeseen anomalies on other components. Critical areas in this respect seem to be the on-board communication and autonomous functions, which caused most of the severe anomalies in the mission with often dangerous side-effects. Completeness, stability, and early availability of a Spacecraft User Manual is very important to avoid overload situations for the flight control team during the final phase of the mission preparation.

Ground Segment. Flexible tools and man-machine-interfaces, well adapted to the often variable needs of mission control, play a very important role in the ability and capability to implement operational work-around solutions. particularly sensitive in this aspect are mission planning tools. For missions of the complexity of EURECA the flight control team should be built up several years before launch, to cope with the workload required for inter-agencies cooperation, database work, FOP preparation and verification activities.

Operations Concept. After extending the flight control team structure for critical phases by a dedicated safety engineer position, the operations scheme used in this missions was well suited to the mission. All encountered anomalies, even those occurring in parallel, could be successfully handled without delaying the mission progress.

3. CONCLUSIONS

The success of the EURECA A1 mission is the proof that the basic operations concept, the ground and space segment design were adequate. Several shortcomings in the system could be identified before and during the mission for which relatively simple solutions can be implemented for a future flight. Since the satellite needs only a relatively small funding in order to be prepared for another flight (about 67.6 MAU for all industrial costs, including launch support), and there are still EURECA slots allocated on NASA's shuttle manifest, a unique opportunity exists to repeat the success of the EURECA A1 mission on a second flight in the near future.

4. REFERENCES

- Domesle, R., H.Hübner and J.Leclerc (1994). On-board Software Maintenance Activities During The EURECA A1 Mission. *EURECA Technical Report, ESA WPP-69*. European Space Agency, Paris.
- Ferri, P. and S.Kellock (1992). A User View of the EURECA Ground Segment. *Proc. of the 2nd SpaceOps Symposium*. Pasadena (CA).
- Ferri, P., H.Hübner, S.Kellock and W.Wimmer (1993). The Joint ESA-NASA Operations for EURECA Deployment and Retrieval. *ESA Bulletin, 76*, 81-90.
- Ferri, P. and W.Wimmer (1994). The Experience with the EURECA On-Board Autonomous Fault Management Functions., *Proc. of the IFAC Symp. on Automatic Control in Aerospace*. Palo Alto (CA).
- Hübner, H and W.Wimmer (1994). Experience with EURECA's On-Board Autonomy from an Operations Point of View. *To be published*
- Innocenti, L. and D.Mesland (1994). EURF Scientific Results: An Overview. *EU Technical Report, ESA WF* European Space Agency, Paris.
- Van Casteren, J. and P.Ferri (1989). Flight Operations, *ESA Bulletin*.
- Wimmer, W. and P.Ferri (1994). Mission Summary. *EURECA Report, ESA WPP-69*. Eur. Agency, Paris.

SCOS II: ESA'S NEW GENERATION OF MISSION CONTROL SYSTEM

M Jones, N C Head
 Flight Control Systems Department, European Space Operations Centre (ESOC)
 Robert Bosch Str. 5
 64293 Darmstadt, Germany

K Keyte, Vitrociset SpA, Rome, Italy
 P Howard, Science Systems Ltd, Bristol UK
 S Lyneskjold, Computer Resources International (CRI), Birkerød, Denmark

ABSTRACT

New mission-control infrastructure is currently being developed by ESOC, which will constitute the second generation of the Spacecraft Control Operations system (SCOS II). The financial, functional and strategic requirements lying behind the new development are explained. The SCOS II approach is described. The technological implications of these approaches is described: in particular it is explained how this leads to the use of object oriented techniques to provide the required "building block" approach. The paper summarises the way in which the financial, functional and strategic requirements have been met through this combination of solutions. Finally, the paper outlines the development process to date, noting how risk reduction was achieved in the approach to new technologies and summarises the current status future plans.

1. INTRODUCTION

This paper describes the new infrastructure for Mission Control Systems which is being produced at the Operations Centre of the European Space Agency in Darmstadt. This infrastructure is the second generation of the Spacecraft Control Operations System (SCOS) and will replace the current generation SCOS I (Mullet et al., 1990) for all new ESOC mission implementations. First candidate client missions are ARTEMIS (a data relay mission), ENVISAT (an earth observation mission) and HUYGENS (a Titan probe). The paper concentrates on the programmatic and the

main functional aspects; technical details related to the implementation techniques and technologies can be found, for example, in Keyte (1994).

2. WHY SCOS II?

In order to provide the context for a discussion of SCOS II and its features it is important to have an understanding of the motivations behind the development of "yet another" set of mission control infrastructure and of the general operational environment in which SCOS II based systems will be used. The main factors which led to the SCOS II development are broadly as follows (Jones et al. 1993):

- financial:

The development of Mission Control systems based on ESA's current generation of infrastructure software (SCOS I) is costly. This is due, at least in part, to the inflexibility of the SCOS I system structure and the resulting difficulty of customising SCOS I software to a mission and of adding mission specific software to the basic system.

- functional:

The increasing complexity of missions requires a corresponding increase in the capabilities of the control systems. For the same reason the effort involved in preparing and monitoring mission operations is increasing.

- vendor independence:

The cost and flexibility of computer hardware for previous systems have been item of concern. The centralised, host-based, architecture of these systems which, resulted in the use of large mainframe computers to support mission operations. This resulted in dependence on the operating system and basic software provided by vendors of the particular host computers chosen, thus effectively tying the Agency to these vendors.

The major drivers for SCOS II can thus be summarised as reduced cost per mission with increased flexibility and portability.

3. THE SCOS II PROJECT: OVERALL APPROACH AND PROGRAMMATICS

The SCOS II project began in 1990 with the general aims outlined in the previous section. A large investment of effort was made to define a comprehensive set of users requirements and associated operations concepts resulting in a very substantial User Requirements Document (URD). This work is outlined in a companion paper (Kaufeler et al. 1994). At an early stage a decision was made to use object oriented analysis, which with its focus on the Problem Domain, encouraged interaction between the

User Requirements work and the software requirements analysis. This led to the need to cope with evolving user requirements and overlapping development phases. How this was resolved in terms of software development approach is discussed by Pujo et al. (1994) and Symonds et al. (1994). The implementation language is C++.

The implementation is proceeding in a series of releases, which will successively add functionality to cover the all areas of the URD. Release 1, due in early 1995 includes the new concept of "system elements" explained in the next section and will have equivalent functionality to the existing SCOS I infrastructure, including in addition telecommand functions (missing from SCOS I) and more modern user interfaces. A "Proof of Concept" prototype was developed and demonstrated in early 1993 to verify the feasibility of the distributed system technology. At the end of 1993 a "telemetry demonstrator" was available, which showed telemetry processing basic functions and associated man-machine interface.

Release 2 (1995-1996) and Release 3 (1997-1998) will add further advanced functionality including areas such as mission planning which have never been treated generically within ESOC before.

4. WHAT IT DOES: THE PROCEDURE-ORIENTED OPERATIONS APPROACH AND SYSTEM ELEMENTS

As in most Operations Centres, ESA mission operations are centred around "procedures" which are executed automatically subject to the occurrence of specified events (usually anomalies) in either the flight or ground segments. Non-procedural (i.e. manual) operations are

reserved for those inevitable cases where appropriate procedures have not been foreseen.

Mission operations engineers usually regard spacecraft as being composed of a number of systems, sub-systems or assemblies. The process of mission control consists of performing actions (either active, controlling ones or passive, monitoring ones) with one or more such systems or sub-systems. Each of these actions is driven by an appropriate procedure.

A particular procedure may "call-up" other procedures to perform some portions of its work. Similarly, a procedure may be called by other, higher level, procedures to perform some actions on their behalf. Loosely speaking, the set of procedure for a mission can be viewed as forming a tree-like hierarchy whose structure is very closely related to the hierarchy formed by the system, sub-system and assembly relationships of the spacecraft itself.

SCOS II infrastructure directly supports the modelling of systems, sub-systems and assemblies. These components are all represented as objects referred to as "System Elements". The relationships between these Elements are stored in the mission database in a tree-like structure (see fig. 1). System

Elements are used in a number of ways in the SCOS II system.

4.1 Abstract Monitoring & Activities

The execution of a typical procedure consists of three major phases:

- setup: checks to ensure that preconditions for execution of the procedure are satisfied and that required tools are available
- execution: use of the tools to perform the activity (this may be a passive, monitoring only activity)
- assessment: check that the results of the activity are as expected and that all required post-conditions are satisfied.

SCOS II System Elements provide support for all three phases, hiding the use of subordinate System Elements from the user once this use has been defined in the database:

- a System Element provides an high level view of the current status and mode of the unit which it represents; initiation criteria for the procedure can be expressed in terms of these

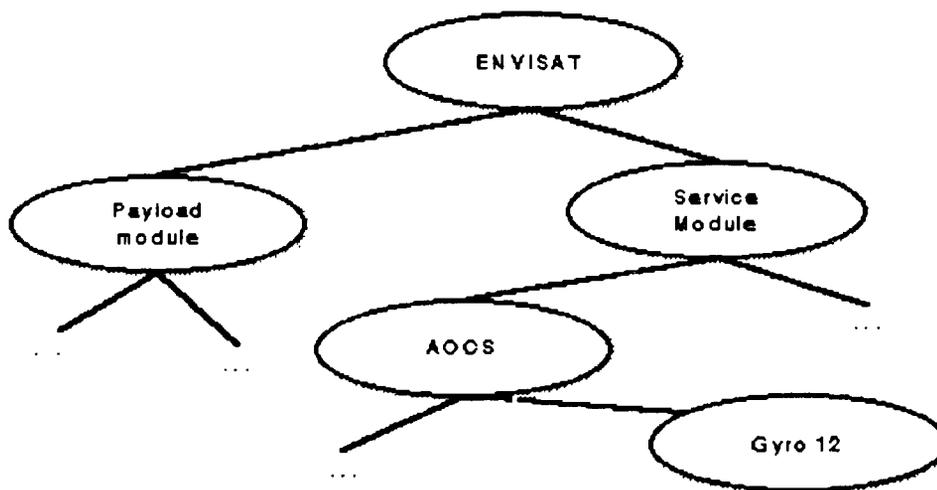


Figure 1 A simple hierarchy of System Elements

values (for example "is the AOCS in fine pointing mode?")

- a System Element provides a set of high level activities which can be initiated either directly or on behalf of other procedures (for example 'perform thruster cat bed preheat')
- a System Element, again based on high level status and mode assessments, allows simple assessment of the success of the activity (for example, 'is the AOCS now in sun-pointing mode with nutation < 0.1 degrees?')

4.2 Event-based procedure initiation

SCOS II provides capabilities for System Elements to signal the occurrence of "Events" (a mode change for example) and to associate "Actions" with these occurrences. One of the types of Action which will be supported is the initiation of a procedure (referred to as an Activity in SCOS II) which may be either diagnostic in nature or may, in the case of unexpected or critical events, take some form of safe mode initiation.

4.3 Building system elements - element templates

Often a spacecraft will have a number of similar devices (gyro's for example) which have an essentially identical set of operational procedures; differences are only to be found in the specific parameters and command encoding details. SCOS II supports the concept of System Element Templates which contain a master definition of the Element behaviour with empty slots for such specific data. Populating the mission database for each of the specific instances of the templated unit is then a

matter of 'filling in the blanks' in the template. This should greatly ease the version control of the database as updates need only be applied once to the template rather than several times. testing of the database will be similarly reduced in cost.

4.4 Element Connections & Dependencies

Many operational constraints and checks in traditional systems are centred around a relatively small number of issues (power status, redundant unit status etc). The configuration of a traditional system to deal with these consumes a significant proportion of the overall configuration effort. SCOS II explicitly supports the concepts of relations between System Elements for (a) power supply and consumption, (b) redundant sets of devices and (c) data routing and forwarding.

These relations, once defined in the database, allow the system to automatically perform many of the processing and control functions which have previously required explicit implementation. Again, this will reduce the cost of configuring the system for a specific mission

4.5 Navigation at the user interface

The System Element hierarchy is also used to provide structure for the user interface navigation facilities; the MMI allows navigation through the database and through the online parts of the system by following the various links between the System Elements. This allows easy movement from say a gyro pack to its power source or to its redundant unit.

4.6 Procedure manipulation

The actual text of the procedures will be made available via the System Elements. For example, when viewing the contents of the AOCS System Element the user will be able to access all AOCS related procedures directly from the MMI rather than via some separate application and a numbering convention to locate AOCS items.

4.7 Integration of ground & flight segments

Perhaps most importantly the concept of System Elements has been extended to allow their use to represent also portions of the mission ground segment (for example ground station equipment, wide area networks, SCOS II workstations themselves). This allows integrated monitor and control of a complete mission system from a single position. A particular advantage of this is the possibility to merge actions for the flight segment with actions for the ground segment in a single SCOS II Activity in the same way as they are merged in the paper procedures of the current systems. An example of such a merged activity might be the AOS (Acquisition of Signal) for a low earth orbiting spacecraft. A simple summary of the steps involved might be:

1. Perform pre-pass dataflow tests (TC to station)
2. verify dataflow tests (in flight & station TM)
3. Transfer orbital elements to antenna controller (TC to station)
4. Select Program Track (TC to station)
5. Wait for notification of receiver lock (in Station TM)
6. Initiate uplink sweep TC to station)
7. Wait for onboard receiver lock (in flight TM)
8. Select Auto Track (TC to station)

Previous mission systems have implemented a variety of *ad hoc* approaches to such combined control and monitoring of flight and ground segments which however have confirmed the benefit of such integration.

5. CUSTOMISATION FOR MISSIONS

The greater capabilities of SCOS II are obtained at the cost of extra information required to set up the system during mission preparation.

To minimise this cost, the System Element concept described in sect. 3 offers an obvious vehicle for implementation of mission specific requirements. System Elements can be viewed as "building blocks" which can serve as a basis for the implementation of these requirements. They can be extended and configured in two different ways (a) by specialising building blocks and (b) use of Operations Language (Baldi et al., 1994):

5.1 Specialising Building Blocks

This is done by a mission specific software engineering team. SCOS II is implemented following an object oriented approach; in particular the System Element is the base of a class hierarchy which allows for progressive, incremental specialisation towards a final System Element representing, for example, an onboard computer for the 'XYZ' mission - see fig 2. This is in fact the genuine software reusability offered by object oriented techniques.

In the long term it is hoped to achieve further reuse of specialised building blocks by sharing them between missions which use the same units in the flight and ground segments. Standardisation of mission

hardware units could thus bring much larger cost savings than any of the measures taken to improve the efficiency of implementation of a single mission system.

5.2 Use of Operations Language

Operations engineers can also perform customisation to make limited changes to existing building blocks. SCOS II allows configuration of many aspects of a System Element through the use of the SCOS II Operations Language. This language is a synthesis of previous languages used in both operations and checkout and allows the production of not only of procedural or algorithmic parts of System Elements (for example command sequences, synthetic telemetry parameters, verification algorithms) but also rule-based parts which allow the identification of Events (described above) leading to the triggering of

Activities. The Operations Language may be either compiled or interpreted; this choice will be made by the operations team, based on the conflicting needs of performance and ease of modification for each System Element.

