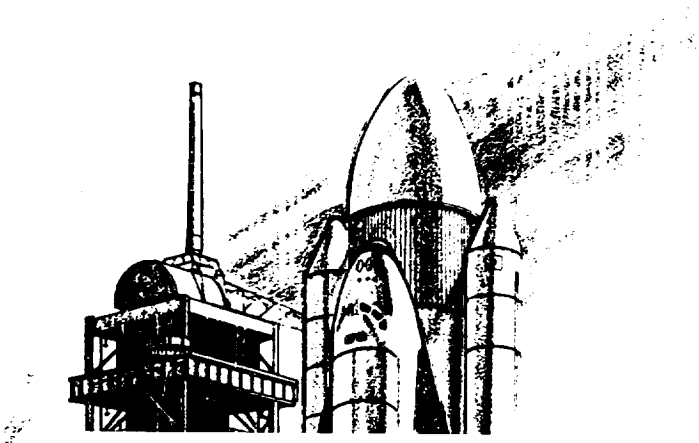
Proceedings of: AF-SD / INDUSTRY / NASA CONFERENCE AND WORKSHOPS ON

MISSION ASSURANCE

VOLUME 2

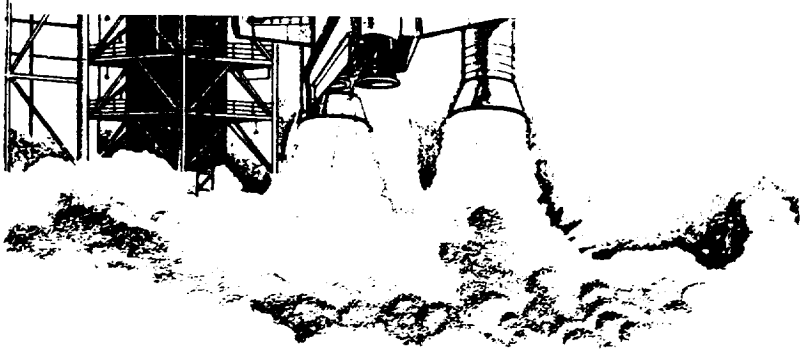


(NASA-TM-101117) PROCEEDINGS OF THE AF/SD/INDUSTRY/NASA CONFERENCE AND WORKSHOPS ON MISSION ASSURANCE, VOLUME 2 (NASA) 646 p

N90-70618

Unclas

00/81 0141485



SECURITY INDUSTRIAL A.

AIA

Aerospace Industries Association

SPACE DIVISION

Space Division

NASA

National Aeronautics and Space Administration

HYATT HOUSE-AIRPORT
LOS ANGELES
7-8-9 JUNE 1983

#141485

TABLE OF CONTENTS

F. Design Assurance Techniques F-1

Long Mission Life Commercial and Government Communication Satellites - A Comparison F-1

Design Assurance for Ground Segments F-19

Automated Design and Analysis Techniques F-161

Fault Tolerant Equipment Design, Test and Reliability F-207

Summary of Issues and Recommendations F-261

G. Materials and Processing Integrity Workshop G-1

Summary of Issues and Recommendations G-87

H. The Acquisition Process H-1

Summary of Issues and Recommendations H-83

I. Software I-1

Software Management I-1

Transition to Ada I-7

Software Development Management I-27

Summary of Issues and Recommendations I-171

J. Test J-1

Summary of Issues and Recommendations J-29

SPECIAL REPORT SR-1

REGISTRANTS R-1

WORKSHOP F

DESIGN ASSURANCE TECHNIQUES

Chairmen:

Ken Hughey
Senior Project Engineer
Launch Systems, Integration
Hughes Aircraft Company

Robert Miller
Chief Engineer
RCA Astro-Electronics

Patricia Anderson
Associate Principal Director
The Aerospace Corporation

Harry Sonnemann
Deputy Engineer for Systems Engineering
and Technology
NASA

Joseph Statsinger
Executive Assistant for Group Vice President
The Aerospace Corporation

Coordinator:

Val Monshaw
Manager, Product Assurance
RCA Astro-Electronics

SESSION 1 — LONG MISSION LIFE COMMERCIAL AND GOVERNMENT COMMUNICATION SATELLITES — A COMPARISON OF REQUIREMENTS AND APPROACHES

Communication Satellite Design Requirements and Design Verification	W. J. Keck, Comsat General
SPACENET	R. Dalebout, So. Pacific Satellite Co.
Defense Communications Satellite System	C. Bredall, The Aerospace Corporation
Approach to Requirements for Long Mission Life Spacecraft	A. W. Weinrich, RCA Astro-Electronics
DSCS III	P. Fishman, General Electric

SESSION 2 — DESIGN ASSURANCE FOR GROUND SEGMENTS

A Catalyst for Ground Segment Mission Assurance	H. Pickover, TRW
Lessons Learned in Software Development for AFSCF's NASA Interface Subsystem	G. L. Wade, FACC
Application of Preplanned Product Improvement (P^3I) to the Consolidated Space Operations Center (CSOC) Program	Maj. J. E. Ford, AF Space Division
Mission Assurance in DSM	R. W. Cosgrove and G. T. Kelton, IBM
DoD Space Shuttle Operations at Vandenberg Air Force Base Launch and Landing Site	P. L. Portanova, The Aerospace Corporation
Integration and Test of the Defense Meteorological Satellite Program Test System	V. E. Olson, The Aerospace Corporation
TPS's As a Design Technique	S. A. Carpenter, Rockwell International
Launch Opertions Design Assurance Considerations	O. H. Fedor, NASA
A Method of Software Safety Analysis	J. G. Griggs, III, Martin Marietta Corp.

WORKSHOP F

DESIGN ASSURANCE TECHNIQUES

SESSION 3 — AUTOMATED DESIGN AND ANALYSIS TECHNIQUES

Discussion of Evolving CAD/CAM Usage	Robert G. Damskey, Hughes Aircraft Co.
CAD/CAM at TRW	John Cairns, TRW
CADDS and Industry Effectiveness	Richard Sharp, Hughes Aircraft Co.
Improved Products Through Integration of Design and Analysis: Finite Element Modeling as an Example	Henry Jacobson and Phil Ness, Boeing
LEADS — An Integrated CAE/CAD/CAM/CAT System	Roy Roberts, LMSC
An Electronic Design Management System	Ray McCann, Prime Computer, Inc.

SESSION 4 — DESIGN ASSURANCE

Design Assurance	Ken Hughey, Hughes Aircraft Co.
Test Failure Trend Management	Arthur W. Ackerman, Jr., Ford Aerospace
Design Assurance Techniques	Panel Discussion

SESSION 5 — FAULT-TOLERANT EQUIPMENT-DESIGN, TEST AND RELIABILITY

Computer-Aided Reliability Estimation Techniques for Fault-Tolerant Systems	S. J. Bavuso, Langley Research Center
Self-Checking Computer Modules	D. E. Erickson and D. A. Geer, JPL
Multiple Microprocessor Flight Control Architectures	G. Hartman and K. Criscoll, Honeywell
IUS Design for Fault-Tolerance	J. H. Keeney, Boeing Company
Mission Assurance through Spacecraft Autonomous Action (Galileo Fault Protection)	P. M. Kobele, JPL
The Need for Design Verification in Fault-Tolerant Systems	K. Levitt, SRI International
Autonomous Redundancy and Maintenance Management Subsystem (ARMMS)	M. Marcucci, JPL

William J. Keck
COMSAT General Corporation

COMMUNICATION SATELLITE DESIGN REQUIREMENTS AND DESIGN VERIFICATION

Commercial communications satellites are purchased for profit purposes and not to advance science. Although the state-of-the-industry is often pressed to provide more capacity and better performance, the associated payoffs are balanced against schedule risk and levels of reliability. The highest reliability is associated with tried and proven designs and these are preferred when they are consistent with the required performance.

Analysis is not a substitute for testing. Testing provides the total proof of performance and includes the design plus the processes used in the manufacture of each component and the assembly. Subtle effects of thermal changes, the vacuum environment, and manufacturing imperfections, are more directly and completely evaluated by testing than by analysis. Each subsystem on each spacecraft should receive full system level testing in the predicted environments.

Spacecraft design features and the total test plan are best negotiated between customer and contractor prior to consummating the contract. Furthermore, these design and test requirements should become part of the contract. Nothwithstanding that formal commitment to design and testing, a flexibility should be provided to customer and contractor for test changes to be introduced by mutual agreement and without the delays associated with contract changes.

COMSAT GENERAL CORPORATION

SYSTEMS TECHNOLOGY SERVICES

DESIGN & TESTING PHILOSOPHY

- Communications Satellites are to make money not to advance science.
- Analysis is a design tool not a design verification.
- Test all systems thoroughly under all predicted environmental conditions.

COMSAT GENERAL CORPORATION

SYSTEMS TECHNOLOGY SERVICES

DESIGN REQUIREMENTS

1. Perform Pre-RFP Feasibility and Trade-Off Studies.
2. Include or Exclude, as applicable, Specific Design Features in RFP.
3. Conduct Pre-Contract Design Negotiations.
4. Use Mission Proved Components.
5. Provide Performance Margins for Each Subsystem.
6. Minimize Spacecraft Complexity.

COMSAT GENERAL CORPORATION

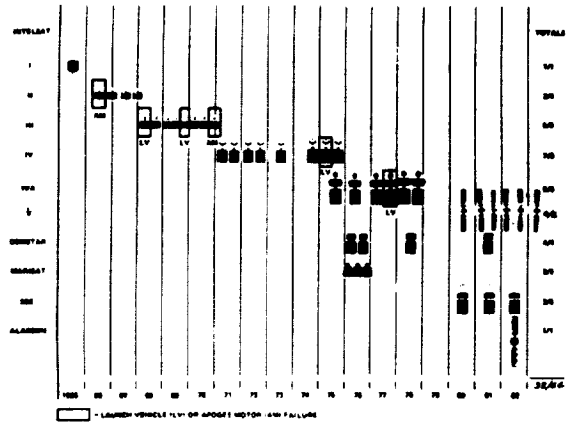
SYSTEMS TECHNOLOGY SERVICES

DESIGN VERIFICATION (Continued)

7. Analyze Components/Subsystems which Perform within Specification Limits but Show Anomalous Characteristics.
8. Participate in Failure Analyses and Problem Solving.
9. Perform Full System Level Tests at the Launch Site.
10. Obtain Sufficient Data to Permit Comparison with In-Orbit Data.
11. Utilize In-Orbit Test Data & Operational Performance as Inputs to Follow-On Designs and Testing.
12. Permit Flexibility in Test Plan and Procedures without Involving the Contract.

COMSAT

SATELLITE LAUNCH EXPERIENCE SUMMARY



COMSAT GENERAL CORPORATION

SYSTEMS TECHNOLOGY SERVICES

DESIGN VERIFICATION

1. Include detailed test requirements in RFP.
2. Conduct pre-contract negotiation of the detailed test plan and incorporate it in the Contract.
3. Review test procedures for adequate and realistic tests.
4. Perform system level tests of all subsystems.
5. Examine test data in Real Time.
6. Test sufficiently to reveal early failure modes.

COMSAT GENERAL CORPORATION

SYSTEMS TECHNOLOGY SERVICES

RECOMMENDATIONS

1. Utilize State of Industry Proved Technology Whenever Possible.
2. Define Detailed Test Plan Prior to Consumating Contract.
3. Provide Test Plan Flexibility to Permit Customer/Contractor Mutually to Revise Plan or Tests w/o Contract Change.

Ron Dalebout
Southern Pacific Satellite Company

SPACENET

The objective of every commercial venture is to obtain a competitive product at a reasonable cost; to be able to sell the product at a price that will allow you to make a profit. To enhance the probability of achieving this objective, in the case of a commercial communications satellite, a framework of circumstances should be implemented that makes it easier to accomplish the production of a reliable and capable satellite in the shortest period of time at the lowest possible cost commensurate with the associated risks.

A set of specific circumstances that will enable attainment of the above objective are:

- Firm Fixed Price Contract
- Performance Vice Design Specification
- Short Production Cycle
- Incentive on Progress and Orbital Life
- Limited Number of Customer Approvals
- Mutual Trust and Respect Relationship

The firm fixed price contract places all of the cost risk with the contractor. The contractor's price allows him the possibility of profit. The customer must support the continuing possibility of profit by assiduously avoiding parallel design efforts, eliminating unnecessary documentation and defining the high cost activities such as testing scope and sparing in detail with the contract. This reduced the cost uncertainty. Additionally, the customer must actively pursue cost avoidance during contract performance by being willing to descope unnecessary efforts.

Defining the requirements necessary to be competitive is being accomplished for a commercial communication satellite using a performance vice design specification exclusively at the system (i.e., satellite) level. This assures the comparative quality of performance without constraining the contractor's latitude for economic design innovation. The customer's obligation is to assure that <u>all</u> of his requirem ents are succintly, simply, and explicitly stated.

The shortest possible realistic production cycle should be agreed to for the simple reason that cost are, in general, directly proportional.

The incentive schedule should reward production progress moderately and long-term on-orbit performance lavishly. Any late delivery negative incentive should be scaled to recompense the customer only for resulting finite costs incurred and should be kept as small as possible.

Keep the customer out of the design and documentation approval cycle as much as is practicable. Limit the approvals to items directly related to system level performance such as:

- Product Assurance Program Plans
- System Level Test Plan
- Special Unit and Subassembly Test Plans
- Satellite Acceptance Test Data
- Satellite On-Orbit Performance Test Data

This area is a significant source of customer-related delays and is a primary genesis of antagonist relationships.

Actively work towards the evolution of a "mutual trust and respect" associated between the customer and contractor. Select your interfacing personnel carefully and always seek a consensus decision. The fundamental requirement is on-orbit system level performance. With heavy incentives on operating mission, the contractor is motivated to make decisions that enhance, not degrade or jeopardize this phase so he can logically be trusted.

The bottom line is that the customer must invest the contractor's motivation into long-term competitive performance on-orbit; intelligently afford him the maximum possible freedom in design and test; maintain a positive reinforcing working relationship; and most importantly allow him a realistic opportunity to make a profit.

Southern Pacific Satellite Company

OBJECTIVE: TO PRODUCE A COMPETITIVE DOMESTIC COMMUNICATIONS SATELLITE AT A REASONABLE COST

Southern Pacific Satellite Company

PERFORMANCE SPECIFICATION

- o SET COMPETITIVE LEVELS
- o ELIMINATE DESIGN SPECIFICATIONS
- o LIMIT TO SYSTEM LEVEL ONLY
- o ASSURE COMPLETE REQUIREMENTS
- o USE SIMPLE, EXPLICIT TERMS

Southern Pacific Satellite Company

SPACENET PROGRAM APPROACH

- o FIRM FIXED PRICE CONTRACT
- o PERFORMANCE SPECIFICATION
- o 30 MONTH PRODUCTION CYCLE
- o INCENTIVE ON PROGRESS AND ORBITAL LIFE
- o LIMITED NUMBER OF CUSTOMER APPROVALS
- o MUTUAL TRUST AND RESPECT RELATIONSHIP

Southern Pacific Satellite Company

INCENTIVE ON PROGRESS AND ORBITAL LIFE

- o 10% OF TOTAL PRICE IN PROGRESS
  - MANAGEABLE MILESTONES EVERY QUARTER
  - EMPHASIZES CRITICAL HARDWARE AND TESTS
- o 26% OF SATELLITE PRICE IN ORBITAL LIFE
  - TOTAL PAID UPFRONT IF FULLY OPERATIONAL
  - PROPORTIONALLY REBATED AS FAILURES OCCUR
  - IF MORE THAN A GIVEN NUMBER, BY TYPE, FAILS THEN TOTAL REMAINING PROPORTIONAL INCENTIVE REBATED
- o GRADUATED LATE DELIVERY PRICE REDUCTION
  - NOT TOO SEVERE

Southern Pacific Satellite Company

FIRM FIXED PRICE CONTRACT

- o MAINTAIN PROFIT MOTIVATION
  - PROHIBIT PARALLEL DESIGN
  - ELIMINATE UNNECESSARY DOCUMENTS
- o DEFINE HI-COST ITEMS IN DETAIL
  - SCOPE OF TESTING
  - SPARING
- o PURSUE COST AVOIDANCE

Southern Pacific Satellite Company

30 MONTH PRODUCTION CYCLE

o PRELIMINARY DESIGN	3 MONTHS*
o FINAL DESIGN AND UNIT FABRICATION (SPACECRAFT CRITICAL DESIGN AFTER 13 MONTHS)	15 MONTHS
o SUBSYSTEM INTEGRATION AND TEST	6 MONTHS
o SATELLITE INTEGRATION AND TEST	6 MONTHS

* VERY SIMILAR TO EXISTING DESIGN

Southern Pacific Satellite Company

BOTTOM LINE

INVEST THE CONTRACTOR'S MOTIVATION INTO LONG TERM COMPETITIVE PERFORMANCE ON-ORBIT. INTELLIGENTLY AFFORD HIM THE MAXIMUM POSSIBLE FREEDOM IN DESIGN AND TEST. MAINTAIN A POSITIVE REINFORCING WORKING RELATIONSHIP. ALLOW THE CONTRACTOR A REALISTIC OPPORTUNITY TO MAKE A PROFIT!!!

Southern Pacific Satellite Company

MUTUAL TRUST AND RESPECT RELATIONSHIP

- o ALL THE COST RISK IS RESIDENT WITH THE CONTRACTOR
- o FUNDAMENTAL REQUIREMENT IS SYSTEM PERFORMANCE
  - ALLOW CONTRACTOR DESIGN AND TEST LATITUDE AT LOWER ASSEMBLY LEVELS
  - CONTROL THROUGH APPROVAL OF SPECIFIC DOCUMENTS
- o AVOID ANTAGONIST RELATIONSHIP
  - SELECT COMPATABLE INTERFACING PERSONNEL
  - REACH DETAILED AGREEMENT ON ALL CRITICAL TASKS

Southern Pacific Satellite Company

LIMITED NUMBER OF CUSTOMER APPROVALS

- o LIMIT APPROVAL TO ITEMS DIRECTLY RELATED TO SYSTEM LEVEL PERFORMANCE
  - PRODUCT ASSURANCE PROGRAM PLANS
  - SATELLITE TEST PLAN
  - SPECIAL UNIT AND SUBASSEMBLY TEST PLANS
  - SATELLITE ACCEPTANCE TEST DATA
  - SATELLITE ON-ORBIT PERFORMANCE TEST DATA
- o SOURCE OF CUSTOMER-RELATED DELAYS
- o CONSTRAIN THE GENESIS OF ANTAGONIST RELATIONSHIP

C. H. Bredall
Aerospace Corporation

DEFENSE COMMUNICATIONS SATELLITE SYSTEM

An introduction to the Defense Satellite Communications System (DSCS) is presented showing the three phases (generations). The Initial Defense Communications Satellite Program (IDCSP) followed by DSCSII and DSCSIII.

When plotted, a linear relationship appears between the design life and year of first launch of these three generations. The MTTF plot is also basically linear with time of first launch. If you believe in extrapolations of such data, a hypothetical future DSCS satellite with first launch in 1993 would have a 14-year design life and a MTTF of 11 to 12 years. The achievement of such life attributes are influenced by such factors as the sophistication of the satellite (for example, parts count), on-orbit environment (for example, man-induced stresses) and the built-in quality of the satellites. A less emphasized factor is "obsolescence." When I was a Customer Engineer with IBM in the early 1950's, I submitted what I thought was an award-winning suggestion for extending the life of an electro-mechanical commutator on a card-sorter. The reply was "Thanks for the suggestion but the commutators are designed to be cost-effective with the projected obsolescence of the sorter." Newer and much improved sorters had already been designed and built. It would be rare indeed if an incremental cost increase is denied because of expected obsolesence of a satellite. Incremental cost denials for other reasons, however, are not all that rare.

Satellite obsolescence appears most likely to occur because of changes in user requirements, new threats or breakthroughs in technology.

DSCS earth terminals during this same 20-year period have also shown design life properties which are linearly increasing with time advancing from a 10-year life for the IDCSP service of 1966 to a 20-year design life for the new GSC-395 SAMT terminal (scheduled for 1985 implementation). An extrapolation of this design life plot indicates a 1993 life of 25 years which is viewed as doubtful, even if obsolesence is not a factor.

The DSCS constellation consists of four active and two spare satellites spaced along the equator. The active satellites are distributed: one in the Atlantic Ocean, two in the Pacific Ocean, and one in the Indian Ocean.

Control of the satellite is accomplished by a combination of the Air Force Satellite Control Facility (AFSCF) and the Satellite Configuration Control Element (SCCE). The AFSCF normally assures the proper orbit, anomaly analysis and correction and housekeeping, such as battery reconditioning. The SCCE normally commands the payload configuration and interfaces with other elements of the Defense Communications System. Either facility has the capability of sending the full command directory and receiving and processing all telemetry.

In this Design Assurance effort there is a constant balancing act going on. The channels of acquisition and direction-giving are well established. However, the period of years leading up to the specifications for the space and ground segments involve a much more complex set of relationships--some formal, some not so formal.

The major satellite design factors are easily identified. An if the satellites were built like automobiles--thousands at a time, repairable in use, and 5-year obsolesence--the design would be easy. The R&D, the user communications needs and the constraints are important factors. What is not readily shown are

the many interactions of these and other factors, including a projection of these factors into the future.

Accordingly, design requires extensive communications, foresight, insight, and the various "ilities" (the newer ones being testability and encurability). An when foresight gets a little hazy when looking 15 to 20 years into the future, a little luck helps a lot.

Constraints are not always obvious. These include funding, politics (including bureaucracy), weight and other booster constraints, environment (natural and man-made), schedules, long life and performance.

So we write a space segment specification and a control segment specification. Should it be functional? Should it be a point specification? If it is functional, the contractor feels a lack of direction and does not have enough information to offer a fixed price and the Government is faced with a series of costly change orders. If it is a point specification with a high degree of configuration detail, the contractor is not free to introduce valuable ideas or exercise his established methods and procedures.

Some examples of the CSCSIII design preliminaries are given. Objectives and concerns differ somewhat between the Government on the one hand and the contractor on the other.

In conclusion, some of the future technology is shown which could be adapted to the future DSCS.

System Description

DSCS SPACE SEGMENT, SUPER HIGH FREQUENCY (7 / 8 GHz)

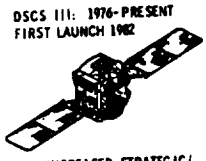
IDCSP: 1964-1973
FIRST LAUNCH 1966

- LONG HAUL STRATEGIC COMM
- 26 SATS
- 4 YR DESIGN LIFE
- EARTH COVERAGE

DSCS II: 1969-PRESENT
FIRST LAUNCH 1971

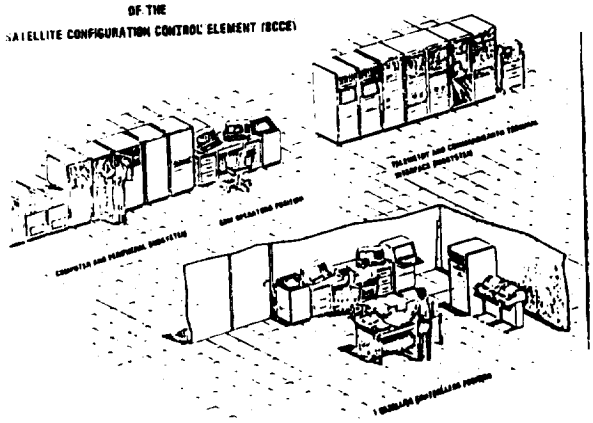
- ADDITIONAL SUPPORT FOR DOD AND STATE
- 16 SATS
- 5 YR DESIGN LIFE
- EARTH AND 2 NARROW BEAMS
- LIMITED ANTI-JAM AND NUCLEAR HARDENING

DSCS III: 1976-PRESENT
FIRST LAUNCH 1982

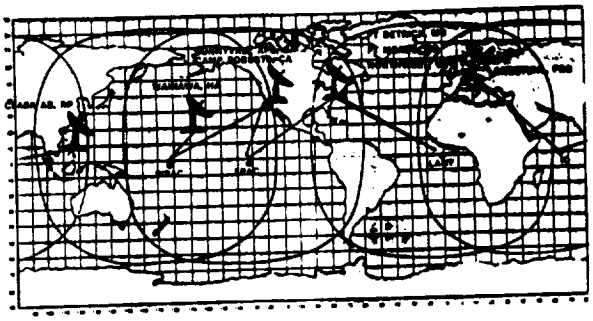


- INCREASED STRATEGIC/ TACTICAL SUPPORT
- 14 SATS
- 10 YR DESIGN LIFE
- EARTH, NARROW AND MULTIPLE VARIABLE BEAMS
- IMPROVED ANTI-JAM AND NUCLEAR HARDENING
- AFSATCOM/SCT

ENGINEERING DEVELOPMENT MODEL (EDM)
OF THE
SATELLITE CONFIGURATION CONTROL ELEMENT (SCCE)

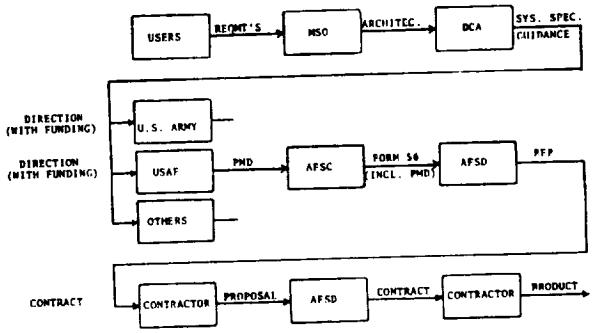


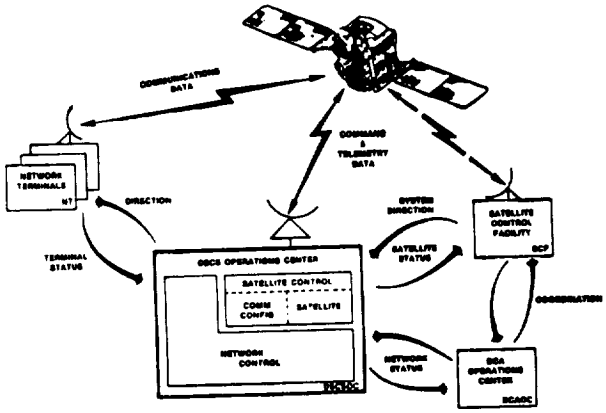
PLANNED DEPLOYMENT
X-BAND COMMAND / TELEMETRY STATIONS



MILITARY COMMUNICATION SATELLITES

DIRECTION & ACQUISITION FLOW





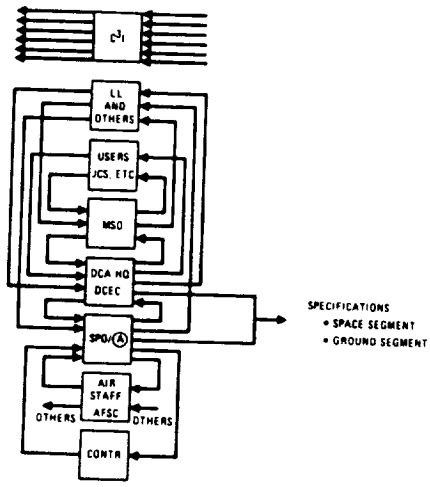
DSCS CONTROL FLOW

MILITARY COMMUNICATION SATELLITES

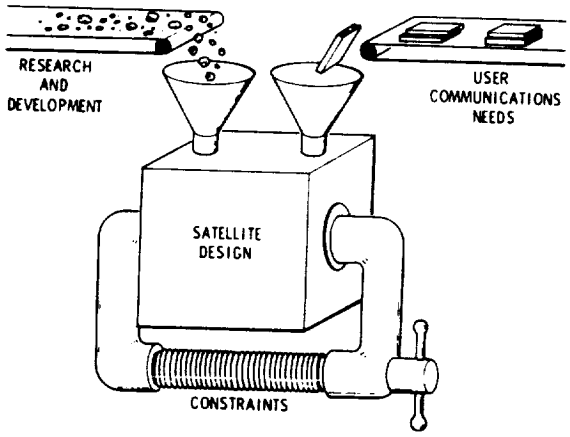
PROGRAM APPROVALS

PHASE (DESCRIPTION)	MILESTONE	NOMINAL APPROVAL
PHASE 0 (CONCEPT EXPLORATION)	0	OSD
PHASE 1 (THRU PDR)	1	DSARC 1
PHASE 2 (CDR, QUALIFICATION AND DEVELOPMENT OF FLIGHT SATELLITES)	2	DSARC 2
PHASE 3 (PRODUCTION	3	DSARC 3

DSCS III Preliminary Discussions 1973-1975



Design Factors



DSCS III

SPACE SEGMENT SPECIFICATION

- FUNCTIONAL SPECIFICATION?
- POINT SPECIFICATION?
- MIX?

DEVELOPING THE RFP

SPACE SEGMENT SPECIFICATION

S/C SIZE/WEIGHT

ONE ON THOR/DELTA 1000# → TWO ON STS 5000#

PRIME POWER

400W → 1000W

RF POWER

60W → 120W

CHANNELS

3 → 6

MBA ANTENNAS 1 - 36" DIA (39 ELEMENT) → 1 - 44" DIA (61 ELEMENT); 2 - 26" DIA (19 ELEMENT)

DSCS III

TRANSPONDERS/SWITCHING

OBJECTIVE:

TO ACHIEVE Ps ≥ 70% AT 7 YEARS

PROVIDE COMMUNICATIONS { INTER-CHANNEL / INTRA-CHANNEL } REDUNDANCY

PROVIDE TT&C REDUNDANCY

PROVIDE CONNECTIVITY FLEXIBILITY ANTENNAS/TRANSPONDERS

SOLUTIONS:

SIX TRANSPONDER CHANNELS

ECH OR MBA	CH. 1	40W	MBA OR GDA
	CH. 2	40W	
	CH. 3	10W	MBA OR ECH
	CH. 4	10W	MBA OR ECH OR GDA
ECH	CH. 5	10W	ECH
	CH. 6	10W	

TT&C

S-BAND SGLS

X-BAND UPLINK ON CH. 1 OR CH. 5 DOWNLINK ON REDUNDANT BEACONS

DSCS III

MULTIPLE BEAM ANTENNA

RECEIVE (NULLING) ANTENNA

HOW MANY ELEMENTS? 37? 61? 91?

- RESOLUTION
- SIZE
- WEIGHT
- POWER

TRANSMIT ANTENNAS

HOW MANY MBA'S? HOW MANY ELEMENTS? 7? 19? 37?

HOW MANY EC HORNS?

GIMBALLED DISH ANTENNA?

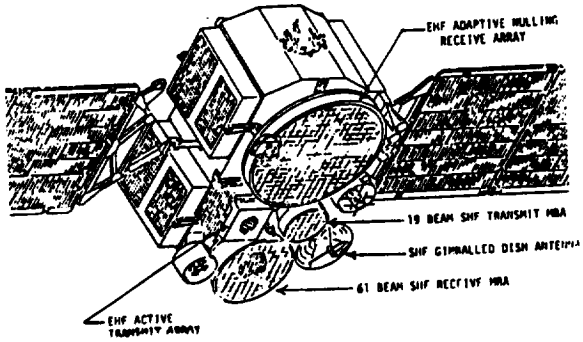
OBJECTIVES/CONCERNS

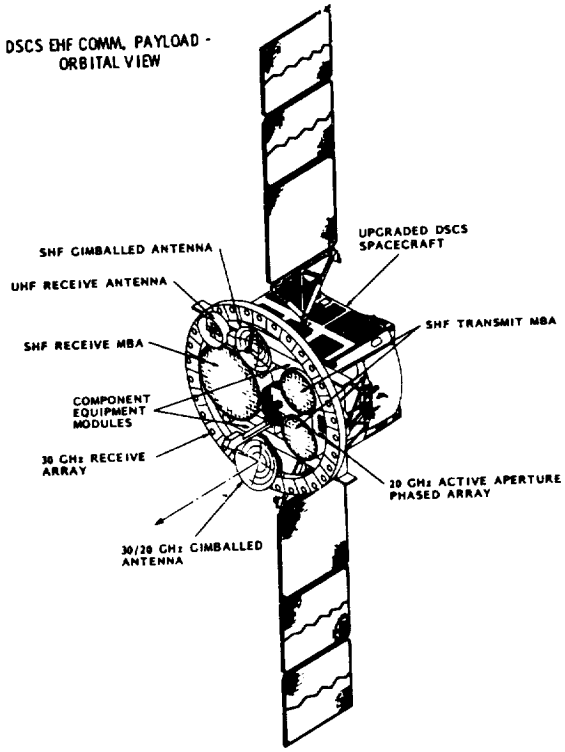
PROCURING AGENCY:

- ASSURE DELIVERY OF SPECIFIED PRODUCT
- AVOID OVERRUNS
- MEET SCHEDULE

CONTRACTOR:

- MEET SPECIFICATIONS
- PROTECT PROFIT
- MEET SCHEDULE





APPROACH TO REQUIREMEMENTS FOR LONG MISSION LIFE SPACECRAFT

A. W. Weinrich
RCA ASTRO-ELECTRONICS

A summary of the key aspects in the approach to requirements for commercial long-life spacecraft was presented. The foundation for establishing requirements is the economics of system cost vs the revenue producing capability of the system. This means striking a balance between the market pressures for higher performance and the technical constraints of available devices and booster capability. The market size and launch cost essentially determine spacecraft size and weight. User terminal size and location establishes coverage and the antenna design. The number of transponder channels is then that which is available with the selected booster size.

Design constraints include state-of-the-art devices and processes. Level S parts are used with selected, well-understood waivers when parts availability becomes a problem. Comprehensive testing is the key to long-life mission assurance. Complex technical problems are resolved with a relatively small number of people selected because of their depth of knowledge and the physics of the device or subsystem in question. Large committees and working groups are avoided.

Commercial contracts offer high orbital incentives and delegate the maximum latitude and responsibility to the contractor in the design and manufacture of the spacecraft. The object is to motivate success on orbit at minimum program cost by establishing large incentives and penalties to the spacecraft contractor.

ASTRO communications satellites

APPROACH TO REQUIREMENTS LONG MISSION LIFE SPACECRAFT

- FUNCTIONAL REQUIREMENTS -- STRIKING A BALANCE
- DESIGN CONSTRAINTS
- DESIGN MARGIN
- COMPREHENSIVE TESTING
- RESOLVING COMPLEX TECHNICAL ISSUES
- COMMERCIAL CONTRACTS -- SUMMARY

RCA Astro-Electronics

ASTRO communications satellites

EVOLUTION OF FUNCTIONAL REQUIREMENTS

- MARKET POTENTIAL
- LAUNCH COST

→ SPACECRAFT SIZE & WEIGHT

- USER TERMINAL SIZE & LOCATION
- SPACECRAFT SIZE & WEIGHT

→ PAYLOAD CONFIGURATION, ANTENNAS & NUMBER OF CHANNELS

- ACCEPTABLE RISK → - USE AVAILABLE POWER DEVICES; - MANAGEABLE INCREASES IN COMPLEXITY
- MUST DESIGN, MANUFACTURE, INTEGRATE & TEST IN 20 MONTHS

RCA Astro-Electronics

ASTRO communications satellites

STRIKING A BALANCE

MARKET PRESSURES

- HIGHER EIRP
- MORE CHANNELS
- BETTER TRANSMISSION QUALITY
- LONGER LIFE

TECHNICAL CONSTRAINTS

- AVAILABLE DEVICES
- BOOSTER CAPABILITY
- BATTERY & SOLAR ARRAY CAPACITY
- REACTION CONTROL SYSTEM & DEVICES

ECONOMICS

- SYSTEM COST
- REVENUE PRODUCING CAPABILITY

RCA Astro-Electronics

ASTRO communications satellites

DESIGN CONSTRAINTS

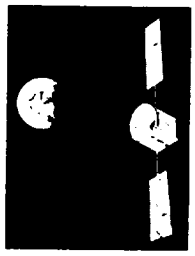
- STATE-OF-THE-ART DEVICES QUALIFIED OR SUFFICIENT UNDERSTANDING OF DEVICE PHYSICS FOR HIGH CONFIDENCE THAT DEVICE CAN BE QUALIFIED
- LEVEL S PARTS WITH SELECTED, WELL-UNDERSTOOD WAIVERS
- NO ACTIVE SINGLE POINT FAILURE (STRUCTURE, AKM, ETC. ARE NOT REDUNDANT!)
- FACTORY TESTABILITY OF SYSTEM, SUBSYSTEMS, SUBASSEMBLIES
- MUST BE PRODUCIBLE WITH PROVEN, QUALIFIED PROCESSES OR SUFFICIENT UNDERSTANDING OF NEW PROCESSES EXISTS TO ASSURE FLIGHT WORTHINESS OF END PRODUCT

RCA Astro-Electronics

ASTRO communications satellites

The Contrasting Points of View

Customer/User



Spacecraft Manufacturer

Launch Operations

RCA Astro-Electronics

ASTRO communications satellites

DESIGN MARGIN IS KEY TO LONG LIFE MISSION ASSURANCE

- TRANSPONDER AMPLIFIER GAIN & GAIN CONTROL
- CONTROL LOOP STABILITY -- PHASE MARGIN
- POWER SUBSYSTEM SIZING -- BATTERIES & SOLAR ARRAY
- MECHANISMS FOR PANEL & ANTENNA DEPLOYMENTS
- ELECTRO-MECHANICAL -- SLIP RING DESIGN & RF SWITCHES
- COMMAND LINKS -- ERROR PROTECTION & REDUNDANCY
- PROTECT POWER BUS FROM CATASTROPHIC FAILURES

RCA Astro-Electronics

ASTRO

COMPREHENSIVE TESTING IS KEY TO LONG LIFE MISSION ASSURANCE

communications satellites

- COMPREHENSIVE FACTORY TESTING OF ALL SUBSYSTEMS AND SUBASSEMBLIES AT BOX AND SPACECRAFT LEVEL
- EXPLOIT COMPUTER-DRIVEN AUTOMATED TESTING TO AVOID TIME PENALTY
- LIFE TEST ISSUES SETTLED OFF-LINE AT COMPONENT (AMPLIFIER OR PROCESSOR LEVEL) BEFORE INITIATING SPACECRAFT DEVELOPMENT
- MINIMUM OPERATIONS ON SPACECRAFT AT LAUNCH SITE
  - SYSTEM ELECTRICAL TEST, SPIN BALANCE, AKM, PYROS
  - NO MECHANICAL OR MECHANISMS INSTALLATION AT KSC

RCA Astro-Electronics

ASTRO

RESOLVING COMPLEX TECHNICAL ISSUES

communications satellites

- IDENTIFY THE KNOWLEDGEABLE INDIVIDUAL(S)
- OBTAIN ASSESSMENT & DECISION
- REVIEW FOR COMPLETENESS
- TRUST THE RESULT

LARGE COMMITTEES CREATE CONFUSION & WASTE TIME

RCA Astro-Electronics

ASTRO

COMMERCIAL CONTRACTS

communications satellites

- OFFER HIGH ORBITAL INCENTIVES
- DELEGATE MAXIMUM LATITUDE/RESPONSIBILITY TO CONTRACTOR IN DESIGN/MANUFACTURING
- REQUIRE STRINGENT ACCEPTANCE & ON-ORBIT TESTS

SUCCESS ON-ORBIT AT MINIMUM PROGRAM COST IS ACHIEVED

RCA Astro-Electronics

Paul Fishman
General Electric

ACHIEVING LONG LIFE

- ANALYSIS
  - WORST CASE ANALYSIS DOWN TO PIECE PART LEVEL
  - FMEA FOR SINGLE POINT FAILURE MODES
- DESIGN
  - REDUNDANCY
  - EXPENDABLES SIZING
  - DERATING
  - REQUIREMENTS ALLOCATION
  - CRITICAL ITEM CONTROL
  - DESIGN MARGIN
- PARTS MANAGEMENT
  - PIPS APPROACH
  - EMPHASIS ON QUAL, TESTING, INSPECTION
- PROBLEM RESOLUTION
  - EMQ CORRECTIVE ACTION APPROACH
  - TIMELY ALERTS
- TESTING
  - EXTENDED LEVELS AND DURATIONS
  - LIFE TESTING

REQUIRES ATTACK
ON ALL FRONTS...
AND LOTS OF T L C

TESTING

REACTION CONTROL SYSTEM THRUSTERS

PARAMETER	NOMINAL REQ'T.	WORST CASE REQ'T.	TEST DEMO	RETEST DEMO
THROUGHPUT (LB-SEC)	28,000	50,000	70,000	70,000
PULSING (CYCLES)	20,000	40,000	40,000*	90,000**
DURATION (HOURS)	1	1	2	2

*NITRIDING AT HIGH TEMPERATURES PRODUCED CRACK FAILURES

**AFTER REDESIGN OF THRUSTER CATALYST-BED RETENTION TO IMPROVE HEAT CONDUCTION/LOWER THERMAL STRESS

ASSURE
WORST-CASE MARGINS
THROUGH TESTING

PARTS MANAGEMENT

PROGRAM INTEGRATED PARTS SYSTEM (PIPS) → YIELDS → HIGHEST RELIABILITY LEVELS

- PMPCB APPROVES PROCUREMENT SPECS
- PROGRAM DERATING POLICY (CONVENTIONAL AND RADIATION)
- PRIME CONTRACTOR ORDERS AND DISTRIBUTES PARTS TO TEAM MEMBERS
- UNIFORM QUALITY SYSTEM MAINTAINED
- TRACEABILITY CONTROLS ASSURED
- PROGRAM-WIDE ALERT SYSTEM

- 100% PRECAP ON ACTIVES
- CRITICAL ITEM CONTROLS
- LOT QUAL PROGRAMS
- DPA PROGRAMS
- RADIATION TEST PROGRAMS
- DATA PACKAGE REVIEWS
- RECEIVING INSPECTION BY USERS

EVIDENCE OF SUCCESS
- LOW REJECTION RATES ON PARTS
- LOW INCIDENCE OF FAILURES AT HIGHER ASSEMBLY LEVELS

ACE GENERAL PURPOSE PROCESSOR

- DESIGN PRACTICES
  - STRINGENT PARTS DERATING REDUCES STRESS
  - COMPREHENSIVE PARTS SCREENING/TESTING
  - HARDENING FOR NUCLEAR SURVIVABILITY
- CPU APPROACH
  - MICROCODE EMULATOR OF EXISTING MINICOMPUTER PERMITS HIGH-CONFIDENCE IN OBJECT CODE, DIAGNOSTICS, AND SPACECRAFT INTERFACES
  - MEMORY PATCH PERMITS READ/WRITE MEMORY UPDATE VIA COMMAND
  - EXCESS CAPACITY
- REDUNDANCY ARCHITECTURE
  - ALL CRITICAL ELEMENTS REDUNDANT AND CROSS STRAPPED

(2)	ROM	(2)	CLOCKS
(2)	RAM	(2)	THRUSTER CONTROLS
(2)	CPU	(2)	WHEEL DRIVES

TESTING AT ALL LEVELS
YIELDS 3800 HOURS ON
PIECE PARTS AT LIFTOFF

PROBLEM RESOLUTION AND CORRECTIVE ACTION

- PROBLEMS ASSIGNED TO EMQ TEAM
- EMQ TEAM APPLIES OTHER REQUIRED EXPERTISE (CONTRACTOR, SUB-CONTRACTOR, SUPPLIER, AND CUSTOMER PERSONNEL)
  - SYSTEM EFFECTIVENESS
  - FAILURE ANALYSIS
  - MATERIALS AND PROCESSES
  - TEST ENGINEERING
  - DESIGN ENGINEERING
  - PARTS ENGINEERING
- EMQ REPORTS STATUS TO PROGRAM/FUNCTIONAL MANAGEMENT AND CUSTOMER
- PROBLEM RESOLUTION REQUIRES ACTION ITEM LISTS WITH RESPONSIBILITY FOR COMPLETION (WHO AND WHEN)

KEY

CORRECTIVE ACTION PLAN
AND ITS TIMELY IMPLEMENTATION TO PREVENT FUTURE REOCCURRENCES

AND

TIMELY ALERTS
TO OTHER POTENTIAL USERS
- GIDEP
- PROGRAM ALERT SYSTEM
- CUSTOMER ALERT SYSTEM

T W T A'S

- REDUNDANCY IN USE
- PRODUCTION READINESS REVIEWS
- PMPCB APPROVALS REQUIRED
- MRB/CAB/FAB INVOLVEMENT
- MANY MPI'S ON MATERIALS, PROCESSES, ASSEMBLIES, SUB-ASSEMBLIES
- MONTHLY STATUS REVIEWS WITH TWTA SUPPLIER AND CUSTOMER

PLUS

- EXTENSIVE TEST PROGRAM ON TUBE, POWER, SUPPLY, AMPLIFIER
  - INCLUDES TV, TC, VIBRATION AND LONG TERM BURN-IN
  - PERFORMANCE PARAMETERS WITH TIGHTENED LIMITS
  - TEST WITNESSING BY VQA
- TEAM APPROACH TO PROBLEM SOLVING
- PRODUCT DATA BOOK REVIEWS WITH CUSTOMER
- FURTHER TESTS AT HIGHER ASSEMBLY LEVELS
  - TRANSPONDER SUBSYSTEM
  - SPACECRAFT
  - ORBITAL

CRITICALITY OF PRODUCT IS UNDERSTOOD
EMPHASIS ON PARTS, MATERIALS, PROCESSES
AND TESTING...TESTING...TESTING....

SPACE DIVISION

SUMMARY

LONG LIFE REQUIRES:

- GOOD PARTS
- CRITICAL ITEM CONTROLS
- ADEQUATE DESIGN MARGIN
- REDUNDANCY
- TESTING
- EFFECTIVE RESPONSE TO PROBLEMS/FEEDBACK
- TLC THROUGHOUT

THE WHOLE IS THE SUM
OF THE PARTS .....
AND
QUALITY VERIFICATION
NEVER STOPS

A CATALYST FOR GROUND SEGMENT MISSION ASSURANCE

H. Pickover
Product Assurance APM
TDRSS Ground Segment
White Sands Ground Terminal, New Mexico

1. OVERVIEW - This paper describes a trouble or problem control system that combines the familiar Quality Assurance hardware and software discrepancy reporting systems, the equally familiar Material Review, Failure Review and Change Control Boards with a highly visible project management developed Trouble Reporting System. The Mission Assurance organizaton uses the Trouble Reporting System to integrate problem tracking and control methods used by various mission assurance disciplines and relies on it to provide the impetus required for problem resolution. Understandably, project technical management is reluctant to get aggressively involved in the several mission assurance reporting systems when each usually deals only with discipline related details. However, project management shows strong interest in a system that covers all aspects of the project. This strong interest is readily converted to increased cooperation from all project quarters which in turn contributes to better utilization of classical mission assurance techniques. The trouble control sequence is started when a trouble or problem is identified. A computer compatible report is initiated and subsequently validated by the initiator's supervisor. The actual trouble is fixed immediately, if possible, or escalated for further consideration. The initial Trouble Report (TR) serves as a basis for opening the standard mission assurance tracking techniques, i.e., Hardware and Software Discrepancy Reports, Failure Reports, etc., which are tracked separately by the several mission assurance disciplines. Disposition, i.e., closure, assignment for further action, etc., is made by a project controlled review board. Mission Assurance collects status changes and reflects these in a Weekly Status Report. This report forms the basis for a weekly management review of status a consideration not lost to all project elements. For project management, the Trouble Reporting System provides:

(1) Visibility into program problems.

(2) A mechanism for assigning action and tracking problem resolution.

(3) A data base for failure trend and logistics analysis.

(4) Historical documentation of problems encountered during various program phases.

For Mission Assurance, the Trouble Reporting System acts as an effective catalyst for performing required mission assurance tasks.

2. THE WSGT SLIDES -

3. TROUBLE CONTRIBUTORS - Any large program will have its share of troubles and, of course, some form of tracking and closure system is necessary to assure management control of trouble resolution. Complicating the control of troubles that all large programs experience, is the development environment represented by the TDRSS Ground Segment. Figure 1 is a simplified overview of this environment. The Figure also illustrates some of the trouble contributors as well as their mainfestations. Hardware and software development for the Ground Segment was accomplished at separate facilities in California, and Integrations and Test at another location in New Mexico. Each activity was headed by separate managers, each reporting to a strong central manager in California. The advantages of telephone systems not withstanding, geographic separation of related activites can be expected to increase communication and interface difficulties between performing sections. Troubles occurring in each of several locations, unless adequately treated, often develop long moment arms, i.e., become more troublesome in the latter phases of project development.

4. TROUBLE CONTRIBUTORS - (continued) - The low unit hardware volume of the Ground Segment (typically six units of a kind) provides a tendency towards using manual production techniques rather than the automated techniques expected in large volume hardware production. While manual production techniques do not necessarily increase the number of hardware troubles, its

inherent variability does contribute to a higher probability for such increases. TDRSS Ground Segment schedule and fixed price contract considerations resulted in concurrent hardware development and manufacturer, a process that typically encourages engineering and production changes. These changes also tend to increase the probability of troubles. Another contributor to hardware associated troubles, and one that is probably the biggest headache for quality personnel, is the questionable workmanship criterion, Best Commercial Practices. This criterion does not significantly increase the number of hardware troubles, but it is the most difficult to control. It is a chronic source of trouble where each trouble requires an inordinate amount of time to resolve, and even more frustrating, it is rarely resolved in a completely satisfactory manner. This criterion is usually mandated in non-flight equipment (i.e., Ground Stations), presumably for economic reasons, and in the opinion of this writer, tends toward establishing the lowest workmanship standards for a workforce involved in hardware production. In the software arena, the largest contributors relate to the requirement to develop hardware an interfacing software concurrently and to the size and complexity of the software system. TDRSS Ground Segment hardware is software driven and the resultant data is computer processed. Interface troubles between hardware and software are not only numerous in this environment, but continue through the integration and test phases of the project. The software system contains over one million machine language instructions, controls over 300 racks of equipment, and processes several thousand parameters in a system of ten major computers. The TDRSS environment and complexity has contributed to approximately 1900 software trouble reports and 2100 hardware problem reports. This magnitude strongly suggests the necessity for early implementation of an effective project trouble control and tracking system.

5. <u>REVIEW BOARD CONCEPT</u> - The heart of the Trouble Reporting System is the review board concept (sort of an extension of the familiar Change

Control Board) which mandates technical decisions by appropriate experts. This concept is further justified by prudent requirements for non-unilateral consideration of a technical problem, request for changes, etc. Review board activities are initiated by Trouble Reports, Change Requests or other formats concerning project related problems. The TDRSS Ground Segment conducted five kinds of review boards in four structured forums. The four structured boards were identified as:

(1) Material Review Board (MRB) - The MRB made decisions as to whether material identified as being discrepant should be rejected, accepted or conditionally utilized pending other actions.

(2) Failure Review Board (FRB) - The FRB reviewed discrepancy reports to initiate actions that properly dispositioned hardware failure problems.

(3) Software Review Board (SRB) - The SRB was established as a review body to process proposed software changes, evaluate software problems, approve fixes and to assign personnel to resolve software problems.

(4) Change Control Board (CCB) - The CCB was established to review and approve formally proposed changes, deviations and waivers to engineering documentation. Two CCBs were utilized for the Ground Segment, a preliminary body at White Sands, and a program CCB in California where project engineering management was concentrated.

The fifth activity, identified for purposes of this paper as a Management Review Board, was less formally structured and in practice was conducted as part of CCB meetings. This body reviewed trouble reports that concerned problems not covered by other review board activities. Each board consisted of a delegated senior representative from each technical area to the extent necessary to

adequately address and subsequently approve of a problem resolution or change request. Also included were Mission Assurance and Project Management representatives. Each board, with the exception of the MRB, convened on a periodic basis using formal agendas as a control mechanism. MRB, except for complex problems, was handled informally, i.e., Mission Assurance coordinated signature efforts with engineering and customer representatives. The relationship of the various review boards to the overall trouble reporting activity is shown in Figure 2.

6. TROUBLE REPORT PROCEDURE OVERVIEW

Scope - The procedure covers the preparation and flow of Trouble Reports initiation through closure.

Purpose - The purpose of the procedure is to establish consistent trouble reporting techniques for problems encountered during various post critical design review phases of project life.

General - The Trouble Report system provides the method for documentation of troubles encountered and it is the vehicle to obtain corrective action. The system is utilized by all project departments. A TR is initiated for all problems, hardware, software, and facilities which have the potential of impacting performance, the overall schedule or operation, and maintenance. Trouble Reports are initiated and submitted to Mission Assurance for logging and subsequent processing through the appropriate review board. Some of the types of problems that are reported include hardware failures, software incompatibilities, interface problems, design problems which impact performance and interfaces, intermittent equipment malfunctions, broken inspection seals, safety problems, operating problems, etc. Normal hardware maintenance items such as cleaning, lubrication, alignment, adjustment, calibration, etc., are not reported on TRs. TR generation serves as the initial step of the trouble reporting mechanism. In the case of a hardware failure, the problem is transferred to a Test Discrepancy Report by Mission Assurance for subsequent failure analysis and action by the Failure Review Board. Each TDR

is then logged and tracked through resolution but the initial TR is not closed until all related mission assurance reports are closed.

A single TR is used to report a given problem area even though the problem resolution may result in more than one discrepancy being identified. An example of this would be a shorted unit failure which causes a power supply failure. In this case, only one TR will be required with both failures noted. If an intermittent problem recurs before closeout of the original TR, it is noted on the original TR only; i.e., a new TR is not generated. TRs are also generated to address Software through the Software Review Board and assigned to the appropriate section for resolution. Sufficient data, as required by software configuration management procedures are attached to facilitate analysis and resolution.

7. TROUBLE REPORT FLOW - A TR is initiated by anyone observing a trouble, problem, or failure. Flow through the TR system proceeds from the initiator's supervisor to the Mission Assurance office and then to the appropriate review board. The review board reviews each TR and its resolution and assigns closeout action, or will assign investigative responsibilities to an appropriate group or individual. When the assigned action has been completed, it is recorded on the TR and given to the Mission Assurance office for status and for submission to the review board. TRs ready for closure will be reviewed at the next review board meeting. TR status is reported to the Program Manager on a weekly basis.

Specific Responsibilities - The initiator of the TR is responsible for the adequate description of the trouble or problem observed, and for providing such information as is required to support investigation and analysis of the problem. The initiator's supervisor is responsible for verification of the trouble and to assure that supportive information is attached to the report. The initiator's supervisor assigns an expected closure date, and submits the TR to the Mission Assurance Office for assignment of a sequence number. The

supervisor or the appropriate manager will assign action for resolution to an individual in his group. The individual who assigned responsibility for resolution takes the necessary steps to confirm the problem and initiate appropriate corrective action, including any required retest. It is also his responsibility to complete the TR form entries relative to the location and description of the problem, including corrective action, taken or required to be taken upon completion of corrective action, the responsible individual signs the TR and transmits it to his manager. Final closure of the TR shall be determined upon the action of the appropriate review board.

8. Trouble Report Format - A TR format has been developed to facilitate entry of its contents into a word processor. The format provides for the placement of information within designated entry fields to allow for various sorting arrangements during processing. The form also has entry areas for uncoded written information. Figure 3 illustrates a complete TR. This TR has been processed through the cycle and closed by a review board.

Defined Entry Fields - The word processor entry areas of the TR form have column identification numbers above the spaces where data is to be entered as well as a description of the data fields.

Trouble Report Number - Columns 1 through 14 are project assigned identifiers comprised of the date, the initiator's location, badge number and the Mission Assurance assigned sequence number representing the number of TRs written to date. Columns 1 thorugh 5 represent the date the TR is written. Column 6 represents the initiator's normal work location. The last three digits of the initiator's badge number are entered in Columns 7 through 9 followed by the sequence number in Columns 10 through 14. In the event an initiator is the customer, the initiator's initials will be used in place of the designated number.

Criticality - The assignment of the criticality code is the

responsibility of the TR initiator with his supervisor's concurrence. This is a one (1) digit code entered in Column 15. The interpretation of the criticality code is as follows:

a) Emergencys (Code 1) - This code is used when the trouble affects performance and the timeliness of the resolution may significantly affect program schedule.

b) Priority (Code 2) - This code is used when the trouble affects performance but the resolution on the Program schedule is also deemed to be minimal when this code is used.

c) Normal (Code 3) - This code is used when the trouble does not affect performance; i.e., test equipment, operations, procedures, training, etc. The impact of the resolution on the Program schedule is also deemed to be minimal when this code is used.

Description - Columns 16 through 53 are reserved for a keyword description of the fault or trouble which is provided by the TR initiator. The first word of the keyword description represents the trouble item and subsequent words, the type or kind of trouble.

Assignment - Columns 54 through 57 are reserved for the identification of the organization and engineer responsible for resolution. This section is completed by the individual assigning action for resolution. An "S" in Column 54 indicates the responsibility for resolution has been assigned to Space Park. The assignments may be changed by review board action if deemed necessary.

Trouble Code - The trouble code shall be entered in Columns 58 and 59 by the action responsible engineer from a set of preassigned codes. For a hardware failure, a trouble code H is appropriate. In this case, the written information area (analysis section) will contain the part name, part number and serial number of the failed unit. Mission Assurance will

use this TR information to generate the Test Descrepancy Report (TDR) and any other required reports.

Disposition Code - At the completion of corrective action, the action responsible engineer will determine the appropriate code to be entered in Columns 60 and 61 from the codes shown on the TR form.

Status Code - When appropriate, a TR is converted to a TDR or SPR with open status and the TR will remain in that status until the TDR or SPR is closed. After review and sign-off by the review board, the appropriate code is entered in Column 62 to indicate the closure of the TR.

Expected Close Date - The expected close date is entered in Columns 63 through 67 by the supervisor assigned the responsibility for closeout. The review board chairman will review and concur with or change the expected close date entered.

Discrepancy Number - The Mission Assurance representative to the review board assigns a TDR number for hardware failures and a Software Problem Report (SPR) number for software problems. For TDRs, a T prefix is entered in Column 68. For SPRs, an S prefix is entered in Column 68. Their respective numbers are entered in Columns 69 through 72.

Uncoded Written Information - There are two areas reserved for written information on the TR form.

Sample Trouble Report - Figure 3 illustrates a completed TR. This TR has been processed through the cycle and closed by the review board.

CONCLUSIONS

Mission Assurance activities in ground station development has customarily been performed with much less rigor than in the space segment environment. Where the ground station evolves to a highly sophisticated hardware and software state (as in TDRSS) the historical levels of Mission Assurance rigor are simply not adequate to control the magnitude of problems experienced.

TDRSS Ground Segment Mission Assurance, in its quest for increased rigor, has relied heavily on the effective trouble reporting procedure described herein. Without this procedure, it is doubtful that the project would have the level of confidence in trouble resolution as it now enjoys.

Contributors To Problems In The Evolution
Of Ground Segment Development

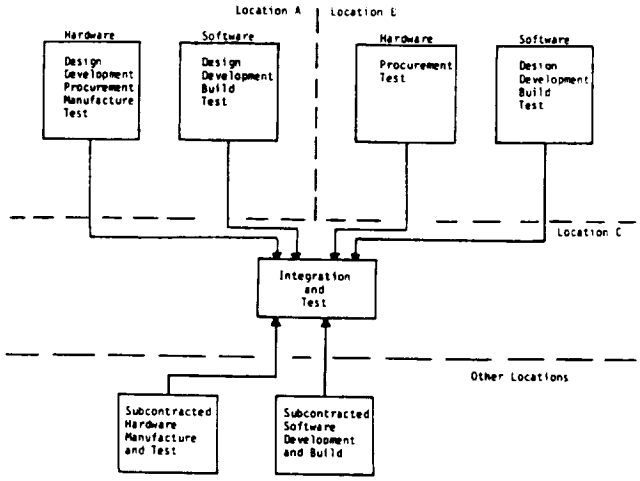
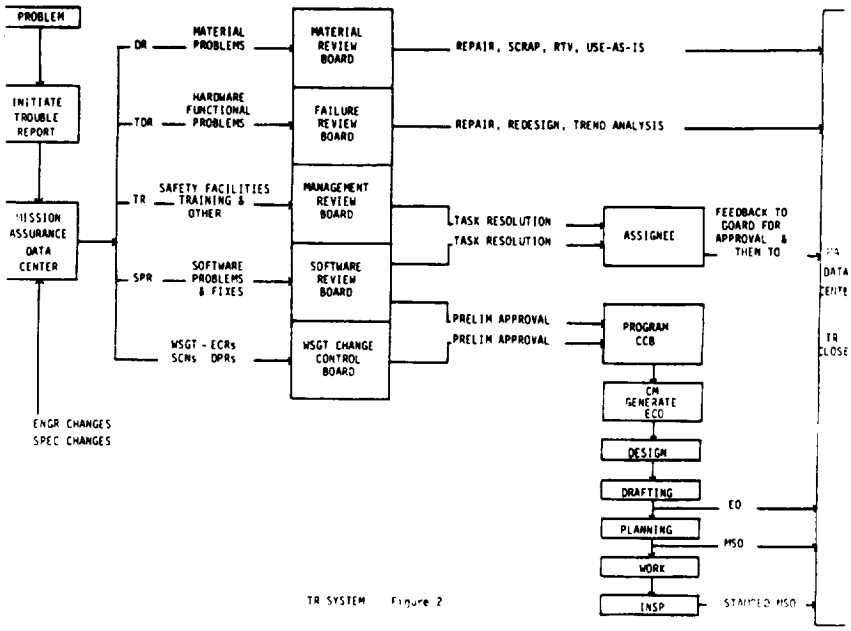


Figure 1



TR SYSTEM Figure 2

TROUBLE REPORT

MO. 03 DAY 16 YR 83

SEQ NO. 3327 CRITICALITY CODE 2

DESCRIPTION: USER IMAG RECEIVER WILL NOT LOCK

CO BADGE T090

INFORMATION: REPORTED BY: T. Sartwell SUPVR: B. O. Smith

TIME	DATE	
1000	3/16/83	PROBLEM OCCURRED
1015	3/16/83	MAINT. - NOTIFIED *
0400	3/17/83	REPAIR COMPLETED *

During GSP 4801 (Routine Verification Test) the MA Receiver for User No. 1 would not lock at Spec. C/N_0 of 48.2 dBHz.

Note: The total power into the receiver was -30.7db and the data rate was 8 Kbps.

ASSIGNED TO: W094 TROUBLE CODE: H01 DISPOSITION CODE: 1 STATUS: 03 EXPECTED CLOSE DATE: 173 DISCREPANCY NUMBER:

CO BADGE

* THESE ITEMS TO BE COMPLETED BY TEST AND COMPUTER OPERATIONS PERSONNEL FOR HARDWARE FAILURES ONLY.

ANALYSIS:

Corrective Action: 31783

1. Input bais and threshold adjusted.
2. User No. 1 verified using Operational Verification Test (SPMAS3) modified for 8 Kbps.
3. No TDR written.

RACK 300
DRAWER A6
SLOT
REPLACEMENT PART
* P/N
S/N
REV
REPLACED PART
S/N

Figure 3

CLOSED BY Signature DATE 3/27/83

APPROVED Signature DATE 3/27/83
APPROVED DATE

LESSONS LEARNED IN SOFTWARE DEVELOPMENT FOR AFSCF'S NASA INTERFACE SUBSYSTEM (NIS)

Gordon L. Wade
NIS Software Development Manager

The NASA Interface Subsystem (NIS) is a new system of the Air Force Satellite Control Facility (AFSCF) data processing segment designed to provide ground support for Space Transportation System (STS) activities. One of USAF's responsibilities in the STS program is to support the Inertial Upper Stage (IUS) missions put in orbit by the Space Shuttle Vehicle. Both vehicles communicate with ground support segments, either directly or via the Tracking and Data Relay Satellite (TDRS of Figure 1). IUS mission operations will partly be conducted by existing AFSCF resources and procedures, and partly by additional data processing resources provided by NIS in a specially configured mission control complex (MCC) within the AFSCF. (Fig. 2)

The requirements, operating concepts, and equipment configuration for NIS evolved over a relatively long period of time. Top level requirements originating in The NIS Functional Requirements Documents included blocking/deblocking of data to/from Goddard Space Flight Center (GSFC) and/or Johnson Space Center (JSC); limit checking, archiving, and display of selected telemetry data from both vehicles; comparison of state vector and attitude data from both vehicles; and support of interactive displays and print capabilities.

Ford Aerospace was selected as the single prime contractor to provide both hardware and software for NIS in a turnkey mode of operation. Economics and timing realities forced a situation where the hardware configuration for both the operational system and a separate dedicated development configuration were well established by the time that a tightly scheduled software development effort was authorized to proceed.

NIS consists of two coupled minicomputers. One, known as Telemetry and Command (TAC), serves as a front-end processor to interface with NASA's NASCOM network. The hardware and a unique framework of operating system software were furnished from an earlier NASA development contract. The TAC Computer Program Configuration Item (CPCI) for NIS consisted of six new NIS applications programs written (of necessity) in assembly language. There are approximately 8000 lines of executable code in the TAC CPCI. The host minicomputer, known as the Decom Interface Controller (DIFC) processor, drives most of the displays, bulk storage, hard copy, and man-machine peripherals, and accomplishes most of the scientific processing. Many new programs were written for the DIFC CPCI using either assembly language or FORTRAN 77. One unique aspect of the DIFC software development was the stipulated adaptation of man-machine interface software furnished by the government from

NASA's Solar Mesospheric Explorer (SME) project. Much of this software had to be significantly modified and extended to satisfy NIS applications and requirements. Another unique aspect was the incorporation of vendor-supplied software for the "downloading" of attached telemetry decommutation equipment (microprocessor- based). There are approximately 70,000 lines of executable code in the DIFC CPCI.

The hardware implementation of NIS included 15 racks of equipment, floor mounted peripherals and desk mounted displays. NIS color displays were installed in the MCC. External interfaces were implemented using fiber optic equipment.

What follows here reflects on the lessons learned in the successful NIS software development that extended over the two-year period from June 1981 through May 1983. Major milestones during that period included many requirement meetings that resulted in a successful System Requirements Review in August 1981, a Preliminary Design Review in December 1981, TAC and DIFC Critical Design Reviews in May and August 1982, respectively, completion of software unit testing in February 1983, and successful support of STS-6 in April 1983 using an interim version of NIS. For discussion purposes, the lessons learned in NIS have been grouped into eight categories.

1. Front-End Preparation. One of the first lessons learned was that front-end planning and good requirements definitions are essential prior to any design effort. Numerous studies and a sequence of 38 technical interchange meetings with the customer and user community were held. These meetings were long and difficult, but resulted in an in-depth understanding and established the requirements for the specification documents. These early efforts provided the necessary foundation for the software architecture. NIS was fortunate in having a strong advocate in the USAF officer ranks to focus these meetings, which led ultimately to a successful formal Systems Requirements Review (a distinctly worthwhile step). Good preparation minimizes later changes to the requirements, and NIS predictably became relatively free of such disruptive, confusing changes in its design, code, and test phases.

An important lesson learned early was to provide the programming staff with formal training on the operating system, the languages, and the hardware & peripherals to be used. Key team members traveled to Colorado for face-to-face orientation on the SME software and operations, and to Florida for a course on the telemetry decommutation equipment. The associated investment is believed to have paid handsome dividends in subsequent performance.

The relatively early availability of a dedicated software development system has already been mentioned. It is an important project advantage for software engineers to be able to get their hands on the hardware at an early stage, if only to get familiar with editors and utilities needed for creation of requirements charts and preliminary design documents.

Finally, a motivated contractor can hardly start too soon in assembling a select software development team. The company assigned the best available people to the job - and in time for adequate preparations to be made.

2. Methodology. NIS software was developed with structured methodology that is evolving (and being documented) within Ford Aerospace and elsewhere. The success of NIS provides added testimony that the method works. The first drafts of two types of document need to be in place very early in the project - the Computer Program Development Plan, and the Software Design and Implementation Standards (so-called in NIS). These important "living documents" should be in the hands of every team member before detailed design starts, and updated thereafter as necessary.

An important element of the methodology is to get the design right before proceeding into coding. In NIS, the use of a program design language (we called it NDL) instead of conventional flow charts helped this process. Being computer-based, the NDL files satisfied some engineers' craving to work on the machine while serving to create a sound, reviewable, readily updated design suitable for direct incorporation into documents. Putting the formal design review charts and specification text on the computer offered many of the same benefits.

The discipline of decomposing the software design into units assignable to individuals, then conducting separate design, code, and unit test walk-throughs on each is definitely worth while. Accompanying this structured development is the valuable Unit Development Folder concept with its complete record of significant activity on the unit. The planning, pacing, status monitoring, communications, training, and general management advantages of the methodolgy were amply demonstrated in NIS.

3. Incremental Integration and Build Testing. The incremental turnover of units to source code control mechanisms, coupled with incremental integration and "BUILD" testing were also

followed to advantage. DIFC software grew in five distinct BUILDs. Early BUILDs served to shake down the BUILD construction process itself (all the way from controlled source code each time via canned job control streams), verify the "backbone" of the architecture, and give the independent test team vital exposure to the software.

Incremental integration and BUILD testing had another benefit. It provided grist for the Software Development Review Board (SDRB) to work on in the form of Software Problem Reports (SPRs). After some early tendencies to take an SPR against their software unit as a personal affront, most software engineers adopted a healthy attitude about this weekly machinery for resolving problems with controlled software in an organized, traceable fashion. Many problems were discovered and resolved at an early date in this manner before larger collections of software became involved.

4. Scheduling. The combination of firm STS program support requirements (i.e., firm delivery dates for NIS software) and a relatively late start for the software development effort caused a very short time span for such development. This was perhaps a blessing in disguise, however, since the associated questions of feasibility forced a planning and scheduling exercise that extended down to individuals and the software units for which they were responsible. The 50 software units formed from the two CPCIs through functional decomposition were laid out on the calendar and generally assigned four milestones each (3 structured walk-throughs plus turnover to the software librarian for configuration control). The 195 milestones thus produced were entered into a computer-based status monitoring and reporting system that served the project very well as a weekly tracking and contingency management tool. The lesson learned is that a similar process should always be performed.

5. The Team. While all the other factors and methods are important, their presence merely allows a software development team to realize its full potential as a group of individuals. NIS was extremely fortunate in its people. Included were three outstanding task leaders, four remarkably gifted subcontractors, and a staff of excellent engineers and programmers. All of the senior personnel were good communicators, teachers, and team workers. Together, their expertise covered every essential technical area. The lesson here is to assemble such a quality team whenever a similar job is to be done.

Organizationally speaking, however, the author feels that such a team has the best chance for success when it can be dedicated to a single project, as was the case in NIS. Minimize the chances for dilution of effort, and the number of times individuals must shift gears to concentrate on the project.

It also helps to have some humorists, some morning types, some evening types, some competitive types, some compulsive helpers, some compulsive teachers, and only a very few with egos near the surface. And it certainly helps to be lucky.

6. Development Resources. NIS was fortunately implemented on a proven, mature set of computers with a relatively capable, robust, stable, and well-supported operating system. The same was true of the languages available for implementation. A software development team has enough problems without continually fighting a flaky, unreliable, poorly documented facility. The difference is very quickly observed and appreciated by insiders, but conveyed to cynical outsiders only with great difficulty.

A dedicated development system is absolutely essential to the success of any large software development effort. The USAF recognized the need for a development system and located it at Ford Aerospace early in the project.

A good, contemporary "programming environment" was actively sought for NIS. This took the form of a UNIX operating system with a rather powerful set of related tools for computer-aided documentation, source code control, briefing chart preparation, graphics, file manipulation, etc. The pre-existence of computer-based graphics tailored to the production of data flow diagrams and structure charts was a key element in the project's success. The only drawback in the case of NIS was the necessity for switching a single processor back and forth on a daily basis from UNIX to the target operating system. Nevertheless, it was better to have had UNIX with this inefficiency than never to have had UNIX at all.

The software development team had an adequate complement of convenient, dependable CRT terminals. In a development effort such as NIS, it seems very clear that the lowest cost solution is for every team member to have his/her own terminal. In fact, in the current economic environment, this author feels that contractors can't even afford to debate the issue.

7. Support. Regardless of how good a team of software developers are, it needs help to succeed on any significant project for today's military aerospace customers. Fortunately, the NIS software team enjoyed such essential support.

A good program management office is essential, not only for its traditional functions, but also for its mitigation of any micro-management tendencies on the part of the customer, and for its constructive comments and criticisms in rehearsals for formal reviews. In the case of NIS, the program management office also satisfied the indisputable requirement for quick reaction to serious facility and hardware performance problems.

NIS successfully avoided the software/hardware personnel contention problem that so frequently emerges to adversely affect progress on system development projects. The mutual understanding and respect that grows from hardware and system engineers knowing about software, and vice versa, is the probable explanation. In any case, NIS software engineers benefited by excellent rapport with the system engineer and various hardware-oriented personnel.

Any team can benefit from the support of a few strategically placed workaholics. NIS happened to have them as the operator of the graphics terminal and packages, the software librarian and Software Development Review Board recorder, and the technical editor for computer-based documentation.

8. Approach. Just below the methodology framework was an implementation approach that yielded numerous lessons.

- Whenever possible, a software development team should acquire its own expertise. It should be as independent as possible in matters of operating systems, interfaced micro-processors furnished software, hardware interfaces, etc.
- Divide the effort as required to get proper granularity, separate and spread responsibilities.
- Promote participative specification writing, design preparation, and design presentation.
- Distribute some of the interesting, challenging, and important tasks to junior personnel.
- Promote a full spectrum of involvement for as many team members as possible (from requirement to fruition).

- Keep task leaders and supervisors technically involved, at least to the extent that they can accurately assess and appreciate the real problems.

- Trust the advice of team experts on technical issues: act accordingly.

- Do participative, ambitious scheduling: prominently display and advertise the resulting plan.

- Set up a highly visible weekly monitoring and reporting cycle.

- Treat software developers as professionals: allow latitude in design and implementation approaches and expressions, balance the methodology rigor with common sense.

- Help make the work rewarding.