6. HARDWARE CONFIGURATION

Initial installation of SCOS II at ESOC will be on a Local Area Network of SUN Sparc 10 workstations running the Solaris 2.3 operating system. However SCOS II is being implemented to be portable across almost any Unix (System V or POSIX compatible) workstation platform. Parts of the system developed to date have been successfully run on SUN IPC and IPX platforms; respectable performance has been achieved without any particular attention to optimisation. Small parts of the system have also been run on Intel 486 based machines

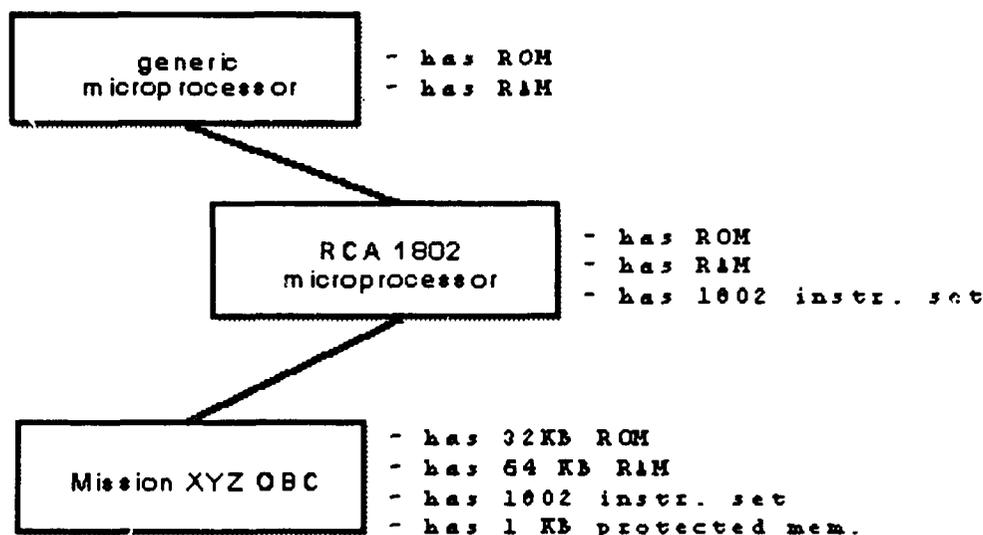


Figure 2 An example of progressive specialisation

(running a largely System V compatible Unix clone); initial indications are that performance is comparable to that of the smaller SUN machines and that this is also a viable platform for missions with low data rates (less than 10 kbits/s) and without exotic science data processing needs.

Although designed to be a distributed system running on large networks of processors SCOS II is also able to run on a single workstation (although obviously no redundancy is available in such a configuration and some performance limitations are to be expected) while still supporting all functions of the distributed system. No software or database modifications are needed to run in this manner. This configuration is known, informally, as "SCOS II-in-a-box".

7. CONCLUSIONS

In conclusion it can be said that the SCOS II project is the first attempt within ESA to provide a highly configurable and reusable software toolbox for building mission control systems. Its main aim has been to reduce costs, increase functionality and achieve vendor independence. To achieve *cost saving* mission specific costs, it uses object-oriented and other modern techniques to increase reusability and allow easy customisation. Greater *functionality* is provided; even in its Release 1 version there is more functionality than in previous ESA mission operations infrastructures and this will improve further with Release 2 and 3 work foreseen in 1995-1998. *Vendor independence* is provided through choice of UNIX and suitable implementation measures to achieve portability. This means that SCOS II could be used for a wide range of missions range from large ones requiring 30-40 workstations and high data rates down to small, low cost missions based on one or

two low-cost platforms. Extension of its use to other areas of the ground segment or mission lifecycle (eg. spacecraft checkout, backup control centres at station, ground segment control) hold out the possibility of further rationalisation and cost saving.

REFERENCES

- Baldi, A., Elgaard, D., Lynenskjold, S., Pecchioli, M., *SCOS II OL: A Dedicated Language for Mission Operations*, (1994), Proceedings of the Third International Symposium on Space Mission Operations and Ground Data Systems, November 1994, Greenbelt, Maryland
- Jones, M., Head N.C., Keyte, K. & Symonds, M., (1993) *SCOS II - ESA's New Generation of Mission Control System*, ESA Bulletin 75
- Kaufeler, P., Pecchioli, M., & Shurmer, I. (1994), *SCOS II - ESA's New Generation of Mission Control System- The User's Perspective*, Proceedings of the Third International Symposium on Space Mission Operations and Ground Data Systems, November 1994, Greenbelt, Maryland
- Keyte, K.P., (1994) *SCOS II - A Distributed Architecture for Ground System Control*, Proceedings of the International Symposium on Spacecraft Ground Control and Flight Dynamics, J. Brazilian Soc. of Mech. Science, XVI(319).
- Mullet, B., Kaufeler, J.F., Bowers, J. & Ahier, M., (1990) *The SCOS: A Configurable Infrastructure for the Control of Spacecraft Using Packets*, ESA SP-308(29)
- Pujo, O., Head N.C. & Jones, M., (1994) *Object Oriented Design, The Software Life Cycle and Software Engineering Standards*
- Symonds, M., Lynenskjold, S., Müller, C., *SCOS II- An Object Oriented Software Development Approach*, (1994), Proceedings of the Third International Symposium on Space Mission Operations and Ground Data Systems, November 1994, Greenbelt, Maryland

CCS-MIP : LOW COST CUSTOMIZABLE CONTROL CENTRE f-6

Christian LABEZIN, Pierre VIELCANET
CISI Aerospace Branch
13 rue Villet
F - 31029 Toulouse Cedex

ABSTRACT

The positioning and station keeping of French national satellites are among the main missions of CNES French Space Agency CNES. The related experience and skills of the Toulouse Space Centre are reknown and often required at international level for a wide range of missions. CISI, a software engineering company, has been contributing during the last 20 years to the development of the French space programmes, particularly in the field of space missions ground control segments. The CCS-MIP system, presented here, is a satellite positioning and station-keeping system designed to answer the CNES multi-mission needs, easily adaptable for a wide range of applications.

INTRODUCTION

National satellites station positioning and station keeping must cope with the following essential quality requirements :

- operational use 24 hours a day,
- safety over critical operations,
- low-cost system operations,
- system robustness and user friendliness,
- system expandability and maintainability.

Most of the systems currently in-use that meet those requirements are maintained and adapted at high costs and are not easily tuned to the actual operational needs.

No doubt that the demand for fast and easy adaptations of control systems is strong in competitive contexts like simple and recurrent space missions with limited budget, or mini-satellites programmes. There is then room in space operations for a flexible system at low recurring cost.

Through their previous developments, CNES and CISI have acquired the necessary skills for the design, the development, the adaptation and maintenance of all the components of a control centre. CISI has therefore developed for CNES the CCS-MIP satellite control centre, in accordance with the above mentioned requirements.

The specific and competitive features of the CCS-MIP, presented in this paper, are based on a tailored architecture hosting simplified basic functions, minimum specific software and extensively reused software applications.

CCS-MIP SUPPORTED FUNCTIONS

This system, designed in agreement with up-to-date mission requirements, has proven to be a sound industrial solution in terms of low cost control centre offering the following functionalities :

- acquisition of telemetry and localisation data,
- telecommands preparation and emission
- telemetry decommutation,
- telemetry display (including minisics),
- telemesure processing,
- integration of orbit and attitude restitution functions,
- logging and monitoring of application and system events.

Figure 1 below shows the main dataflows between these functions and the basic interactions with the satellite database.

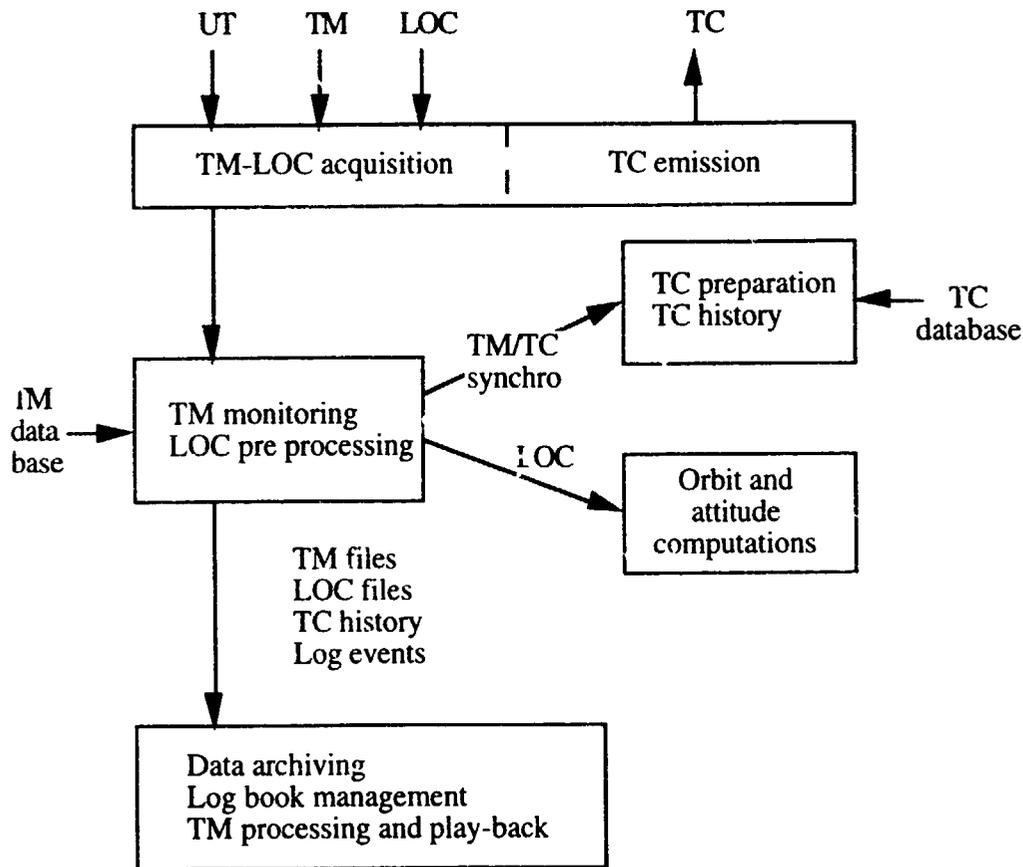


Fig. 1 : Ground control segment main functions

TECHNICAL DRIVERS

The major technical drivers have been identified in light of CNES and CISI deep experience in the development of satellite ground segments and more particularly of control centres. Due to the rapid technical evolution in this domain, the CCS-MIP has been implemented with the following design options :

- distributed computer architecture, as shown in figure 2, including real-time processors for satellite and ground stations monitoring and control, and off-line processors for operational data preparation, satellite data archiving and processing and attitude/orbit computations.
- use of a reliable and compatible hardware chosen among the first rank computer vendors
- centralisation of the processing and decrease of the number of necessary operational workstations
- use of X terminal stations

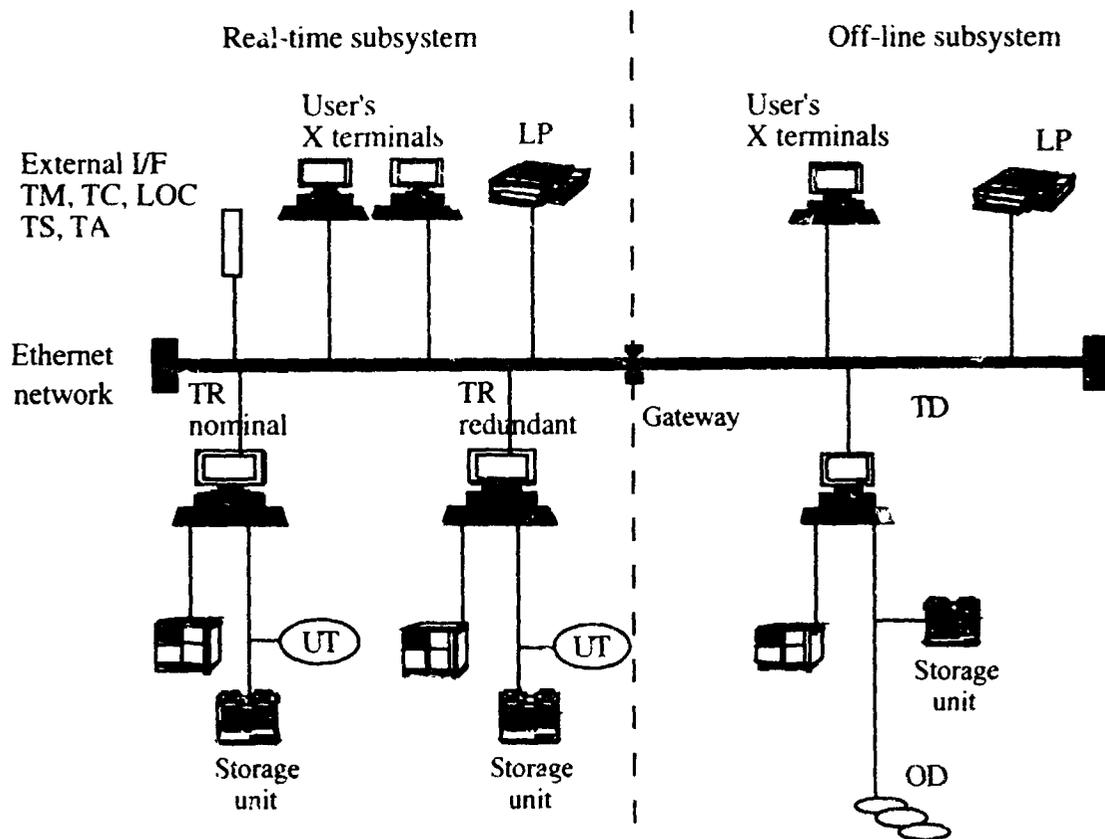


Fig. 2 : CCS-MIP generic architecture

Legend :

TR : Real-time
TD : Off-line

TA : Teleactions
TS : Telesurveillance

UT : Universal time
OD : Optical disc

- selection of stable and broadly used industrial standards for operating system (Unix system V), network protocols (Ethernet, TCP/IP, FTP) and graphical user interface (X Windows, OSF/MOTIF),
- use of mature software packages and reuse of existing software
- design of simple and guiding dialogues, self-explanatory graphical displays and on-line user-friendly help facilities,
- important parameterisation of the system, external interfaces adaptability and functional modularity,
- multi-satellite missions capability, for several satellite systems (SPACEBUS, EUROSTAR...).

These choices have resulted in the availability of a simple and customizable platform to be enhanced in order to meet most specific requirements. For instance, the needs of a mini-satellite program can be taken into account within short delays and at a very attractive cost.

CCS-MIP DEVELOPMENT METHOD

The development method used for the CCS-MIP project is depicted in figure 3. Two main stages appear in the CCS-MIP project lifecycle : the product definition and design stage with strong involvement of CNES and CISI engineers and the production stage performed by CISI. The production line refers to the stable W (hardware and software) development model.

This industrial effort has been supported by CISI Quality organisation conforming the ISO 9001 standards for studies, turn-key developments and software maintenance.

The product definition and design stage is rather innovative due to the adoption and systematic use of value analysis techniques.

Requirements screening, reviews of the specifications, trade-off on candidate architectures, assessment of available technologies and components have been performed (and iterated whenever necessary), in order to reach a valuable solution meeting the real needs under severe cost (and risk) reduction constraints.