- Let the software products speak for themselves whenever possible.

LESSONS LEARNED IN SOFTWARE DEVELOPMENT FOR AFSCF's NASA INTERFACE SUBSYSTEM (NIS)

Gordon L. Wade
NIS Software Development Manager
Satellite Control Facilities Operation

Ford Aerospace & Communications Corporation

NIS OVERVIEW

- New computers to be installed at STC for IUS and DoD spacecraft support
- NASA Interface Subsystem (NIS) to provide
  - Data exchange between NASA and AFSCF communications and computer systems
  - Orbiter data needed by MCC users to conduct real-time STS payload operations
  - Specialized data processing functions related to IUS flight operations
- Complexity of interfaces and security considerations dictate distributed processor configuration

Ford Aerospace & Communications Corporation

STS SUPPORT OVERVIEW (WITH TDRSS, 1983 - 86)

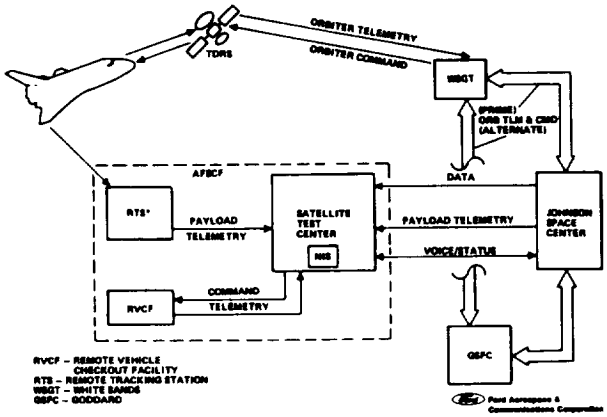


FIGURE 1

NIS OVERVIEW (CONTINUED)

- Pre-established hardware
- Long requirements evolution
- Two coupled minicomputers
  - TAC front end for NASCOM network interface
    GFE hardware and operating system software
    New NIS applications tasks

    DIFC host
    Displays, bulk storage, hard copy, MMI
    Scientific calculations
    Nucleus of MMI software is GFE (SME project)
    Vendor-supplied software downloads telemetry decom equipment

Ford Aerospace & Communications Corporation

STC STS DATA PROCESSING OVERVIEW

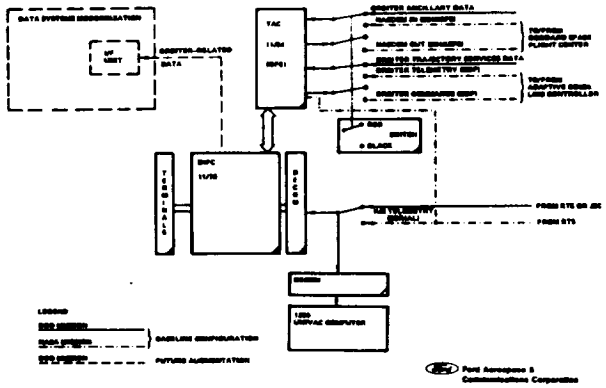


FIGURE 2

FACTORS IN THE NIS SOFTWARE DEVELOPMENT SUCCESS

- Front-end preparation
- Methodology
- Incremental integration and build testing
- Scheduling
- The team
- Development resources
- Support
- Approach

Ford Aerospace & Communications Corporation

FRONT-END PREPARATION

- Reasonably good requirements definition
  - Long and difficult requirements meetings
  - Early TIMs with customer and user community as software design developed
  - Strong project officer in USAF
  - Meaningful SRR
- Specific staff training on target system
- Early access to target hardware for software development
- Select development team

Ford Aerospace & Communications Corporation

INCREMENTAL INTEGRATION AND BUILDS

- Incremental integration
  - Control of source code
- Build construction on development system
- Build testing on target machine
  - Independent test team
- Organized handling of problems with SDRB

Ford Aerospace & Communications Corporation

METHODOLOGY

- Structured methodology
  - Standard Ford Aerospace (Tailored)
- Living CPDP & SDIS documents
  - Unit development folder concept
  - Program design language instead of flowcharts
  - Structured design complete before coding, CDR
- Computer-based design documents
- Walk-throughs: design, code, unit test
- Unit milestones for status monitoring & management

Ford Aerospace & Communications Corporation

THE TEAM

- Results no better than aggregate potential of individuals
  - Outstanding task leaders
  - Gifted senior people - right expertise at right time
  - Good communicators, teachers, teamworkers
  - Great staff of junior people
- Dedicated organizational unit
- Unusual esprit de corps

Ford Aerospace & Communications Corporation

SCHEDULING

- Extremely "success-oriented"
  - 24 month development period
- Detailed planning and scheduling tasks
  - Down to assignable unit level (195 milestones, for 50 units)
  - Weekly status

Ford Aerospace & Communications Corporation

DEVELOPMENT RESOURCES

- Proven equipment
- Excellent development system
  - Target operating system
  - Languages
- UNIX
- Computer based graphics
- Convenient, dependable CRT terminals

Ford Aerospace & Communications Corporation

SUPPORT

- Strong support groups are vital to success
- Program management office
  - Balanced technical direction/latitude in traditional functions
  - Constructive review comments/criticisms
  - Optimal buffering of customer concerns
  - Quick reaction to facility/hardware performance problems
- Good working relationships with hardware & systems engineers
- Workaholics where you need them
  - Graphics terminal operator
  - Software librarian & SDRB recorder
  - Technical editor for computer-based documentation

Ford Aerospace & Communications Corporation

APPROACH

- Acquire own expertise whenever possible
- Division of effort, separate responsibilities
- Promote participative spec writing, design effort
- Distribute interesting, challenging, & important tasks
- Promote a full spectrum of involvement
- Keep task leaders & supervisors technically involved
- Trust team expertise
- Do participative, ambitious scheduling
- Set up highly visible weekly monitoring & reporting
- Treat software developers as professionals
- Make the work rewarding
- Let the software products speak for themselves

Ford Aerospace & Communications Corporation

APPLICATION OF PRE PLANNED PRODUCT IMPROVEMENT (P^3I) TO THE CONSOLIDATED SPACE OPERATIONS CENTER (CSOC) PROGRAM

JAMES E. FORD, Major, USAF
CSOC Aquisition Branch
AF Space Division

DESCRIPTION:

Many DOD program are experiencing substantial delays in placing weapon system in the field. This translates into systems which frequently approach absolence before implementation is complete. A potential solution in many cases is the application of Preplanned Product Improvement (P^3I). P^3I is an acquisition concept which programs resources to accomplish the orderly and cost-effective phased growth of a system's capabilities, utility and operational readiness. Careful application if this initiative to programs which are long-term, subject to need change, require performance growth, or have near-term, operational milestones, can result in significant savings. Prime benefits include: shortening of deployment time; greater control of program funds; system upgrades accomplished in a sequentially planned and budgeted manner; and a reduction of logistical and support problems.

P^3I is not a program itself, it is a management approach to system development which plans for change. This approach uses a "stepped" requirements process in which reduced capability is accepted sooner because the system is planned for subsequent upgrades. A conventional application might be to an aircraft; the initial design would use available technology, but would reflect planning for sophisticated upgrades in terms of future volume, weight and power provisions. Development of new modules or components directly infers the need to identify additional R & D funding up front, and DOD level guidance already supports this need. A broader application of P^3I principles applies to Command, Control, Communications and Computer (C^4) systems. In programs of this nature, the process of pre-planning an incremental or phased growth approach in itself meets the basic intent of P^3I. While funding wedges for R & D of high technology elements might be required, in many instances the incremental process provides ample opportunity to use newer technology while remaining within the state of the art (i.e. no R & D required). In short, the process of pre-planning a program to phase its growth is a basic application of P^3I, and R & D funding wedges may or may not be required. In either case, some cost increase can be expected in the design phase to accommodate the planning process and to incorporated those provisions which allow the flexibility to accommodate changes and/or growth.

Literature on P^3I is still relatively scarce, and specific examples of ongoing applications are rarer yet. Information should be compiled at Product Division level and made readily available to program offices. Air Force, Command and Product Division focal points have been identified, and are listed below:

Air Force	Lt Col Bastien/RDXM/277-3040(AV)
AFSC	Lt Col Hernandez/SDDE/858-3316(AV)
Space Division	Mr. Musser/ALM/833-1966(AV)

IMPLEMENTATION:

To be effective, P^3I must be applied early in a program, and be fully supported and understood by the responsible Program Director. Application of P^3I to a specific program should not be forced upon the Program Director, but must derive from his understanding of what it is, and his belief that it is appropriate for his program. Education is required, and this should take the form of promulgation of Command level policy, followed by Product

EXAMPLE: Application of P^3I to CSOC Program.

The Consolidated Space Operations Center (CSOC) will consolidate operation elements of the Air Force Satellite Control Facility and the National Aeronautics and Space Administration (NASA) Space Shuttle mission control. This provides a redundant capability to perform these essential functions in a separate, secure location (Colorado Springs, Colorado). A new facility is being built to house these primary operational capabilities, and has been designed to allow growth for future missions. The CSOC will have a complete Communications capability which will connect it to the existing NASA and Air Force telemetry, tracking and commanding networks.

CSOC has been designed for implementation in operational increments. These are designated Incremental Support Capabilities (ISC's), as shown below:

ISC 1:	Backup Mission Control Complex (MCC)
ISC 2:	1st MCC
ISC 3:	2nd MCC
ISC 4:	3rd MCC
ISC 5:	Shuttle Mission Planning capability
ISC 6:	Shuttle Mission Readiness capability
ISC 7:	Shuttle Mission Control capability

These ISC's are time phased, beginning with ISC 1 in 1985 and culminating in ISC 7 in 1990. This phasing of operational capability in itself represents an application of P^3I, but for CSOC a more specific application can be described; communications. The communications capabilities of the system are known, but the requirements must be implemented over a long period of time and are subject to upgrading.

The contract approach for CSOC Communications is to implement P^3I as part of the basic design process. The contractor is required to design a communication system which meets the total known operational need (through ISC 7), plus provide flexibility for additional future growth. The design must then accommodate the known operational increments and be installable, testable, and usable at each ISC. A key design goal is modularity, such that each subsequent installation of additional communications capacity does not disrupt ongoing operations. Yet another provision of this process calls for a design which can meet requirements with currently available equipment, but which has provisions for converting to more advanced equipment for later ISC's. A specific area which is being considered is to implement early ISC's using conventional wiring technology but allowing for possible future use of fiber optics.

Implementation of P^3I to CSOC communications began with the system specificaiton where growth, modularity and capacity requirements were stated. These are functional requirements, so the contractor still retains the design latitude to meet the requirements in his own way. Continuous emphasis and evaluation of contractor planning for P^3I is accomplished through specific contracting mechanisms. The proposal has a separate section which addressed P^3I, and all technical factors were evaluated for consistency with the P^3I approach described. In this way, P^3I considerations played an important role in the selection of the Communications Contractor.

The Statement of Work calls for specific analysis of P^3I during the design process, and every technical review specified by MIL-STD-1521A requires an assessment of P^3I implementation. In this way, the entire development process has been tailored to recognize P^3I imperatives, including adherence to a funding profile which is staggered to be consistent with the ISC's. Costs to support upgrades are also identified as a early part of the contractual effort, and the Program Office has the opportunity to assess the validity of the enhancement and seek funding wedges while the program is still in the development stage.

Other contractual mechanisms used to emphasize P^3I on CSOC communication include: a testing program geared to accept usable increments of communications capability to support a given ISC; application of Award Fee provisions which allow subjective evaluation of contractor success at P^3I implementation; and frequent reference to lessons learned from other P^3I programs as evaluated by the Program Office supported by P^3I experts (most notably the expertise available at DSMC). In summary, P^3I has been emphasized, planned for and is measurable in its application to the CSOC program, and especially to CSOC Communications.

Divison guidance. Local guidance should include provisions for screening all ongoing programs and identifying potential candidate programs. Programs identified must make their own assessment of the value to be derived from P^3I application, and the decision to apply must receive Command level support, particularly in obtaining revised program direction and early budget emphasis as required.

The application of upgrades at pre-planned points in time requires the Program Director to develop an acquisition strategy that addresses P^3I early in the system development process. Drafters of PMD's must address the applicability of P^3I, and specific contracting approaches must follow. Some basic elements of a typical approach would include:

AREA	INTENT
Acquisition Plan	- address P^3I in a separate section - assess applicability/impact
Specification(s)	- phased requirements - growth provisions - emphasis on modularity and capacity
RFP/Proposals	- evaluate P^3I during source selection
Contractual Tasks	- develop P^3I design criteria - evaluate during reviews - assess at development testing - build to increments/phased upgrades
Incentives	- P^3I incentive applied - continuous emphasis and appropriate award of special incentives (i.e. Award Fee)
Expert Assistance	- AF, Command and Product Division focal points - Defense Systems Management College - Industry
Lessons Learned	- other P^3I programs

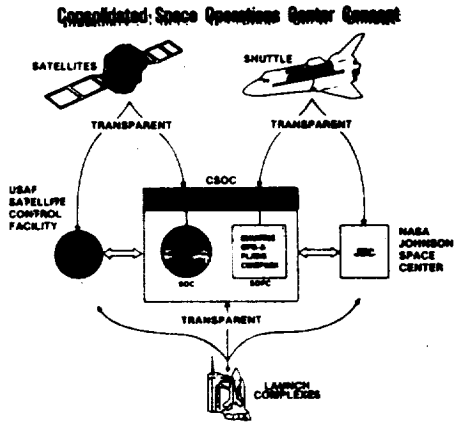
Program Directors must identify funding impacts, if any, early. Designation of a program for P^3I applicability requires Product Division, Command and Air Force support in the budget cycle for funding wedges so identified.

CSOC P^3I

APPLICATION

OVERVIEW

- CSOC PROGRAM DESCRIPTION
- P^3I GENERIC DESCRIPTION/GUIDANCE
- P^3I TAILORED APPLICATION TO CSOC

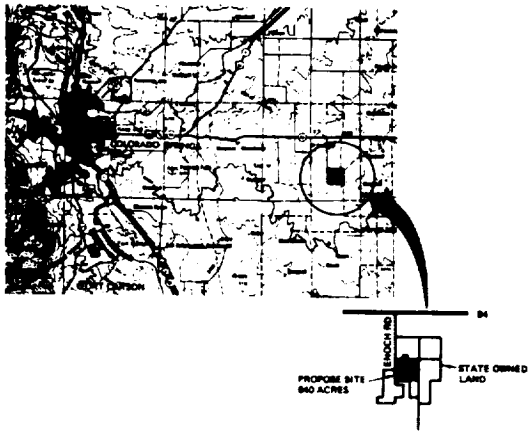


PRE-PLANNED PRODUCT IMPROVEMENT (P^3I)

"P^3I IS AN ACQUISITION CONCEPT WHICH PROGRAMS RESOURCES TO ACCOMPLISH THE ORDERLY AND COST EFFECTIVE PHASED GROWTH OF A SYSTEM'S CAPABILITIES, UTILITY AND OPERATIONAL READINESS."

--DEPSECDEF MEMO, 6 JUL 91

CONSOLIDATED SPACE OPERATIONS CENTER SITE



PROBLEM

SUBSTANTIAL DELAYS IN PLACING SYSTEMS IN THE FIELD WHICH MEANS SYSTEMS ARE FREQUENTLY APPROACHING OBSOLESCENCE BEFORE THE COMPLETION OF PRODUCTION RUNS.

SOLUTION: P^3I

PLAN AND BUDGET FOR IMPROVEMENTS IN ADVANCE OF DETAILED KNOWLEDGE OF CRITICAL SYSTEM OR COMPONENT CHANGES.

SOLUTION RESULTS

- THE ACQUISITION AND DEPLOYMENT TIME IS SHORTENED.
- GREATER CONTROL OF PROGRAM FUNDS IS EXERCISED.
- SYSTEM UPGRADE IS ACCOMPLISHED IN A SEQUENTIALLY PLANNED AND BUDGETED MANNER.
- REDUCED LOGISTICAL AND SUPPORT PROBLEMS.

CSOC P^3I STRATEGY

- o ACQUISITION PLAN
- o RFP/PROPOSALS
- o CONTRACTUAL TASKS
  - o DESIGN CRITERIA
  - o REVIEWS
  - o DEVELOPMENT TEST
  - o INCREMENTAL IMPLEMENTATION
- o INCENTIVES
- o SPECIFICATION CAPACITY REQUIREMENT
- o SOURCE SELECTION CRITERIA
- o EXPERT ASSISTANCE
  - o DSMC
  - o INDUSTRY
- o LESSONS LEARNED
  - o OTHER SPO'S

DEPSECDEF P^3I POLICY

- o ESTABLISH ORGANIZATIONAL FOCAL POINTS
- o EXAMINE MAJOR PROGRAMS FOR APPLICATIONS
- o PROMULGATED P^3I IMPLEMENTATION PLAN
- o REFLECT P^3I IN PROGRAM ACQUISITION STRATEGIES

CSOC P^3I STRATEGY CONTRACT SPECIFIC

FACILITIES → MEETS MOST STRATEGY OBJECTIVES

SOC → DSN DEPENDENT/MEETS SOME OBJECTIVES (PROCESSORS)

SOPC → STRONG NASA IMPACT/INTEGRABLE TO MEET ALL OBJECTIVES

COMM → LEAD CSOC EQUIPMENT ACQUISITION
MEETS ALL STRATEGY OBJECTIVES

SECURITY → MEETS SOME OBJECTIVES

TRAINING → STRONG APPLICABILITY

P^3I APPLICATION CRITERIA

- o LONG TERM SYSTEM
- o THREAT OR NEED LIKELY TO CHANGE
- o TECHNICAL PERFORMANCE MUST GROW WITH TIME
- o NEED FOR NEAR TERM, PARTIAL CAPABILITY
- → o SERVICE COMMITS TO HIGHER INITIAL FUNDING

- o P^3I APPROPRIATE AND BENEFICIAL FOR CSOC1

CSOC COMMUNICATIONS

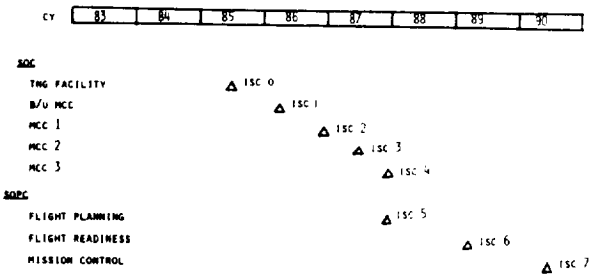
- o SCOPE
  - o TOTAL COMMUNICATIONS PACKAGE FOR CSOC
- o BUILDS COMM SYSTEM TO "INCREMENTAL SUPPORT CAPABILITIES"
- o TWO PHASE ACQUISITION APPROACH
  - o PHASE ONE
    - COMPETITIVE CONCEPT DEVELOPMENT
    - PRELIMINARY SYSTEM DESIGN
  - o PHASE TWO
    - SELECT BEST PHASE ONE DESIGN
    - BUILD, ACTIVATE AND TEST

PRIMARY P^3I THRUSTS

- o TECHNOLOGY
  - o R&D FUNDING WEDGES
  - o NEW TECHNOLOGY FITTED LATER
- o SYSTEM GROWTH PROVISIONS
  - o INCREMENTAL CAPABILITIES/PHASING
  - o DESIGN MODULARITY
  - o OBSOLESCENCE AVOIDANCE
  - o ADAPTIVE TO UPGRADING/CHANGE/EVOLUTION
- o ACQUISITION STRATEGY EMPHASIS
  - o APPLIED EARLY AND THROUGHOUT DEVELOPMENT CYCLE
  - o DETAILED CONTRACTUAL IMPLEMENTATION
  - o SPECIFIC INCENTIVES

CSOC

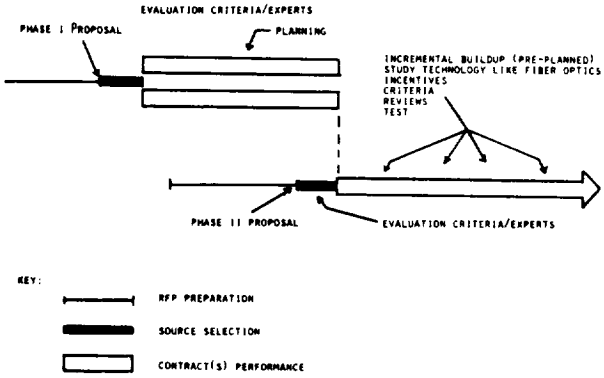
INCREMENTAL SUPPORT CAPABILITIES



REFERENCES

- DOD-D-5000.1 "MAJOR SYSTEMS ACQUISITIONS"
- DEPSECDEF MEMO "IMPROVING THE ACQUISITION PROCESS THROUGH P^3I"
- P^3I IMPLEMENTATION PLAN
- DOD ACQUISITION IMPROVEMENT PROGRAM (DSMC)
- ACTION 2 "P^3I" (DSMC)
- P^3I : DO WE REALLY UNDERSTAND IT? (AFISC - TAKEN FROM TIG BRIEF)
- P^3I (DR HY LYON - TEXAS INSTRUMENTS, INC)
- THE P^3I INITIATIVE - IDEALISM VS REALITY (DR LYON - T.I INC)
- AMERICAN DEFENSE PREPAREDNESS ASSOCIATION SEMINAR AND WORKSHOPS "P^3I FOR MAJOR WEAPONS SYSTEMS ACQUISITIONS"
- CONCEPTS "A CULTURAL CHANGE : P^3I" (LTC MORROW, USA - SUMMER 82)

COMMUNICATIONS CONTRACT P^3I SPECIFICS



S U M M A R Y

- GIVE P^3I A GENUINE LOOK
- TAILOR P^3I TO MAKE IT YOUR VERSION
- P^3I IS MUCH MORE THAN UP FRONT R & D (RESEARCH & FUNDING)
  - INCREMENTAL OR PHASED APPROACH IS P^3I BY ITSELF
  - APPLICATION/EMPHASIS THROUGHOUT ACQUISITION PROCESS IS THE KEY
- P^3I CURRENTLY OFFERS CREATIVE OPPORTUNITIES
  - MODEST AMOUNT OF LITERATURE
  - VERY LITTLE DOD GUIDANCE
  - DO YOUR OWN THING!

PAGE 48 INTENTIONALLY BLANK

MISSION ASSURANCE IN DSM

Richard W. Cosgrove, IBM
George T. Kelton, IBM

I. INTRODUCTION

The Command and Control Segment (CCS) of the Air Force Satellite Control Facility (AFSCF) is being modernized under a project known as Data System Modernization (DSM). The size and complexity of the system to be installed, as well as the requirement to transition while maintaining system operations, demands a development technique which provides constant customer-contractor interaction from the development of system requirements through acceptance of the system. Detailed test planning through a concept of requirements allocation with traceability to the lowest level of development and test is the key to mission assurance in DSM.

Satellite Control Facility Background
Foil #1

The AFSCF was established in the 1960's to support national space test programs. It provides services for telemetry, tracking and commanding of DOD, non-DOD, and occasionally space vehicles of other nations. In support of external users, it provides data processing, distribution and routing services as agreed upon. It is responsible for management of design, operation and maintenance of the worldwide network.

Why Modernize?
Foil #2

The changing roles and missions of space programs from that of space tests in the 1960's to space operations, as well as advances in satellite technology, have placed ever increasing support requirements on the AFSCF network.

This increased network loading requires increased availability of the Remote Tracking Station (RTS) resources to perform telemetry, tracking and control functions in addition to increased data processing capability for space vehicle contact support. Increased work load, coupled with an objective to reduce operating costs dictates a modernization of the data system architecture and an upgrading of the data processing hardware and software.

II. AFSCF DESCRIPTION

In order to understand the size and complexity of the Data System Modernization program, a brief description of the existing SCF system is necessary.

The AFSCF System
Foil #3

The AFSCF is a worldwide, highly distributed facility which is divided into five major segments. The Facilities Segment provides the physical, electrical and environmental elements that are required at the Remote Tracking Stations and the Satellite Test Center. The Support Segment includes the timing equipment located at the RTSs and the STC. The major function of the Communications Segment is to provide communications between all other segments of the AFSCF system, NASA, and other external users. This includes communications security and data recording functions for the AFSCF. The Telemetry, Tracking and Command (TT&C) Segment is comprised of antennas, receivers, transmitters, and signal conditioning equipment which perform pre-launch satellite checkout, acquisition and tracking of satellites, reception of telemetry and ranging data, transmission of commands to the satellite, and transmission to the Satellite Test Center (STC) of telemetry and ranging data via the Communications Segment. Lastely, the Command and Control Segment, which is the segment addressed by DSM, contains the data processing, display, control, distribution and interface hardware and software necessary for Mission Control Complex (MCC) operations, range planning and scheduling, range resource configuration control, and the overall management of the AFSCF.

AFSCF Network
Foil #4

The global operations of the AFSCF network are comprised of twelve antennas at seven Remote Tracking Stations. These facilities are located at Vandenberg AFB, Greenland, United Kingdom, Guam, Hawaii, New Hampshire, and in the Indian Ocean, the Remote Vehicle Checkout Facility (RVCF) at Cape Kennedy AFS, plus the Satellite Test Center at Sunnyvale, California

PAGE 50 INTENTIONALLY BLANK

AFSCF Hardware/Operations Today
Foil #5

At the STC, program office and AFSCF support personnel plan, integrate, schedule and control the total network for multiple satellite support. Each program has a support team which operates from an assigned MCC at the STC as if it were the only program using the network.

Pass planning is developed on the Flight Support Computers, mainly based on data obtained during the last contact with the same satellite, with output being placed on transfer tapes. These plans include the development of telemetry modes, schedules, and tables; orbit determination and ephermeris generation, including antenna pointing data; organization of station instructions; and the development of vehicle commands. The transfer tapes are then hand carried to the Emulator Buffer Computer (EBC) for transmission to the RTS during the pre-pass contact support phase.

The function of the EBC is to provide realtime data support during all contact phases. During pre-pass, data from the transfer tapes is transmitted to the RTS designated to support the pass. During the pass, realtime telemetry, tracking and commanding data is received from the RTS. Telemetry messages are unpacked and formatted for output to MCC printers and displays, along with tracking and commanding information. Post-pass activities include receiving additional data as well as the processing of that data.

The network interface communications computers provide formatting and protocol links to computers at other agencies such as NASA, Eastern Test Range and Space Division in El Segundo.

The Controls and Display subsystem, including the C and D Computer, provides the man-machine interface via consoles containing display monitors and keyboards. These consoles are utilized by mission control personnel to monitor pass operations and to provide command and control capability through voice links with the RTS.

At the RTS, the RTS military Tactical computer (mTc) receives pre-pass data and conducts data transfer tests. During pre-acquisition, the mTc transfers pre-pass data from disk to core and copies the entire contents of core to disk for use in recovery operations in the event of a pass failure. The mTc then passes pointing data to the antenna and, following acquisition, processes the tracking data received from the antenna. Commands are sent from the computer by the operator at the Station Operator Console, who is in voice contact with the Mission Controller in the MCC, and then the commands are monitored on the downlink for verification. In addition, the mTc processes a subset of the downlink telemetry, records TT&C data and retransmits any requested "missed" data to the EBC.

III. DSM DESCRIPTION

Data System Modernization
Foil #6

One of a number of major improvement activities within the AFSCF, the Data System Modernization program is a full scale engineering development effort to upgrade the AFSCF's data system.

DSM will:

a. Centralize necessary data processing capabilities at Sunnyvale AFS by eliminating the data processing capability, command and control equipment and associated operations and maintenance personnel at the Remote Tracking Stations (RTSs).

b. Provide direct command and control capability from the MCCs through a new control and display system, thereby reducing the staff required in the computer areas.

c. Provide fourth generation automatic data processing equipment together with upgraded and modern software.

d. Convert to new output displays, thereby conserving paper resources and reducing data storage costs.

e. Provide increased capacity for the currently defined requirements and growth for future requirements.

DSM Hardware Description
Foil #7

Virtually the entire Command and Control Segment is replaced by DSM. The CCS is comprised of a new Data Distribution System and 10 separate computer complexes; eight Mission Control Complexes (MCC), the Range Control Complex (RCC), and the System Development and Test Laboratory (SDTL) at the STC, and

Remote Tracking Station/Remote Vehicle Checkout Facility Control and Status Equipment (RCSE) at the RTS s and the RVCF.

Each of the computer complexes is comprised of commercially available, state-of-the-art systems consisting of dual mainframes with shared peripherals and several redundant smaller computers which are used as front end processors. The systems are configured in a "kernel" architecture, which is identical in each of the complexes.

This "kernel" architecture is a symmetrical configuration that concurrently supports both the high priority task of realtime spacecraft contact support and lower priority planning and evaluation. The planning and evaluation processor can be switched to the higher priority realtime contact task within 85 seconds, thus achieving a 0.9994 contact support mission reliability. The kernel CPUs share dual IBM Series/1 computers, which serve as front end processors for telemetry data, a shared disk system averaging 5.5 gigabytes of storage per MCC, six additional Series/1 computers linking the system to the AFSCF network, and other peripherals as needed, such as printers, tape units and displays. Most equipment is supplied by IBM commercial divisions except for the Sanders displays and the Harris Contact Support Equipment Group (CSEG) units. Variations in MCCs are accommodated by the addition of hardware and by the use of an appropriately sized CPU. The RCC, utilizing similar equipment, manages the range resources and controls RTS/MCC connectivity within the STC.

Operations in DSM are summarized as follows. Pass planning will be developed on the Planning and Evaluation Processor. Concurrently, and utilizing shared peripheral resources, real-time contact support will be conducted on the Contact Support Processor. Ground control directives and vehicle commands are routed through the front end interface units, which provide interfacing and protocol and apply appropriate security measure to the Data Routing Group (DRG). Connectivity with the desired RTS resource is provided by DRG switching which is under control of the Range Control Comples (RCC). Data received by the RCSE is processed and acted upon accordingly. This includes the control of the antenna as well as providing the timed release of commands to the space vehicle.

On the return link, in normal operations the RCSE passes gathered antenna data and receiver status, as well as selected command echo data, to the controlling MCC for real time display and interaction. This same data is passed to the RCC via the back-up narrowband link for its role of monitoring range resources. Telemetry data is passed via the wideband link directly to the MCC for health and status monitoring and user dissemination.

DSM Software
Foil #8

The benefits provided by a common hardware architecture are continued in the software development approach. Common software will be developed for use by each of the 10 complexes which will be augmented by software for each MCC.

Approximately 6.5 million source lines of code (SLOC) will be delivered by the DSM program, of which about 5.0 million consist of IBM licensed programs and IBM OS/MVS. Of the remaining 1.5 million SLOC, about 20% can be lifted directly or lifted and modified from IBM Federal Systems Division products. The remaining new code will be developed, mainly utilizing the Jovial J73 language, as separate software modules which are referred to as Computer Program Configuration Items (CPCI). The CPCIs are the result of subdividing system requirements into logical, functional partitions, which allows somewhat independent development of pieces of the system.

IV. DSM IMPLEMENTATION

Foil #9

As in the development of any product, the goal of the DSM organization is to provide a product which satisfies the stated requirements and ensures satisfactory performance in the application of that product. To this end, IBM has sought out subcontractors and employees who have experience in previous space related efforts and established a program organization with total performance responsibility and effective control over the resources needed to satisfy system requirements.

The methodology developed to assure mission performance in DSM is one which has dissected the system requirements into manageable units to

allow modular hardware and software design by the systems development organization as well as to provide the test organization with the capability to verify each of these requirements at various levels of test.

Within DSM, the Test and Integration (T&I)department is organizationally independent of the System Development department. This provides the necessary authority to function as a pseudo-customer.

T&I will utilize several resources in the performance of its task, one of which is significant enough to warrant at least a brief description. The Systems Development Laboratory (SDL), located in Gaithersburg, Md. is the primary facility for the development, integration and verification of DSM. It employs a characteristic set of DSM hardware augmented by emulation and driver equipments which provide the capability to simulate the AFSCF environment.

V. REQUIREMENTS

Foil #10

The first step in the development of any system is the generation of system level requirements. The second step is to allocate these requirements into logical design units. These units are called configuration items (CIs). The third step is to develop a detailed specification for each configuration item that explicitly delineates how each requirement is met by the design to be implemented.

The approach taken in DSM was to develop increasingly lower levels of documentation that provide allocation and traceability of detailed customer requirements from initial specifications to individual CI/CPCI design specifications. Each level of documentation also contains a requirements verification matrix that provides upward traceability as to where in the test cycle each requirement will be verified for and by the customer.

System Level Requirements
Foils #11,12

An extensive effort was undertaken by the customer user community that resulted in detailed system requirements specifications for DSM. These requirements form the foundation for CCS design and development.

The AFSCF Specification for the Command and Control Segment ("A" Specification) was then derived from the user defined detailed requirements specifications. This document established the performance, design, development, and test requirements for the Command and Control Segment. The intent of this document is to provide:

a. A concise functional description of the CCS.

b. Functional CCS requirements.

c. A definition of interfaces between the CCS and the other AFSCF segments.

d. A specification of military standards, directives, and studies that pertain to the design, development, and implementation of this segment.

An appendix to the "A" Specification contains CCS work load requirements that provide a defined load based on a set of activities that must be supported by the CCS. It also provides a quantitative basis for the defined load in terms of such activities as: maximum allowable run time, maximum storage, and minimum response time.

Within the "A" Specification, requirements are traceable from system functional requirements to CCS functional area and/or configuration item (CI/CPCI) design characteristics.

Requirements Allocation
Foils #13,14

The requirements delineated in the "A" Specification were allocated by the DSM System Engineering organization to individual CIs/CPCIs using two documents:

The Requirements Allocation and Traceability Matrix (RA&TM) provides traceability of requirements from systems level documentation and allocates these requirements to lower level specifications. Every requirement is traceable to a parent requirement in the "A" Specification. The purpose of this document is to verify that all requirements found in or derived from the CCS requirements documents have been included in the "A" specification and that these requirements have been allocated properly to lower-level specifications.

The Performance Requirements Allocation Matrix (PRAM) partitions and allocates performance requirements and resources needed to fulfill the

functional requirements described in the "A" Specification to the Configuration Items (CIs) and Computer Program Configuration Items (CPCIs)

CI/CPCI
Program Development Specificaitons
("B" Specifications)
Foil #15,16

"B" Specifications are used to state the requirements for the development of individual CIs/CPCIs. Each "B" Specification must be in sufficient detail to describe effectively the functional characteristics that each CI/CPCI must achieve. It contains a requirements traceability matrix that traces requirements from the "A" Specification and a requirements verification matrix describing how and when each requirement will be verified.

CI/CPCI
Program Product Specifications
("C" Specifications)

The "C" Specification for each CI/CPCI establishes the complete identification of that CI/CPCI including: requirements, storage allocation, detailed design, and method of qualification. It also contains a requirements allocation matrix and a requirements verification matrix.

VI. TEST

Foil #17

The primary goal of mission assurance will be achieved by the contractor in a series of tests called Development Test and Evaluation (DT&E) which are designed to first verify and then demonstrate to the customer, each individual requirement detailed in the "A" Specification. DT&E is preceded by development tests used to evaluate algorithms and design alternatives. However, it is in DT&E that the capabilities to meet system requirements and perform mission operations are verified. Therefore it is the DT&E program that will be discussed here.

The DSM program test philosophy, including the goals to be achieved and the method used to ensure that achievement is described first. Next, the implementation of the test program to accomplish these goals is discussed. This requires a definition of the various test levels and how they relate to the accomplishment of the overall goals.

In this discussion special emphasis will be placed on formal CCS system testing, since this is where the final contractor demonstration of mission assurance is made.

Following the DT&E test phase, individual STC Vehicle Operations (VOs) will conduct a series of independent mission assurance tests. This phase of testing is called Operations Test and Evaluation (OT&E).

Philosophy
Foil #18

The CCS test philosophy can be described by delineating the goals to be accomplished and the method used to ensure compliance.

The primary goals of the CCS test structure can be summarized in three key points:

a. Verify and demonstrate that the CCS system adheres to the letter of the requirements.

b. Verify and demonstrate that the implemented design of the CIs/CPCIs fully adheres to the approved design specification.

c. Assure that the final product performs in accordance with the operational expectations for both normal and abnormal conditions.

Foil #19

The method used to ensure compliance has been designed to allow evaluation and verification of the CCS as it evolves and demonstration of capabilities at the system test level. For the purposes of this discussion verification means the evaluation of a requirement for an envelope of conditions including: nominal, limit, and error conditions.

Demonstration means the retest of a previously verified requirement in a total system configuration to demonstrate system operability and performance. A demonstration of a requirement will not retest the full envelope of conditions, but will normally be limited to a nominal condition. The method used to verify requirements as the CCS evolves is as follows:

a. A "shall" is the lowest level of delineation of a CCS requirement. It is interpreted as "you shall do the following." In the "A" Specification, shalls have been identified down to the partial sentence level.

b. The Verification Cross Reference Matrix (VCRM) is used to map system requirements (shalls) to test verification requirements. There is a VCRM in the "A" specification and in all "B" and "C" specifications. This matrix identifies the requirements (shalls), at what level of testing the requirement is to be verified, and the method used to verify the requirement.

c. Shall verification testing has been planned so that the verification will take place as early as feasible in the evolution of the CCS. This gives early insight into the development of the CCS. This plan is documented by the various VCRMs. A shall is verified by conducting tests of nominal, limit, and error conditions. These tests will verify that the requirement has been fully and properly implemented.

d. System level test methodology allows for the testing of each system shall at the earliest feasible point in the evolution of the CCS. It must be recognized, however, that some shalls require a complete system before evaluation can take place. Additional testing is planned that will retest many of the previously verified shalls in a full system environment. This technique ensures that the stated goals of CCS testing can be fulfilled while allowing for problem diagnosis and resolution throughout the test cycle.

Implementation
Foil #20

The CCS DT&E program has been implemented by dividing it into four levels designed to verify and obtain customer acceptance of assigned shalls at the earliest feasible point in the development cycle. The first three levels are contractor conducted test activities. These include component tests, CI/CPCI compliance tests, and initial system tests. The fourth level is a customer conducted DT&E systems tests. After the four levels have been described, initial formal systems testing will be described in more detail. This activity has been singled out because it is the final verification of shalls conducted by the contractor.

Foil #21

Each DT&E level is concerned with all the goals of testing, but concentrates on one or more of those goals as its primary objective. The description of each DT&E level is subdivided into individual test programs designed to most effectively meet its primary goals. The description of each level and its associated test program will include the primary goals of the test, the CCS elements to be tested, who is responsible for planning, execution, evaluation, witnessing, and acceptance, the relationship of this test to other tests and levels, and the location of test execution.

Foil #22,23

a. The primary goals of component testing are to ensure that the implementation correctly reflects the design specificaitons. This initial level of testing is designed to verify individual hardware/software components. These tests include Internal Readiness Tests (IRTs) and Preliminary Qualification Tests (PQTs) for software; and Component Tests (COTs) for hardware.

Software IRTs are internal development organization tests on increments of a CPCI prior to the start of formal shall verification in a PQT. Once the software has passed an IRT it is made available to the Test and Integration organization for informal initial CI/CPCI integration testing. There is no formal shall verification during this test activity. The T&I organization generates the test plans and conducts the tests. The development organization evaluates the results and both organizations determine successful completion of the test. IRT test plans are made available to the customer for review on an informal basis to provide early feedback of customer requirements and concerns. ALL IRTs are executed in the SDL.

Foil #24

Software PQTs are conducted in the SDL on software functions that are critical to the CPCI. These tests are used to verify shalls mapped to individual CPCIs. A successfully completed CPCI PQT is the basis of acceptance of a CPCI by the T&I organization for formal system testing. PQT planning and execution is the reponsibility of the T&I organization. But, since PQTs are designated as formal shall acceptance tests, the contractor must submit a separate test plan/procedure for each CPCI to the customer. The customer must approve PQT test plans prior to the start of testing. Both organizations evaluate the results, but, the customer has final determination of a successfully completed test.

Hardware COTs are internal development engineering tests on hardware components prior to assembly into a configuration item. Tests are conducted on subcontractor premises or at the SDL. The System Development Organization is responsible for planning, execution, and evaluation of the tests. The T&I organization reviews the test plans and may informally witness the test.

Foil #25

b. The primary goals of CI/CPCI compliance tests are to demonstrate to the customer that CI/CPCI implementations correctly reflect their B specifications. Compliance tests are defined here to include Formal Qualifications Tests (FQTs) of software and Compliance Tests (CLTs) for hardware.

Foil #26

Software FQTs are conducted on each CPCI immediately after it has successfully completed PQT testing. It is a complete test of the CPCI in a continuous test period prior to functional configuration audit. All FQTs are run in the SDL. The T&I organization is responsible for planning, executing and evaluating FQTs. The customer approves the test plans prior to the start of testing, witnesses the test, and determines successful completion of the test.

Hardware CLTs are conducted on CI's to demonstrate to the T&I organization and the customer that the hardware is in compliance with CI requirements specifications. These tests are planned and executed by the development organization based on plans previously reviewed and approved by the T&I organization and the customer. Both the T&I organization and the customer witness the test and participate in its evaluation. The customer determines when the test has been successfully completed. These tests are conducted either at a subcontractor facility or in the SDL.

Foil #27

c. Initial system tests are conducted by the contractor to ensure that the CCS adheres to the customer requirements as delineated in the "A" and "B" specifications, and to ensure that the final product performs in accordance with the customer's operational expectations.

Integration tests of CIs/CPCIs are defined to be preliminary tests of CPCIs or portions of CPCIs and associated CIs that contain increments of natural or logical entities (i.e. telemetry, tracking). The objectives of these tests are to verify data flow through the evolving system and to allow early detection of interface problems across CIs/CPCIs. These tests will be performed using software increments that have passed IRT and hardware components that have passed CLT. The T&I organization has total responsibility for these activities which are conducted in the SDL. Since no shalls are formally verified in these tests, customer approval of the test plans is not required.

Foil #28

The first formal systems level tests are called Initial Development Test and Evaluation (I-DT&E) which are conducted in the SDL. An I-DT&E is conducted for each operational configuration to be delivered to the STC. For these tests the SDL will be configured as near to the operational configuration and environment to be verified as is practical. Prior to start an I-DT&E, the software components to be used must have passed PQT/FQT and the hardware components must have passed CLT. All shalls that have not been previously verified to and by the customer must be verified during this activity. In addition, many of the previously verified shalls will be retested as part of a series of operability tests designed to assure CCS ability to support the designated mission it is configured to perform. This is the first contractor test designed to verify total system mission assurance. Test planning, execution, and evaluation is the responsibility of the T&I organization. The customer is responsible for test plan review and approval, witnessing the test, and final acceptance of a successful test activity.

A selected portion of the I-DT&E test program conducted at the SDL will be rerun at the STC. The formally qualified CI/CPCIs will be tested in the actual operational configuration and environment. This is the final contractor executed test and is designed to verify mission operability goals. Since I-DT&E testing is the final contractor test of mission assurance it will be discussed in greated detail later in this section.

Foils #29,30

d. Final integrated CI/CPCI tests also known as Final Development Test and Evaluation (F-DT&E) will be conducted immediately following completion of I-DT&E. The tests will be conducted by a customer test team to independently verify compliance with the A specification. These tests will focus on interfaces, system level end-to-end tests, and multiple satellite support capabilities. The contractor generates test plans and supports execution of these tests. The customer approves test plans, executes the tests, and evaluates test results.

Foil #31

As a foundation to a more detailed discussion of I-DT&E, the primary goals of this activity are restated. They are to perform integrated CI/CPCI tests designed to provide a comprehensive verification of performance requirements, and to perform operability tests designed to demonstrate that the CCS meets the operational expectations of the customer.

Initially each I-DT&E will be performed at the SDL using a configuration that resembles the operational configuration as closely as practical. Following hardware installation at the STC, the software used for I-DT&E at the SDL will be installed in its operational configuration at the STC. Selected I-DT&E tests will then be rerun using other CCS/AFSCF segments that are available.

Foil #32

a. The objectives of I-DT&E can best be stated by dividing them into objectives to be run at the SDL and objectives to be run at the STC.

SDL objectives are to ensure that each operational configuration test is as thorough as possible before CPCIs are installed in the STC and to perform engineering tests where hardware/software design support is readily available.

The STC objectives are to ensure that each operational configuration can meet its requirements while utilizing other equipment/functional areas of the STC and to demonstrate that each operational configuration meets customer expectations.

Foil #33

b. The I-DT&E test strategy is to verify that each operational configuration meets its requirements specifications. Whenever feasible, these tests will be conducted in the SDL to allow for verification of requirements as early as possible in the test cycle, to provide early customer visibility into test progress, to minimize I-DT&E test time required at the STC, and to enable STC testing to be primarily concerned with system operations and mission assurance testing.

c. The methodology used to optimize testing in the SDL and STC in support of I-DT&E goals is to test CI/CPCI integration at the SDL, initially test each operational configuration at the SDL (load and special tests will be defined later in this presentation), and use a test driver computer at the STC to allow many tests to be performed without impacting AFSCF operations.

Foil #34

d. The following are the four types of tests run under I-DT&E:

The CCS system initialization tests will be designed to establish all CCS system initial condition interfaces, to initialize the CCS to a point where all displays are active and ready for operations, and to verify shalls associated with CCS system initialization.

Foil #35

The CCS system mission operability tests will be designed to verify the ability of the system to support vehicle operations including the generation of mission, track/orbit, telemetry, and command support plans; execution of a contact support plan; processing, archiving, and evaluation of track, telemetry, and command data; planning and execution of simple and complex maneuvers; simultaneous support of two satellite vehicles; support of launches and launch rehersals; initialization and utilization of external interfaces; and error recovery. These tests will be used to verify shalls associated with system operability that have not been previously verified. In addition, this test will also be used to retest previously verified shalls in a systems operational environment.

Foil #36

The CCS system load test will be designed to verify and demonstrate the shalls specified in the CCS work load appendix to the 'A' Specification.

The CCS special tests will be designed to verify all shalls that cannot be effectively tested during any other test activity.

Operations Test and Evaluation (OT&E)
Foils #37,38

After completion of DT&E for each operational configuration the CCS system for that configuraiton is made available for operations. The Mission Director for that operation then starts independent mission assurance tests. This phase of testing is divided into two sequential activities. The first is called Initial Operations Test and Evaluation (IOT&E) and the second is called Formal Operations Test and Evaluation (FOT&E).

In order to protect satellite operations during this phase of testing the Mission Director conducts parallel activities in two independent facilities. The original facility continues to support satellite operations while the new facility is undergoing continued mission assurance testing. Upon successful completion of this activity, operations will be switched to the new facility and the old facility will be deactivated.

This activity will not be discussed in detail since the contractor supports this activity in a problem resolution mode and does not participate in its test planning.

VII. SUMMARY
Foil #39

The modernization of such a large and complex system, including active transitioning, is not a simple task. The USAF has, from system definition, recognized the need to completely and accurately define every detailed requirement. The DSM team has responded to this challenge with a system approach that utilizes proven hardware and software to the greatest extent possible and by the establishment of a system which allocates both functional and performance requirements to manageable entities. Traceability of each requirement is provided to allow verification at the earliest possible time in the development schedule. Continuing demonstrations are provided, of the compatibility of each component as the system is incrementally assembled. Through this concept, mission assurance will be provided in DSM.

FOIL 1

AIR FORCE SATELLITE CONTROL FACILITY (AFSCF)

- ESTABLISHED IN '60s TO SUPPORT NATIONAL SPACE PROGRAMS
- PROVIDES FOR TELEMETRY, TRACKING, AND COMMANDING
- SUPPORTS DOD, NON-DOD, AND OCCASIONALLY FOREIGN SPACE VEHICLES
- PROVIDES DATA PROCESSING, DISTRIBUTION AND ROUTING SERVICES FOR EXTERNAL USERS

FOIL 2

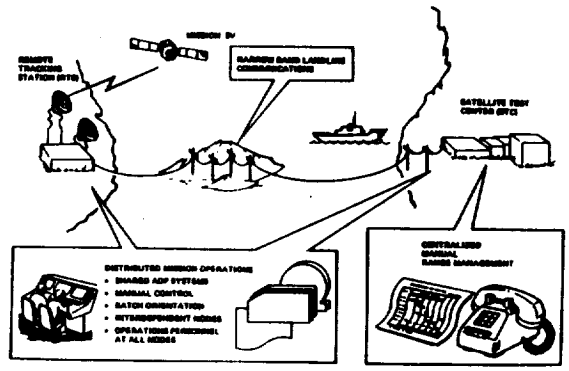
REASONS FOR MODERNIZATION

- CHANGING SPACE PROGRAMS ROLES AND MISSIONS
- TECHNOLOGY ADVANCEMENT
- INCREASE LOAD
- REDUCE OPERATING AND MAINTENANCE COSTS

FOIL 3

AFSCF SYSTEM

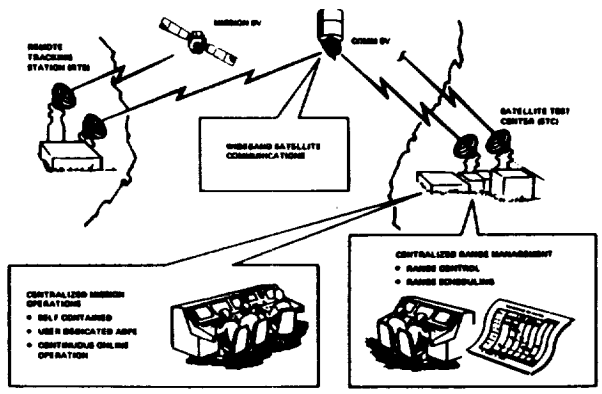
- WORLDWIDE, HIGHLY DISTRIBUTED FACILITY
- FIVE SEGMENTS
  - FACILITIES
  - SUPPORT
  - COMMUNICATIONS
  - TELEMETRY, TRACKING AND COMMANDING
  - COMMAND AND CONTROL



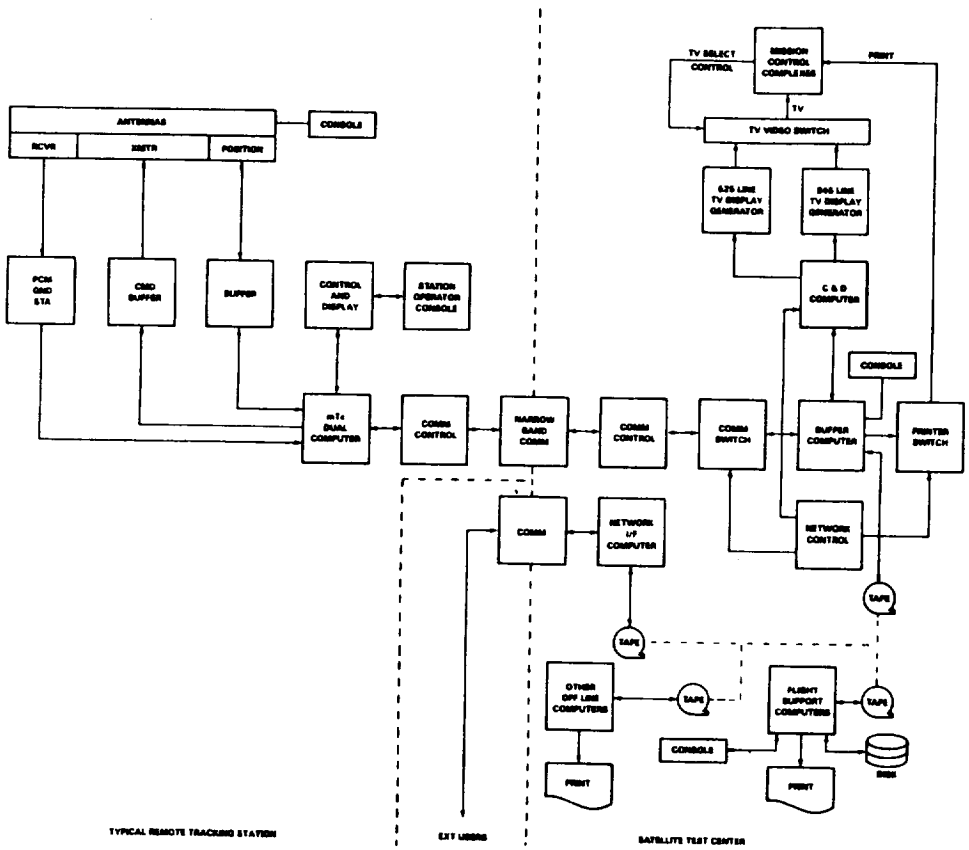
FOIL 8

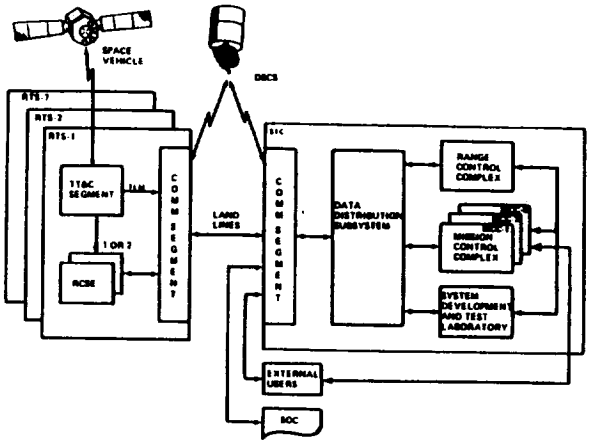
DATA SYSTEM MODERNIZATION

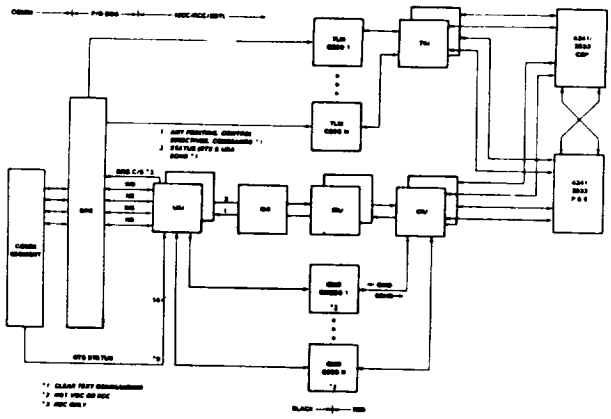
- CENTRALIZES DATA PROCESSING CAPABILITY THROUGHOUT THE NETWORK AT THE STC
- PROVIDES DIRECT COMMAND AND CONTROL CAPABILITY
- PROVIDES FOURTH GENERATION ADPE
- PROVIDES SOFTWARE DEVELOPED WITH MODERN DEVELOPMENT TOOLS
- REDUCES STAFFING AT RTS
- PROVIDES FUTURE GROWTH CAPABILITY
- MODERNIZATION OF DATA SYSTEM ARCHITECTURE

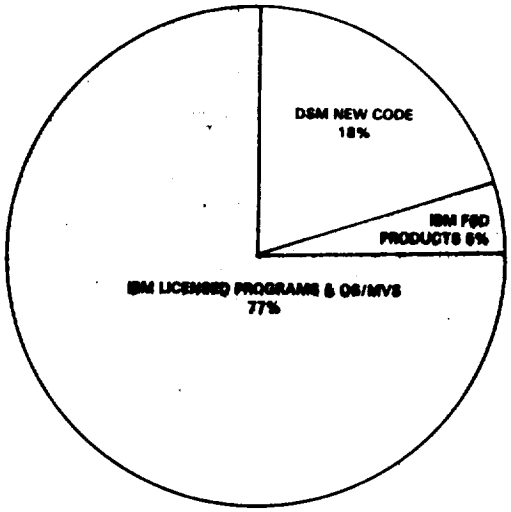


FOIL 5


ANTENNAS
RCVR
XMTR
POSITION
CONSOLE
PCM GMD STA
CMD BUFFER
BUFFER
CONTROL AND DISPLAY
STATION OPERATOR CONSOLE
DUAL COMPUTER
COMM CONTROL
NARROW BAND COMM
COMM CONTROL
COMM SWITCH
BUFFER COMPUTER
PRINTER SWITCH
COMM
NETWORK I/F COMPUTER
NETWORK CONTROL
TAPE
TAPE
OTHER OFF LINE COMPUTERS
TAPE
FLIGHT SUPPORT COMPUTERS
TAPE
CONSOLE
DISK
PRINT
PRINT
TV SELECT CONTROL
MISSION CONTROL COMPLEXES
PRINT
TV
TV VIDEO SWITCH
625 LINE TV DISPLAY GENERATOR
945 LINE TV DISPLAY GENERATOR
C & D COMPUTER
CONSOLE
TYPICAL REMOTE TRACKING STATION
EXT USERS
SATELLITE TEST CENTER


SPACE VEHICLE
DSCS
RTS-3
RTS-2
RTS-1
TT&C SEGMENT
TLM
1 OR 2
RCSE
COMM SEGMENT
LAND LINES
COMM SEGMENT
DATA DISTRIBUTION SUBSYSTEM
RANGE CONTROL COMPLEX
MISSION CONTROL COMPLEX
SYSTEM DEVELOPMENT AND TEST LABORATORY
EXTERNAL USERS
SOC



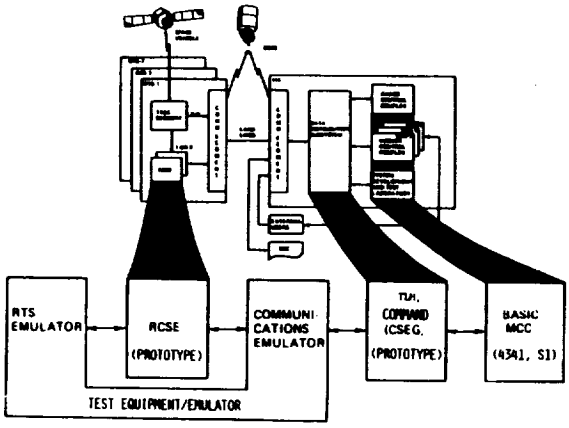


FOIL 9

DSM IMPLEMENTATION

- UTILIZE SUBCONTRACTOR AND EMPLOYEES WITH RELATED SPACE COMMAND AND CONTROL EXPERIENCE
- DSM METHODOLOGY
  - REQUIREMENTS ALLOCATION
  - TRACEABILITY
- INDEPENDENT TEST ORGANIZATION
- SYSTEM DEVELOPMENT LABORATORY

SDL INTEGRATION TEST CONFIGURATION



FOIL 10

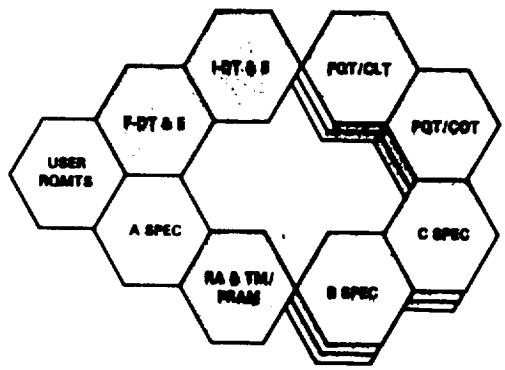
REQUIREMENTS

- DEVELOPMENT OF SYSTEM REQUIREMENTS
  - GENERATION OF SYSTEM LEVEL REQUIREMENTS
  - ALLOCATION OF REQUIREMENTS TO LOGICAL DESIGN UNITS
  - DEVELOPMENT OF DETAILED DESIGN SPECIFICATION
- DATA SYSTEM MODERNIZATION (DSM)
  - PROVIDE ALLOCATION OF REQUIREMENTS AND TRACEABILITY
  - PROVIDE UPWARD TRACEABILITY OF REQUIREMENTS VERIFICATION

FOIL 11

SYSTEM LEVEL REQUIREMENTS

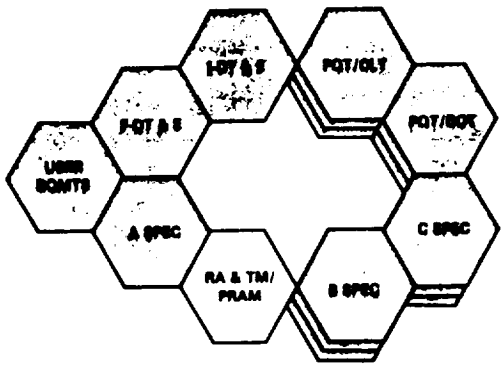
- CUSTOMER REQUIREMENTS
  - DETAILED USER REQUIREMENTS
- A SPECIFICATION
  - DERIVED FROM DETAILED REQUIREMENTS SPECIFICATIONS
  - INTENT OF DOCUMENT
    - PROVIDE A CONCISE CCS FUNCTIONAL DESCRIPTION
    - ESTABLISH FUNCTIONAL CCS REQUIREMENTS
    - DEFINE INTERFACES WITH OTHER AFSCF SEGMENTS
    - SPECIFY OTHER APPLICABLE DOCUMENTS
  - CCS WORK LOAD REQUIREMENTS
  - TRACEABILITY



FOIL 13

REQUIREMENTS ALLOCATION

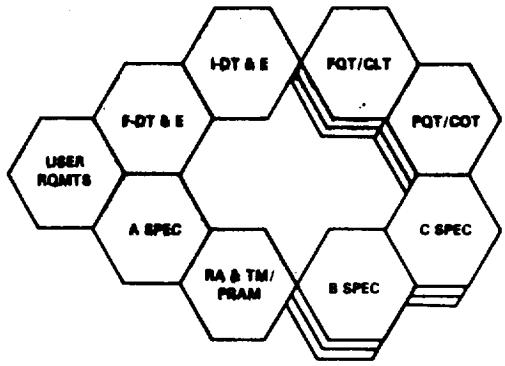
- REQUIREMENTS ALLOCATION AND TRACEABILITY MATRIX
  - TRACEABILITY OF FUNCTIONAL REQUIREMENTS FROM SYSTEM LEVEL DOCUMENTATION
  - ALLOCATION OF FUNCTIONAL REQUIREMENTS TO B AND C SPECIFICATIONS
- PERFORMANCE REQUIREMENTS ALLOCATION MATRIX
  - PARTITIONS AND ALLOCATES PERFORMANCE REQUIREMENTS AND RESOURCES NEEDED FROM "A" SPECIFICATION TO CIs/CPCIs



FOIL 15

REQUIREMENTS/DESIGN SPECIFICATIONS

- CIs/CPCIs PROGRAM DEVELOPMENT SPECIFICATIONS ("B" SPECIFICATIONS)
  - ONE PER CI/CPCI
  - DESCRIBE FUNCTIONAL CHARACTERISTICS
  - REQUIREMENTS TRACEABILITY MATRIX
  - REQUIREMENTS VERIFICATION MATRIX
- CIs/CPCIs PROGRAM PRODUCT SPECIFICATIONS ("C" SPECIFICATIONS)
  - COMPLETE IDENTIFICATION OF CI/CPCI
    - REQUIREMENTS
    - STORAGE ALLOCATION
    - DETAILED DESIGN
    - METHOD OF QUALIFICATION
  - REQUIREMENTS TRACEABILITY MATRIX
  - REQUIREMENTS VERIFICATION MATRIX



FOIL 17

TEST

- DEVELOPMENT TEST AND EVALUATION (DT&E)
- TEST PHILOSOPHY
  - GOALS
  - METHOD USED TO ENSURE COMPLIANCE
- IMPLEMENTATION
  - TEST LEVELS
  - EMPHASIS ON CONTRACTOR FORMAL SYSTEMS TESTS

FOIL 18

TEST GOALS

- CCS SYSTEM ADHERES TO LETTER OF REQUIREMENTS
- IMPLEMENTED DESIGN FULLY ADHERES TO DESIGN SPECIFICATIONS
- FINAL PRODUCT PERFORMS IN ACCORDANCE WITH CUSTOMER OPERATIONAL EXPECTATIONS

FOIL 19

SHALL VERIFICATION AND DEMONSTRATION

- "VERIFICATION" vs "DEMONSTRATION"
- DEFINITION OF A "SHALL"
- VERIFICATION CROSS REFERENCE MATRIX (VCRM)
- SHALL VERIFICATION
- SYSTEM LEVEL DEMONSTRATION

FOIL 20

IMPLEMENTATION

- FOUR DT&E TEST LEVELS
  - COMPONENT TESTS
  - COMPLIANCE TESTS
  - INITIAL SYSTEMS TEST
  - FINAL DEVELOPMENT TEST AND EVALUATION

FOIL 21

DT&E LEVELS

- PRIMARY GOALS OF EACH TEST
- CCS ELEMENTS INCLUDED IN EACH TEST
- RELATIONSHIP TO OTHER TESTS
- LOCATION OF TEST EXECUTION

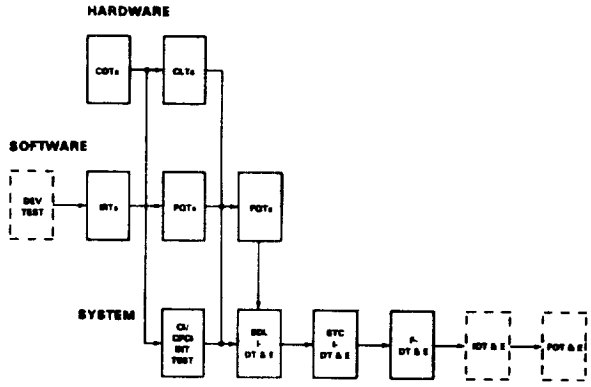
FOIL 22

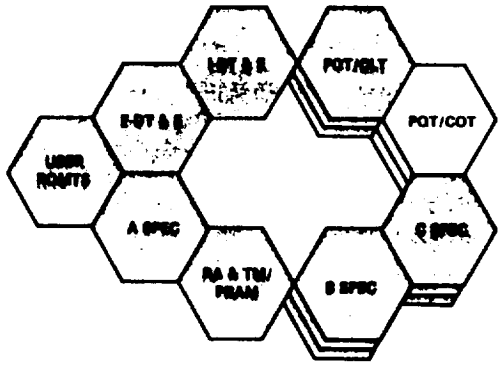
COMPONENT TESTS

- TESTS

  SOFTWARE
  - INTERNAL READINESS TESTS (IRTs)
  - PRELIMINARY QUALIFICATION TESTS (PQTs)

  - HARDWARE
    - COMPONENT TESTS (COTs)
- GOALS
  - DESIGN SPECIFICATION IMPLEMENTATION
- CCS ELEMENTS INVOLVED
  - INDIVIDUAL HARDWARE/SOFTWARE COMPONENTS
- TEST EXECUTION LOCATION
  - SDL OR SUBCONTRACTOR FACILITY

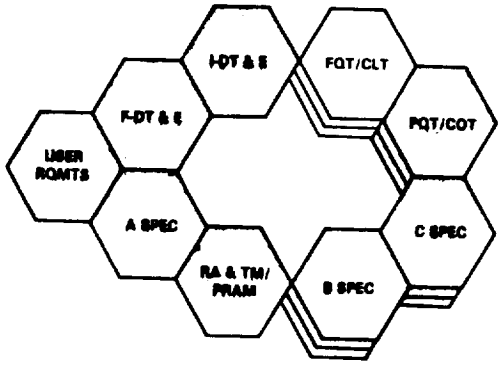




FOIL 25

COMPLIANCE TESTS

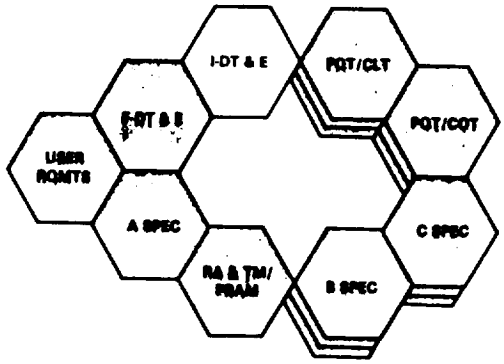
- TESTS
  - SOFTWARE
    - FORMAL QUALIFICATION TESTS (FQTs)
  - HARDWARE
    - COMPLIANCE TESTS (CLT's)
- GOALS
  - DESIGN SPECIFICATION IMPLEMENTATION
  - ADHERANCE TO B SPECIFICATION
- CCS ELEMENTS INVOLVED
  - INDIVIDUAL HARDWARE/SOFTWARE CONFIGURATION ITEMS
- TEST EXECUTION LOCATION
  - SDL OR SUBCONTRACTOR FACILITY



FOIL 27

INITIAL SYSTEMS TESTS

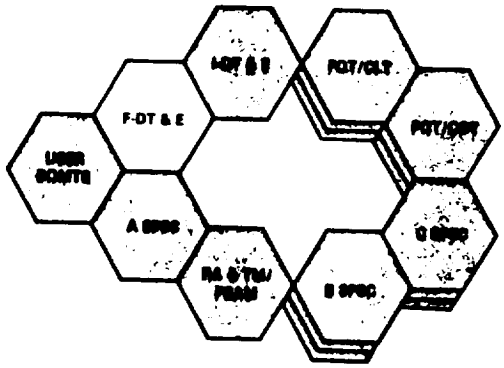
- TESTS
  - CIs/CPCIs INTEGRATION TESTS
  - INITIAL DEVELOPMENT TEST & EVALUATION (I-DT&E)
    - FIRST CCS TOTAL SYSTEM TESTS IN SDL
    - RETEST IN OPERATIONAL CONFIGURATION AT THE STC
    - FINAL CONTRACTOR MISSION ASSURANCE TEST
- GOALS
  - ADHERENCE TO REQUIREMENTS AS DELINEATED IN THE A AND B SPECIFICATIONS
  - MEET OPERATIONAL EXPECTATIONS
- CCS ELEMENTS INVOLVED (I-DT&E)
  - CPCIs THAT HAVE PASSED PQT/FQT
  - CIs THAT HAVE PASSED CLT
- TEST EXECUTION LOCATION
  - IN THE SDL AND IN THE STC



FOIL 29

FINAL DEVELOPMENT TEST AND EVALUATION (F-DT&E)

- TEST
  - FINAL DEVELOPMENT TEST AND EVALUATION (F-DT&E)
    - END TO END SYSTEMS TESTS
    - CCS INTERFACES
    - MULTIPLE SATELLITE TESTS
- GOALS
  - INDEPENDENT A SPECIFICATION VERIFICATION
  - MISSION OPERABILITY
- CCS ELEMENTS INVOLVED
  - CCS SYSTEM THAT HAS PASSED I-DT&E IN THE STC
  - TESTS RUN AT STC BY CUSTOMER TEST TEAM



FOIL 31

I-DT&E GOALS AND TEST CONFIGURATION

- GOALS RESTATED
  - VERIFICATION OF FUNCTIONAL/PERFORMANCE REQUIREMENTS
  - MEET THE OPERATIONAL EXPECTATIONS OF THE CUSTOMER
- TEST CONFIGURATIONS
  - SDL
    - RESEMBLES OPERATIONAL AND ENVIRONMENTAL CONFIGURATION AS CLOSELY AS PRACTICAL
  - STC
    - USES OPERATIONAL CONFIGURATION AND OTHER CCS/AFSCF SEGMENTS

TEST OBJECTIVES

- SDL
  - TEST EACH OPERATIONAL CONFIGURATION AS THOROUGHLY AS POSSIBLE BEFORE INSTALLING IN THE STC
  - CONDUCT TESTS WHERE HARDWARE/SOFTWARE DESIGN SUPPORT IS READILY AVAILABLE
- STC
  - TEST OPERATIONAL CONFIGURATION UTILIZING OTHER EQUIPMENT/STC FUNCTIONAL AREAS
  - VERIFY OPERATIONAL CONFIGURATION MEETS CUSTOMER EXPECTATIONS

TEST STRATEGY

- VERIFY EACH OPERATIONAL CONFIGURATION MEETS ITS REQUIREMENTS SPECIFICATION
- TEST IN THE SDL WHENEVER FEASIBLE TO:
  - ALLOW VERIFICATION OF REQUIREMENTS AS EARLY AS POSSIBLE
  - ALLOW CUSTOMER EARLY VISIBILITY INTO TEST PROGRESS
  - MINIMIZE TIME REQUIRED AT STC
  - EMPHASIZE OPERATIONS TESTING AT STC

TEST METHODOLOGY

- TEST CIs/CPCIs INTEGRATION IN THE SDL
- EXECUTE INITIAL TESTS OF OPERATIONAL CONFIGURATIONS IN THE SDL
- USE TEST DATA DRIVER AT STC TO MINIMIZE AFSCF OPERATIONS IMPACTS
- WHENEVER FEASIBLE PERFORM LOAD & SPECIAL TESTS IN THE SDL

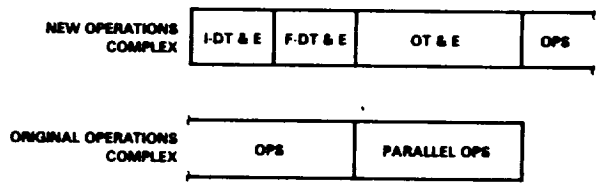
OPERATIONS TEST AND EVALUATION (OT&E)

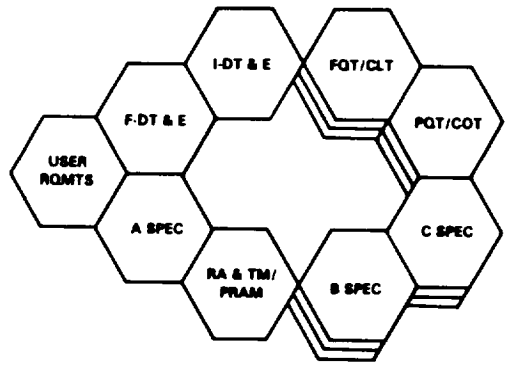
- TESTS
  - INITIAL OPERATIONS TEST AND EVALUATION (IOT&E)
  - FORMAL OPERATIONS TEST AND EVALUATION (FOT&E)
- GOALS
  - INDEPENDENT USER VALIDATION OF MISSION ASSURANCE
  - PROTECT SATELLITE OPERATIONS
- CCS ELEMENTS INVOLVED
  - CCS SYSTEM THAT HAS PASSED F-DT&E
- TEST LOCATION
  - STC PARALLEL OPERATIONS

FOIL 34-36

I-DT&E TEST TYPES

- SYSTEM INITIALIZATION TESTS
  - ESTABLISH INTERFACES
  - INITIALIZE CCS SYSTEM
  - VERIFY SYSTEM INITIALIZATION SHALLS
- MISSION OPERABILITY TESTS
  - SUPPORT VEHICLE OPERATIONS
  - LAUNCH AND LAUNCH REHERSALS
  - EXTERNAL INTERFACES
- LOAD TESTS
  - A SPECIFICATION WORK LOAD APPENDIX
- SPECIAL TESTS
  - ALL SHALLS NOT VERIFIED ELSEWHERE





DOD SPACE SHUTTLE OPERATIONS

AT VANDENBERG AIR FORCE BASE

LAUNCH AND LANDING SITE

Peter L. Portanova

Director, System Requirements
Shuttle Activation & Operations Directorate

The Aerospace Corporation

ABSTRACT

This paper discusses the rationale for the existence at Vandenberg Air Force Base, CA of a launch and landing site for the NASA-developed Space Shuttle Vehicle (SSV), and outlines the operational flow for which the USAF Space Division is responsible. An overview of Space Shuttle Vehicle processing at Vandenberg Air Force Base Launch and Landing Site is presented.