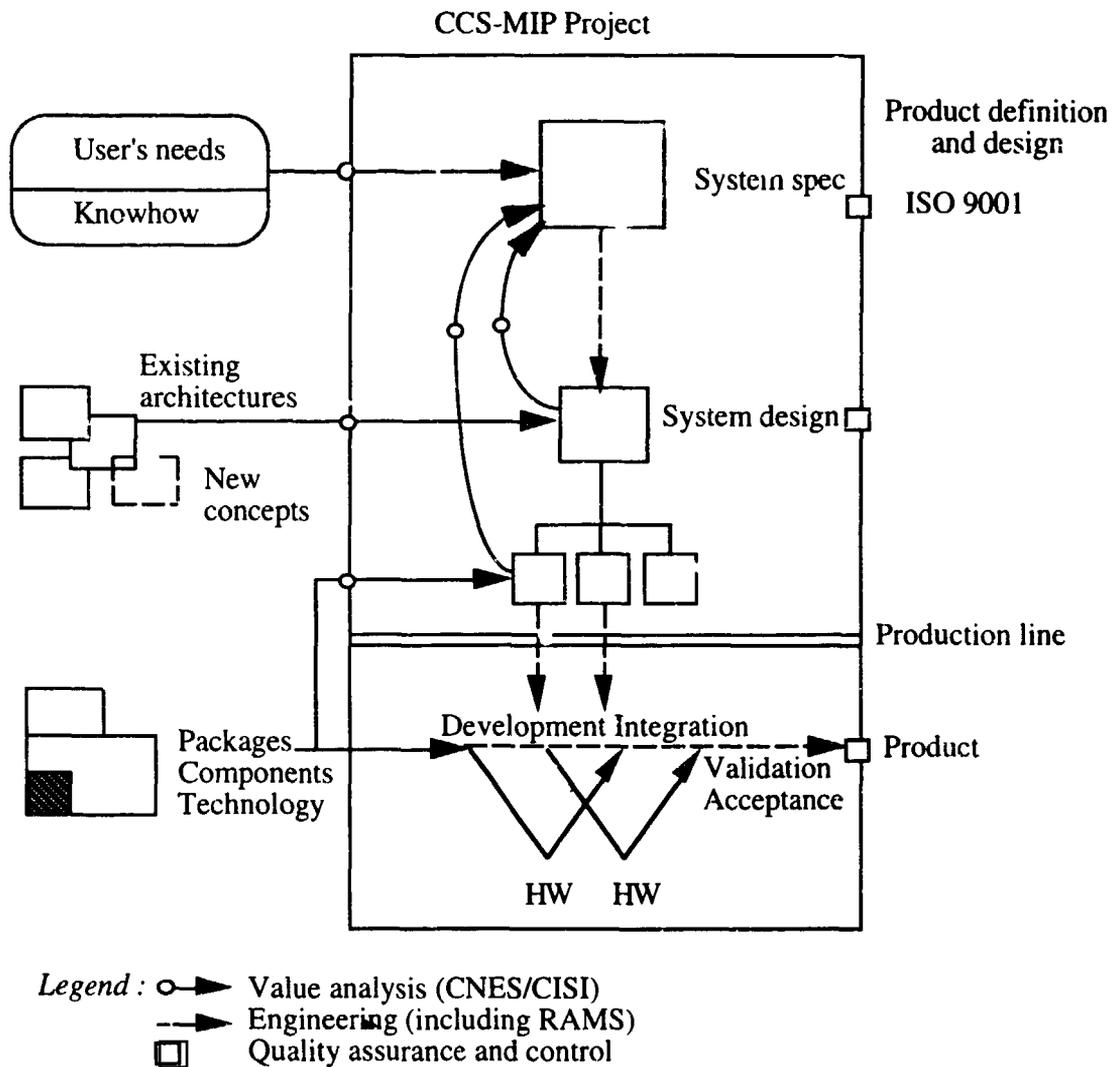


Fig. 3 : CCS-MIP development method

CONCLUSION : A REUSABLE SOFTWARE AND HARDWARE PLATFORM

The CCS-MIP answers in a very efficient way the functional objectives, operational needs, and quality requirements of the new control centre generation. In order to reach this goal, CNES and CISI analysed, understood and often simplified these requirements to get a well integrated solution meeting the user's needs. For instance, the resulting right-sized system is considerably easier to operate than its predecessors.

This early and global review of all requirements (optimal analysis approach) turned out to be very effective. This successful approach has been greatly supported by the experience and skills of the customer and contractor teams.

CCS-MIP is currently fully operational for TDF1 and TDF2 station keeping and is ready to use for the TURKSAT satellite positioning mission.

ACKNOWLEDGEMENT

We want to thank here all participants to the CCS-MIP effort, more precisely CNES engineers, initiators of the development method reported in this paper, who have greatly contributed to the definition and global design of this new control centre and CISI engineers who successfully implemented the CCS-MIP system.

THE IUE SCIENCE OPERATIONS GROUND SYSTEM

Ronald E. Pitts and Richard Arquilla
Computer Sciences Corporation Astronomy Programs
1000A Aerospace Road
Lanham-Seabrook, MD 20706

ABSTRACT

The International Ultraviolet Explorer (IUE) Science Operations System provides full real-time operations capabilities and support to the operations staff and astronomer users. The components of this very diverse and extremely flexible hardware and software system have played a major role in maintaining the scientific efficiency and productivity of the IUE. The software provides the staff and user with all the tools necessary for pre-visit and real-time planning and operations analysis for any day of the year. Examples of such tools include the effects of spacecraft constraints on target availability, maneuver times between targets, availability of guide stars, target identification, coordinate transforms, e-mail transfer of Observatory forms and messages, and quick-look analysis of image data. Most of this extensive software package can also be accessed remotely by individual users for information, scheduling of shifts, pre-visit planning, and actual observing program execution. Astronomers, with a modest investment in hardware and software, may establish remote observing sites. We currently have over 20 such sites in our remote observers's network.

INTRODUCTION

The International Ultraviolet Explorer launched in January 1978, makes ultraviolet spectral observations of astronomical objects in the wavelength range 1150-3200 Å. The satellite occupies a moderately elliptical geosynchronous

orbit centered approximately over northern Brazil. IUE is three-axis stabilized by a special attitude control system using the remaining two of six original gyroscopes, the Fine Sun Sensors, and (at times) the Fine Error Sensor (FES) star tracker. This system can provide arcsecond pointing accuracy and stability. All space-borne activities are controlled by IUE's On Board Computer (OBC), which utilizes 8 Kb of memory.

The spacecraft is commanded jointly in real-time from the IUE Telescope Operations Center (TOC) (Science) and the IUE Operations Control Center (IUEOCC) (Engineering) at Goddard Spaceflight Center for 16 hours each day, and from the European Space Agency's Villafraanca del Castillo Satellite Tracking Station near Madrid, Spain for 8 hours a day.

By mid 1990 the original Experiment Display System (EDS) in the TOC was in need of replacement and the main ground system computers in the IUEOCC were reaching the limits at which further software enhancements and work-a-rounds could be installed. With NASA approval, Computer Sciences Corporation created a working group in late 1990 to design a new science operations Ground System. The group presented its system design to NASA in early 1991. The plan was approved and the new Telescope Operations Control Station (TOCS) system went on-line in early 1992.

Commands issued by the Telescope Operator (TO) from the TOCS workstation are transmit-

ted to the Xerox Sigma 5 (prime) or Sigma 9 (backup) mainframes in the IUEOCC as a series of parameters and calls to encoding procedures. The Sigma computer generates the necessary standard command sequences and data blocks which are then uplinked via a dedicated antenna at the NASA Wallops Island Tracking facility. The spacecraft can be seen from the Wallops site 24 hours a day. All commanding is routinely monitored and under the general supervision of the Operations Director who, along with the engineering staff, have display consoles in the IUEOCC. Return telemetry from the spacecraft is received at the ground station and forwarded to the Sigma computer. Spectral image data are reconstructed, archived to data tape and disk, and sent to the TOC via a NASCOM link for display and quick-look analysis by the Science Operations staff and Guest Observer (GO). The GO may adjust his/her plans in real-time, based on the real-time data and staff advice.

The overall reliability of the ground system is crucial since the IUE has no on-board tape recorder and the SEC Vidicon spectrographic cameras use a "destructive" read. Quick-Look analysis is also critical because there is no on-board "exposure" meter and the IUE daily transits the outer fringes of the Van Allen belts.

SYSTEM CAPABILITIES

The Telescope Operations Center complex has three main functions:

- It initiates science instrument, tracking, and maneuver commands to the spacecraft.
- It allows quick-look data analysis and real-time data quality assessment.
- It provides real-time and pre-visit planning tools to the guest observer and staff.

The commanding capabilities are built into a work station window environment which is shown in Figure 1. The standard configuration consists of a large window dedicated to image display and smaller window "buttons" rapidly accessed by a mouse. The image window accepts spectral and star-field coordinate overlays and provides position information, such as a star's position in the star tracker field of view, to the command software via the cursor. The buttons contain extensive menus of the commonly used commands, complete with relevant arguments. A command is selected by the mouse cursor and displayed on a command line at the bottom of the window prior to being transmitted to the mainframe.

Both the main display window shown in Figure 1 as well as the other displays used on the TOCS are generated using the various widget tools of the Interactive Data Language (Research Systems Inc). This allows the generation of complex graphical displays with a minimum of coding compared with a standard X-windows tool kit.

Three buttons activate especially useful sub-windows which allow the staff to efficiently command the spacecraft for extensive periods with little keyboard input. One window, shown in Figure 2, allows the TO to "type ahead" a sequence of anticipated commands for a particular operation, so that each command can subsequently be selected and sent to the Sigma computer as needed. The second button stores a log of previously used instructions, and any command can be recalled and reissued to the spacecraft. The third button retrieves coordinate and other astronomical information for a specific target from the TOCS database for storage in an image science header or as input for maneuvering calculations. This option has eliminated the need for target list tapes.

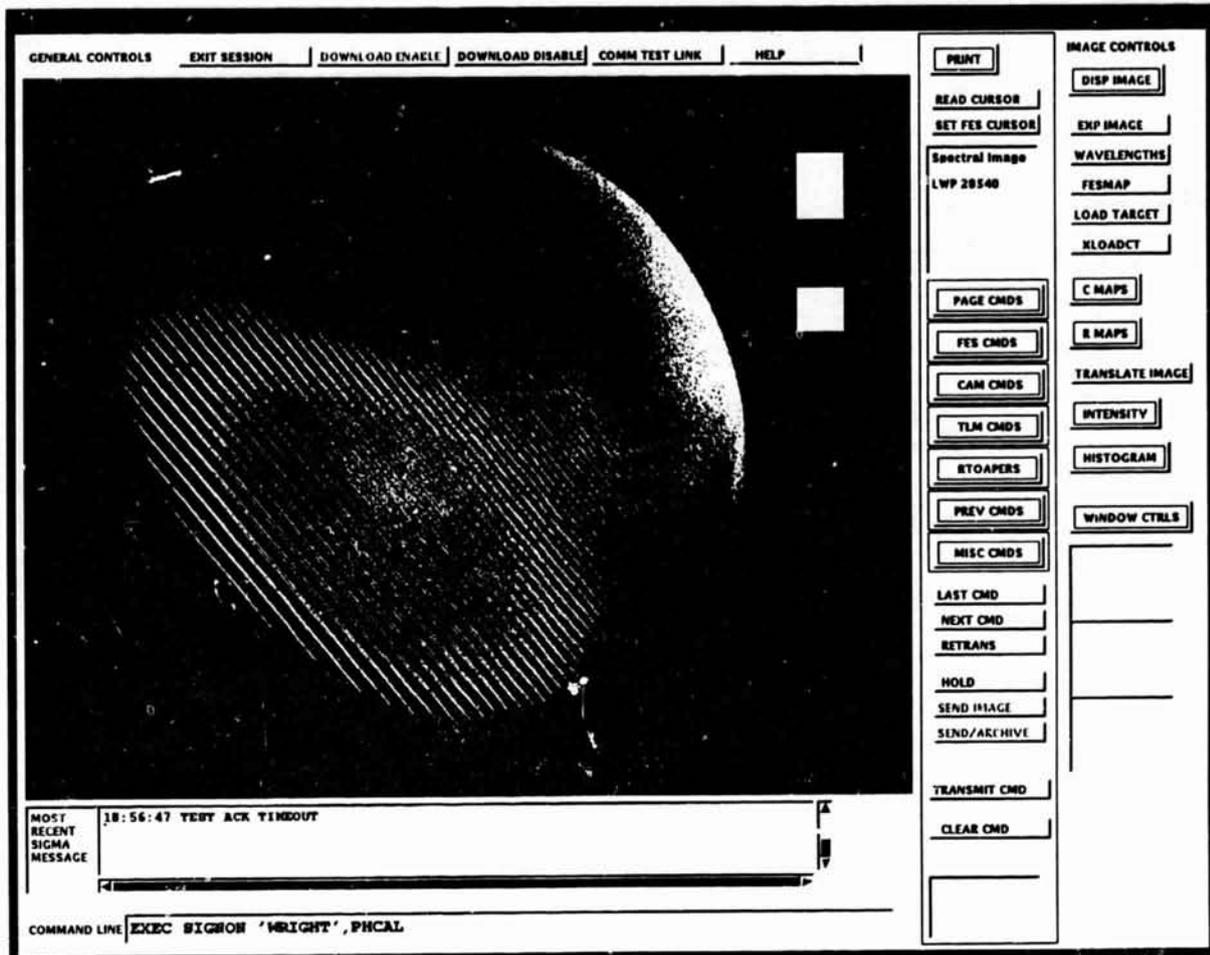


Figure 1: The Main TOCS display window used by the Telescope Operator to initiate commands to be sent to the IUE spacecraft. The main image window is displaying a high dispersion echelle spectrographic image. An accompanying 16-level color scale (only partially visible in the black and white image) allows a visual determination of the exposure level of the image.

Additional command station buttons call routines to analyze or manipulate raw spectral images or FES images used for target acquisition. These images are accompanied by wavelength or position overlays, the latter fully capable of simulating the features of the FES field of view. These quantities may also be derived from cursor output. Both basic and more sophisticated code is available to select subregions of an image for close examination or to enhance contrast in real-time to identify faint spectral features or stars. Basic spectral analysis functions

include histogram plots of pixel exposure levels in selected regions and intensity plots of levels versus wavelength to define the shape of the spectrum. These plots are placed in temporary subwindows containing their own manipulation functions. A sample of an intensity plot is shown in Figure 3.

The workstations store read-down images for several days so that data from previous observing sessions can be examined. Periodic purges are performed automatically. This is a useful

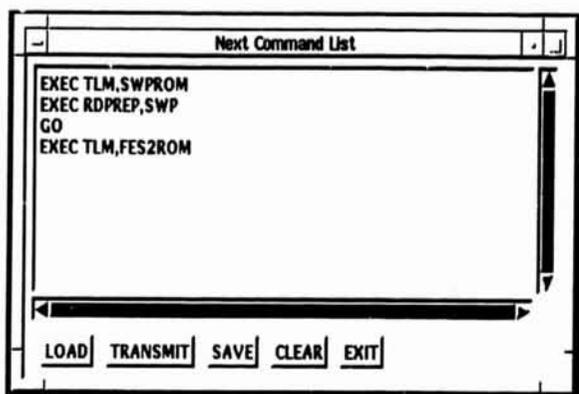


Figure 2: The Next Command Sub-Window.

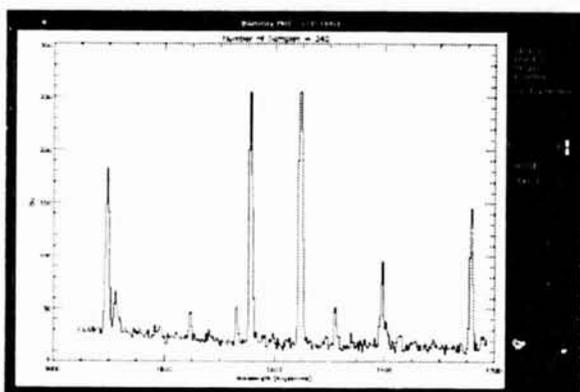


Figure 3: A sample intensity plot as a function of wavelength is shown from a low dispersion IUE image plot. The object is the emission line rich star, RR Tel.

feature for programs monitoring variable objects.

Using offline support software, images can be forwarded to an account on the support station dedicated to Guest Observer use and accessed via a VT 1300 color X-terminal. This account, in addition to facilitating communications with other systems, creates a window environment having the same image analysis capabilities as the TOCS. Thus the observer may examine data in detail without impacting operations. The user may also produce laser printer hardcopy of

the image and corresponding data plots. This feature has greatly reduced the use of expensive camera film for this purpose.

A fundamental strength of the ground system is the existence of a diverse set of offline software, in both window and menu format, available to both the staff and observer for scheduling, planning, and altering (in real-time) an observing timeline. The code is installed on the primary and backup command workstations, and on a VAX 4000 computer.

All software not directly accessible from the main TOCS window display (*i.e.* the offline software), is called from an auxiliary button window which can be called up on any X-window terminal. This is shown in Figure 4. This keeps the software organized so that the Resident Astronomer does not have to remember specific command call syntaxes. Any required input parameters to these program modules are either automatically loaded from a database area, or small window widgets specifically created for typed input. Most of this code is also available to remote users in a dedicated menu-driven account which requires only the equivalent of a VT100 terminal.

ANTENNA ANGLES	FPM	PHCAL LOG	SAOCAT
ANTENNA PATTERN	GUIDESTAR CATALOG	PHCAL TARGETS	SD TIME
ADSFES	IUE PARALLAX	POSITIONS	SERENDIPITY
BETADOT	LIGHT	PRECESSION	SKYMAP
CHECK DATA	LOST ATTITUDE	PRIORITY PROCESS	STREAK
EARTH BETA ANGLE	MANEUVER	PYAD50	TARGETYEAR
EARTH OCCULTATION	NFESGDE	REMOTE DOWNLOAD	TRANSFORMS
FES DISTORTION	NWGUIDE	RA HELP	USSBS TARGETS
FES MAGNITUDE	OBC ERROR	RA WINDOW	HELP
		REPLAY	EXIT

Figure 4: The RA button window used to call Auxiliary Software programs.

The offline software uses lunar, solar, and or-

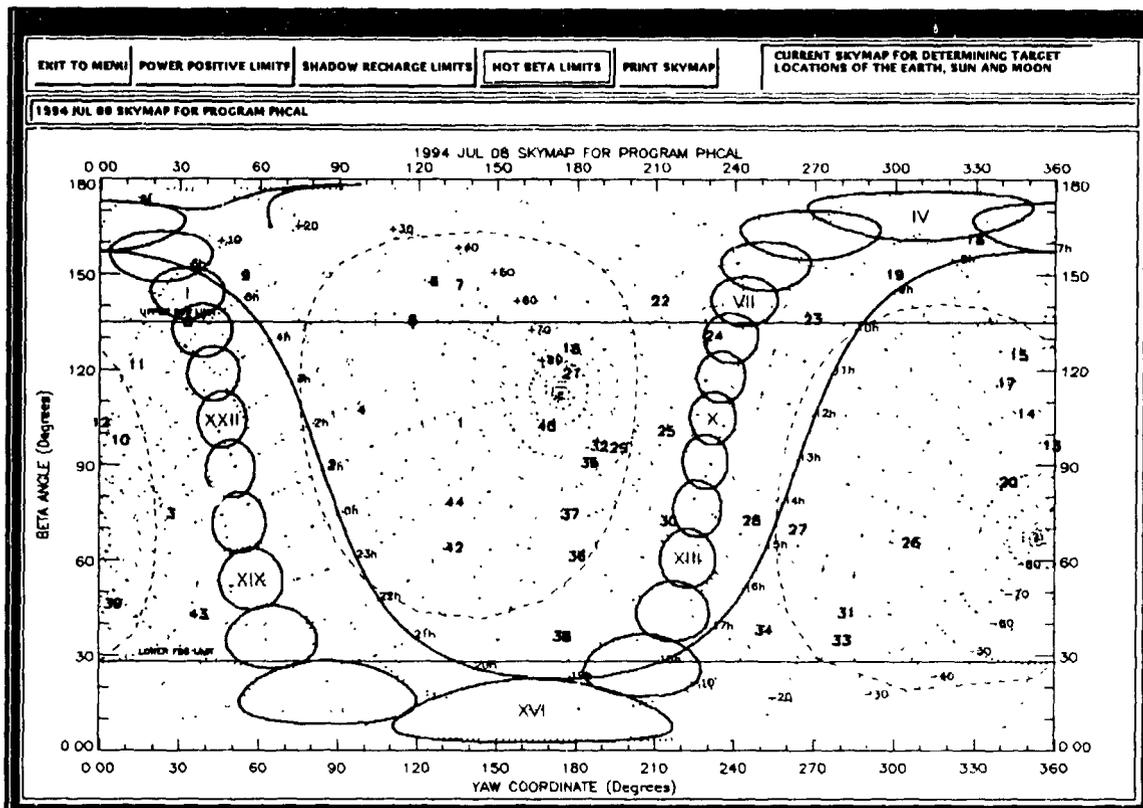


Figure 5: A sample IUE Skymap showing various IUE constraints. The numbers scattered about the map refer to targets of the observer's program. The circles and Roman numerals represent the Earth at various UT times during the day. The dashed lines are the outer limits of the Earth Avoidance Zone (an advisory limit). The dotted grid is the standard right ascension and declination coordinates. The inverted V-shaped line in the upper left-hand corner of the map is the moon's path as seen by IUE over a 24-hour period. The map is plotted as β (the supplement of the Solar angle as seen in IUE's coordinate system) versus the spacecraft yaw angle. In this frame of reference, spacecraft maneuvers from target to target can be plotted as combinations of vertical and horizontal line segments.

bital ephemerides to examine a target's availability and acquisition properties for any time of the year. Users identify potential sun angle (power and control) constraints, earth or lunar occultations, S-band antenna pointing (telemetry signal strength) problems, and the orientation of an extended target relative to the spectrograph apertures in order to schedule observing time or adjust programs in progress. The user can display a skymap (see Figure 5) showing the positions of program targets, and calcu-

late maneuver times between any pair of objects. The observer may obtain an updated observing schedule, read observatory policies, and submit observing forms remotely.

A subset of the software pertains to fine acquisition of targets. The most commonly run program uses the Hubble Guide Star and Smithsonian Astrophysical Observatory catalogs to search for stars having particular properties in proximity to a desired object and to construct a

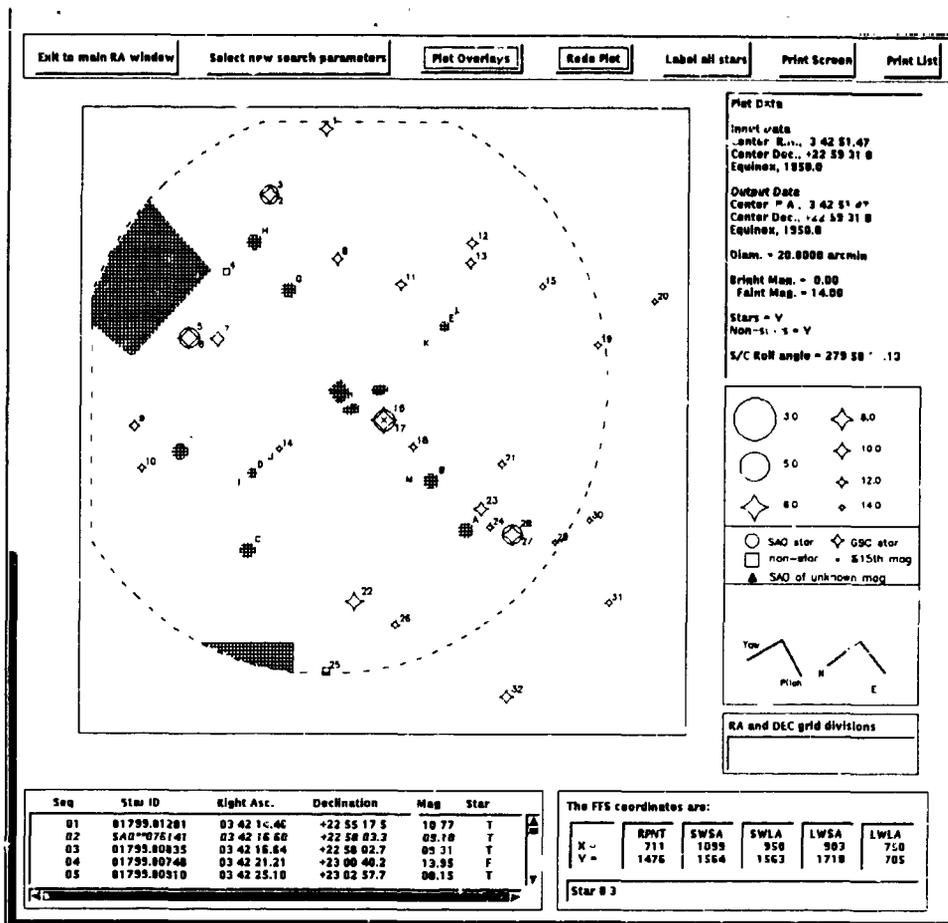


Figure 6: A sample IUE Guide Star Map Plot for the FES field of view. Both HST Guide Star Catalog (various sized diamonds) and SAO Catalog (circles) star designations are plotted. Various related information can be displayed including star identifications, magnitudes, coordinates, and the positions of the stars in the FES field of view with the target star at various standard locations (e.g an aperture).

detailed simulation of the star tracker field of view. The "front-end" of this software has been customized for the IUE environment (a sample is shown in Figure 6). Users may determine the positions of stars in this field and use the information to select guide stars for a long observation, prevent target misidentification, or choose stars from which to perform fine offset slews to a target. Other code provides menus of all relevant coordinate transformations which may be required during an acquisition. This software and planning code are essential to maintaining IUE's well known real-time operational flexi-

bility, scientific efficiency, and productivity.

A final but useful set of software permits the user to interactively enter, append, and edit data into a disk file and subsequently mail the file to relevant Observatory tasks and to the ESA station. This capability speeds dissemination of information to Observatory personnel and enhances the efficiency of data processing.

REMOTE OBSERVING WITH IUE

With the availability of modern workstations,

the development of the Internet, and increasing budget constraints, the IUE project decided to replace the original custom IUE Remote Observing Station equipment with a new flexible, but low cost system suitable for a wide variety of sites. This replacement was carried out concurrently with the development of the TOCS.

Design and implementation of the IUE Remote Observing package was driven by five major considerations:

- **Low cost.** The original custom IUE remote observing system, developed in the early 1980's by Prof. Donald York at the University of Chicago required custom dedicated equipment costing \$15,000 - \$20,000 per site. With the development of relatively low cost workstations and graphical software over the last few years, the design goal became the replacement of a custom system with off-the-shelf hardware which would likely already be available at a number of research and teaching institutions.
- **Ease of use.** With the introduction of X-Windows several years ago, it became possible to design a graphical interface for the user. Clearly marked buttons and a graphics interface could replace command-line oriented approaches. Thus a new user could very quickly become proficient in use of the software package. A secondary goal was to have the interface resemble, as closely as possible, the same interface being developed for the TO at the TOCS. Thus a GO being familiar with the interface used at the Observatory would have little trouble with a graphical interface of the same general design. The Remote GO Display screen is identical to the TO's Display shown in Figure 1 except for the

absence of command related buttons and functions.

- **Easily maintainable software.** One of the problems with the original EDS equipment was that it contained thousands of lines of assembler code running on a PDP 11/35 computer. It was extremely difficult to make more than very minor changes to the code and major enhancements to the system were not practical. The new system used the commercially available Interactive Data Language and C. This allows relatively easy expansion and software enhancements. In addition, the workload can normally be shared between both the prime and backup stations allowing for more inexpensive equipment.
- **System Security.** With the rapid expansion of Internet and the development of computer hacking, system security is of prime importance to computers used for real-time spacecraft commanding. The solution was a hardware controlled one-way communications bridge. The workstations can download images and files to the VAX 4000 support station for transfer to a remote observing site, but nothing can be initiated from the support station or any other computer to connect with the TOCS workstations.
- **Ready availability to the remote GO of information critical to on-going real-time IUE observing.** The original system required over 5 minutes to transmit an 8-level black and white image to the remote observing site. The upgraded system in current use now transmits a copy of the actual image displayed on the TOCS, in full color and resolution. With FES images, the remote observer can use a mouse to determine the exact location of the target

or offset star in FES coordinates. The TO in turn can enter these coordinates so that the cursor appears at the exact specified location. Thus there is no ambiguity between TO and GO on the desired object's location in the FES field-of-view. The GO can also conduct quick-look analysis on the spectral images during the shift, independent of the TO. Thus neither the GO nor TO is slowed down by spacial separation at different sites, and maximum observing efficiency is maintained.

The number of IUE remote observing sites has steadily grown since its introduction in the Spring of 1992 and now numbers over 20 sites. Temporary sites are also possible. Remote observing sessions were conducted from the American Astronomical Society meeting in Washington earlier this year.

ENHANCEMENTS IN PROGRESS

While the original system design is complete and is fully functional, the Observatory is interested in eliminating the transfer of images from the Sigma computer to the IUESIPS processing computer using 9-track computer tapes. Direct image downloading from the TOCS station to the IUESIPS computer would bring both greater efficiency and require less time for image processing. Work is underway, on a time-available basis, to develop this enhancement, determine its practical feasibility, and implement it if possible. If on-going tests prove successful, this enhancement should be on-line by the end of 1994.

SUMMARY

The present IUE Science TOCS system was installed in 1992 as a replacement of the antiquated original Experiment Display System,

which had been used since launch. The new system is low cost, reliable, and uses off-the-shelf hardware and commercial software (*i.e.* the Interactive Data Language and C) as a basis for a flexible, easily maintainable, and expandable Science Telescope Operations commanding system. By periodic updating of such a system rather than relying on a custom static system installed at launch, it is possible to maintain a large number of relatively inexpensive remote observing sites around the country or around the world. If used in small future real-time missions, this approach should allow widescale easy access to real-time observing with a modest size operations budget. This allows a greater share of available monies to be used for data analysis and interpretation by GOs. It also allows them to perform the observing from their own institution, greatly decreasing the interruptions in their schedules while providing the Observatory with schedule flexibility for inserting Targets of Opportunity and other unforeseen interruptions associated with spacecraft and the real-time observing mode. A complete technical description of this software (Pitts, 1994) is nearing completion and should be available by the end of the year.

REFERENCES

Pitts, R. (1994). *IUE Science Operations Control Center (SOCC) Telescope Operations Control Station (TOCS) Software Maintenance Guide*, (NASA DOC # 510 - 3 SMD/0192) NASA Goddard Spaceflight Center, Greenbelt, MD (in preparation).

Research Systems Inc., 777 29th Street, Suite 302, Boulder Colorado, 80303.