SUMMARY

Our adventures in space, both those of the U.S. Air Force and NASA, will rely on the Space Transportation System (STS) for the next several decades. The immensity and complexity of this program challenges every facet of engineering and management.

Space Shuttle operations will be located at both the east coast and the west coast of the United States to provide orbital inclinations and corresponding launch azimuths required for projected payload requirements.

Kennedy Space Center FL STS operations began with the first launch of a Space Shuttle on 12 April 1981. Vandenberg Launch and Landing Site STS operations will begin with its initial launch capability of a Space Shuttle on 15 October 1985 with initial operational capability scheduled for February 1987.

The Vandenberg Launch and Landing Site (VLS) is an integral part of the national STS capability. Much of the VLS hardware, software, and procedures are NASA common, but there are some unique design, integration, activation and operational requirements at VLS, which are detailed below.

Due to a reduction in the projected VLS launch rate, and in order to maximize cost effectiveness, KSC processing capabilities will support early year VLS mission requirements.

INTRODUCTION

Space Shuttle flights will be launched from two locations: Kennedy Space Center, Florida and Vandenberg Air Force Base, California Launch and Landing Sites. These sites were selected based upon both NASA and DOD needs for projected payloads and requirements for no land mass over-flight during the boost phase of the Space Shuttle launch. Vandenberg Air Force Base (VAFB), CA was selected as the Western Launch and Landing Site for the NASA developed Space Shuttle Vehicle in order to obtain the most desirable combination of launch azimuths and planned orbital inclinations (Figure 1). Launches from Vandenberg Launch and Landing Site (VLS) will support both NASA and DOD missions, with classified or non-classified payloads, launched between azimuths of 158° and 208°. This launch capability is intended to accommodate polar and near polar orbital missions. The VLS will provide an initial launch capability of four launches per year and up to ten evenly spaced launches per year during full operational capability.

The VLS reference mission, to be used for development criteria, is a payload delivery and retrieval mission, consisting of a delivery mission of a modular spacecraft weighing 14,515 kg (32,000 lb) in a 278 km (150 nmi) circular orbit at 98° inclination, and a retrieval of a passively cooperative, stabilized spacecraft weighing 10,205 kg (22,500 lb) from a similar coplanar orbit and returned to Vandenberg. The mission length, including contingencies, will be seven days. The allowable spacecraft dynamic envelope is 4.6 m (15 ft) diameter by 18.3 m (60 ft) long.

The USAF Space Division is responsible for acquisition, activation and operation of the DOD ground support systems for the STS at VAFB and nearby Port Hueneme, which will be used to recover the expended Solid Rocket Boosters. In keeping with the STS implementation philosophy, maximum practical use of existing VAFB facilities and NASA-developed ground support equipment and software is emphasized. Checkout and operations of the vehicle at the launch pad will essentially be the same as those at Kennedy Space Center, with any differences attributed to mission or geographic constraints.

Current STS launch planning calls for Kennedy Space Center to perform Orbiter processing for the first three VLS launches, and solid rocket booster processing and parachute refurbishment for all VLS launches. VLS facilities have been optimized to utilize resources at other locations such as NASA/KSC for the early missions. It is planned that full launch and landing capabilities will be available when mission requirements for capability beyond four launches per year is necessary.

The Space Shuttle Vehicle (SSV) consists of four major elements: a reusable manned Orbiter, three main engines, an expendable External Tank (ET), and two reusable Solid Rocket Boosters (SRBs). The basic Space Shuttle Vehicle characteristics are as in Figure 2.

The VLS ground support systems which support the Space Shuttle operations are divided into discrete geographical and functional Station Sets (Figure 3). Each geographical Station Set includes the aggregate of facilities, support equipment, operating and maintenance

procedures and personnel required to perform all specified activies within each station set, such as Runway, Launch Pad, etc. The functional Station Sets include a totally functional element design procured as a system and then installed at appropriate locations, for example, Communications, Vandenberg Launch Processing System, Utilities, Logistics, etc. All required facilities, support equipment, computers, software, personnel, and procedures included within geographical and functional Station Sets are allocated in accordance with their respective Station Set Specifications. All ground Computer Application Programs are allocated to the test sequence software system in accordance with its system specification.

Space Shuttle operations flows at VAFB are depicted in Figure 4, and detailed below in roughly chronological order.

NORTH VANDENBERG OPERATIONS

Mate/Demate Facility. When the Orbiter arrives on a modified Boeing 747 Shuttle Carrier Aircraft (SCA), it will be towed to the Mate/Demate Facility and the Orbiter removed from atop the SCA by use of an Orbiter Lifting Frame (OLF). The OLF was designed and developed uniquely for VLS and performs the same functions as the KSC Mate Demate Device. Subsequent to removal from the SCA, the landing gear is extended and the Orbiter lowered to the concrete ramp and then towed via the towroute to the Orbiter Maintenance and Checkout Facility.

Landing Facility. The Orbiter will arrive at the runway at North Vandenberg Air Force Base (NVAFB) either on SCA or returning from orbit. The landing facility consists of a 200 foot wide by 15,000 foot-long concrete runway with a high-intensity approach lighting system and edge threshold. Both Tactical Air Navigational and a Microwave Scanning Beam Landing System transmits accurate directional and glide slope information to the Orbiter during the landing phase. As soon as an Orbiter lands and rolls to a stop, all necessary post-landing equipment is positioned around the vehicle to safe, secure, and supply the required purges, cooling, and electrical power preparatory to flight crew egress and subsequent towing to the Orbiter Maintenance and Checkout Facility (Figure 5).

Orbiter Maintenance and Checkout Facility. As a result of an optimization study, the Orbiter will be processed/checked out by KSC for the first three VLS launches in the KSC Orbiter Processing Facility and ferried to VLS by SCA. This checkout will minimize processing the Orbiter at the VLS Orbiter Maintenance and Checkout Facility (OMCF) (Figure 6).

Upon arrival at the OMCF, the Orbiter will be positioned on jacks and leveled, allowing the necessary positioning relative to the work platform for accessibility to the Orbiter. When fully operational, this facility will perform ordnance safing, main engine purge and drying, venting of high pressure gases, draining of the hypergolic manifolds and cryogenic systems, payload removal as required, and a data dump of on-board computers to ground recorders.

In addition, the hypergolic modules consisting of Orbital Maneuvering Subsystem (OMS), the Forward Reaction Control System (FRCS) and Aft Reaction Control System (ARCS) can be removed, and sent to KSC for processing if required.

Additional operations that are performed at the OMCF include payload removal, complete internal and external Orbiter inspection, checkout of the fluid and electrical systems, repair of thermal protection system tiles, and reinstallation of flight crew equipment and hypergolic modules. The Orbiter payload bay is reconfigured for the next flight by removing provisions from the last flight and installing electrical/mechanical systems including payload retention fittings for the next mission. After inspection, testing, maintenance, and repair is completed, the payload bay doors are closed, access panels and ports closed out, and the Orbiter lowered off its jacks onto a transporter (Figure 7) for towing to South VAFB launch pad, 17 miles distant. Vandenberg's OMCF has a second major work area to conduct payload safing and deservicing which is separated from the Orbiter by a blast wall.

Presently, the only payload planned to be placed in the Orbiter payload bay while the Orbiter is horizontal in this facility will be the European Space Agency Spacelab. All other scheduled civil and military payloads will be placed into the Orbiter while in its vertical position on the launch pad. Also included in the OMCF, but contained within a separate building will be an Orbiter Thermal Protection System (TPS) refurbishment facility. This facility will enable TPS repair and production of any required replacement tiles.

Vandenberg Launch Processing System. The Vandenberg Launch Processing System (VLPS) consists of a network of ground computers, data links, displays, controls, hardware interface devices, and ground computer software to process the Space Shuttle Vehicle from landing through launch (Figure 8). The VLPS is a functional station set and is divided into three subsystems: Checkout, Control and Monitoring System (CCMS), Central Data System (CDS); and Record and Playback System (RPS). Maximum use of existing NASA software, automation, and management information support will be utilized. The VLPS used at VAFB is functionally identical to that at KSC.

One set of VLPS hardware located in Bldg. 8510 is provided to process the Orbiter at NVAFB. Another set of VLPS hardware, located at South VAFB in the Launch Control Center adjacent to the launch pad, will accommodate launch operation functions.

Hypergolic Maintenance and Checkout Facility. The activation of this facility has been deferred as a result of the aforementioned optimization study. All OMS modules/Line Replacement Units (LRU) repair and refurbishment are to be performed at KSC.

Flight Crew Systems Facility. The Flight Crew facility is located at NVAFB where existing buildings are being modified to provide sleeping and medical facilities and flight crew equipment checkout. Also included will be a mission equipment kit building primarily devoted to equipment required to support cargo bay activities on a mission to mission basis.

Logistics Facility. This facility consists of a central supply facility which will house and control all flight and ground support systems component

spares and material required to support VLS operations. Also forming a part of this facility is an Orbiter Space Shuttle Main Engine (SSME) shop and an Orbiter brake and tire shop. This facility will also support SSV Line Replacement Unit (LRU) maintenance and checkout, and material management.

SOUTH VANDENBERG OPERATIONS

While the Orbiter is processed at NVAFB, the ET and SRBs are processed at South Vandenberg Air Force Base (SVAFB) facilities.

Launch Control Center. This facility is an existing two-story, concrete blockhouse located approximately 1100 feet from the launch pad. The Launch Control Center houses the VLPS hardware, communication systems, closed circuit television, hazard monitors, payload operations areas, and accommodations for technical support personnel. This station set provides the capability to check out, control and monitor the Space Shuttle Vehicle at the launch pad, and payload operations in the Payload Preparation Room.

External Tank Processing and Storage Facility. The ETs are delivered by NASA barge from Michoud, LA. through the Panama Canal to a modified harbor at South Vandenberg, removed from the barge and towed to the External Tank Checkout Facility (TCF) (Figure 9). The ET consist of two tanks joined together by a collar-like intertank to form one large propellant storage container 47 m (154.2 ft) long and 8.4 m (27.5 ft) in diameter. The forward and aft tanks are for liquid oxygen and liquid hydrogen respectively. It weighs approximately 69,000 lb empty and approximately 1,630,000 lb at launch. Upon ET arrival at the TCF, the range safety receivers and decoders will be installed and checked out. Ordnance will then be installed, and thermal protection closeout initiated. Final ET checkout will be performed at the launch pad. In addition to the capacity for storing up to four of these tanks while maintaining internal blanket pressure, the facility has a checkout cell where inspection, cleaning, leak, humidity, range safety equipment checks can be conducted, and spray-on foam insulation applied at the closeout ports. The harbor, tow route, and processing facility are part of the TCF.

SRB Refurbishment and Subassembly Facility. SRB component storage, aft boost assembly, and limited SRB component checkout are accomplished at this station set (Figure 10). The overall length of the SRB is 45.5 m (149.2 ft), and the diameter is 3.7 m (12.2 ft). Each of the four SRM segments weighs up to 167,375 kg (369,000 pounds). The SRM propellant segments are received by rail and transferred to the storage area of the station set. The SRB forward assembly and aft skirts with TVC components installed are delivered by C5 aircraft from KSC where refurbishment, subassembly, and comprehensive checkout have been accomplished. The aft skirt is mated to the aft solid propellant segment and nozzle to form the aft booster assembly, and the aft and forward assemblies are connected together with electrical cabling at VLS for a limited end-to-end electrical systems checkout.

Launch Pad. The launch pad station set includes facilities, support equipment and services required to assemble, integrate, checkout and launch the Space Shuttle (Figure 11). Facilities include launch mount, exhaust ducts, Access Tower (AT), the Mobile Service Tower (MST), a mobile Shuttle Assembly Building (SAB), a mobile Payload Changeout Room (PCR), a fixed multi-cell Payload Preparation Room (PPR),

and storage and holding areas for hypergolic and cryogenic propellants and gases. In addition, sound-suppression water storage and deluge systems, and an ice suppression system are included. This Station Set is used for assembly, integration, checkout, and launch of the SSV. The launch pad operation is supported from the Launch Control Center and divided into the following major phases:

1. Space Shuttle Vehicle Assembly and Interface Verification. The SSV assembly begins with the MST in position at the Launch Mount. Preparations for stacking of the SRM segments consist of transporting forward and aft assemblies and individual booster segments to the launch pad by a self-propelled, rubber-tired transporter (Figure 12). At the launch pad, the environmental covers and transporter tie-down restraints are removed, the SRB handling device installed, and grounding systems verified. Holddown-post release hardware is then pretensioned. Once support fixture alignment is confirmed, the first aft booster assembly is hoisted from its transporter and positioned on its support fixture using the MST bridge crane (Figure 13). The second SRB aft booster assembly is similarly installed and the alignment verified. The center segments, forward segments, and forward assemblies of each SRB are then alternately installed with leak checks performed to verify SRB seal integrity as each segment or assembly is installed. After completion of the SRB motor segment stacking, additional alignment checks are made to ensure that excessive loads are not induced into the ET when it is mated to the SRBs. SRB tunnel cables used to cover the electrical harness are installed and verified. Destruct ordnance is then installed but left in a safed condition using safe-arm devices.

During these operations, the ET and SRB-ET interface hardware is moved to the pad. Once physical verification of cabling is complete, SRB tunnel covers are installed and the SRB-ET interface hardware installed and checked out. The SRBs are then ready for ET mating.

The SAB, a high-bay structural steel building, moves on two sets of double tracks and mate with the MST to seal off the launch pad and provide a benign environment for ET and Orbiter erection, mating, and processing before launch.

The ET is transported from the TCF to the launch pad (Figure 14). The ET is lifted clear of the transporter, rotated to the vertical position and stabilized using the MST and SAB cranes (Figure 15). The ET-SRB aft stabilizing struts are installed and the ET lowered until the ET weight is transferred to the SRB forward attach fittings and secured.

The Orbiter is transported 17 miles from the OMCF to the launch pad on a rubber-tired transporter. At the launch pad, the Orbiter is lifted clear of the transporter by the MST and SAB cranes, rotated to the vertical, stabilized, and translated to the ET (Figure 16). After the two aft structural attachments are made the forward attachments are secured. This procedure transfers the Orbiter weight to the ET-SRB assembly. Following structural mating of the Orbiter, the Orbiter-ET and Orbiter-launch mount interfaces are established and verified, and (TPS) closeout starts.

Following establishment and verification of the various interfaces and environmental conditioning, ground power is applied to the vehicle. A Shuttle Interface Test (SIT) is

performed to verify interface integrity and systems compatibility. The VLPS provides the stimuli/response verification, monitor status, and provides control and recording of test data. The SIT ends with the vehicle and support equipment configured for final ordnance installation, connections and verification activities.

2. Space Shuttle Vehicle Servicing. The SAB is rolled back prior to start of hazardous servicing. Hazardous servicing of the Orbiter and SRBs is controlled remotely by VLPS, and commences when all nonessential personnel have cleared the pad area following service preparations. Orbiter servicing consists of the loading of ARCS, FRCS, and OMS hypergols, plus the loading and pressurization of ARCS, FRCS and Main Propulsion System (MPS) helium bottles. When servicing is complete, ground crews return to the pad and disconnect APS servicing lines and install flight panels.

A Shuttle Range Safety System open loop test is performed, flight code plugs installed, and a closed loop test performed. This is followed by a power on/off stray voltage check of ordnance circuits, and then final connection of ordnance.

All ground services and platforms are secured, and at T-36 hours MST rollback to launch position occurs, and ground interfaces are re-established. The ET H_2 vent arm and GO_2 Vent Arm are positioned and interfaces connected. All ET vent arm interfaces are verified and leak checked. The Orbiter crew cabin access arm is moved back into position, and the clean room environment re-established.

3. Payload Installation. The Payload Preparation Room (PPR), which is part of the Launch Pad Station Set, consists of three checkout cells each with fixed platform levels, an erection room, an airlock and a transfer tower (Figure 17). During the early years, only two of the three checkout cells will be activated. The payload enters the PPR through the airlock and moves into the erection room for payload removal from transporter by overhead crane, rotation to vertical, and transfer to a checkout cell. Payload integration and checkout, ordnance installations, and fuel loading (if required) are performed in the checkout cell. The complete integrated cargo assembly including the payload strongback is moved to the transfer tower. The transfer area provides a support base from the erection room 75-ton bridge crane to the 75-ton transfer tower hoist. The transfer tower transfers the payload from the strongback to the Payload Ground Handling Mechanism (PGHM). The PGHM with cargo translates the cargo from the PPR tower to the PCR using the overhead rails. The PCR, which is a mobile structure, rolls to the Orbiter on the launch mount, and cargo installation into the Orbiter bay is performed utilizing the PGHM. Since the PPR and PCR facilities are located approximately 800 feet within the launch pad area, no payload retest within the PCR is required prior to transfer of the payload to the Orbiter payload bay.

4. Countdown and Launch. Precount operations place the SSV, support equipment and launch facility in a position of readiness to start the cryogenic loading and proceed to countdown. A final walkdown is performed. All non-flight hardware is removed, and SRB flight batteries and final ordnance installed. Final preparation for cryogenic loading is performed. Following retraction of the PGHM, the Orbiter payload bay doors are closed and seals verified.

The Payload Changeout Room doors are closed, and all mechanical and electrical interfaces between the PCR and launch pad are disconnected and stowed for movement. Access platforms and other movable support equipment are also retracted/stowed. The PCR is then moved back to the PPR at approximately T-12 hours.

Upon completion of the precount operations, the SSV is in a configuration either to proceed directly into the launch countdown or to be placed in a standby status for up to 24 hours. Countdown preparations can be held within this time.

The countdown is entirely under VLPS control except for the final seconds when operations are controlled by the onboard computer. The final countdown commences at approximately T-12 hours with loading of cryogenic propellants into the ET. The MPS/ET loading begins with LO_2/LH_2 chill down of the propellant transfer lines, and followed by slow fill to chill the ET. When slow fill is completed, fast fill is initiated. Fast fill is continued until the specified levels are reached, at which time a slow fill is again initiated to complete the MPS/ET loading. Upon completion of the cryogenic loading, the flight crew and passengers proceed to the Orbiter cabin where they secure and leak check the hatch and take their positions for launch. The terminal countdown is started after the crew and passengers report they are secured for launch. The terminal countdown includes performance of the flight crew checklist, verification of the guidance, navigation and control alignment, SSV subsystems status checks, transferring of SSV electrical power from ground to vehicle internal, and obtaining the required clearances for launch.

During the terminal count the following operations are also performed: termination of ET propellant replenishment, final flight pressurization of ET LO_2 and LH_2 tanks, retraction of the ET LO_2 tank vent arm, retraction of the Orbiter crew cabin access arm, arming of the required ordnance circuits, and the actions/verifications associated with the engine start sequence and the liftoff functions. Finally, the main engines are started. When the desired main engines thrust level is achieved, the SRB's are ignited, the holddown bolts and umbilicals are pyrotechnically released, and liftoff occurs.

At liftoff, operational command and control is transferred to the flight crew and NASA/JSC Mission Control Center. During flight, VLS will continue to provide range support, flight safety, contingency operations support, and monitoring of flight SSV status to prepare for landing and maintenance operations.

5. <u>Launch Pad Refurbishment, Reconfiguration and Facility Resupply</u>. Following liftoff, ground systems are deactivated and secured. Safety and preliminary damage inspections are conducted and when conditions permit, the SRB Refurbishment and Storage Facility and ET Storage Facility are reopened for normal work and the next launch pad cycle for launch begins.

Launch pad refurbishment consists primarily of inspection of the facility and equipment repair or replacement, as required. The first action required in refurbishment is to wash down the MST and pad structure as soon as possible after launch to remove the acid fallout resulting from the SRB exhaust products.

The crew cabin access arm is inspected and aligned after each launch to

restore the clean room environment. The ET umbilical plates are removed and refurbished by disassembly, cleaning and replacing of seals and components. All non-reusable items such as pyrotechnic cables in the SRB holddown posts are replaced during refurbishment.

SRB Retrieval and Disassembly Facility. When the SRBs have expended all solid propellant, they separate from the ET and parachute back to the ocean. Beacons on the SRBs are activated at impact and provide guidance to the retrieval vessels. A nozzle plug is lowered into the water from the retrieval vessel and inserted by divers into each SRB nozzle. Air is pumped into the SRB to displace water and cause the SRB to float horizontally on the water surface. The SRB frustums are recovered, and parachutes wound onto reels on the retrieval vessel. The tugboat and the retrieval vessel each tow one expended SRB to the U.S. Naval Station at Port Hueneme (Figure 18). KSC will provide one retrieval vessel and recovery crew for the first three VLS launches, and the USAF will provide and outfit the leased tug. The USAF is responsible for management of the retrieval operations.

At Port Hueneme, two straddle-lift carriers remove each SRB from the water and place them on special rail car dollies riding on a leveled, continuously welded track. The ordnance is safed and any remaining thrust vector control system hydrazine is drained. The rail cars are towed into an initial wash facility and then into the main building for complete SRB disassembly. The forward and aft assemblies are sent back to the wash facility for insulation removal by a high-pressure water spray and subsequent drying. The empty propellant segments are shipped back to the manufacturer for installation of new propellant and the forward skirt, aft skirt, parachutes, and remaining components are sent to KSC for refurbishment.

Transportation, Communications and Utilities Station Sets. There are additional functional station sets for auxiliary services; these include the Transportation Station Set which consists of the roads and special transportation equipment to move the SSV elements around Vandenberg; the Communications Station Set which provides voice, TV and data transmission within and among the station sets and control centers; and the Utilities Station Set which provides electrical power and distribution as well as fire suppression water, and sanitary sewers to the geographical station sets.

Facility Verification Vehicle/System Test Vehicle (FFV/STV). The ground support system capability to support STS operations at VLS will be verified during the Activation phase. FVV/STV is one of the major tests conducted during the Activation. The FVV will consist of Orbiter 101 (Enterprise), a lightweight External Tank, and an inert set of Solid Rocket Motors assembled to a flight set of SRB structures. One aft skirt in the stack will be the structural test article. The System Test Vehicle (STV) is essentially a mechanical payload.

The objectives of the FVV/STV test are:

(1) Verify operational and handling interfaces between the various station sets and FVV/STV hardware. Of primary interest in operational interface verification is the demonstration of the vehicle erection and mating technique as well as the transportation routes.

(2) Validate operational plans and procedures.

(3) Ground crew training.

Each element of the FVV/STV will arrive at VLS and be processed through its respective station set. This paper will outline only FVV/STV testing at the launch pad.

FVV/STV launch pad operations will consist of SRB stacking, ET erection and Orbiter mating, and STV installation. Structural dynamic response (Twang) tests will be performed following the completion of SRB stacking. After completion of the Twang tests, the Forward Assemblies are stacked and the Influence Coefficient Test is completed. The ET will be positioned in the SAB, lifted, rotated to the vertical position, and mated with the SRBs stacked on the launch mount. The Orbiter will then be positioned, attached to the SAB lifting support equipment and lifted, rotated to the vertical position, and then mated to the ET/SRB stack on the launch mount. Operations methodology, clearances, and interfaces will be verified during the erection and mating operation.

The STV will be used as the payload verification vehicle for the PPR/PCR. The STV will provide checks for all clearances, interfaces and mechanical handling capabilities in the PPR/PCR. Finally, the STV will be transferred from the PPR to the PCR. The PCR will move foward through the SAB to the launch mount and transfer the STV into the Orbiter bay. Payload servicing, maintenance and equipment installation/ removal within the payload bay will be checked. Then the STV will be removed from the Orbiter bay to the PCR, and from the PCR to the PPR to verify the capability to remove a payload.

Upon completion of all FVV/STV operations, the Orbiter 101, SRBs and ET will be destacked and returned to their respective station set. Orbiter 101 will be returned to NASA by the SCA.

VLS Integrated Assessed Timeline. Vandenberg Launch and Landing Site Shuttle Turn- around Analysis Report (VSTAR) is a formal presentation of results of KSC STAR and analytical studies of the ground turnaround processing capability under development by the USAF at Vandenberg Air Force Base (Figure 19). Those analyses and associated activities are performed by, or under auspices of, the VLS Shuttle Turnaround Analysis Group (VSTAG). The Launch Pad Station Set integrated assessed timeline have been developed by VSTAG (Figure 20).

Critical review of all tasks necessary to the ground processing effort is conducted on a continuing basis by the VSTAG. The reader is cautioned to review the ground rules that were used in the analysis before using Figure 19 and Figure 20 for other planning. Throughout the duration of the VLS program, continuous effort will be directed towards reducing the turn-around time. This will be accomplished through an enhancement program and may include suggested redesign, design changes, requirement changes, and additional or new support equipment.

VLS Program Plan. The activation of VLS facilities has been underway for several years. Figure 21 is an outline of the completely activated and partially activated facilities. The VLS initial launch capability of the Space Shuttle is 15 October 1985.

REFERENCES

1. "Space Transportation System Specification, Performance and Design Requirements, VAFB Ground Support System," Volume 2, SSS-STS-100.

2. "Space Transportation System Ground Support System Integration - Ground Operations Plan", Martin-Marietta Corporation

3. "Station Set Specifications"

4. Vandenberg Shuttle Turnaround Analysis Report, VSTAR 08

5 SRB Processing Plan, United Space Boosters, Inc.

6. "Ground Support Systems to Payload Interfaces" Volume 1, IFS-PL-001 Martin Marietta Corporation.

7. Conversations with Members of the Technical Staff, Shuttle Activation and Operations Directorate, The Aerospace Corporation, May 1983

FIGURES

Figure No.

1. Launch Azimuths and Orbital Inclinations
2. Space Shuttle Vehicle
3. Vandenberg Station Set Geographic Locations
4. Vandenberg Space Shuttle Operations Flow
5. Orbiter Servicing Convoy
6. Orbiter and Maintenance Checkout Facility
7. Orbiter on Transporter
8. Vandenberg Launch Processing System
9. ET Processing and Storage Facility
10. SRB Refurbishment and Subassembly Facility
11. Launch Pad
12. Solid Rocket Motor Transporter
13. SRB Stacking And Alignment
14. ET on Transporter
15. ET Erection and Mating
16. Orbiter Erection and Mating
17. Payload Preparation Room
18. SRB Retrieval and Disassembly Facility
19. VLS Integrated Assessed Timeline
20. Launch Pad Integrated Assessed Timeline
21. VLS Program Plan