EFFICIENT MISSION CONTROL FOR THE 48-SATELLITE GLOBALSTAR CONSTELLATION

by

Dan Smith
Loral AeroSys
7375 Executive Place
Seabrook, Maryland 20706

ABSTRACT

The Globalstar system is being developed by Globalstar, Limited Partnership and will utilize 48 satellites in low earth orbit (See Figure 1) to create a world-wide mobile communications system consistent with Vice President Gore's vision of a Global Information Infrastructure. As a large long-term commercial system developed by a newly formed organization, Globalstar provides an excellent opportunity to explore innovative solutions for highly efficient satellite command and control. Design and operational concepts being developed are unencumbered by existing physical and organizational infrastructures. This program really is "starting with a clean sheet of paper."

Globalstar operations challenges can appear enormous. Clearly, assigning even a single person around the clock to monitor and control each satellite is excessive for Globalstar (it would require a staff of 200!). Even with only a single contact per orbit per satellite, data acquisitions will start or stop every 45 seconds! Although essentially identical, over time the satellites will develop their own "personalities" and will require different data calibrations and levels of support.

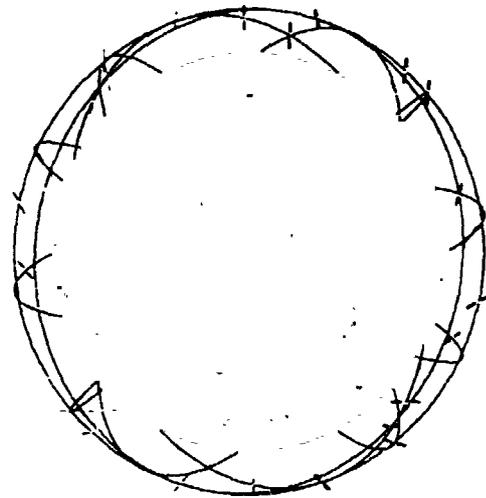


Figure 1: Globalstar Constellation

This paper discusses the Globalstar system and challenges and presents engineering concepts, system design decisions, and operations concepts which address the combined needs and concerns of satellite, ground system, and operations teams. Lessons from past missions have been applied, organizational barriers broken, partnerships formed across the mission segments, and new operations concepts developed for satellite constellation management. Control center requirements were then developed from the operations concepts.

This paper concludes by summarizing the applicability of these engineering processes and concepts to future missions of different magnitudes.

BACKGROUND

The growth in demand for telecommunications services over the last 20 years has been phenomenal. The projected growth over the next 20 years is expected to be even more dramatic.

Cellular phone systems are spreading across many parts of the world and, in some areas, are the primary mode of phone communications. Traditional cellular systems, however, may not be cost-effective in areas of low population density, very rugged terrain, or limited infrastructure.

Satellite-based systems now developing will bring affordable cellular-type voice and data communications to all regions of the world. A constellation of satellites can cover the globe with a network of moving cell sites. Ground-based systems coordinate handoffs between these moving cells in a manner similar to how handoffs are now coordinated when a vehicle moves between ground-based cell sites.

By using satellites, coverage is provided over most of the earth's surface. By using low earth orbit (LEO) satellites instead of geosynchronous satellites, time delays for the transmission to/from the satellites become imperceptible and power requirements are reduced to the point that hand-held phones can be developed.

The Globalstar system will utilize 48 satellites organized in eight planes of six satellites each. Eight additional satellites will be placed in phasing orbits as spares.

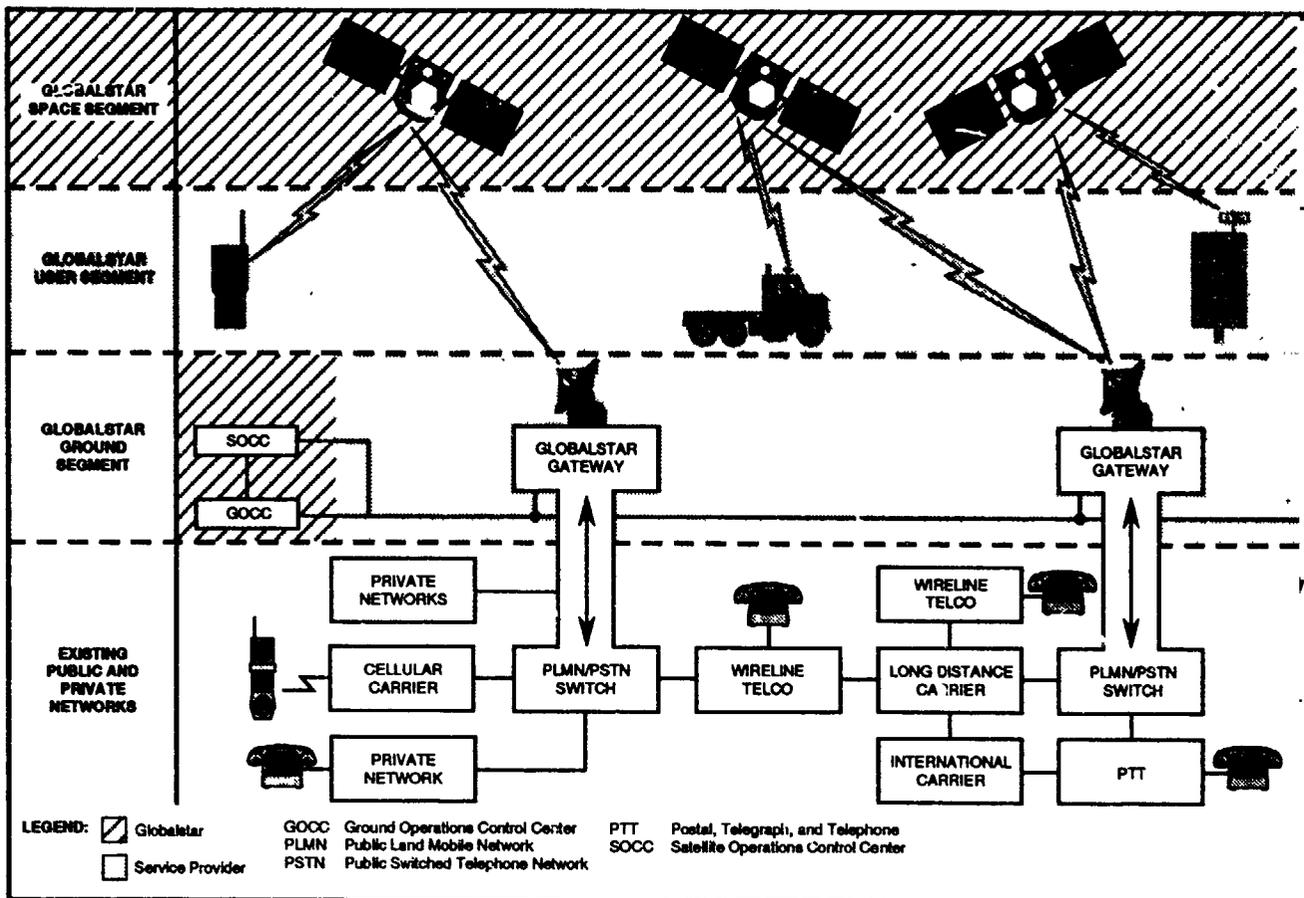
The satellites will each be at an altitude of approximately 1400 kilometers in circular orbits inclined at 52 degrees. This orbit selection concentrates coverage in the middle, most populated latitudes, thereby increasing the level of overlapping coverage in order to expand system capacity in those regions and to strengthen system robustness. Should a satellite be lost, there is still total coverage over most regions.

Voice and low-rate data traffic will be routed from hand-held phones through one or more passing satellites and then to ground-based gateways. The gateways will switch the calls into existing public and private phone system networks (Figure 2). With this approach, the Globalstar system takes maximum advantage of existing switching systems, networks, customer bases, and billing systems. In addition to voice communications, the Globalstar system will provide position determination, paging, and messaging services.

The design and development of the Globalstar system is well underway, with satellite launches to begin in 1997 and the full constellation to be in place by the end of 1998. Loral AeroSys is under contract to develop the satellite operations control centers (SOCCs) and to provide operations support. Other contracts are in place for space segment development, for gateways and their control systems, and for the Globalstar phone units.

COOPERATIVE ENGINEERING DEVELOPMENT APPROACH

Efficient control of the Globalstar satellite constellation requires innovations beginning with the system definition and extending through to operations concepts and system design.



LORAL TPR002-002m

Figure 2: Globalstar System Configuration

In many traditional systems, a satellite is designed and specs are then given to the ground system developer who, under a different contract, builds a control center and provides it to the operations contractor to determine how to use it for a satellite they had only limited prior knowledge of. Globalstar, however, is being developed as a joint effort between system level, satellite, ground, and operations teams. A "trusted contractor" approach is employed where each participant is looked to as the expert in a particular field. In evolving this approach, Loral altered some of the traditional customer-contractor process flows while maintaining the necessary levels of progress oversight. Through working groups, relationships established between contractors, and other formal and informal concepts developed by Loral and the

Globalstar, LP contractors, a total system concept has evolved and a WIN-WIN mentality has developed.

One example of the joint engineering approach is demonstrated in how the telemetry formats were designed. The Loral ground segment development personnel applied "lessons learned" with over 30 satellite systems to develop a list of telemetry stream characteristics of past systems which increased system complexity or were found to be of little use. In some cases the satellite manufacturer thought they had been "adding a feature" and in others the ground complexity was matched by a spacecraft complexity and both could be eliminated. The operations team is involved in specifying the resolution needed for specific on-board parameters and in defining

the sample rate for the parameters. In some cases, the operations team has influenced the quantity, location, and types of on-board sensors. Spreadsheets and a common data base agreed to by all contractors allow for convenient information exchange. On-board data reduction replaces engineering tape recorders and allows graphs of a full orbit's data to be generated within seconds of the start of a new contact. All-in-all, the system cost and complexities have been reduced, more capabilities have been designed into the system, and the operations team will be provided access to telemetry data which best meets their defined needs.

A series of similar engineering efforts have been performed in the areas of ground system antenna site design, command formats, onboard autonomy, orbit determination approaches, and in a series of very specific spacecraft configuration areas. Evolving design decisions are incorporated into the operations concepts only after possible impacts to other areas are addressed.

In all cases, the concepts for operations are determined as a joint effort between the ground system developers and the operations personnel. In many cases, the satellite team is consulted to validate assumptions and to critique ideas. The end-goal of the effort is to meet all system objectives while limiting both the development and the lifecycle operational costs. With Globalstar, these costs play a critical role in determining the overall profitability of the enterprise.

OPERATIONS STRATEGIES

A set of strategies and plans for providing efficient management of the constellation have evolved along with the detailed operations concepts. Detailed SOCC

requirements were developed from these ideas. Collectively, the strategies characterize the uniqueness of the Globalstar's efficient mission operations approach:

1. Process only the data needed. Through satellite autonomy and innovative data handling techniques, the nominal anticipated real-time monitoring and control period per satellite has been reduced to once per orbit, or about 10 minutes out of every 114 minutes. Data for other viewable portions of the orbit is stored at the remote ground stations and only processed if a need arises, much like a flight recorder on an airliner. If no problems are encountered, the remote site data is deleted after several days without ever being transferred to the SOCC.

2. Concentrate on the satellites with problems. Automated software monitoring of the satellite subsystems will allow some satellite contacts to occur without any human monitoring. All data streams received will be monitored by the software and only selected contacts will be monitored by the operations staff. Monitoring will take place at the parameter, satellite subsystem, and full satellite evaluation levels. The operations team will be able to define the evaluation criteria and to regularly update the checks performed to reflect differences between satellites and the increase understanding of the satellites' performance characteristics. This level of automation will be used to reduce the burden on the operations personnel for monitoring of healthy satellites and will allow additional time for working with satellites requiring special attention. The automated capability will be controlled so that every satellite is still observed at a minimal rate. The actual observation level for monitoring the constellation can be throttled based on factors such as problem histories, learning curves, constellation size,

and even operation shifts.

3. *Make the best of the "few minutes per orbit" contact.* The objective is to utilize the limited real-time contact data to the fullest extent possible and to minimize the off-line analysis efforts required for other time periods. To orient the operator regarding the next pass, a contact log report will be generated indicating the times of the contact, the planned commanding activity summary, and any outstanding issues to be closely monitored. This report is updated with actual data throughout the pass and goes to the master pass log at the end of the pass. Automated procedures will allow planned operational steps to be executed without intervention. The use of on-board telemetry reduction allows for critical parameters to be collected at commandable intervals and downlinked as a data set during the pass. This information, which may cover an extended time period, is immediately viewable as plots and reports on the user's screen. With these plots, the user can rapidly assess the performance of parameters of interest over the entire previous orbit or, for example, during the critical seconds of a thruster firing. If necessary, immediate remedial action can be initiated should the stored data confirm a suspected anomaly.

4. *Take advantage of the large number of satellites.* Management of 56 satellites should not require 56 times the effort of managing a single satellite. There are several areas in which the large number of satellites is actually an advantage. A new method of looking for possible problems is to plot the data from many satellites on top of each other (aligned for equator crossing, time of day, land mass location, etc.) and to look for outliers. In effect, there are 55 control satellites for each satellite being evaluated. Additionally, theories regarding environmental factors can quickly be tested

by looking for common reactions across multiple satellites. Having many satellites will facilitate the establishment of an anomaly resolution data base which can be searched by satellite or component. Procedures developed through lengthy analysis can often be applied to other satellites which experience similar problems at a later time and some common problems can be corrected on the ground prior to future launches.

5. *Monitor more than one satellite at a time.* Operations personnel will be able to monitor up to 6 satellites per workstation. With the concepts of multiple satellite monitoring, efficient display of information is crucial. A number of innovative displays have been developed to support constellation management. Map-based displays annotated with satellite status information create a high level system display, with satellite icon selection available to go to detailed information levels. Additional table-based displays support the monitoring of small groups of satellites and detailed text and graphical displays are used at the individual satellite level. Users will be able to tailor their screen definitions to best match their responsibilities and work approach.

6. *Automate the system configuration.* One contact per orbit equates to about 13 contacts per day per satellite, and about 700 contacts per day for the entire constellation. The total system is data driven and automatically reconfigured. Remote sites process all data received and log it to local disks or send it to the SOCC as directed in established setup tables.

Within the SOCC, the allocation of satellites to user workpositions and the configuration of the system to support the data streams will be automated. Operators will use generic terms to specify what satellites they

wish to monitor. One operator may want to watch all satellites for which commanding is planned, while another may only want to watch satellites for which critical parameters are found to be out of limits. A wide variety of selection criteria have been identified which, together, can support a wide range of operations concepts.

7. *Manage for the mission lifetime.* The true goal of the operations team is to maximize the amount of time during which each satellite carries revenue bearing communications traffic. Many steps are in place to make the day-to-day operations efficient. The operations team will also work to extend the mission life of the system and the individual satellites. An on-line performance data base is maintained for the life of each satellite, beginning with assembly line testing results and calibrations. The anomaly history data base will allow problems to be tracked against time. Operational workarounds may be found to extend component life on many satellites based on information gathered from a few. For Globalstar, data exchanges between the satellite operations centers and the center which manages the phone traffic level and quality will allow for the development of joint operations procedures to maximize revenues and extend mission life.

CONTROL AND MONITORING OF 56 SATELLITES

Including on-orbit spares, the Globalstar operations team will be controlling 56 satellites from a single control center. As shown in Figure 3, the actual number of satellites normally viewed at a single time is considerably less. The level of satellite autonomy reduces the amount of time each satellite must be observed. On-board data storage allows for collection of critical

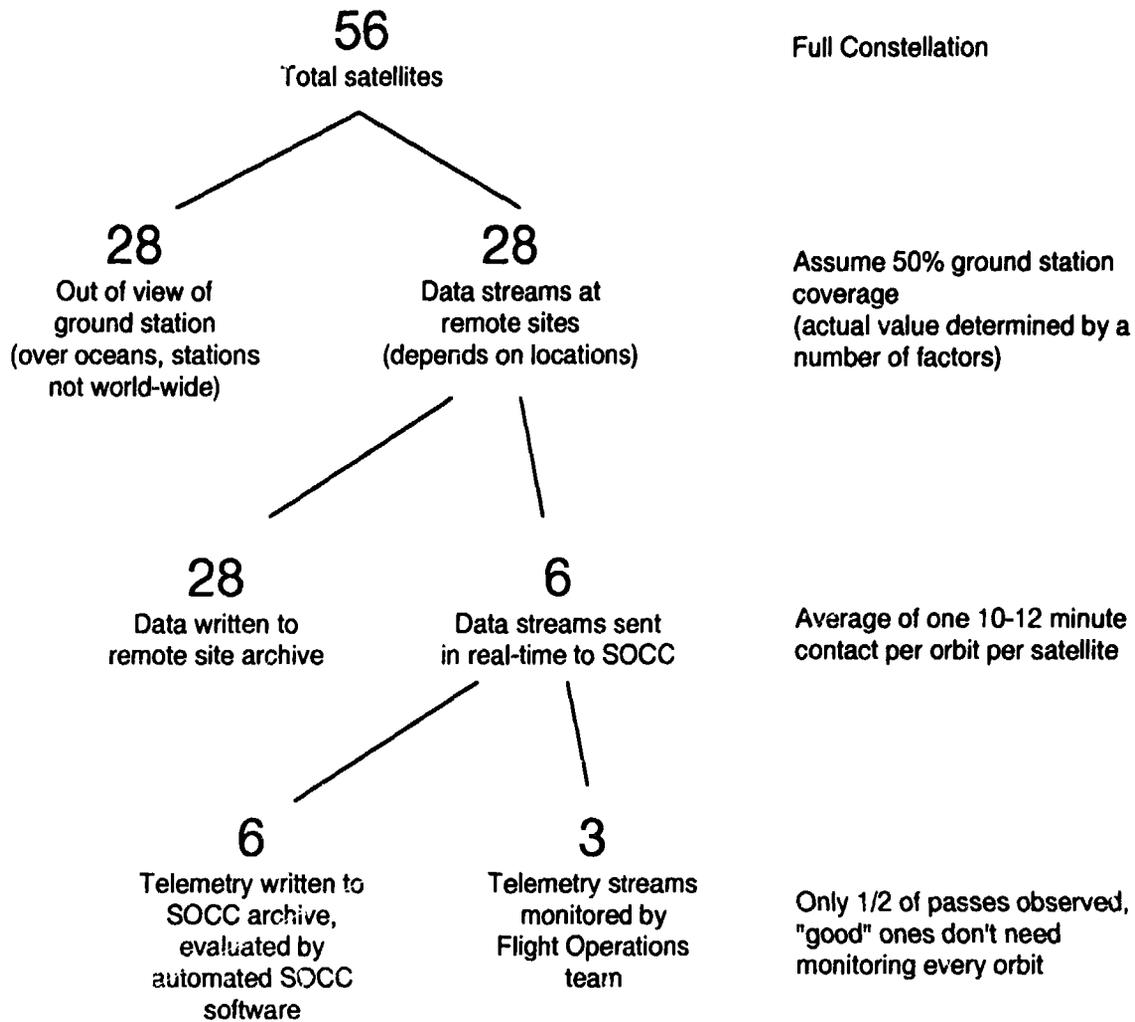
performance data over the entire orbit. A distributed flight recorder concept, implemented across the network of ground stations, provides a data resource should problems be identified. Automation within the control center reduces the burden on the flight operations team and allows some contacts to be monitored only by the software. Collectively, these strategies allow a very small operations team to efficiently control and monitor the entire constellation.

As many as 12 satellites will be dispensed from a single launch vehicle. The system is sized to accommodate the high level of monitoring of each satellite during launch in addition to the routine operations which must continue. Anomaly investigation and resolution, orbit maneuvers, and infrequent large software and data loads to the satellites also require additional support. Operations personnel have been involved in sizing these efforts, determining the number of workpositions required, and supporting the facility design to best accommodate the variations expected in support requirements.

CONCLUSIONS AND APPLICABILITY TO OTHER MISSIONS

Efficient mission control is not just an operations issue - it must be designed into the satellite and ground system from the beginning.

Cooperative processes between contractors during the early concurrent engineering phase of system-level design can provide significant payoffs in terms of system capability and implementation and operations costs. The processes developed by the Globalstar team, involving multiple contractors around the world, have proven extremely successful.



LORAL TPR002-001m

Figure 3: Globalstar "Point in Time" Operations

Process, technical, and "lessons learned" exchanges between government agencies and the private sector benefits all. In the case of the Globalstar mission, NASA lessons learned have been studied and mature operations concepts have been adapted and combined with new ideas to create the innovative approaches necessary to efficiently manage a very large satellite constellation.

Problem solving approaches and solutions will obviously vary depending on the application. The specific strategies developed for Globalstar help the overall system work effectively, and may be applicable to other systems. What is clearly applicable from the

Globalstar effort is the understanding that new organizational and engineering approaches can lead to tremendous benefits.

The processes of trusted contractors, cooperative concurrent engineering across development segments, and a cross-contractor team approach, applied by a set of organizations with a broad base of disciplined engineering skills will lead to systems which are better engineered to meet the combined objectives of the mission and the individual goals of the supporting teams.

REFERENCES

Gore, A. (March 21, 1994). *The Global Information Infrastructure - Remarks prepared for delivery by Vice President Al Gore*. International Telecommunications Union.

Hatfield, D. (1993). *A Discussion of the Globalstar Satellite-Based Mobile Telecommunications System*. Hatfield Associates, Inc. Boulder, Colorado.

MINI ALL-PURPOSE SATELLITE CONTROL CENTER (MASCC)

Gérard ZAOUCHE

Centre National d'Etudes Spatiales (CNES),
31055 Toulouse CEDEX, FRANCE

ABSTRACT

A new generation of Mini All-purpose Satellite Control Centers (MASCC) has been developed by CNES (F). They turn out to be easily adaptable to different kinds of satellites, both Low Earth Orbital or Geostationary.

The features of this MASCC allow both standard satellite control activities, and checking of passengers experiments hosted on a space platform.

In the different environments in which it may be used, MASCC provides standard broadcasting of telemetry parameters on animated synoptics (curves, bargraphs, alphanumeric displays, ...), which turns out to be a very useful and ergonomic medium for operational teams or satellite specialists.

Special care has been taken during the MASCC development about two points :

- **automation of all routine tasks**, allowing automated operation, and limiting human commitment to system supervision and decision making,
- **software adaptability**.

To reach these two main objectives, the MASCC design provides :

- a simple, robust and flexible hardware architecture, based on powerful distributed workstations,
- a table-driven software architecture, easily adapted to various operational needs. Satellite characteristics are described in a

central Data Base. Hence, the processing of telemetry and commands is largely independent from the satellite itself.

In order to validate these capabilities, the MASCC has been customized to several types of satellites and orbital platforms :

- SPOT4 : French new generation of remote sensing satellite,
- TELECOM2 : French geostationary TV and telecommunication satellite,
- MIR : Russian orbital platform.

MASCC development has been completed by the third quarter of 1993.

This paper will provide first a description of the MASCC basic functions, of its hardware and software design. It will then detail the increased automation capability, along with the easy adaptation of the MASCC to new satellites with minimal software modifications.

Key words : MASCC, Satellite Control Center, workstations, adaptation, flexibility, automation.

INTRODUCTION

The Satellite Control Centers are a component of the "satellite(s) ground segment" unit.

Their role is to provide for technical monitoring and control of the satellite and its passengers (through housekeeping telemetry reception, location data and telecommands transmission) and for platform, passenger or payload

management based on planning requests from User Mission Centers.

In addition to the usual criteria for performance, reliability and robustness required from a unit on which the satellite is entirely dependent, the system features the following criteria :

- low development, operating and maintenance costs ;
- very short spaces of time to availability and easy implementation ;
- ease of adaptation to developments concerning either the satellite or the mode of operation used ;
- ergonomic and high quality presentation of data for specialists using sophisticated analysis tools.

The Control Center (MASCC) has been specially developed to meet requirements for rapid adaptation and low operating costs.

The MASCC product was developed directly from work carried out to set up the SPOT 4 Control Center (the French sun-synchronous Earth Observation satellite). This means that it has been able to benefit from development work financed for the SPOT 4 project (high processing capacity, automated operation, ergonomic media for information displays, etc.). The MASCC is equipped with an enhanced range of options and standard features in order to meet requirements for adaptability to different satellites with greater ease.

The MASCC features all the usual Control Center functions and is well adapted for use with non-geostationary satellites such as observation satellites or mini satellites, etc.

It may also be adapted for use with geostationary satellites (with occultation of tasks related to movement and limited satellite visibility).

MASCC FUNCTIONS

Preparation functions

Other than satellite related activities, the MASCC provides for the preparation and generation of the following :

- TeleCommand sets with numerous transmission attributes (burst transmission, operator acknowledgement, time tagging to earliest and latest point, etc.); Command sets stored in an internal library either for real time transmission or for insertion in a "TeleCommand Plan" ;
- synoptic data visualisation with real time display of the various parameters to be viewed in a user-friendly form (curves, bar graphs, active symbols or text, digital display systems, etc.). These synoptic displays are stored as autonomous files.
- parameters (either to be computed in real time from telemetry parameters or previously computed) derived from computation law description through a readily accessible interpreter language.

Preparatory functions to real time activities

The MASCC provides for the preparation of a "TeleCommand Plan" (PDC) to be transmitted to the satellite.

This PDC is generated by inclusion of the TeleCommands stored in an internal library and of TeleCommands from dedicated subsystems (payload management, orbitography, etc.) within the MASCC computer or distant computers.

The TeleCommands are planned within the PDC in compliance with :

- a forward visibility chronogramme of stations in the case of non-geostationary satellites ;
- operational constraints related to the satellite (masked antenna, etc.) ;
- management constraints related to satellite use (transmission of particular Commands at particular points, class of Command to be used in preference to other classes, etc.).

This "PDC" preparation may be carried out in manual mode under operator control or

in automatic mode working from sequencing algorithms.

Real time functions

The MASCC provides all the standard real time functions of a Control Center.

The following list gives the main real time functions provided for :

- interfacing with satellite monitoring stations with automatic testing and station selection (capacity for simultaneous dialogue with two stations) ;
- acquisition and storage of CNES and NASA station location data ;
- acquisition, processing and display of CNES station remote control data ;
- acquisition, processing, display and storage of telemetry data :

Acquisition may be carried out simultaneously with two stations.

Processing involves decommutation, implementation of the transfer function, implementation of conditions of significance, application of monitoring functions, processing of the variant part, computation of predicted parameters, computation of prepared parameters via interpreter programme.

- parametered transmission of TeleCommands through transmission of prepared PDC or library stored Commands. This may occur :

. either in manual mode, Command by Command as the operator implements decisions, with wait time for acknowledgement of each Command ;

. or in sequenced automatic mode with wait time for acknowledgement of the previous Command ;

. or in automatic mode but in burst transmission at the maximum rate of the uplink connection, with no wait time.

- display of :

. synoptics on which all parameters may be shown (telemetry, remote control, computed parameters, etc.) ;

. alerts (parameter values, operational subsystems, etc.) ;

. parameters as required with capabilities for display and modification of monitoring thresholds, transfer functions, etc. ;

. TeleCommand data (Commands under implementation, causes of wait or anomaly, etc.) ;

. log of all main events occurring during operation.

- distribution of all data received via Internet network to an other MASCC (or several MASCC) dedicated to TX display of a large number of synoptics for monitoring phases such as launch or difficult manoeuvres, etc. ;
- saving of data generated at particular points on a redundant computer ;

Non real time management functions

MASCC features a set of analysis and operational management functions for :

- TeleCommand management after transmission to the satellite by means of a report on each Command transmitted, recycling of non transmitted Commands and generation of a log of Commands describing events involving transmissions to the satellite ;

- Command library management (display, elimination, etc.) ;

- re-run of acquired telemetry with parameterisation of dates, length of time and speed (required rate of transmission : normal, accelerated or step by step) ;

- verification of time-tagging consistency between on-board time and station time-tagging, with re-tagging of acquired telemetry if necessary ;

- storage and compression of acquired telemetry data in a Technical Data Base ;

- extraction and display (curves) of telemetry data over different periods ;

- transmission of acquired data to networks if required (telemetry data, location data, log, etc.).

MASCC DESIGN

Hardware design

Hardware architecture is designed to provide for :

- the type of performance required by Control Center missions ;
- a high level of reliability and availability ;
- ready implementation of further developments ;
- failure management through modular design.

An Ethernet connected work station and X terminals have been opted for.

A satellite is managed at any given moment by one workstation even though several satellite programmes may be operating within one workstation. This option means that the system can be readily modified to cope with developments in satellite families and that redundancies can be managed with ease in high-availability systems.

The workstations selected are :

- RISC architecture HP 9000 series 700 (712, 715 or 735) ;
- SYSTEM V compatible UNIX operating system (HP / UX)

Preference has been given to standard equipment :

- X terminals
- graphics interface : X11, OSF-MOTIF, ILOG library, PV-WAVE ;
- coding : C and C ++ ;
- internal data exchange on TCP/IP, NFS
- External data exchange on X25, FTP.

Hardware options and MASCC function capabilities may lead to different architecture configurations depending on the type of requirement. The diagrams below show different implementation possibilities, depending on mission requirements (Figure-1).

Adaptation to special or temporary needs (improved data distribution, arrival of a new satellite within a family, connection to a dedicated system, etc.) may be carried out without disrupting the existing architecture through the addition of workstations or TX, and if necessary through re-dimensioning of the network.