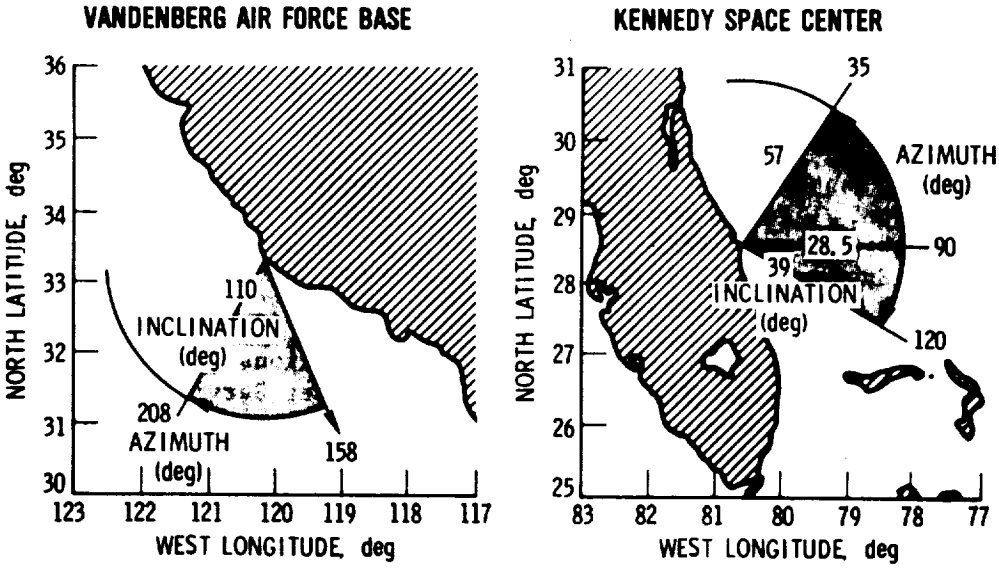


Figure 1. Launch Azimuths and Orbital Inclinations

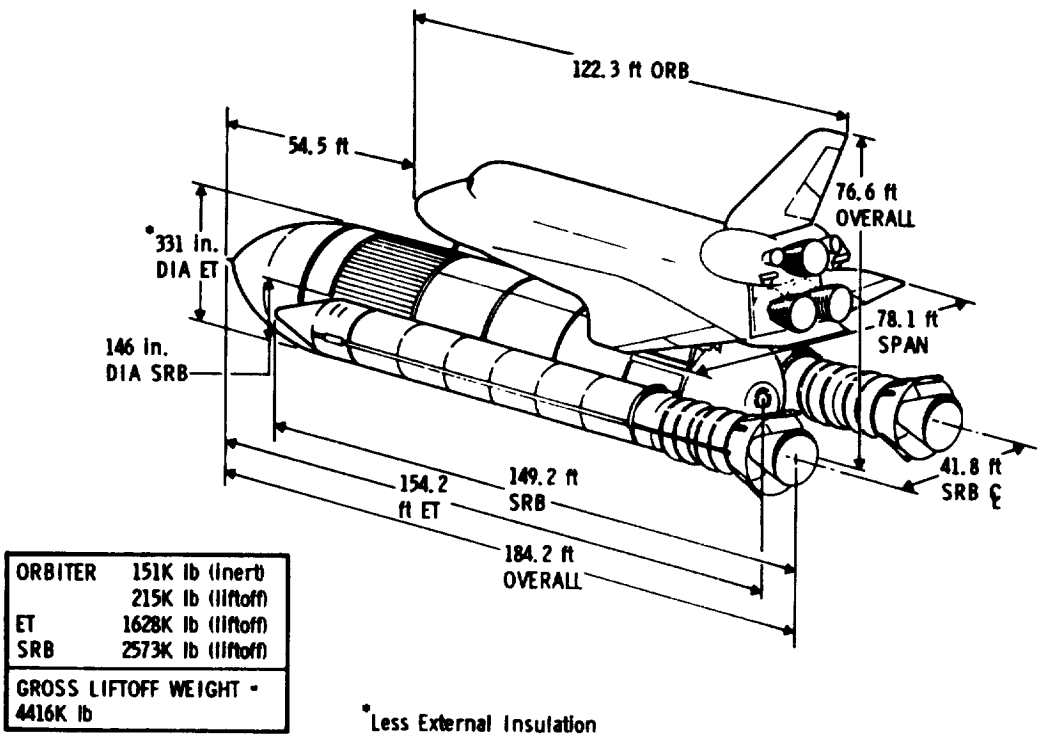


Figure 2. Space Shuttle Vehicle

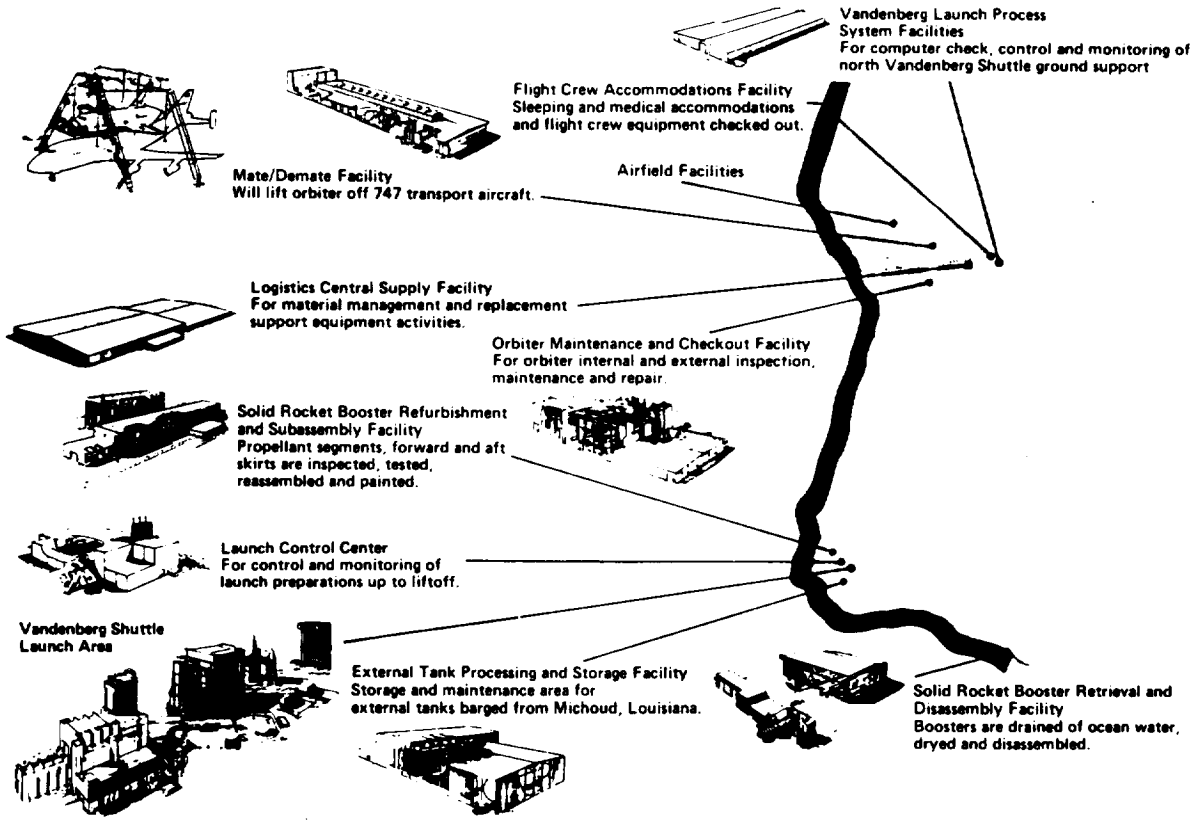


Figure 3. Vandenberg Station Set Geographic Locations

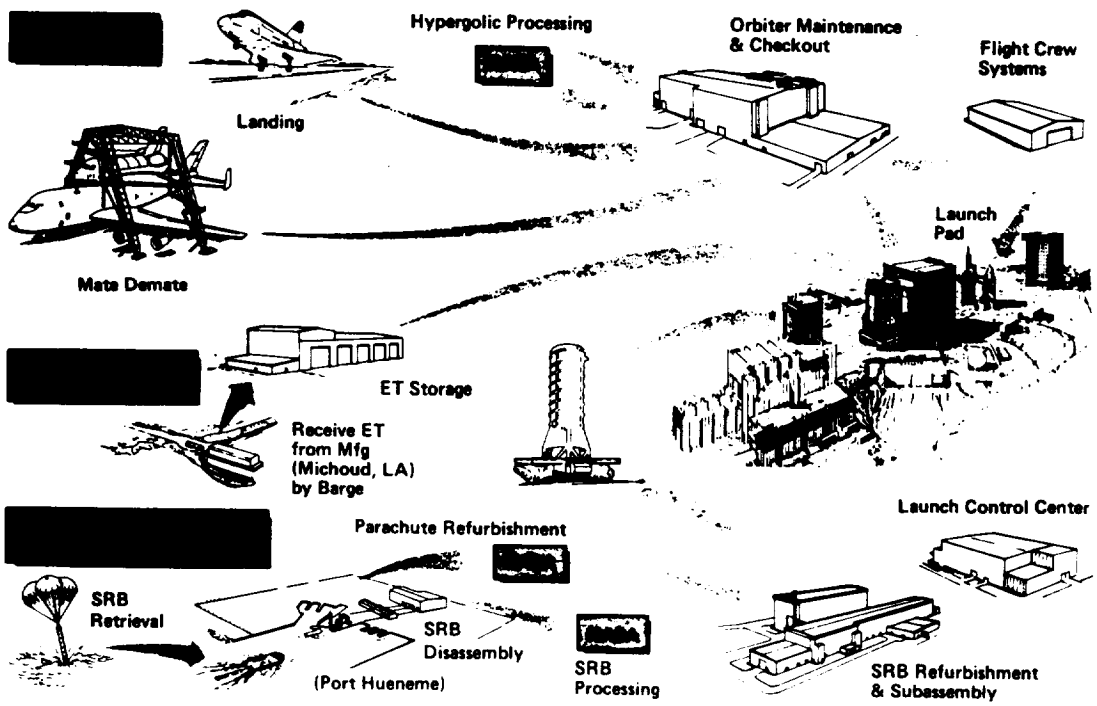


Figure 4. Vandenberg Space Shuttle Operations Flow

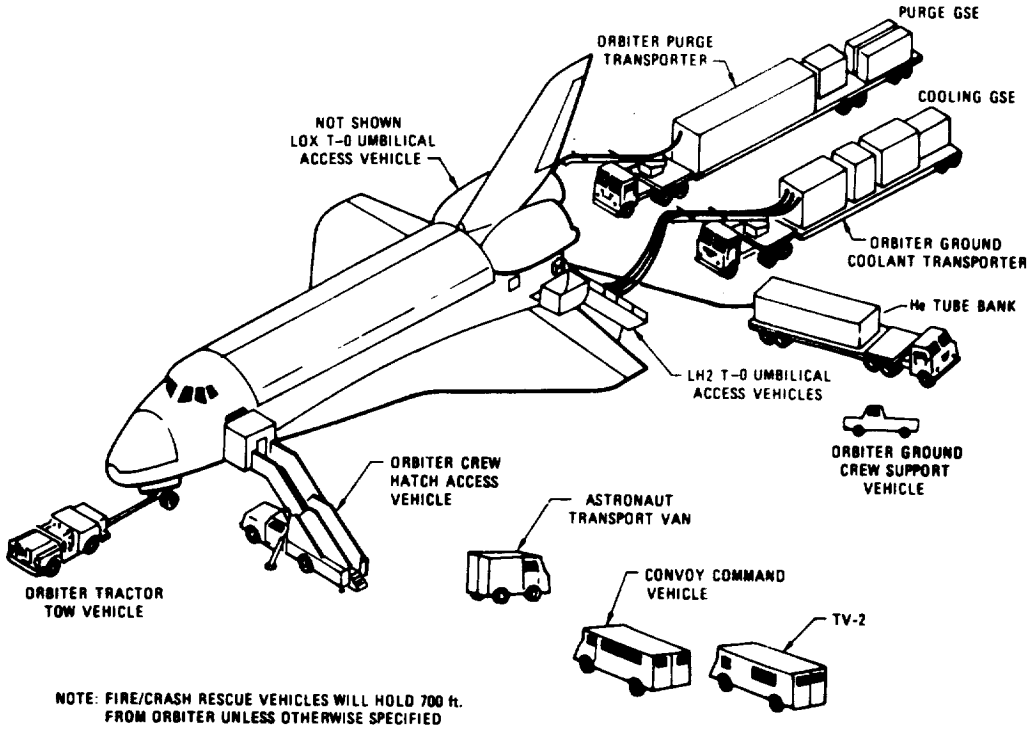


Figure 5. Orbiter Servicing Convoy

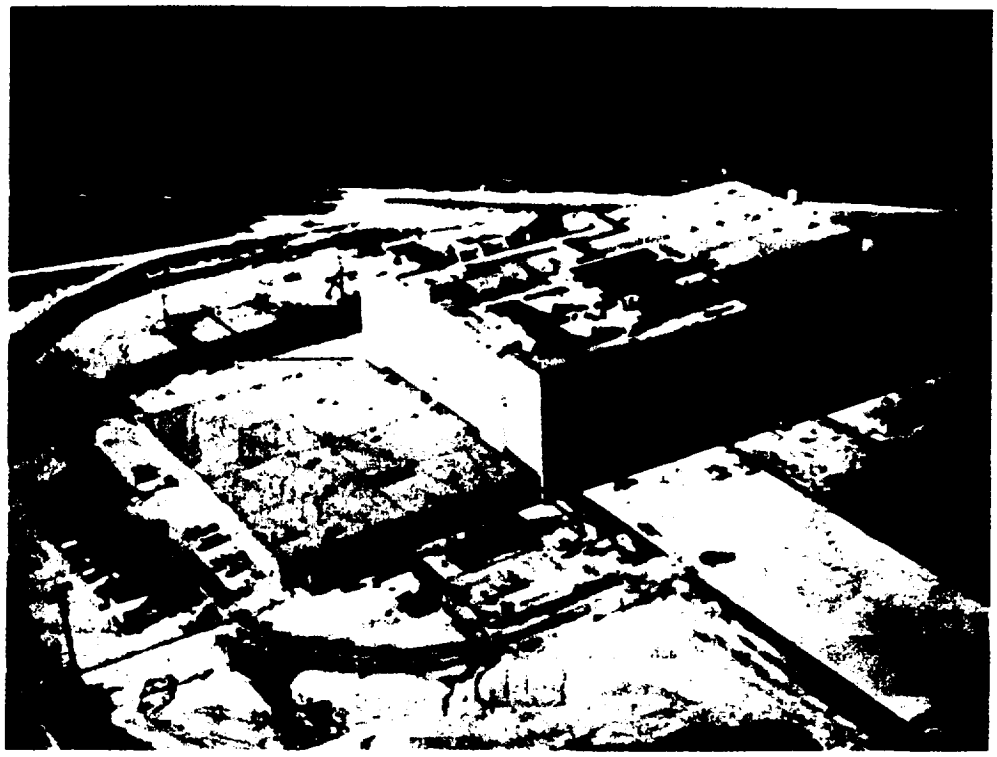


Figure 6. Orbiter and Maintenance Checkout Facility

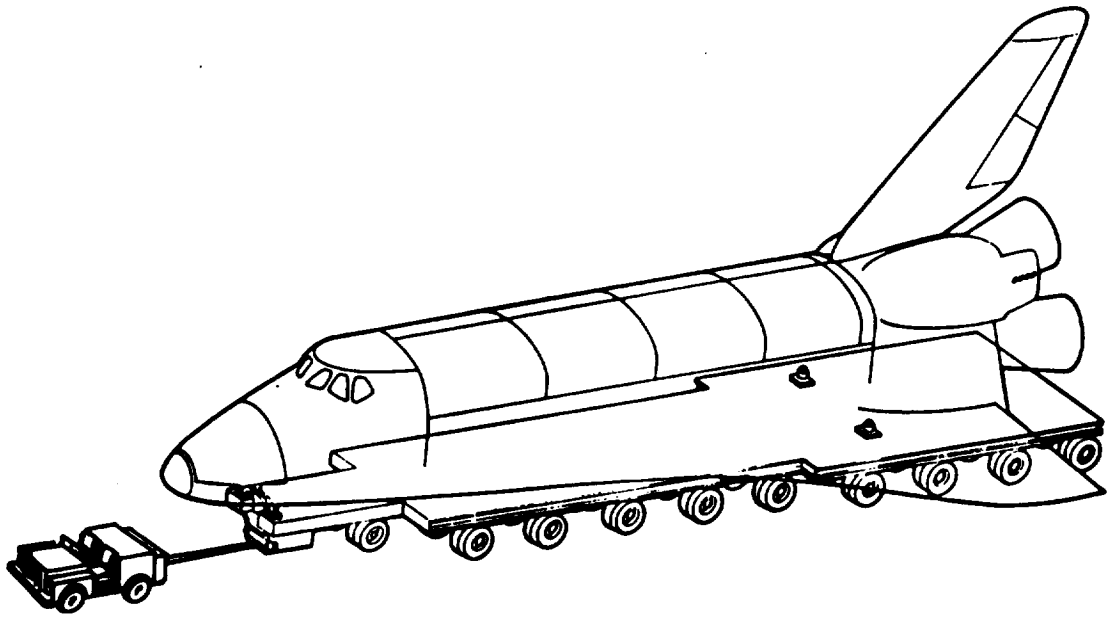


Figure 7. Orbiter on Transporter

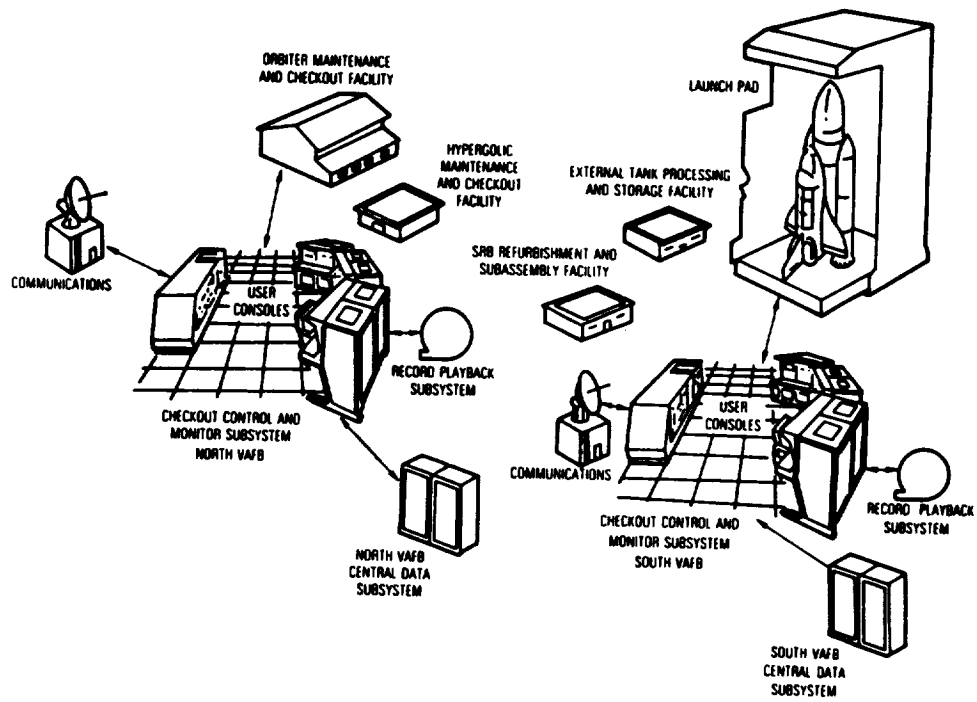


Figure 8. Vandenberg Launch Processing System

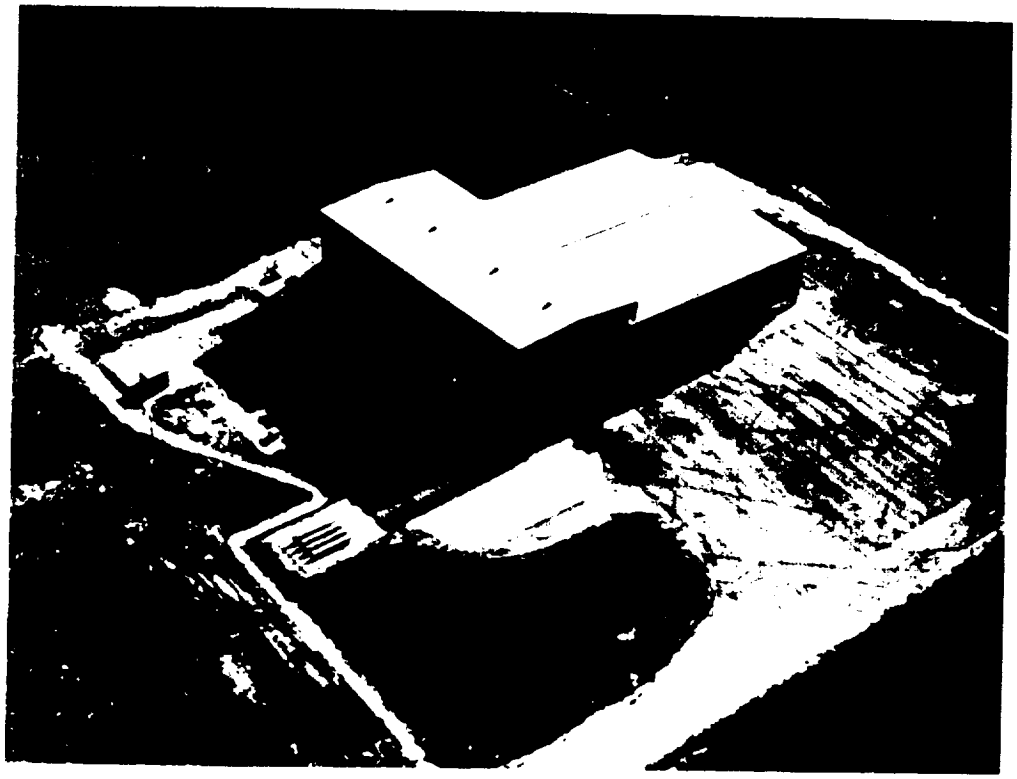


Figure 9. ET Processing and Storage Facility

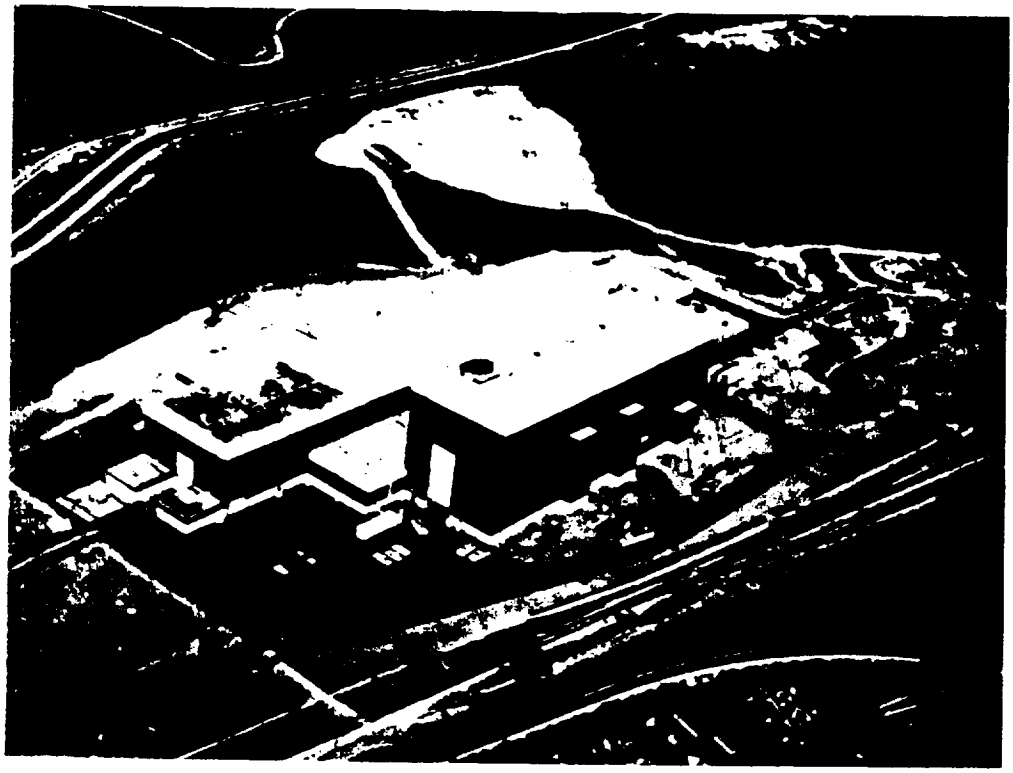


Figure 10. SRB Refurbishment and Subassembly Facility

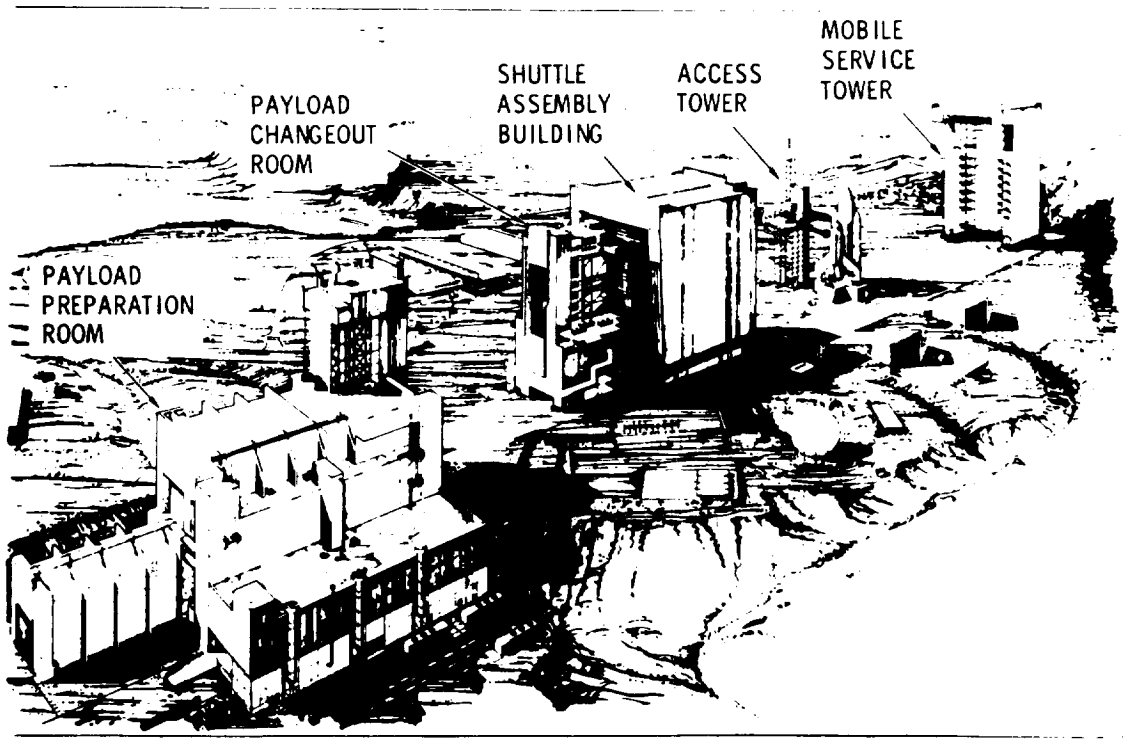


Figure 11. Launch Pad

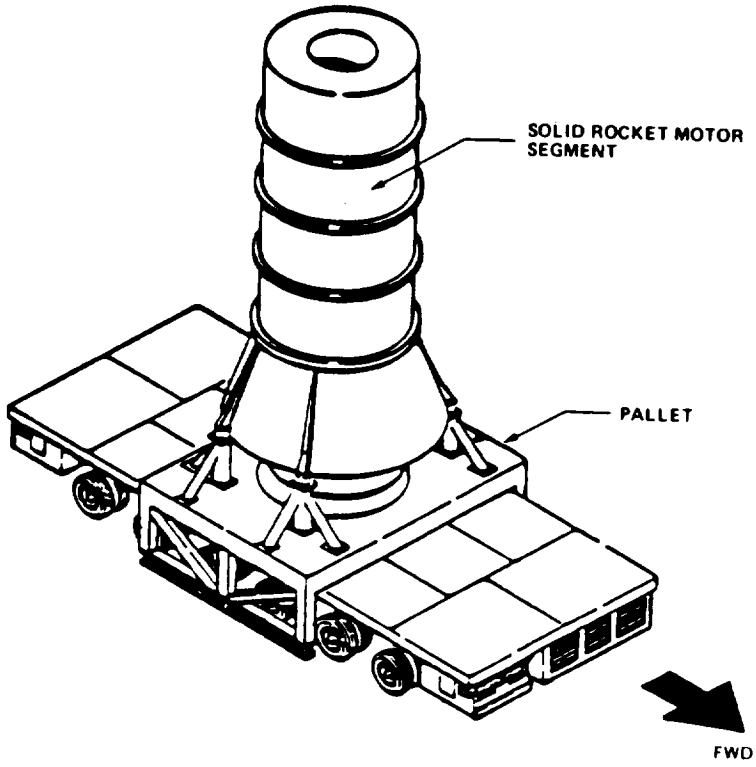


Figure 12. Solid Rocket Motor Transporter

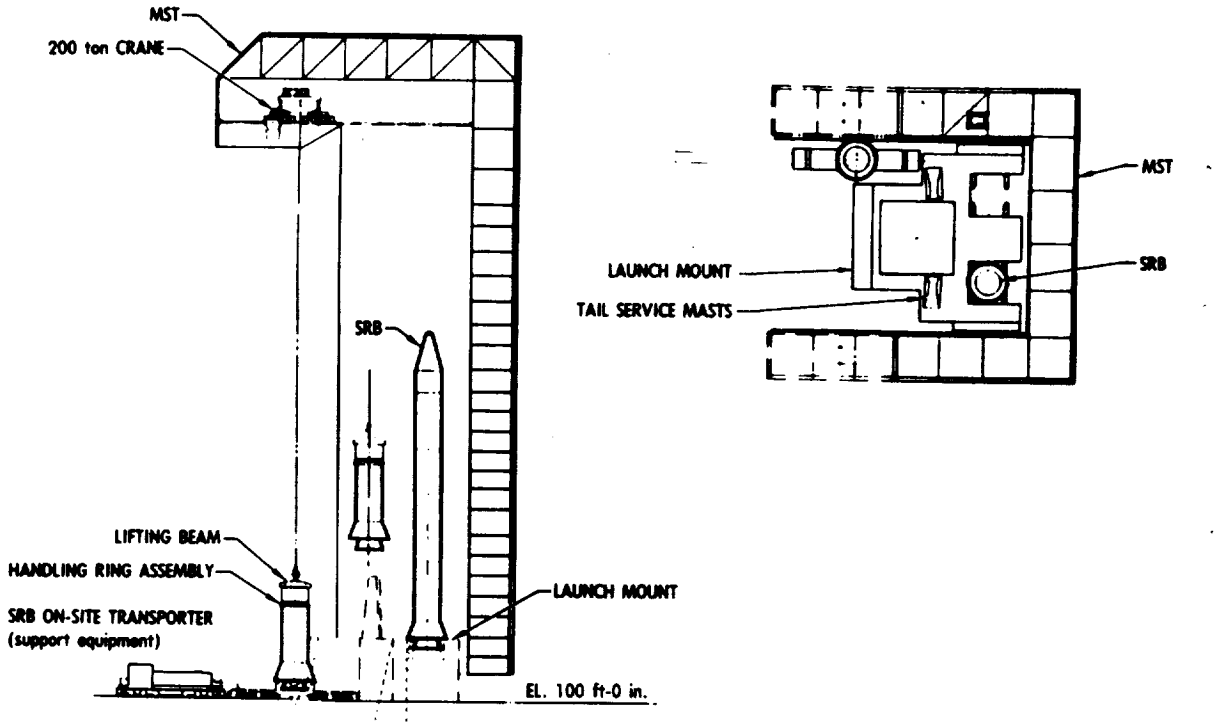


Figure 13. SRB Stacking And Alignment

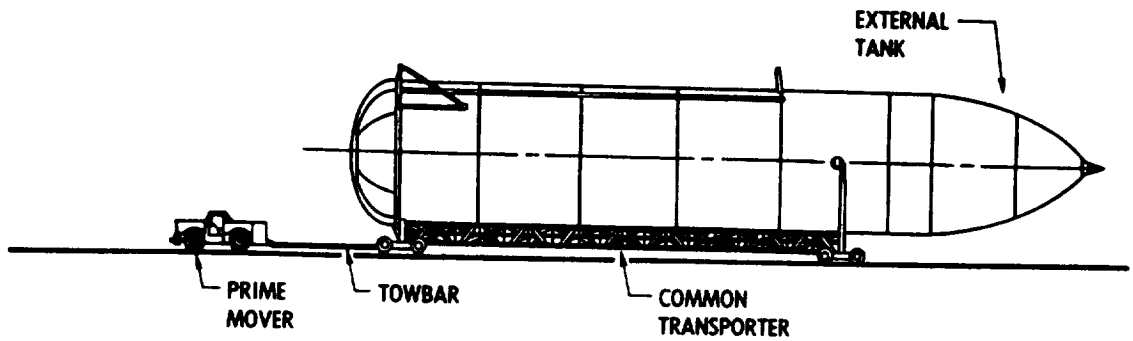


Figure 14. ET on Transporter

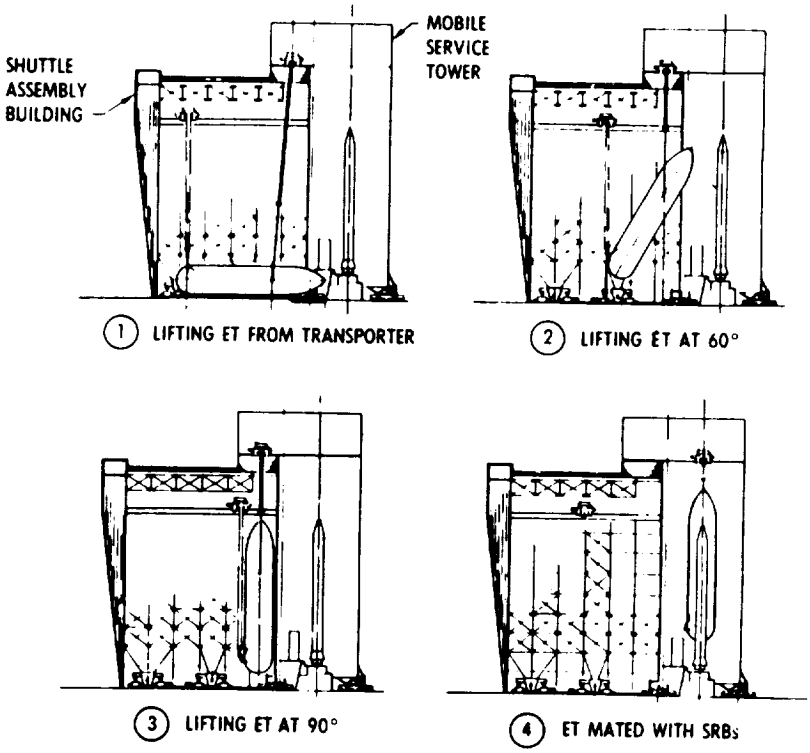


Figure 15. ET Erection and Mating

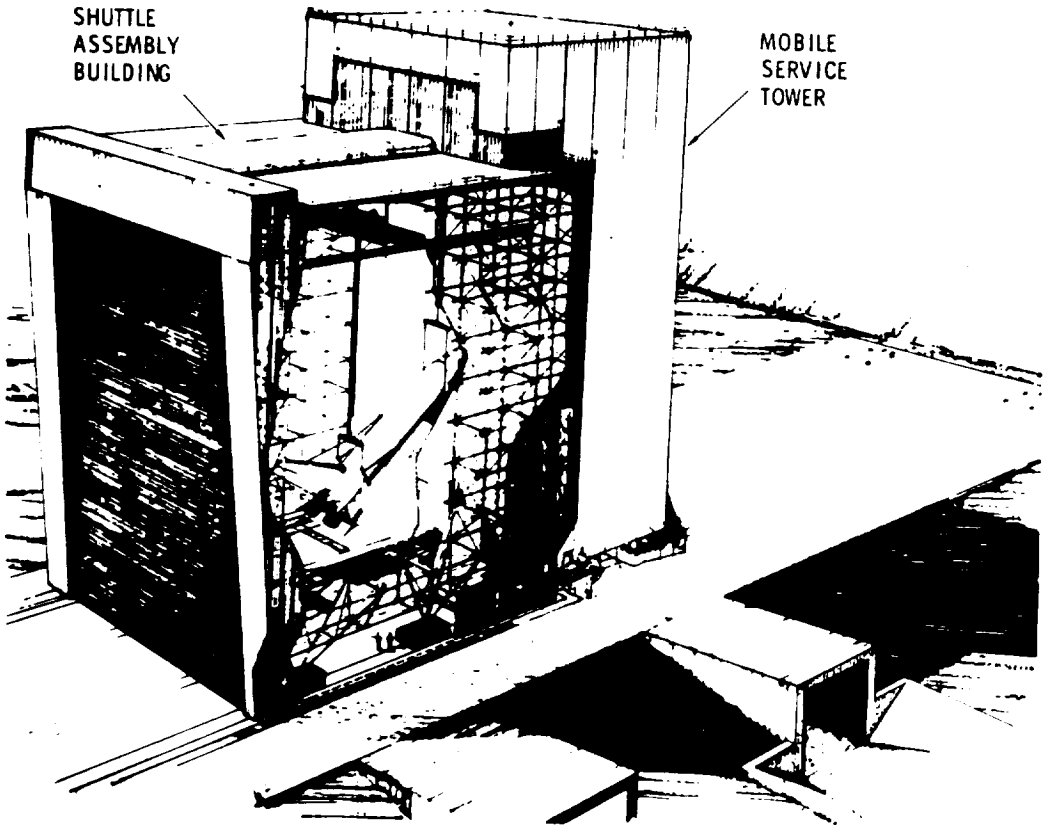


Figure 16. Orbiter Erection and Mating

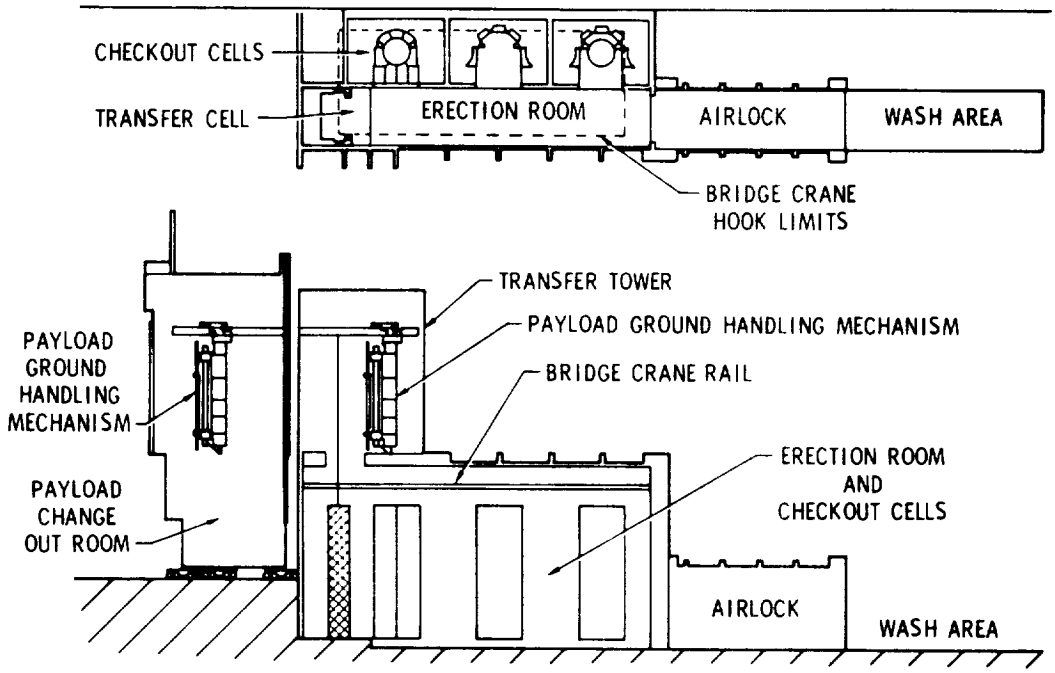


Figure 17. Payload Preparation Room

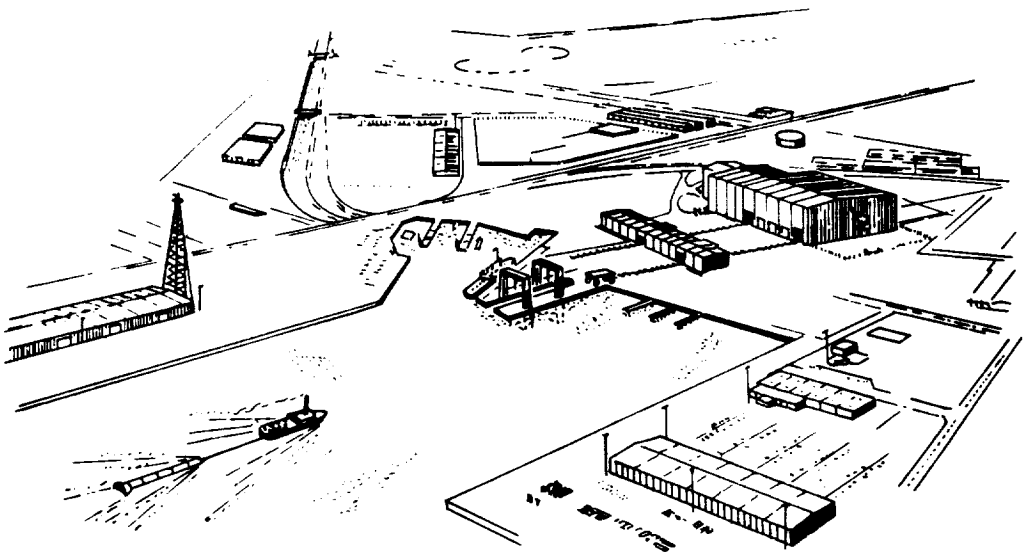


Figure 18. SRB Retrieval and Disassembly Facility

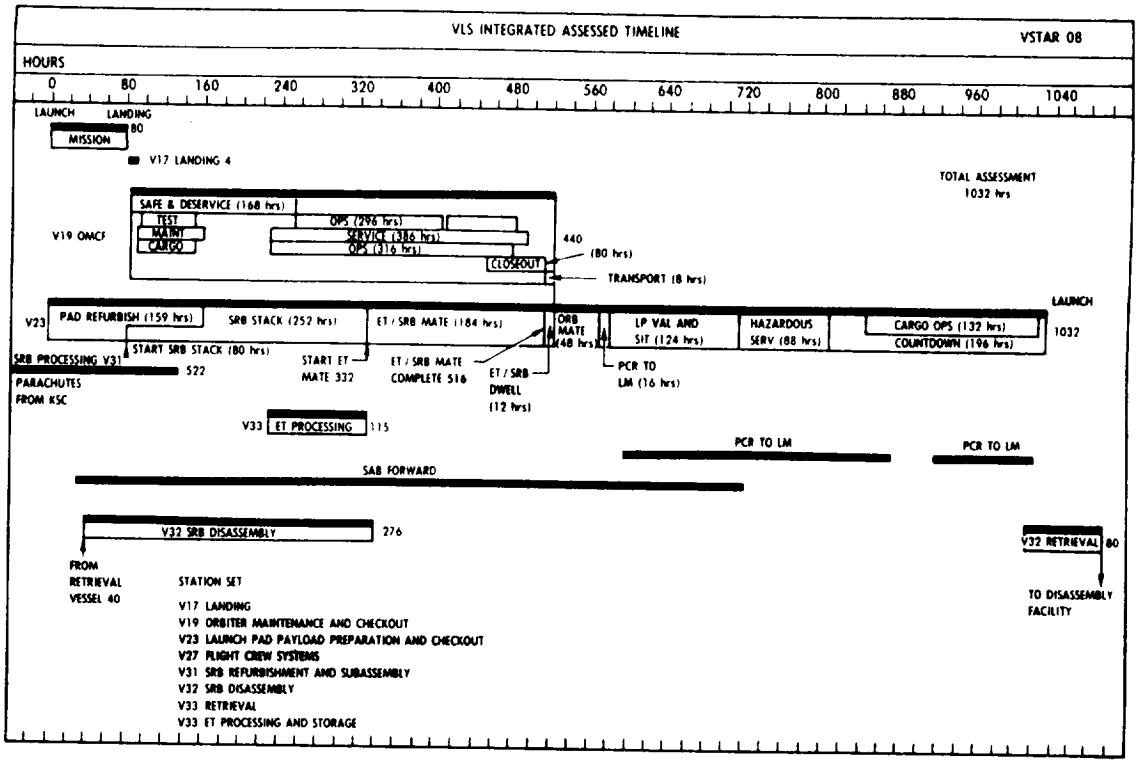


Figure 19. VLS Integrated Assessed Timeline

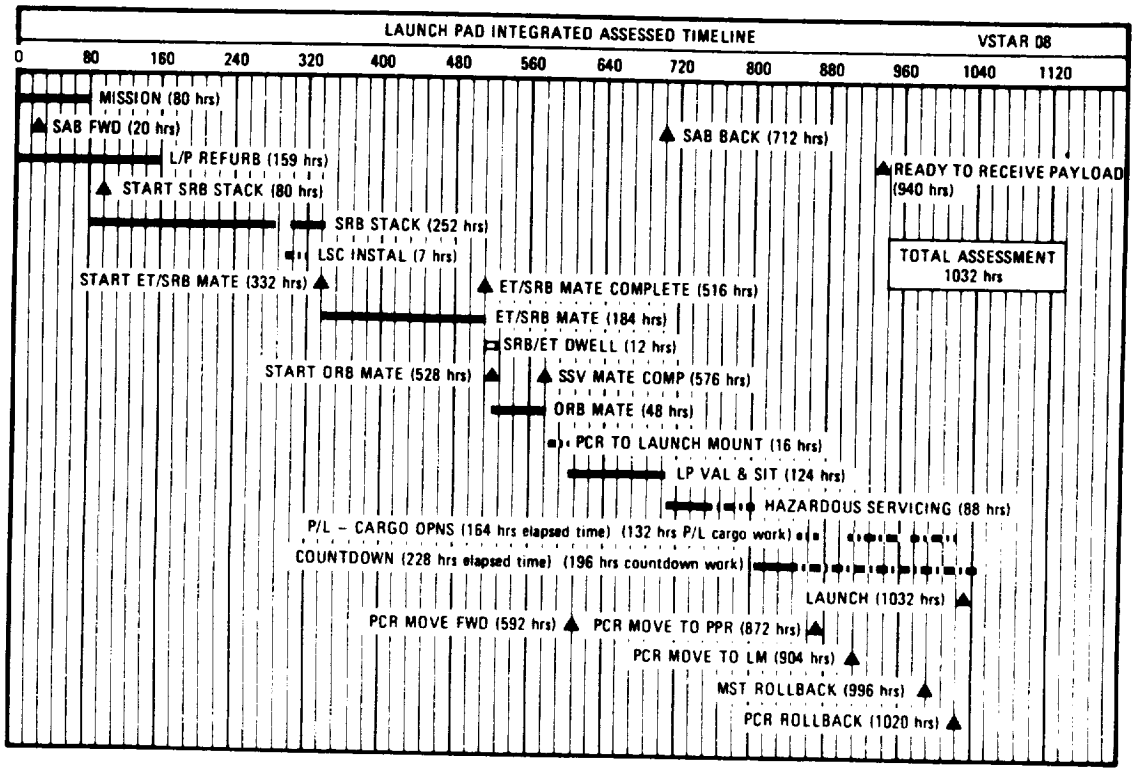


Figure 20. Launch Pad Integrated Assessed Timeline

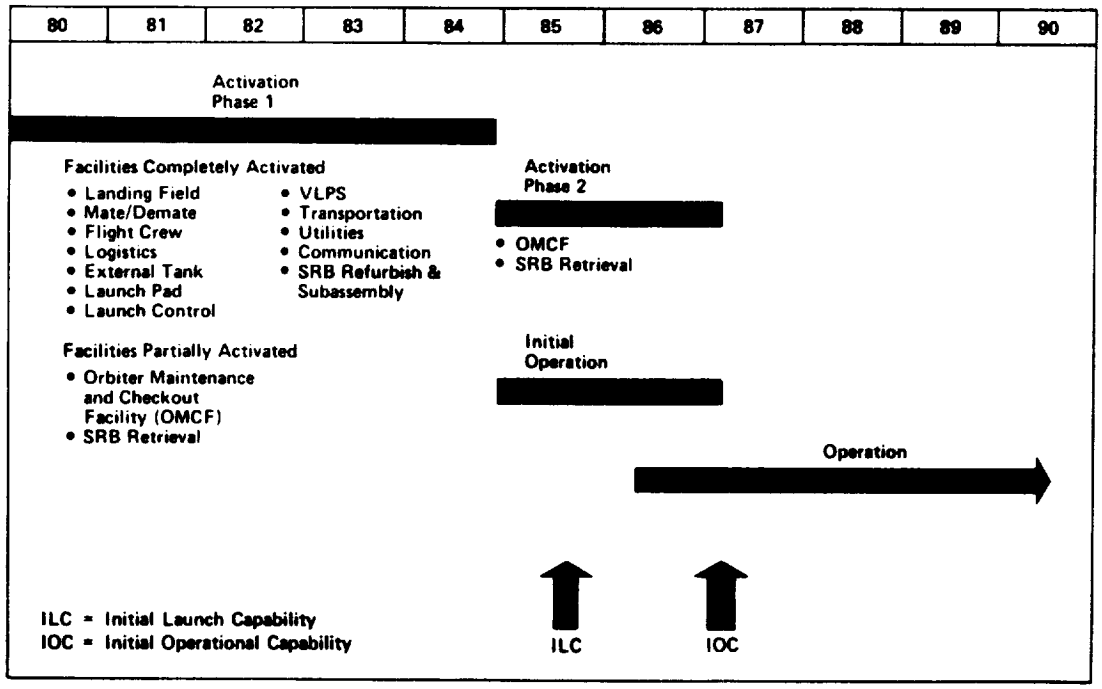
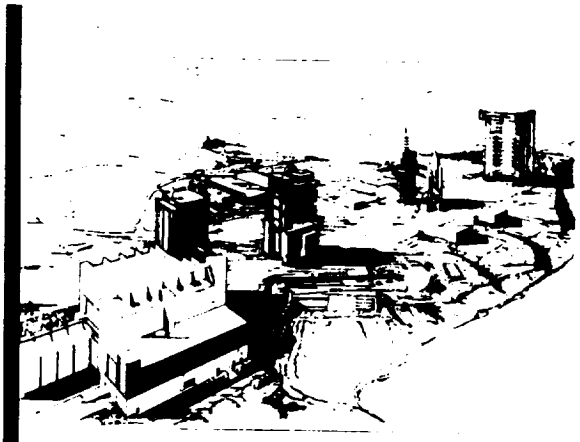
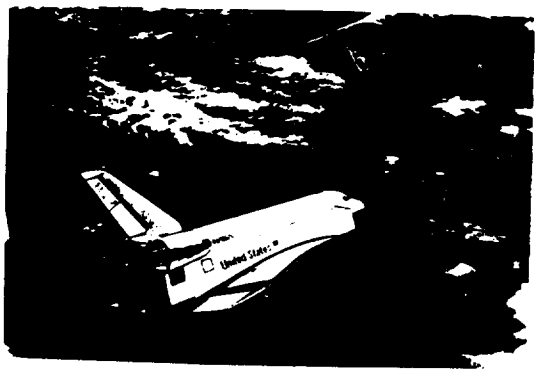