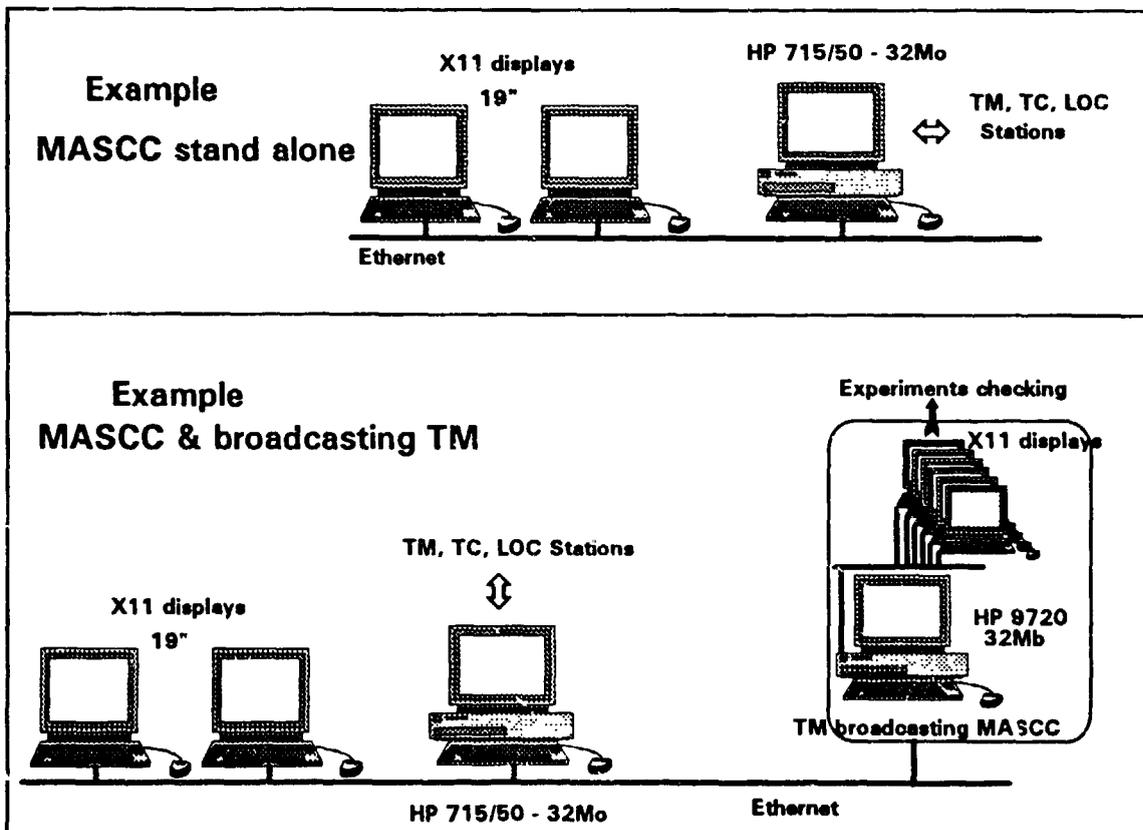
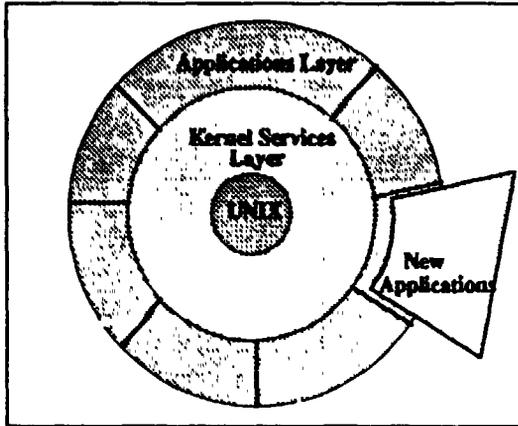


Figure-1: MASCC implementation examples

Software design

Software architecture is made up of two layers above the system layer (HP-UX, OSF-MOTIF) :

- kernel services layer
- applications layer



Kernel Services Layer

This layer forms the basis for the applications layer and enables the applications software and the UNIX system to operate autonomously.

In the case of transfers to new system versions, test operations and maximum cover of test cases will involve this layer (while the applications layer will in general be only slightly modified).

The kernel services layer is made up of :

- Libraries providing for standardisation and access to common services :
 - . file management (ASCII for easy editing) ;
 - . message management (transfer of messages into files, windows or printers);
 - . interprocess communications management (message files, shared memory, semaphore) ;
 - . inter-machine communications management
 - . time-tag management (time shift capability)
- utilities allowing for simple MASCC management :
 - . log use (sort, fusion, etc.) ;
 - . hardware and software configuration control ;

. Digital Data Storage (DDS) ;

. time synchronisation on different computers ;

- an Agenda which forms the basis of all MASCC automation :

. simple and ergonomic work programme editing

. automatic execution of applications defined in a work programme

. activation and de-activation of applications on a local or distant computer through the network ;

. re-run capability in cases of incorrect termination of an application.

Platier	Editer	Procedure	Journal de Bord	Semaine
CCS1				
CCS2				
BTB1				
DMGS				
Station ASIS				
Debut fenetre 07:12:00	Duree fenetre 2h	PGT n°	Repose : 00:00:00	Cocher Passage Ink Index

Figure-2: An Agenda work plan scheme

Applications Layer

Strict selection procedures and rules for the design of the MASCC applications layer have been laid down to ensure easy parameterisation and adaptation of the product to different satellites and different modes of operation.

Scissoring by "procedure"

Each operational function is carried out by a software package called a "procedure". These procedures are in overlay above the Kernel Services layer.

A procedure is an autonomous task designed to run without operator intervention and to be executed on one of the workstations. It may be activated automatically by the Agenda (Figure-2) or manually by the operator through an ergonomic man-machine interface.

One of the most important design concepts consists in excluding the automation

constraint from procedure development, as this generally makes software programmes more complex and disrupts the design process. Automation and flexibility in task activation are dealt with by the Agenda. Moreover, over-riding of overall automation may be activated at will or in the case of any anomaly, however slight. The other fundamental design concept ensues from the first and provides for the ready modification of "procedure" sequencing, depending on the requirements of the operating team, and if necessary for additional processing (in the form of a "procedure") without disrupting existing tasks and without the need for modifications to existing software for purposes of integration. Adaptation to operating requirements is dynamic in nature.

Building a work plan for one or more days involves defining the required procedures on the Agenda work schedule (Figure-2). Skeleton work plans are available for the routine operating mode.

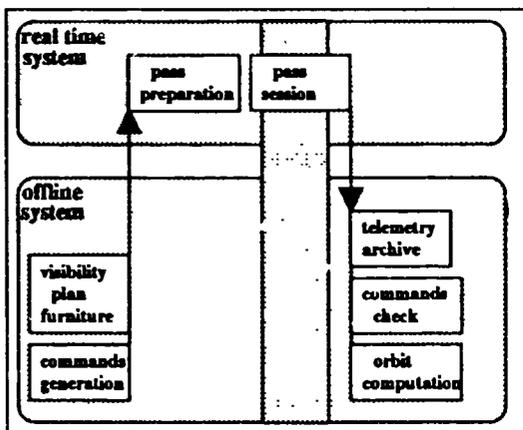


Figure-3: daily routine procedure sequence

Screen Ergonomics

The Man-Machine Interface is implemented on X11 / Motif standards. High resolution screens (19" X11 displays) are available for the implementation of all animated synoptic windows and alphanumeric windows. Ergonomic specifications (based on the OSF / MOTIF style guide) have been

drawn up to ensure that MASCC screens are of uniform design.

MASCC screens are readily adaptable to presentation requirements formulated by operating personnel. Screen features may be modified by means of resource file configuration and parameterisation tables.

Independence from the satellite

Processing carried out by the software is described in tables (System Data Base).

These tables are formatted on the basis of files describing all satellite data (system constraints, parameters, processing, computation laws, transfer functions, monitoring, etc.).

When the processing required to take on a new satellite can be described in these files, the software programmes will take the new satellite into account without the need for modification.

All software modifications observed in different MASCC versions have always been justified by specific satellite behaviour within a specific mode, and have always been kept to a minimum. As a general rule, nominal satellite modes are easily described in the files available.

These files may be supplied by the satellite designer or from data bases containing all the relevant satellite-related data.

Use of ASCII files (SR6-10 format)

Systematic use has been made of ASCII files in SR6-10 format, whether for external or internal interface files (Command, telemetry or satellite description files, etc.).

The kernel services library provides for highly flexible use of these files for the various applications.

The modifications made to files during the MASCC integration and validation phases within a ground segment have been greatly simplified thanks to the use of standard UNIX tools such as "grep", "awk", etc.

Any errors in comprehension at the interface between two units may be bypassed until corrections are made and do not block integration. Data sets and test cases are readily produced (e.g. random telemetry files, etc.), so that system reliability is enhanced by wide test coverage.

Independence between Real Time Processes

Independence between non real time tasks is assessed through the concept of "procedures" and the use of Agenda. The Agenda synchronises procedures which involve data exchange, generally through files.

On the other hand, the tasks running simultaneously in real time (telemetry processing, command transmission, window or synoptic displays, acquisition of network blocks, etc...) are activated in a unique procedure and exchange data through high rate message queues.

The MASCC has implemented a protocol providing for independence between the data-generating tasks and data-consuming tasks : the producer / consumer protocol.

A task providing data (Network interface, telemetry, ...) doesn't know tasks using it (synoptic display, Telecommand, ...).

For instance, a synoptic displays Telemetry parameters and the receiving station. These data are processed by two different functions : "Telemetry" and "Network Interface". The synoptic needs them for display. Hence, the "synoptic" function subscribes to these parameters through the producer/consumer protocol, without knowing who is producing them. Once

initialized, "Telemetry" and "Network Interface" functions know through the protocol tables, which parameters to provide, at which frequency and to which receivers.

This mechanism allows a large genericity between Real time tasks and supports easy modifications of managed data and addition of new consuming tasks, delocated or not.

ADAPTABILITY CAPABILITIES

In order to validate MASCC adaptability, a number of goals were set :

- adapt to a Low Orbit Satellite;
- adapt to a Geostationary Satellite;
- adapt to On Board Experiments monitoring;
- carry out the corresponding modifications in the shortest delay;
- display the results on an unique platform.

The easiest adaptation consisted in adjusting MASCC to the French observation satellite SPOT4, MASCC being directly issued of SPOT4 Control Center design.

The Geostationary Satellite used for this test was TELECOM2, French satellite for telecommunication and television broadcasting.

The adaptation, here, consisted in describing Telemetry in the System Data Base. Some particularities of this telemetry, such as DWELL data, required software modifications. Details of parts modified and corresponding efforts are provided in the following table.

Modification details table

Task	Technical description	Modification size	Effort
Telemetries differences analysis and choice of a solution			4 days
"System Data Base" adaptation	specific tool (awk)	entirely developped	4 days

Task	Technical description	Modification size	Effort
Adaptation for the archiving telemetry format	specific tool (run task)	entirely developed	4 days
Adaptation of TC2 synoptics	specific tool (awk)	entirely developed	4 days
TM treatments modification	taking into account : - new format (21 frames) - dwell characteristics - bits inversion	7 sets modified (1000 lines) 4 interface files (80 lines)	6 days
Man Machine Interface telemetry re-run modification	new look for the MMI		4 days
Telemetry re-run control	taking into account : - new format (21 frames) - new frequency (1,2s)	2 sets modified (20 lines)	2 days
General information display	dwell informations display new format display	2 sets modified and one data file (20 lines)	1 day
Synoptics files creation and modification	checking TM modification		3 days
Integration, validation			5 days
Installation, acceptance			4 days

Modifications have been performed by two skilled MASCC designers.

The last adaptation dealt with the Telemetry provided by the Russian platform MIR, for an off-line analysis of parameters (On board Experiments, vehicles rendezvous data, ...). Telemetry is available on a diskette for tests in France or on an Ethernet network within the TSOUP facilities. Both description of this Telemetry in the System Data Base and efforts in reordering events were required. This adaptation, which has been performed during two months, is now running in TSOUP facilities.

All these adaptations were performed without modifying the central structure of MASCC. Man Machine Interfaces were configured to manage display requirements and available supports.

CONCLUSIONS

MASCC development was completed in late 1993.

The system is now operating in the main CNES control rooms when launches are carried out, to distribute telemetry data to the "spacecraft" specialists.

Today, the MASCC is being considered as a replacement for the SPOT 2 and SPOT 3 Control Centers now operating, involving minimal cost and with up-to-date hardware ensuring low cost maintenance.

The MASCC is included within the Control Center selection list being reviewed by CNES for its mini-satellite programme.

MASCC enhancements are planned to coincide with the implementation of the new SPOT 5 generation of Earth Observation satellites and others.

REFERENCES

1. Fratter, I. 1992. Agenda: a task organizer and scheduler. In SpaceOps 92.

Author Index

A

Abedini, Annadiana - OP.2.a
Aguilera, Christine - SD.2.f
Alcott, G. - SE.6.c
Alexander, Scott - SE.1.a
Altunin, Valery I. - OP.5.a
Ames, Charles - SD.2.a
Arquilla, Richard - OP.3.f
Aslam, Tanweer - SD.4.a
Auernheimer, Brent - SD.2.a
Ayache, S. - OP.4.a, OP.4.c

B

Baize, Lionel - OP.3.a
Baker, D.F. - *OP.5.d
Baker, Paul L. - SD.5.a
Baldi, Andrea - OP.4.b, SE.3.a
Barro, E. - *SD.2.b
Beach, Edward - SE.2.a
Beckman, R. M. - OP.1.c
Bell, Holland T. - SE.1.c
Benjamin, Ted - SD.1.d
Bennett, Toby - DM.2.d, DM.3.a,
DM.3.g
Berry, Thomas - MM.2.i
Beser, Eric - SD.5.b
Bianco, Giuseppe - *OP.1.e
Biesiadecki, Jeff - SE.1.a
Boreham, Charles Thomas -
OP.1.a
Bote, Robert - MM.2.i
Boyer, Jeffrey S. - SD.2.c

Braun, Armin - *OP.1.f
Brenot, Jean-Marc - OP.1.b,
OP.4.c

Brittinger, Peter - MM.2.a
Bromberg, Daniel E. - DM.2.h
Brooks, Jr., Robert N. - OP.2.b
Buckley, Brian - OP.2.c
Butler, Madeline J. - SE.6.a
Butler, Scott A. - *SD.3.g
Buxbaum, K. L. - MM.1.b
Byrne, Russell H. - DM.2.h

C

Calanche, Bruno J. - DM.1.a
Calvin, Richard - MM.2.i
Cameron, G. E. - OP.6.a
Cardenas, Jeffery - DM.2.i
Carper, Richard D. - SE.4.d
Carraway, John - MM.1.a
Carter, Leslie E. - *SD.2.d
Cayrac, D. - OP.4.a, OP.4.c
Challa, M.S. - *OP.5.d
Chapman, K. B. - OP.1.c
Cheuvront, Allan R. - MM.2.b
Choudhary, Abdur Rahim -
SD.3.a
Christ, Uwe - DM.1.b
Clotworthy, Tim - DM.3.i
Conaway, B. - DM.1.g
Corrigan, Jim - OP.5.b
Cox, C. M. - OP.1.c
Cox, Nagin - SE.1.a
Crysel, William B. - MM.2.1

Cuevas, O. O. - OP.1.c
Cureton-Snead, Izeller E. - SE.1.c

D

Davenport, William - SD.4.f
Davis, Don - DM.3.a
Davis, Randy - MM.3.e
De Saint Vincent, A. - MM.3.a
Debatin, K. - SE.4.a
Del Bufalo, A. - *SD.2.b
Demelenne, B. - *OP.3.b
Desai, Vishal - DM.1.c
Deutschmann, J. - *OP.5.d
Dias, William C. - MM.2.n
Diekmann, Frank J. - DM.2.a
Douard, Stéphane - OP.1.d
Dowling, Jason - DM.3.g
Drake, Brian C. - SE.1.b
Duff, Gary - SE.5.e
Duncan, Elaine F. - MM.2.1
Dunford, E. - OP.6.b
Durand, Jean-Claude - DM.2.j

E

Easton, C. R. - SE.3.b
El-Boushi, Mekki - SD.2.e
El-Ghazawi, Tarek A. - DM.3.b
Elgaard, Dennis - OP.4.b

F

Fatig, Michael - SD.3.b, *SD.3.c
Ferri, Paolo - DM.3.j, OP.3.c
Fishman, T. - MM.1.d
Flora-Adams, Dana - SE.2.b

MM - Mission Management, OP - Operations, DM - Data Management, SE - Systems
Engineering, SD - Systems Development, * Presented in Poster Session