Figure 21. VLS Program Plan

DOD Space Shuttle Operations at Vandenberg Launch and Landing Site





Peter L. Portanova
Director, System Requirements
Shuttle Activation and Operations Directorate
The Aerospace Corporation

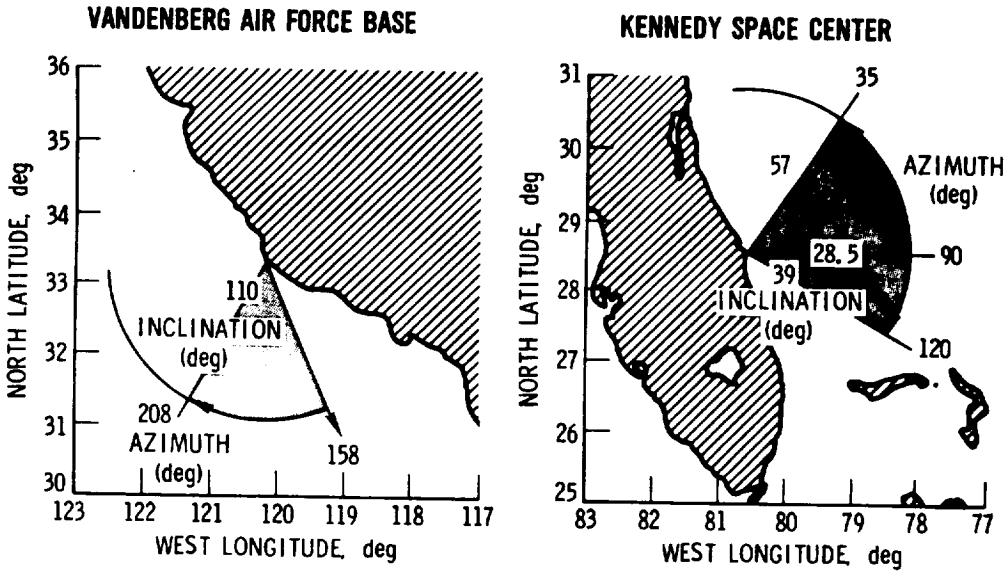


Figure 1. Launch Azimuths and Orbital Inclinations

- LAUNCH AZIMUTHS AND ORBITAL INCLINATIONS FROM VANDENBERG AIR FORCE BASE ARE COMPLIMENTARY TO THOSE AVAILABLE FROM KENNEDY SPACE CENTER

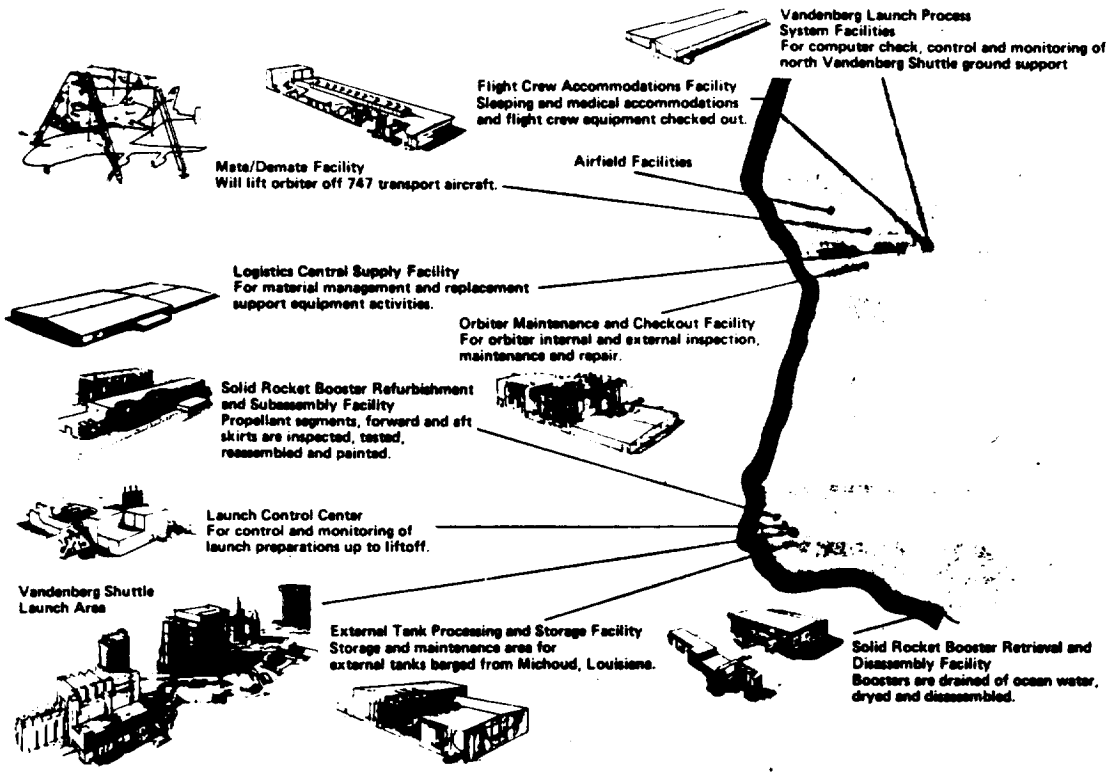


Figure 3. Vandenberg Station Set Geographic Locations

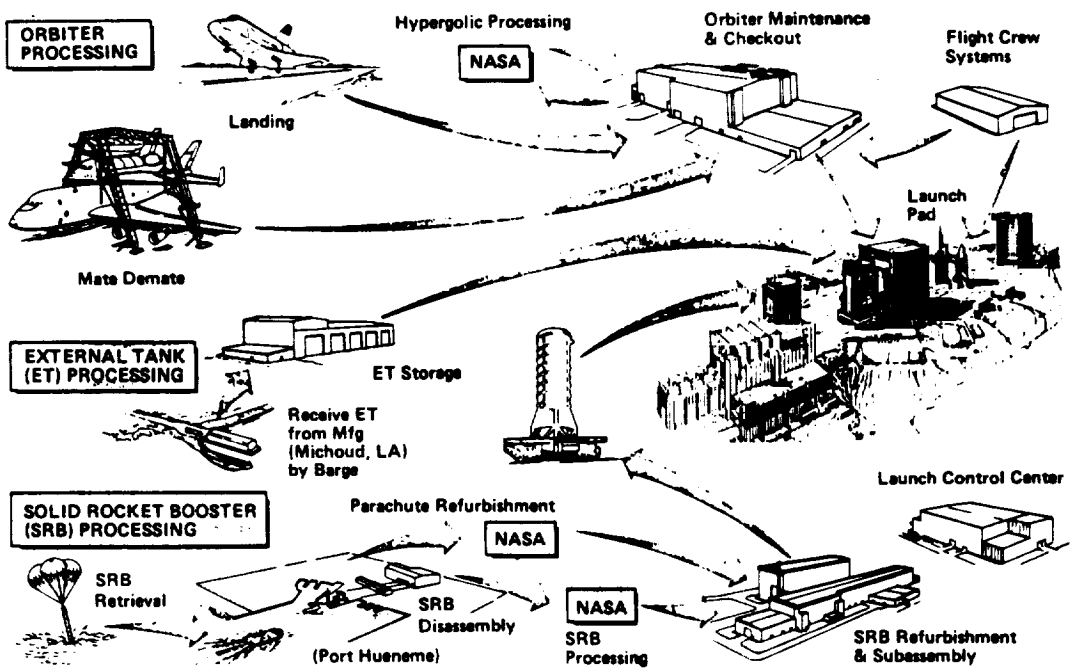


Figure 4. Vandenberg Space Shuttle Operations Flow

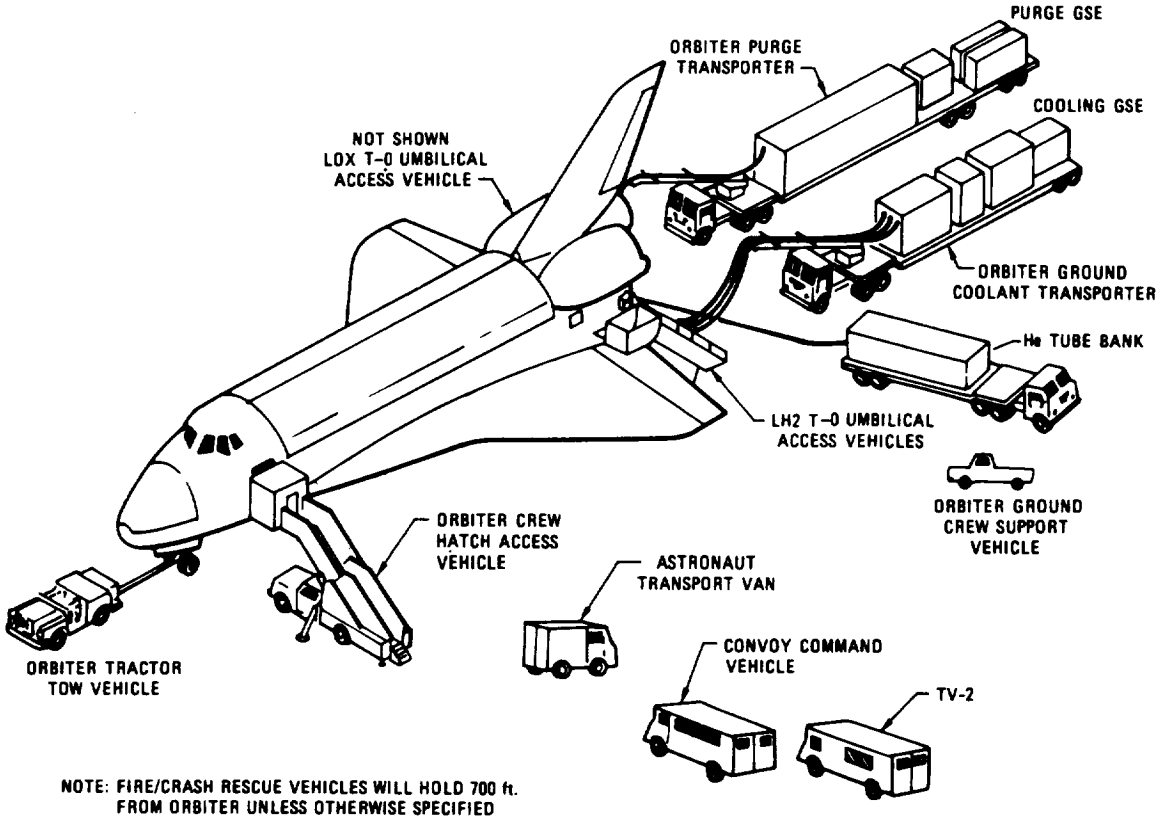


Figure 5. Orbiter Servicing Convoy

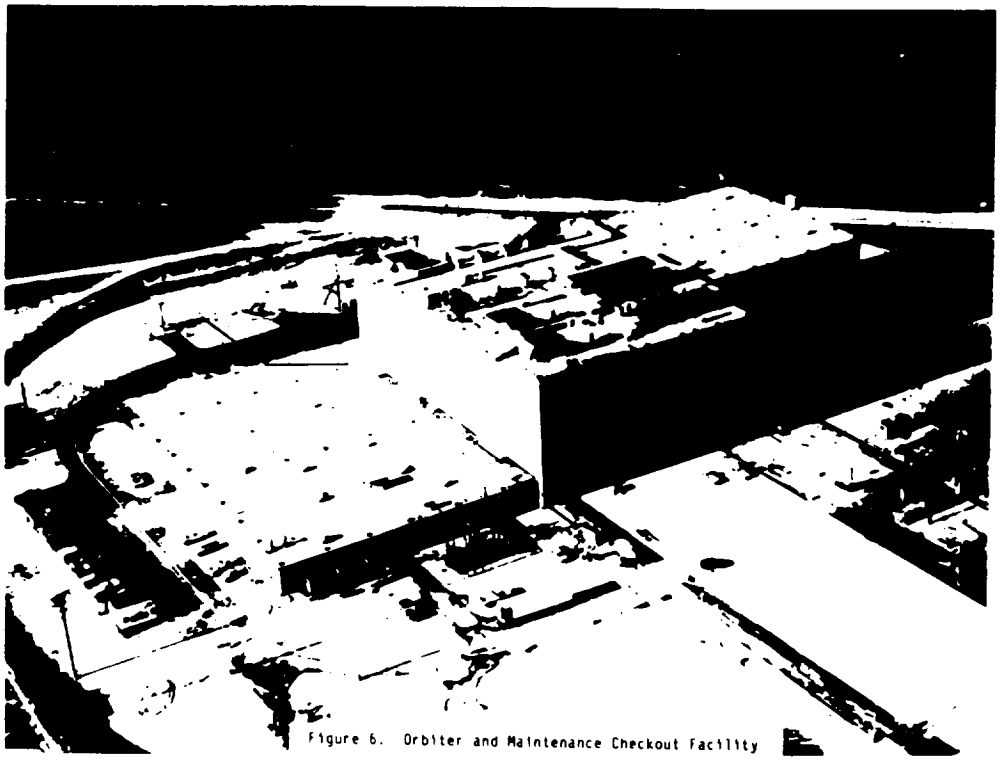


Figure 6. Orbiter and Maintenance Checkout Facility

NORTH VANDENBERG AIR FORCE BASE

- 0 MATE/DEMATE
  - / UNIQUE DESIGN
  - / MATE/DEMATE ORBITER TO/FROM SHUTTLE CARRIER AIRCRFT MODIFIED 747
- 0 LANDING FACILITY
  - / 15,000 FOOT RUNWAY WITH NAVIGATIONAL AIDS
  - / POST LANDING OPERATIONS
    - o UPON COMPLETION OF LANDING ROLLOUT THE ORBITER IS MET BY A CONVOY OF SERVICING VEHICLES
    - o ORBITER INSPECTION
    - o EGRESS FLIGHT CREW AND HOOKUP ORBITER SERVICING CONVOY
    - o TOW ORBITER TO OMCF

HYPERGOLIC MAINTENANCE AND CHECKOUT FACILITY

- / ACTIVATION HAS BEEN DEFERRED

NORTH VANDENBERG AIR FORCE BASE

ORBITER MAINTENANCE & CHECKOUT FACILITY (OMCF)

- / HIGH BAY HANGAR-TYPE AREA
- / MULTI-LEVEL STEEL ACCESS PLATFORM
- / MEZZANINE AND ENVIRONMENTAL ENCLOSURE - CLASS 100,000
- / HAZARDOUS VAPOR DETECTION SYSTEMS
- / 60 TON BRIDGE CRANE WITH TWO 37.5 TON TROLLEY HOISTS FOR PAYLOAD HANDLING
- / ONE 10 TON BRIDGE CRANE, ONE 15 TON BRIDGE CRANE FOR MODULE HANDLING
- / LOW BAY STORAGE & OPERATIONAL SUPPORT AREA
- / HARDENED CLASS 100,000 HIGH BAY PAYLOAD DESERVICING AREA
- / PERFORM ORBITER LIMITED MAINTENANCE & CHECKS
- / REFURBISH THERMAL PROTECTION SYSTEMS
- / PAYLOAD REMOVAL & DESERVICING, SPACELAB INSTALLATION
- / INSTALL AND REMOVE MISSION EQUIPMENT KITS
- / INSTALL AND REMOVE HYPERGOLIC MODULES & COMPONENTS
- / ORBITER PREPARATIONS AND TRANSPORT TO LAUNCH PAD

NORTH VANDENBERG AIR FORCE BASE

- 0 FLIGHT CREW SYSTEMS FACILITY
  - / EXISTING BUILDING 8505 FOR FLIGHT CREW ACCOMMODATIONS
  - / PROVIDES LIVING QUARTERS, COMMODITIES AND OTHER CREW ACCOMMODATIONS SUPPORT CREW TRAINING AND MISSION BRIEFINGS/DEBRIEFINGS
  - / EXISTING BUILDING 6710 FOR FLIGHT CREW EQUIPMENT
- 0 TRANSPORTATION STATION SET
  - / ROADS AND SUPPORT EQUIPMENT, I.E., RUBBER TIRE TRANSPORTERS, ETC.
  - / PROVIDES THE TRANSPORTATON ROUTE FROM NVAFB TO SVAFB AND BETWEEN STATION SETS
- 0 LOGISTICS FACILITY
  - / CENTRAL SUPPLY FACILITY - 191,300 SQUARE FEET
  - / INVENTORY CONTROL, ACQUISITION, STORAGE AND DELIVERY OF EQUIPMENT, SPARES, REPAIRABLES, MATERIALS AND CONSUMABLES
  - / MANAGEMENT OF WORK CONTROL SYSTEMS AND MAINTENANCE DOCUMENTATION

NORTH VANDENBERG AIR FORCE BASE

- 0 VANDENBERG LAUNCH PROCESSING SYSTEM (VLPS) BUILDING 8510
  - / NORTH VLPS CONTROL CENTER
  - / CHECKOUT, CONTROL AND MONITOR SUBSYSTEM - MOD COMP II
  - / CENTRAL DATA SUBSYSTEM - HONEYWELL 66/80
  - / RECORD AND PLAYBACK SUBSYSTEM
  - / SOFTWARE DEVELOPMENT
  - / CHECKOUT, CONTROL AND MONITOR NVAFB OPERATIONS
  - / DATA BASE ADMINISTRATION
  - / TEST DATA ANALYSIS

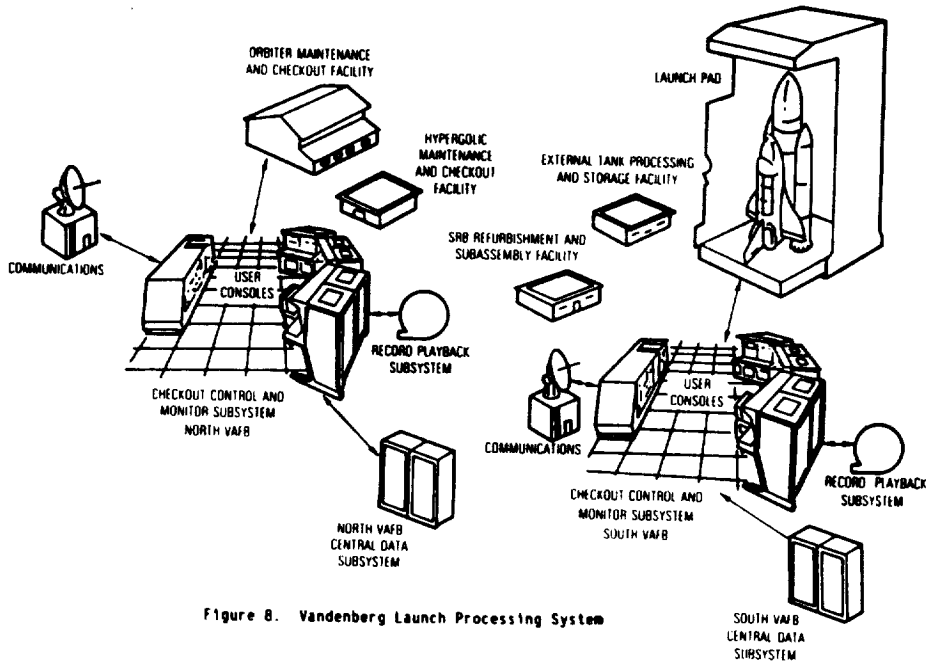
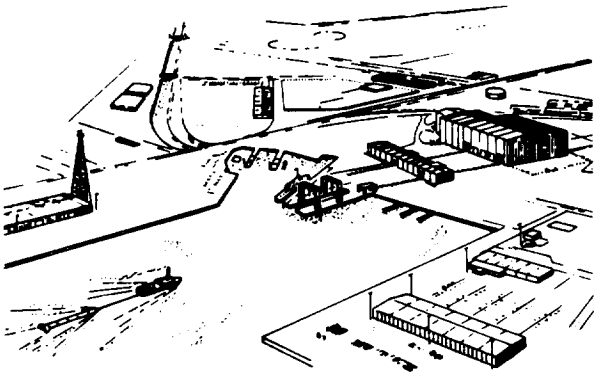


Figure 8. Vandenberg Launch Processing System

SOUTH VANDENBERG AIR FORCE BASE

- o SRB RETRIEVAL AND DISASSEMBLY FACILITY
  - / LOCATED AT PORT HUENEME
  - / CONCRETE RETRIEVAL WHARF AND SLIP
  - / WASH BUILDING
    - o HYDROLASER CLEANING
    - o POTABLE AND DEIONIZED WATER
    - o SURFACTANT WASH AND RINSE
  - / DISASSEMBLY BUILDING
    - o 40 TON AND 16 TON CRANES
  - / WATER TREATMENT FACILITIES
    - o 20,000 GALLON DEIONIZED WATER SYSTEM
    - o 100,000 GALLON INDUSTRIAL WASTE SYSTEM
  - / INITIAL WASH AND DRAIN RECOVERED SRB
  - / DISASSEMBLE FORWARD AND AFT SKIRTS, FRUSTRUM, AND COMPONENTS
  - / FINAL WASH, DRY AND PRESERVE COMPONENTS
  - / RETURN EMPTY SRM TO MANUFACTURER TO BE REFURBISHED AND REFILLED
  - / SHIP FORWARD SKIRT, AFT SKIRT, PARACHUTES AND OTHER SRB COMPONENTS TO KSC FOR REFURBISHMENT



SRB Retrieval and Disassembly Facility - Port Hueneme Harbor

SOUTH VANDENBERG AIR FORCE BASE

- o SRB REFURBISHMENT AND SUBASSEMBLY FACILITY
  - / HIGH BAY MAJOR ASSEMBLY BUILDUP AND STORAGE AREA
    - o 200 TON BRIDGE CRANE
    - o 125 TON FIXED HOIST
  - / STORAGE AND SUPPORT AREA
  - / LOW BAY AND RECEIVING/INSPECTION, ASSEMBLY AREA
    - o TWO 15 TON BRIDGE CRANES
  - / ADMINISTRATIVE/SUPPORT AREA
  - / RECEIVE, INSPECT, SOME CHECKOUT AND STORE MAJOR SRB COMPONENTS
  - / MATE AFT SKIRT ASSEMBLY/NOZZLE EXTENSION WITH AFT SRM SEGMENT
  - / AFT AND FORWARD ASSEMBLY ELECTRICAL SYSTEM CHECKOUT

SOUTH VANDENBERG AIR FORCE BASE

- o LAUNCH PAD
  - / PAYLOAD PREPARATION ROOM
  - / PAYLOAD CHANGEOUT ROOM
  - / SHUTTLE ASSEMBLY BUILDING
  - / LAUNCH MOUNT
  - / ACCESS TOWER
  - / MOBILE SERVICE TOWER
  - / SUPPORT EQUIPMENT BUILDING HYPERGOLIC
  - / EMERGENCY EGRESS SYSTEM
  - / STORAGE AREAS FOR CRYOGENIC AND HYPERGOLIC PROPELLANTS AND GASES
  - / ICE SUPPRESSION SYSTEM

- / SSV BUILDUP CYCLE
  - o MST IS ROLLED FORWARD TO THE LAUNCH MOUNT
- / SRM STACKING
  - o SRM SEGMENTS ARRIVE AT THE LAUNCH PAD ON SELF-PROPELLED RUBBER-TIRED TRANSPORTERS
  - o AFTER PREPS, THE FIRST AFT BOOSTER ASSEMBLY IS HOISTED FROM ITS TRANSPORTER AND STACKED USING THE MST BRIDGE CRANE
  - o SECOND SRB AFT BOOSTER ASSEMBLY IS SIMILARLY STACKED AND ALIGNMENT VERIFIED
  - o CENTER SEGMENTS, FORWARD SEGMENTS AND FORWARD ASSEMBLIES ALL ALTERNATELY STACKED AND JOINTS LEAK CHECKED
  - o ADDITIONAL ALIGNMENT CHECKS
  - o DESTRUCT ORDNANCE INSTALLED AND IN SAFED CONDITION
  - o SRB TUNNEL CABLES VERIFIED, TUNNEL COVERS AND INTERFACE HARDWARE INSTALLED
  - o READY FOR ET MATING

- / PAYLOAD INSTALLATION
  - o PAYLOAD PREPARATION ROOM CONSISTS OF THREE CHECKOUT CELLS, ERECTION ROOM, AIRLOCK AND TRANSFER TOWER
  - o ONLY TWO CELLS WILL BE ACTIVATED DURING THE EARLY YEARS
  - o PAYLOAD ENTERS PPR THROUGH AIRLOCK, MOVES INTO ERECTION ROOM, LIFTED CLEAR OF TRANSPORTER BY OVERHEAD CRANES, ROTATED TO VERTICAL AND TRANSFERRED TO CHECKOUT CELL
  - o WHILE IN THE CELL THE PAYLOAD IS INTEGRATED, TESTED, AND ORDNANCE INSTALLED - FUEL LOADING IF REQUIRED
  - o OVERHEAD CRANE TRANSFERS THE COMPLETED CARGO INCLUDING STRONGBACK TO TRANSFER TOWER
  - o TRANSFER TOWER TRANSFERS PAYLOAD FROM STRONGBACK TO PGHM
  - o PGHM TRANSLATES THE CARGO FROM PPR TO PCR USING OVERHEAD RAILS
  - o PCR ROLLS UP TO THE ORBITER ON THE LAUNCH MOUNT
  - o CARGO INSTALLATION INTO THE ORBITER BAY USING PGHM

- / ET ERECTION AND MATING
  - o THE SAB IS ROLLED FORWARD TO THE LAUNCH MOUNT AND MATED WITH MST TO SEAL OFF THE LAUNCH PAD AND PROVIDE A BENIGN ENVIRONMENT - SIMILAR TO KSC/VAB
  - o TOW ET ON TRANSPORTER TO PAD
  - o ET IS LIFTED CLEAR OF THE TRANSPORTER
  - o ROTATED TO THE VERTICAL POSITION AND STABILIZED USING MST MST AND SAB CRANES
  - o MATED TO THE SRB

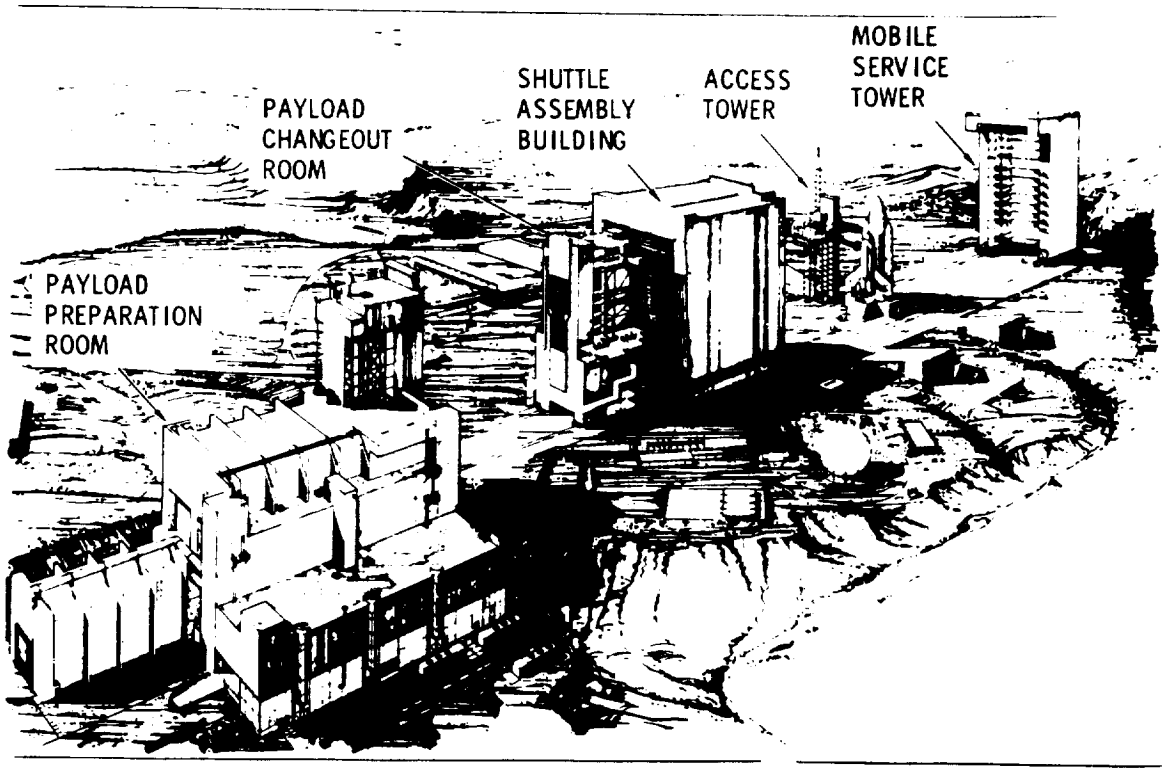


Figure 11. Launch Pad

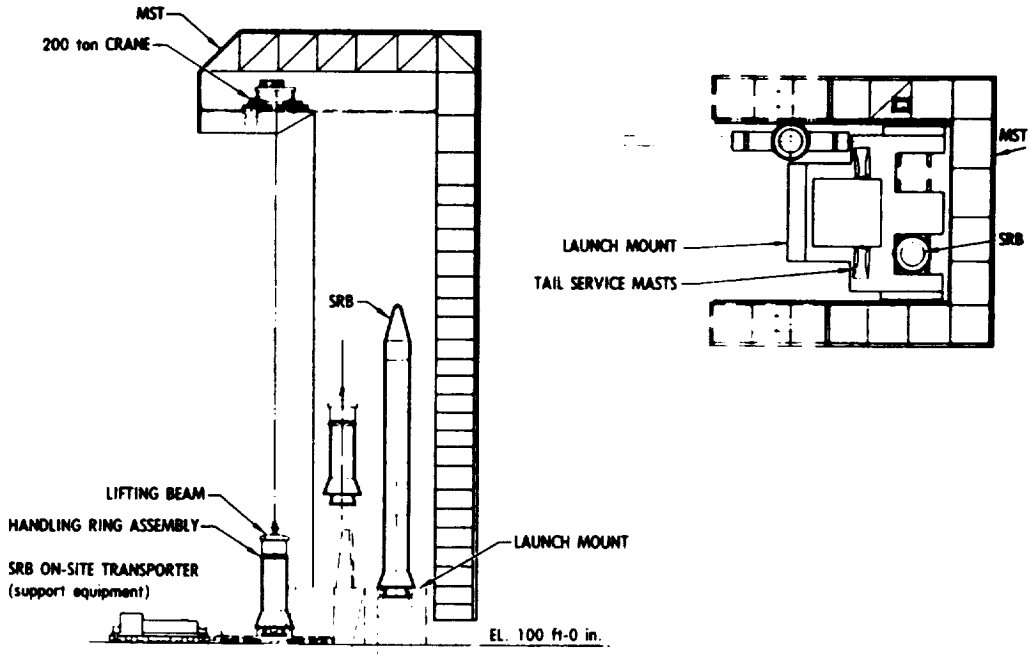


Figure 13. SRB Stacking And Alignment

- LAUNCH PAD (CONT'D)
  - / ORBITER ERECTION AND MATING
    - o ORBITER IS TRANSPORTED 17 MILES FROM OMCF TO THE LAUNCH PAD ON RUBBER-TIRED TRANSPORTER
    - o ORBITER IS LIFTED CLEAR OF THE TRANSPORTER BY MST AND SAB CRANES
    - o ROTATED TO THE VERTICAL AND STABILIZED
    - o MATED TO THE ET
    - o ALL INTERFACES ARE VERIFIED
    - o ENVIRONMENTAL CONDITIONING AND GROUND POWER APPLIED TO SSV
    - o SHUTTLE INTERFACE TEST IS PERFORMED
    - o FINAL ORDNANCE INSTALLATION, CONNECTIONS AND VERIFICATION NOW READY
    - o SAB ROLLED BACK PRIOR TO START OF HAZARDOUS SERVICING
    - o PAD CLEAR OF NON-ESSENTIAL PERSONNEL
    - o REMOTELY CONTROLLED BY THE VLPS
    - o AFT RCS, FORWARD RCS AND OMS HYPERGOLS ARE LOADED
    - o HELIUM BOTTLES ARE LOADED AND PRESSURIZED
    - o GROUND CREW RETURN AFTER HAZARDOUS SERVICING COMPLETE