Author Index

Fogel, Alvin J. - DM.3.c
Folk, J. - DM.1.g
Fong, W. - DM.1.g
Forman, Michael L. - SD.1.a
Frogner, Bjorn - DM.2.m
Frye, Stuart - *DM.2.b

G

Gaasbeck, James Van - OP.2.c
García-Pérez, Raúl - MM.2.c
Garman, C. - SE.6.c
Gasquet, A. - MM.3.a
Gershman, R. - MM.1.b
Ghaffarian, Kam - SE.5.b
Ghuman, Parminder - DM.3.g
Giannini, F. - SD.3.e
Golden, Constance - SE.3.c
Good, Andrew - MM.1.g
Gordon, Julie - SD.5.d
Gotanda, T. - *SE.1.d
Grebowsky, Gerald - DM.3.i
Green, William B. - DM.2.c
Gregg, Watson W. - OP.2.i
Grieser, William H. - OP.4.i
Guerrero, Ana Maria - SE.2.b,
SD.2.f
Guffin, O. T. - MM.3.b

H

Hageman, Barbara H. -
*SD.3.h, *SD.5.c
Hagopian, Jeff - MM.1.c
Hall, Dana - SD.3.d

Happell, Nadine - MM.2.f
Harbaugh, Randy - SE.4.b
Harrell, Linda - *DM.1.d
Harris, Jonathan C. - DM.2.d
Harvey, Raymond J. - MM.2.d,
SE.6.d

Hayes, Tony - SE.5.e
Haziza, M. - OP.4.a, OP.4.c
Hazra, Tushar K. - SD.1.a
Head, N. C. - OP.3.d
Hei, Jr., Donald J. - SE.1.c
Hell, Wolfgang - DM.3.d
Heuser, Wm. Randy - SD.1.b
Hogie, Keith - SE.5.b
Holdaway, R. - OP.6.b
Hooke, Adrian J. - SE.3.d, SE.3.e
Hope, Sharon A. - SD.2.h
Hornstein, Rhoda Shaller -
SE.1.c
Hosack, Beryl - DM.2.e
Howard, P. - OP.3.d
Hübner, H. - OP.3.c
Hudson, Richard K. - *SE.6.b
Hughes, Peter M. - OP.4.d
Hull, Larry - MM.3.e
Hurd, William J. - SE.1.c
Hurley, Daniel - SE.2.b

I

Illmer, N. - OP.4.e
Incollingo, Marco - DM.1.b
Ishihara, Kiyoomi - *DM.1.e

J

Jain, A. - OP.4.e
Jeffries, Alan J. - SE.3.e, SE.4.d
Jiang, Jianping - *SD.2.d
Joe, Chester - SD.2.f
Joensson, Rolf - MM.2.e
Joli, Jean-Pierre - *MM.3.c
Jones, M. - OP.3.d, SE.4.c

K

Kan, Edwin P. - DM.3.e, OP.2.d
Karlín, Jay - SE.5.a
Kasperovich, Leonid - *DM.2.f
Kaufeler, Jean-Francois -
SE.3.f, SE.3.i
Kaufeier, P. - OP.4.f
Keehan, Lori - OP.2.e
Kelly, Angelita C. - SE.1.c
Keyte, K. - OP.3.d
Knoble, Gordon - DM.3.b,
SD.4.a, SE.6.c
Koeberlein, III, Ernest - DM.3.f
Koslosky, John T. - *SD.3.h,
*SD.5.c, SE.2.a
Krall, Laura A. - SD.2.h
Kruse, W. - DM.3.h
Kuiper, Thomas B. - OP.5.a
Kuntz, Jon - *OP.4.g
Kwadrat, Carl - MM.2.f
Labbe, X. - MM.2.g

MM - Mission Management, OP - Operations, DM - Data Management, SE - Systems Engineering, SD - Systems Development, * Presented in Poster Session

Author Index

Labezin, Christian - OP.3.e
 Labrune, Yves - MM.2.g
 Landshof, J. A. - OP.6.a, SE.6.d
 Larduinat, Eliane - SD.4.b
 Larrison, Gus - SD.4.g
 Lee, Young H. - SD.2.a
 Lehtonen, Kenneth - DM.2.g
 Levine, Allen J. - MM.1.f
 Levitt, D S. - SD.3.f
 Lightfoot, Patricia C. - SE.1.c
 Linick, Susan H. - MM.2.m
 Liu, Simon - MM.3.e
 Loubeyre, Jean Philippe -
 *OP.2.f

Louie, John J. - SD.2.f, SE.2.b
 Luczak, Edward C. - OP.4.d
 Ludwinski, J. M. - MM.1.b
 Lynenskjold, Steen - OP.3.d,
 OP.4.b, SD.3.i

M

Macé, Guy - OP.6.c
 Mackey, William - SE.5.b
 Mahmot, Ron - SE.2.a
 Maldague, Pierre F. - SD.2.e
 Malina, Roger F. - OP.2.a
 Mandl, Daniel - SE.5.a
 Manning, Evan - *DM.1.f
 Mao, Tony - DM.3.i
 Markley, Richard W. - DM.2.h
 Marshall, M. H. - SE.6.d
 Martelli, Andrea - OP.6.d
 Masters, W. C. - *OP.5.c

Maxwell, Theresa - MM.1.c
 McCaleb, Frederick - SD.4.a
 McKinney, J. Franklin -
 MM.2.b
 McLaughlin, Bruce A. - SD.4.c
 McOmber, Robert - DM.1.h
 McPherson, P. H. - OP.6.b
 Message, Philip - SD.4.f
 Messent, David - SE.1.b
 Michael, K. - DM.1.g
 Mies, L. - OP.4.e
 Miko, J. - DM.1.g
 Miller, Kevin J. - OP.4.h, SD.2.f
 Miller, W. H. - DM.1.g
 Mirchandani, Chandru - SD.5.d
 Moore, J. Michael - SD.5.a
 Mueller, Karl L. - MM.2.e
 Müller, Christian - SD.3.i
 Murphy, Elizabeth D. - *SD.2.d
 Murphy, Susan C. - SD.2.f,
 SE.1.a, SE.2.b
 Myers, C. R. - SD.3.f

N

Nakata, Albert Y. - MM.2.n
 Nakayama, K. - *SE.1.d
 Nanzetta, Kathy - DM.3.g
 Natanson, G. A. - *OP.5.d
 Nelson, Bill - OP.2.g
 Nester, Paul - SD.4.a
 Neuman, James C. - MM.2.b
 Newhouse, M. - MM.3.b
 Newsome, Penny A. - SD.4.d

Nguyen, Diem - SD.5.d, SE.5.b
 Nicklas, Randolph - SE.5.c
 Nickum, William G. - SD.1.a
 Niézette, M. - SE.4.c
 Nilsen, Erik N. - SD.4.c
 Norcross, Scott - OP.4.i
 Nye, H. R. - SE.3.g

O

O'Mullane, W. - SE.4.c
 O'Reilly, John - DM.2.m
 Oberto, Jean-Michel - MM.2.h
 Odubiyi, Jidé - SD.1.c
 Ondrus, Paul J. - SE.4.d
 Otranto, John F. - SD.4.d

P

Pace, Marco - SE.3.a
 Paczkowski, B. G. - MM.1.b
 Pajerski, Rose - SD.3.d
 Panem, Chantal - SD.2.g
 Parrod, Y. - MM.3.a
 Patt, Frederick S. - OP.2.i
 Pavloff, Michael S. - *OP.5.e
 Pecchioli, Mauro - OP.4.b, OP.4.f
 Peccia, N. - OP.2.h, SD.3.e
 Pender, Shaw Exum - DM.3.f
 Pendley, R. D. - SD.3.f
 Perkins, Dorothy C. - SE.4.e,
 SE.6.a
 Pietras, John - SE.3.j
 Pignède, M. - SE.3.h
 Pingitore, Nelson V. - *SE.6.b
 Pitt, Karl J. - MM.1.f

MM - Mission Management, OP - Operations, DM - Data Management, SE - Systems
 Engineering, SD - Systems Development, * Presented in Poster Session

Author Index

Pitts, Ronald E. - OP.3.f
Pollmeier, V. M. - *OP.5.c
Potter, William - SD.4.b
Poulter, K. - SE.4.c
Powers, M. - DM.1.g
Prettyman-Lukoschek, Jill -
MM.2.i

Pritchard, Jim - DM.3.b
Pujo, Olivier - MM.3.d

R

Rabenau, E. - DM.3.h
Rackley, Michael - SE.5.a
Rahman, Hasan - DM.2.i
Ramchandani, C. - SE.6.c
Rao, Gopalakrishna - MM.2.i
Real-Planells, B. - SE.3.h
Reed, Tracey - MM.1.c
Reese, Jay - OP.2.e
Reeve, Tim - SE.1.a
Reiss, Keith - OP.6.e
Richmond, Eric - SD.4.e
Rider, James W. - MM.2.l
Robinson, E. - MM.1.d
Rocco, David A. - MM.2.j
Rodríguez, Iván - SE.4.c
Rohrer, Richard A. - MM.2.k
Romero, R. - SE.4.c
Rosenberg, Linda H. - *SD.3.g
Rosenberger, John - SD.5.a
Rossi, F. - *SD.2.b
Rouff, Christopher - SD.4.f
Roussel, A. - MM.2.g
Rudd, Richard P. - MM.2.m

S

Salcedo, Jose - OP.4.j
Sank, V. - DM.1.g
Sary, Charisse - MM.3.e,
*OP.4.k
Scales, Charles H. - SE.1.c
Scheidker, E. J. - SD.3.f
Schlagheck, Ronald A. - MM.2.l
Schön, A. - OP.4.e
Schulz, Klaus-Jürgen - DM.1.b
Schwarz, Barbara - SE.2.a
Sheppard, Sylvia B. - *SD.3.g,
SE.5.d
Shi, Jeff - DM.2.d, DM.3.i, SD.5.d
Shigeta, T. - *SE.1.d
Shirah, Gregory W. - OP.4.d
Short, Jr., Nicholas M. - DM.3.a
Shurmer, I. - OP.4.f
Sigman, Clayton B. - *SD.3.h,
*SD.5.c
Sillers, J. - SE.6.c
Simons, Mark - SD.4.g
Sinclair, Craig - SD.3.d
Smith, Dan - OP.3.g
Smith, S. R. - SE.3.h
Smith, Simon T. - MM.3.d
Soderstrom, Tomas J. - SD.2.h
Solomon, Jeff - DM.3.g
Sonneborn, George - SE.6.a
Sørensen, Erik Mose - DM.3.j
Speciale, Nick - DM.2.d
Spradlin, Gary L. - MM.2.m

Sridharan, Ramaswamy -
MM.1.d, SE.5.e

Stallings, William - SE.3.i
Standley, Shaun - MM.1.e
Starbird, Thomas J. - OP.4.j,
SD.2.e

Starkey, Paul - MM.3.d
Stern, Daniel C. - MM.1.f
Stoffel, A. William - OP.4.l
Stokes, Grant - MM.1.g
Stoloff, Michael - SE.3.j
Strobel, Guenter - MM.1.h
Sturms, Jr, Francis M. -
MM.2.n

Suard, Norbert - DM.2.j
Sugawara, Masayuki - *DM.1.c
Symonds, Martin - SD.3.i
Szakal, Donna - SE.4.b

T

Tai, Wallace S. - MM.2.n
Talabac, Steve - DM.3.k
Thomas, C. W. - OP.1.c
Timmermans, R. - SE.4.c
Toft, Mark - MM.2.i
Torgerson, J. Leigh - OP.2.b
Troendly, Gregory M. - SD.1.a
Truszkowski, Walter F. -
SD.1.c, *SD.2.d

U

Uhrig, Hans - SE.3.j

MM - Mission Management, OP - Operations, DM - Data Management, SE - Systems
Engineering, SD - Systems Development, * Presented in Poster Session

Author Index

Uhrlaub, David R. - SE.2.c
Ulvestad, James S. - MM.1.i

V

Vallone, Antonio - SD.4.e
Valvano, Joe - DM.1.i
Varghese, Thomas - *OP.1.e
Vassallo, Enrico - SD.4.h
Vielcanet, Pierre - MM.2.g,
OP.3.e

Viggh, H. - MM.1.d
Vo, D.-P. - OP.4.c
Voglewede, Steven D. - DM.2.k

W

Weinberg, Aaron - DM.1.h
Welch, Dave - SE.3.k
Welling, John - DM.3.g
Werking, R. D. - SD.3.f
Wessen, Randii R. - SE.5.f
Wheadon, J. - SE.4.c
Whitgift, D. - SE.4.c
Whitworth, G. W. - OP.6.a
Wickler, Martin - *MM.3.f
Wiercigroch, Alexandria B. -
DM.2.1

Wimmer, W. - OP.3.c
Wiseman, Andy - MM.1.d,
SE.5.e

Wobbe, Hubertus - *OP.1.f
Wolff, Thilo - MM.3.d
Volken, Pamela R. - OP.5.a
Woodward, Robert H. - OP.2.i

Y

Yamada, Takahiro - SE.4.f
Yamamoto, K. - *SE.1.d
Yeh, P.-S. - DM.1.g
Yokokawa, Y. - *SE.1.d
Younger, Herbert - DM.2.m
Yven, Clet - *MM.3.c

Z

Zaouche, Gérard - OP.3.h
Zawacki, Steven J. - SD.2.e
Zeigenfuss, Lawrence B. - SE.4.e
Zillig, David J. - DM.1.h,
DM.1.i, SD.1.d

Zupke, Brian S. - SD.2.h

MM - Mission Management, OP - Operations, DM - Data Management, SE - Systems
Engineering, SD - Systems Development, * Presented in Poster Session

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.			
1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE November 1994	3. REPORT TYPE AND DATES COVERED Conference Publication	
4. TITLE AND SUBTITLE Third International Symposium on Space Mission Operations and Ground Data Systems		5. FUNDING NUMBERS 500	
6. AUTHOR(S) James L. Rash, Editor			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Goddard Space Flight Center Greenbelt, Maryland 20771		8. PERFORMING ORGANIZATION REPORT NUMBER 94B00135	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Washington, D.C. 20546-0001		10. SPONSORING/MONITORING AGENCY REPORT NUMBER CP-3281 Part 1	
11. SUPPLEMENTARY NOTES This Proceedings is composed of two volumes.			
12a. DISTRIBUTION/AVAILABILITY STATEMENT Unclassified-Unlimited Subject Category 17 Report available from the NASA Center for AeroSpace Information, 800 Elkridge Landing Road, Linthicum Heights, MD 21090; (301) 621-0390.		12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) Under the theme of "Opportunities in Ground Data Systems for High Efficiency Operations of Space Missions," the SpaceOps '94 symposium included presentations of more than 150 technical papers spanning five topic areas: Mission Management, Operations, Data Management, System Development, and Systems Engineering. As stated in the executive summary of the symposium proceedings, the papers "focus on improvements in the efficiency, effectiveness, productivity, and quality of data acquisition, ground systems, and mission operations. New technology, techniques, methods, and human systems are discussed. Accomplishments are also reported in the application of information systems to improve data retrieval, reporting, and archiving; the management of human factors; the use of telescience and teleoperations; and the design and implementation of logistics support for mission operations."			
14. SUBJECT TERMS Data Handling, Telemetry Processing, Mission Planning, Orbit Determination, Standards, Modeling, Communications Networks, Communications Systems, Ground Based Data Acquisition Systems, Spacecraft Communications, Spacecraft Command, Spacecraft Tracking, Spacecraft Control (Communications), Expert Systems		15. NUMBER OF PAGES 693	
		16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT Unlimited