SOUTH VANDENBERG AIR FORCE BASE

- LAUNCH CONTROL CENTER
  - / EXISTING TWO LEVEL REINFORCED CONCRETE BLOCKHOUSE
  - / HOUSE THE VANDENBERG LAUNCH PROCESSING SYSTEMS
  - / CHECKOUT, CONTROL AND MONITOR: SPACE SHUTTLE VEHICLE
    SUPPORT EQUIPMENT
    LAUNCH PAD FACILITIES
  - / CLOSED CIRCUIT TV MONITORING
  - / COMMUNICATIONS
  - / RECORD AND PLAYBACK
  - / TIMING AND COUNTDOWN SYSTEM
  - / ORBITER COMMUNICATIONS AND TRACKING SYSTEMS

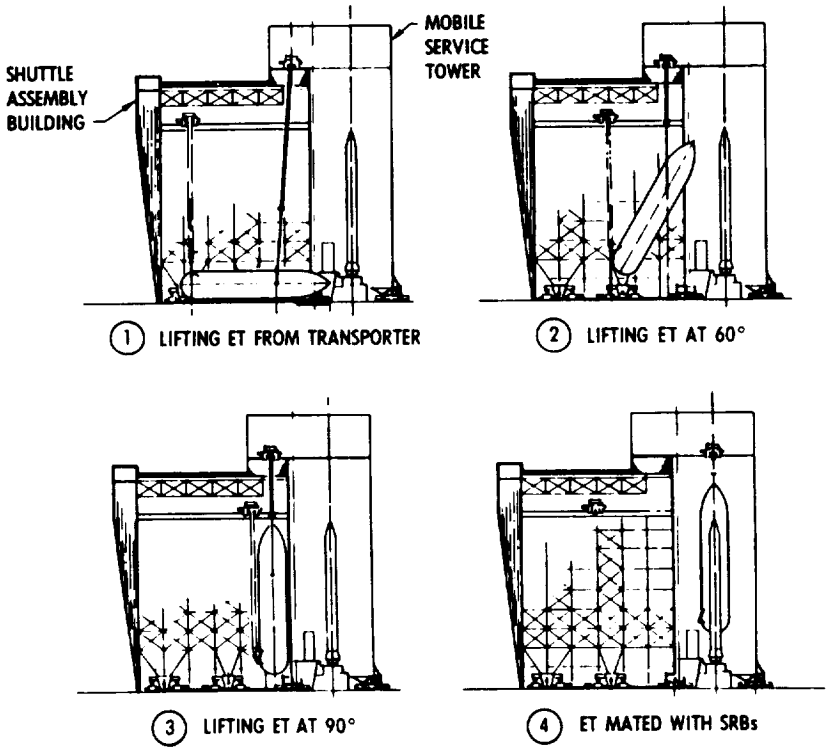


Figure 15. ET Erection and Mating

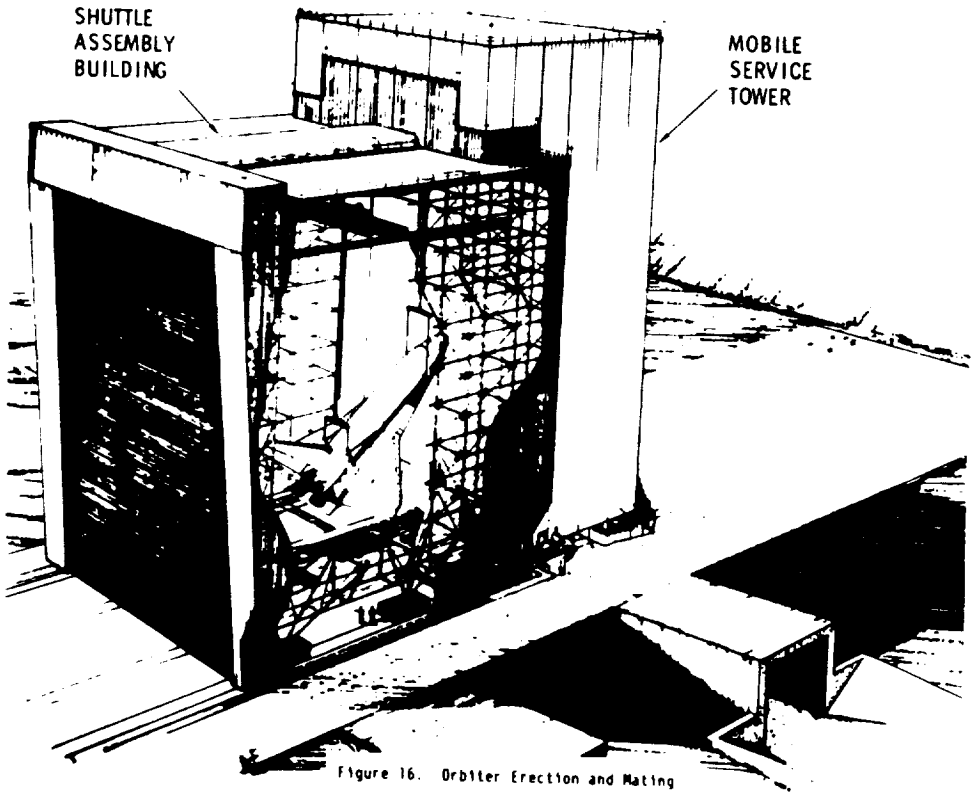


Figure 16. Orbiter Erection and Mating

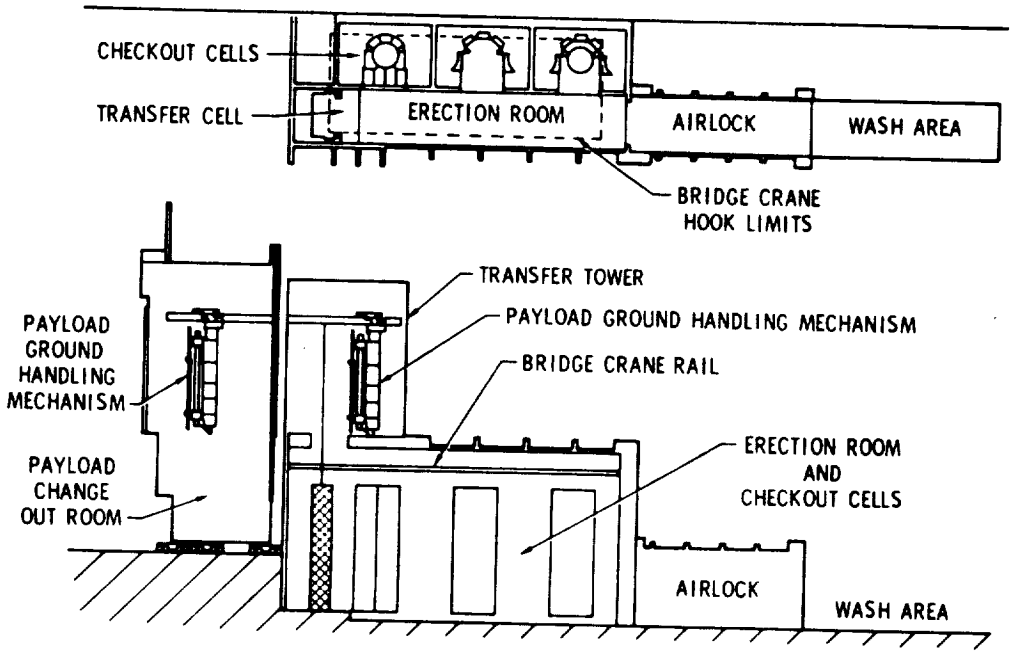


Figure 17. Payload Preparation Room

SOUTH VANDENBERG AIR FORCE BASE

- EXTERNAL TANK PROCESSING AND STORAGE FACILITY
  - HIGH BAY CHECKOUT CELL
  - FOUR STORAGE CELLS
  - LOW BAY ADMINISTRATIVE, STORAGE AND EQUIPMENT AREA
  - RECEIVE ET BARGES FROM MICHOUD AT SVAFB DOCKING FACILITY
  - RANGE SAFETY SYSTEM COMPONENTS INSTALLATION AND SYSTEM CHECKOUT
  - ET FALLS BALLISTICALLY INTO THE SOUTH PACIFIC OR INDIAN OCEAN, DEPENDING ON MISSION
  - NO ATTEMPT IS MADE TO RECOVER ET

Facility Verification Vehicle/Systems Test Vehicle (FVV/STV)

- THE FVV / STV WILL BE USED DURING ACTIVATION TESTING AND PRIOR TO RECEIPT OF THE FLIGHT ELEMENTS
- FVV CONSIST OF:
  - ENTERPRISE (Orbiter 101)
  - LIGHTWEIGHT EXTERNAL TANK
  - TWO INERT SOLID ROCKET BOOSTERS
- STV IS ESSENTIALLY A MECHANICAL PAYLOAD
- TEST OBJECTIVES
  - DEMONSTRATE ERECTION AND MATING OF THE ORBITER, EXTERNAL TANK AND SOLID ROCKET BOOSTERS AND PAYLOAD HANDLING
  - VERIFY OPERATIONAL AND HANDLING INTERFACES BETWEEN THE VARIOUS FACILITIES AND FVV/STV
  - VALIDATE OPERATIONAL PROCEDURES
  - GROUND CREW TRAINING

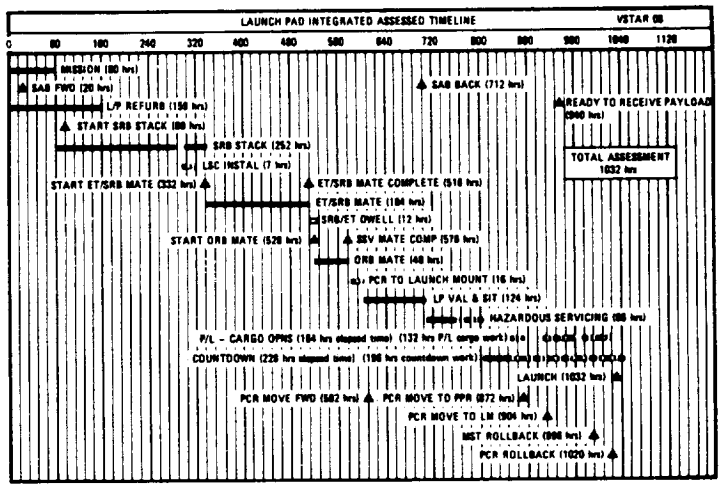


Figure 20. Launch Pad Integrated Assessed Timeline

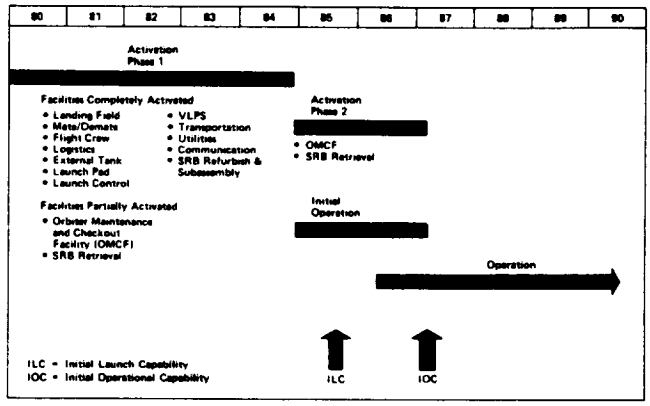


Figure 21. VLS Program Plan

INTEGRATION AND TEST OF THE DEFENSE METEOROLOGICAL SATELLITE PROGRAM GROUND SYSTEM

Vernon E. Olson
Director for Mission Operations
Defense Meteorological Satellite Program

The Aerospace Corporation
El Segundo, California

Launch and successful on-orbit operation of a satellite requires successful operation of each element in a complex system: the factory test equipment, launch site ground support equipment, the booster, the satellite itself, and the satellite ground command and control system. The Defense Meteorological Satellite Program recently completed a successful launch of the first of a new series of satellites supported by a new ground command and control system. The acquisition of this new ground system and the methods employed to assure mission success will be discussed. The approach used in this acquisition was to initiate several parallel hardware and software developments. The government assumed responsibility for monitoring these developments and for the system level integration and test. Final system integration was accomplished on site by a team of government and contractor engineers under the direction of the government.

1. Program Description.

-The Defense Meteorological Satellite Program (DMSP) provides daily worldwide meteorological data to military users. Placed in circular near-polar orbits, the satellites carry a payload of various types of meteorological sensors. The primary sensor, the operational linescan system, provides global cloud imagery in both the visible and infrared spectra. A wide variety of other mission sensors can be deployed to satisfy selected user requirements for specific meteorolgical data. Data from selected sensors are transmitted directly to tactical users while data from all sensors are stored in onboard tape recorders for later transmission to strategic weather data centers.

2. Satellite Command & Control

-The functions of satellite command and control are tracking, orbit determination, acquisition and processing of status telemetry, telemetry analysis, mission planning, command determination, and the transmission and verification of real time commands and computer loads. Each of these functions must be accomplished in a timely and reliable manner to assure mission success. Success is measured by the ability of the system to deliver payload data to the users and by the ability of the system to respond to satellite anomalies. Tracking and orbit determination are of time critical importance immediately after launch to establish satisfactory achievement of a stable orbit as well as following orbital manuevers. However, in general, the tracking and orbit determination functions are not time critical in responding to satellite anomalies. Proper response to satellite anomalies demands first of all that the data presented to the analyst be of the highest integrity under all conditions such as noisy lines and even during unexpected system conditions such as a tumbling satellite. Reliability in telemetry processing is of almost

equal importance in that contact times with ground stations may be limited and the ability to recover telemetry at a particular station contact may be vital to a command and control decision. After the analyst has decided on a course of corrective action the ground command and control system must generate the required command sequences in a timely manner and reliably uplink them to the satellite when it is in contact with a ground station. A "low orbiter" such as a DMSP satellite compounds the command and control problem since the satellite is visible from the ground network only about 15 minutes of each 100 minute orbit. For such satellites a reliable real-time telemetry and command system is essential.

3. Ground System Description.
-The DMSP ground system consists of eight interconnected minicomputers and 16 microprocessor based special purpose processors in a central facility of 6000 square feet. Each of four minicomputers (each with four special purpose processors) is configured to service the telemetry and command requirements of a single satellite while it is in contact with a ground station. Commanding is accomplished through either of two program dedicated ground stations by transmitting the command stream to the remote sites via a communications satellite. There is no storage of the commands at the remote site and the command stream is uplinked to the satellite in a "bent-pipe" mode. Telemetry streams are returned to the central site without processing at the remote site in a similar "bent-pipe" mode. Two minicomputers are configured to accomplish the mission planning and command generation functions including the preparation of routine and special purpose computer loads for each of the satellite's six onboard processors as well as maintaining memory images for each of the onboard processors. A single minicomputer is allocated for the post real-time telemtry analysis function and a single minicomputer with ground versions of the onboard processors serves as a flight vehicle simulator. Local interaction with the computers is by 20 color-graphics terminals and 30 monochrome terminals. Remote users at the satellite manufacturer's factory and the launch site also have access to any of the computers at the central facility. Software is allocated to six major subsystems to accomplish the above mentioned functions. Development required about 250,000 lines of high order language and 25,000 lines of assembly language. Total cost of ground system including the new facility was approximately 30 million dollars.

4. Integration Approach.
-Since parallel developments were undertaken by five contractors and one government agency, the challenge during the early development phase was to establish and control the interfaces. The satellite was in assembly and test during this time period and the interface definition was found to be somewhat fluid and incomplete thereby causing some late changes in the ground system. In the interest of economy, no single contractor was given the overall responsibility to integrate the ground system nor to integrate the ground system with the satellite system. The government, namely the DMSP program office, retained the integration responsibility. The government formed a team within the program office to monitor subsystem tests at the contractors' development facilities and insisted that discrepancies and deficiencies be documented at this early stage so that those affecting integration

at higher levels could be corrected. Occasionally contracts were adjusted to allow the interface specification to change, the deficiency to be corrected later, or even corrected on the other side of the interface by another contractor if it appeared to be more expedient.

Early on it was decided that a high fidelity satellite simulator was required to test the system and train the operators prior to attempting integration with the satellite at the factory or launch site. The contract for the simulator was let early and it was the first major subsystem to be delivered. As the software elements were completed at the contractors' facilities the central site was prepared to accomplish multi-element integration with the installation of the operational hardware and the flight vehicle simulator.

At this point a government-contractor task force was formed on site to direct and expedite the multi-element integration. System level tests were conducted at the earliest possible time with as near to expected operational conditions as allowed by the availability of operational databases. System level testing in this context meant tests with the simulator at the central facility and finally, through communications links implemented for that purpose, with the satellite at the launch facility. The test plans and procedures were developed by starting with the system requirements as explicitly specified and then enhancing them based on the extensive operational experience of the task force members as well as knowledge gained during the integration and test of the satellite. This test, known as the Command System Test, was first conducted at launch minus six months and involved sending commands and computer loads for all major operational scenarios as well as processing all telemetry modes. The Command System Test was run even though there were many known deficiencies with the ground system in order to uncover any incompatibilities at the earliest possible time. The test required about one week to complete and was run three times as changes were incorporated in the ground system. Discrepancies noted at all levels of testing for all subsystems were tracked by the task force management and decisions were made on a daily basis regarding the priority and impact of implementing the fixes. Fixes were grouped into "builds", tested, and where appropriate regression tests were conducted.

Data integrity was established by conducting a test to verify correct processing at each of several intermediate points for all data types processed by the system. Even with some automation of the test data analysis, the data integrity test required at least five days to complete. The criticality of data integrity caused the test to be repeated four times during the iterative process of installing fixes and retesting.

5. Lessons Learned.

-In developing large systems it is vital to establish and control interfaces as early as possible and at the lowest level possible. Extra effort expended in this aspect of program development is repaid many times over in time and effort saved during the test and integration process.

-The satellite simulator, developed at less than 10% of the ground system cost, proved invaluable as a debugging aid, verifying the critical satellite interface before commanding "live", as well as allowing crew

training and operational procedures refinement.

-Special communications interfaces developed to allow commanding the satellite at the launch site proved to be very worthwhile. Certain command sequences which could not be verified against the simulator because of incomplete modelling did not work when attempted with the satellite. Discovery prior to launch allowed an orderly restructure of the command sequences and allowed the early orbit testing to proceed on schedule.

-A government team can direct the integration and test of a large computer based satellite command and control system in a time critical environment if adequate numbers of personnel are available who have detailed knowledge of the ultimate user requirements as well as technical knowledge and experience in large system integration and if the contractors are cooperative and willing to work as members of a team.

Defense Meteorological Satellite Program

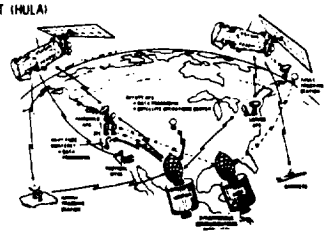
MISSION-OPERATIONAL MILITARY SATELLITE METEOROLOGY

DEPLOYED CONFIGURATION

- SATELLITES IN 450 nmi POLAR ORBIT (98.7°)
- SUN-SYNCHRONOUS, THREE-AXIS STABILIZED
- GROUND FACILITIES
  - OFFUTT AFB, MONTEREY, CA
  - LORING, FAIRCHILD, KAENA PT (HULA)
  - TACTICAL VANS, WORLDWIDE
- SHIPBOARD RECEIVING TERMINALS
  - JFK/ENTERPRISE
  - INDEPENDENCE/MIDWAY
  - KITTY HAWK/AMERICA
- LINKS
  - AMSAT RELAY

SENSORS

- VISIBLE/IR CLOUD IMAGERY
- MISSION SENSORS



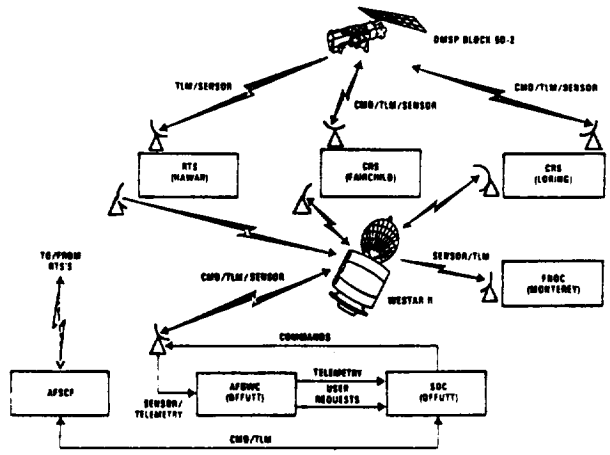
Telemetry Types Processed at Satellite Operations Center

- EQUIPMENT STATUS TELEMETRY (EST)
  - 3 DATA RATES (2, 10 and 60 kb/s)
  - 40 MAJOR FRAME FORMATS
  - 28 SUBCOMMUTATED SUBFORMATS
  - CONTAINS ANALOG, DISCRETE, AND COMPUTER DATA
- STORED MISSION TELEMETRY
  - 2 DATA RATES (1.33 and 2.66 Mb/s)
  - 3 LINE FORMATS
  - 7 FRAME FORMATS
  - CONTAINS IMAGERY, CAL, SENSOR, AND EST DATA

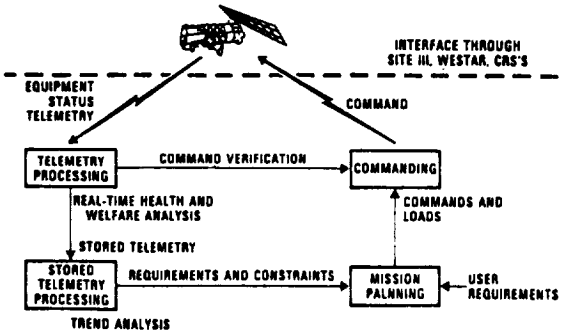
Satellite Command and Control

- TIME CRITICAL FUNCTIONS
  - TELEMETRY ANALYSIS
  - COMMANDING
- ANOMALY RESPONSE
  - DATA INTEGRITY
  - RELIABLE TELEMETRY PROCESSING
  - RELIABLE COMMANDING
  - RAPID COMMAND GENERATION
- LOW ORBITER
  - LIMITED CONTACT TIME (5-15 min)
  - ANOMALY RESPONSE CRITICAL

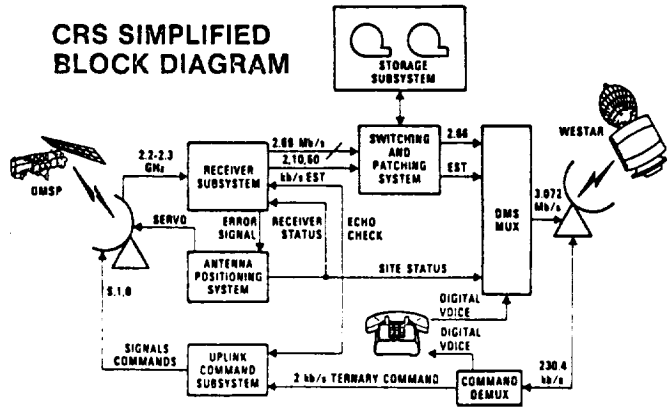
DMSP GROUND SYSTEM CONFIGURATION



SATELLITE OPERATIONS GROUP OVERVIEW



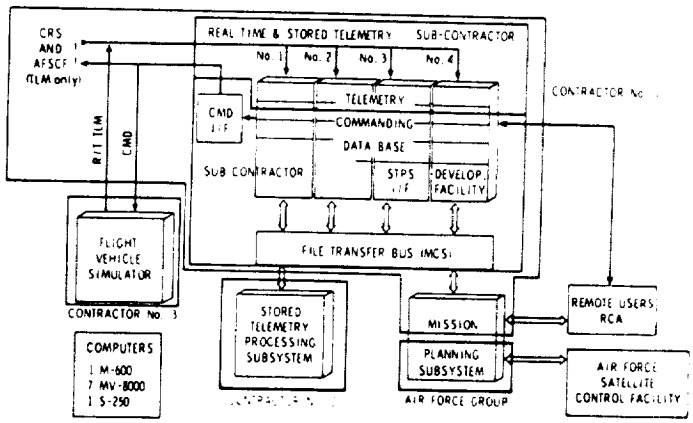
CRS SIMPLIFIED BLOCK DIAGRAM



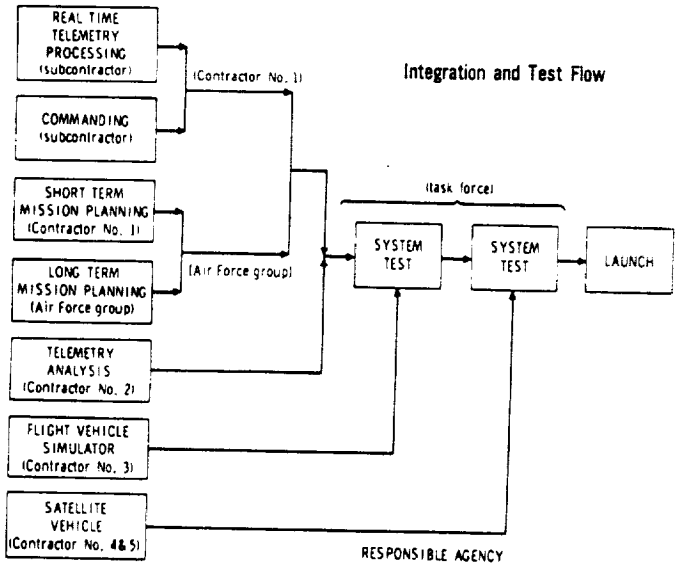
DMSP Ground System Requirements

- PROVIDE TIMELY METEOROLOGICAL DATA TO USERS
- SUPPORT MAINTENANCE OF SATELLITE HEALTH
- PROVIDE RAPID COMMAND AND CONTROL RESPONSE

Ground Segment Integration DMSP



Integration and Test Flow



Lessons Learned

- INTERFACE DEFINITION AND CONTROL IS VITAL
- SATELLITE SIMULATOR WORTH THE INVESTMENT
  - VERIFY CRITICAL COMMANDING INTERFACE
  - EXCELLENT GROUND SYSTEM DEBUGGING AID
  - CREW TRAINING
  - OPERATIONS PROCEDURES REFINEMENT
- PRE-LAUNCH COMMANDING OF SATELLITE FROM COMMAND AND CONTROL CENTER BENEFICIAL
  - INTERFACE VERIFICATION
  - PROBLEMS UNCOVERED INVOLVING TIMING OF CERTAIN COMMAND SEQUENCES

TPS'S AS A DESIGN TECHNIQUE

Sharon A. Carpenter, Mgr.
Automatic Subassembly
Test Development Engrg.
Collins Government Avionics Division
Rockwell International

The Navstar Global Positioning System (GPS) is a space-based system that will revolutionalize navigation.

When the GPS system is operational in 1987, it will consist of a network of Navstar satellites circling the earth, and thousands of individual user equipment receivers. These receivers will translate satellite signals into navigation information so that the user will know where he is within 10-20 meters anywhere in the world, any time of the day, in any kind of weather. Information will include three dimensional location, velocity to 0.1m/sec. and precise time to nanoseconds. The GPS information will be displayed in a common grid or, other system that the customer may desire.

Rockwell International's Collins Government Avionics Division helped pioneer the effort to make GPS fully operational. Participating in the Phase I concept validation program, our Generalized Development Model (GDM) proved the feasibility of high dynamic, aided anti-jam operation. Now the Division is in Full-Scale Development (FSD) of user equipment for the GPS Joint Program Office.

The Full Scale Development Program calls for design of the user equipment hardware and software, production prototype build, extensive in-plant formal testing, Development Test and Evaluation, Initial Operational Test and Evaluation on each of the host vehicles, and a production proposal.

Multi-level commonality has been designed into the Collins family of joint service GPS user equipment to insure that the system will be easy to produce and maintain in a wide variety of applications. For the host vehicle applications shown, Collins user equipment sets will have 82% of all system software, 75% of all Line Replaceable Units (LRU's) and 94% of all Shop Replaceable Units (SRU's) in common. The result of this unique Collins design will be low life-cycle costs.

In order to meet this objective of commonality and yet maintain low life-cycle costs, much emphasis was given to minimize problems in hardware and software development. There were many tools and techniques that could be applied, but each of them had to be analyzed to measure their worthiness to the project at hand. At all times of the hardware and software development, it became necessary for a person to wear many hats in order to meet not only the performance specifications of the end product, but the needs and requirements of the manufacturing, test, and logistic groups.

Much interfacing between these groups became necessary. This can be very labor intensive and frustrating to the design engineer attempting to meet his own goals and schedules.

This paper will present some of the in-plant tests that GPS designers have used and how they minimized risk in meeting the goals of commonality and low life-cycle costs. It will then focus on the Test Program Set development which was accomplished in parallel with the FSD design activity. By presenting both design engineering in-plant test and Test Program Set development for the SRU's, a better understanding can be gained as to the simultaneous techniques/methods that must exist to assure completion of the mission assigned.

In-Plant Tests:

The in-plant test objectives for hardware/software design were identified as:

- 0 Early identification/resolution of problems.
- 0 Formal test would be a verification of known performance.
- 0 Major field test problems would be eliminated.
- 0 All equipment types fully qualified for intended host vehicle application.

Steps to be taken by Rockwell-Collins to reduce risk were identified:

Software:

- 0 Simulation Capability Developed
- 0 Simulated Real-Time Test System
- 0 Early Checkout on Target Machine

Hardware:

- 0 Testability Analysis
- 0 Component/Breadboard Testing

Component Level:

- 0 Feasibility and Characterization testing at the Component Level to reduce overall risk
- 0 Examples:
  - Custom RF Signal Attenuator for Satellite and Jammer Power Level Control
  - Crystal Filter Testing
  - RF Mixer Testing for Phase Delay

Breadboard/Brassboard Testing:

- 0 All critical RF design areas breadboarded and tested.
- 0 Performance Data Testing accomplished prior to Planar Board layout stage

---

- 0 Complete Engineering Models, For Form, Fit, and Function
- 0 Preliminary Environmental/EMI
  - Included shock, vibration, cargo bounce, hammerblow and thermal testing
- 0 Reliability Growth (5000 Hours/Equipment Type)
  - Burn-In exercises LRU over environmental stress conditions to eliminate infant mortality in the hardware

---

Software/Hardware Integration:

- 0 Extensive Simulation Capability
- 0 Dry-Run Prior to Formal Test

Software simulation capability was developed using the Univac Computer. The performance results were determined, presented to JPO, and algorithm testing completed. Here we were able to identify problems before the user equipment software was developed. The key point here was to consider special cases as well as normal conditions. This analysis gave validity to the results we had earlier determined.

As the heart of the receiver, the Code Generator is a good example of how Rockwell-Collins implemented the risk reduction steps identified. Approximately one year prior to the contract

award, the new technology and complexity of this circuit were identified as potentially high risk areas. Plans were undertaken to break the circuit down into five main chips for easier test capability. All chips were tested at the component level using the Sentry 7 tester, then tested as a unit for temperature, voltage and clock frequency. A simulator was also used to compare expected results in the memory with what the Code Generator memory contained. All of these tests plus more provided a high degree of confidence that the Code Generator performed as prescribed.

In all of these tests, test plans, design and coding walk-throughs were accomplished to guide the direction of the design. Throughout this full-scale development phase, it must also be remembered that the needs of other groups had to be interlaced with the design in order to meet the final objective.

Test Program Set Development

One of these areas was the Test Program Set development of the SRU's for depot testing. The objective of the test program was to ascertain the operational readiness of the unit under test. The prime contract had specific maintainability goals which related to the ability to diagnose faults in the hardware when it is deployed. The TPS development, concurrent with design development is meant to assure the Government of that accomplishment. This ties the TPS development into a design verification technique. In addition, software and hardware is being designed in the TPS development. This parallel development involves not only minimizing risk to the TPS product but identifying how it may impact the prime equipment design.

The TPS's not only became a tool to prove test capability in the field but a tool to enhance the design development phase. This was achieved by the design engineer's recognition of the benefits these tests gave to the verification of the hardware. Initial awareness of these requirements and incorporation into the front-end of the design minimized down the road work-arounds and redesign.

Many of the methods/processes used in the design of Test Program Sets were implemented in the user equipment design areas, eg. test plans, simulations, walk-throughs, and validation tests.

Whether Test Program Sets are done in parallel with design development or upon completion of prime product design, testability for this assignment needs to be performed early. Because the TPS effort was in parallel, there are specific verification means of testability. This is not to say the product did not have testability; but testability needs are dependent upon the capability of the test system utilized. This parallel TPS effort has made it easier for design engineering to become acquainted with what unique testability guidelines are required for card level test before any redesign or relayout in the next phase of the program occurs.

As stated earlier, the objective of doing Test Program Sets was to ascertain the operational readiness of the unit under test. This in itself is a design assurance technique because so often in card level testing, the design itself is challenged as to its functionality. On the GPS program, ten SRU's were identified for this task. The ten were chosen to be good examples of the complexity of the design, but were also cards which would have the least likelihood for mass changes. This latter objective was to minimize a large amount of changes for later TPS effort. Half of the cards

were mainly digital in nature and the others analog. The targeted systems for implementation were the AN/USM-410 EQUATE system for the analog cards and the AN/USM-465A for the digital cards.

The philosophy for fault validation testing was defined per MIL-STD-2077. Briefly this states "When the SRU contains more than ten nonrepairables, isolation to groups of four or less shall be pos- sible for 80% of the possible faults. Isolation to groups of eight or less must be possible for 95% of the possible faults. Isolation to groups of ten or less must be possible for 100% of the possible faults".

In order to maximize the utility of the Test Program Set development, design engineering was provided the capability to test their digital cards in-house using our test programs rather than generating another set of test equipment. Because the group performing the TPS development was versed in automatic card-level testing in the factory environment, they were able to utilize in-house equipment to measure the effectiveness of the tests for the digital cards. This simulation provided engineering test capability which aided their design troubleshooting and gave us an opportunity to see more cards, enhancing the validity of the test program design. This effort required a close interaction between design engineer, TPS engineer, and lab technician. It yielded commonality of test and a good identification of problem areas early-on where changes could still be made with minimum risk.

The simulator employed for this activity helped identify race conditions, and design faults, and provided minimal breadboarding.

The analog cards were simulated in a lab environment to match their targeted system. This was to minimize the amount of time necessary to integrate hardware and software on an off-site system. Again, as in the design of the prime product, special situations were observed along with normal ones.

Utilization of the targeted test systems for depot testing had not been accomplished in-house prior to this activity. To assure a good response to the Test Program Set development, one digital and one analog card were chosen to have accelerated schedules. This approach was decided upon to minimize problems that could be encountered in all ten Test Program Sets.

In addition, throughout the process, a Design Policy Manual was developed to ensure the procedures that were successful could be followed for future TPS development.

This Design Policy Manual was used to standardize a set of instructions, along with design aids to ensure a common starting point for all TPS's. The goals for the instructions and guidelines contained in the manual were:

- Standardize the TPS Design Approach
- Minimize the Developer's Task Planning Time
- Present All Pertinent References
- Define Terms and Tasks
- Provide Task Descriptions for External Communications
- Provide a "Look Back" Reference as the Project Progresses
- Reduce Program Costs Through Balanced Implementation and Design Discipline

Upon being assigned the development task of a Test Program Set (TPS) for one of the specific SRU's (Shop Replaceable Units), the engineer collected all pertinent information--schematic, parts list, layout, production test requirements and specifications. It became necessary to maintain good communications with the designer so that additional information on changes were obtained in a timely manner. Updated drawings, scheduled releases, circuit analysis assistance, and more, were needed throughout the TPS effort.

When the data had been collected, the preliminary design effort took place. The engineer analyzed the data and determined the best approach for testing the board. As a result of the circuit analysis, a Test Flow Diagram and Test Plan were developed. These preliminary documents defined the test program and later became part of the English Language Test Document.

Two separate test plan outlines had been developed for the GPS TPS effort, one for the AN/USM-410 and one for the AN/USM-465A. Questions and concerns were covered on these outlines and required answers before proceeding.

The most important part in the design of a Test Program Set is the analysis of the circuit. The more effort spent on the front-end of a project establishing direction, the better able the engineer was able to meet the end objective. Definitions were made as to how the circuit would be broken out and the desired results determined. If an engineer does not know what he expects, he cannot determine the path to get there, nor the inputs required by others.

In the case of Test Program Set development, the Test Strategy Report was created to outline the test requirements, delineate problems or constraints anticipated in implementing a test design, and to provide a general overview of the test approach. This plan was reviewed in a walk-through with peers, similar to the concept review which was held with the customer.

The Concept Review was held soon after the design effort began and upon completion of the overall test approach, which was reflected in the Test Strategy Report. This review was held to ensure that the test design approach conformed to the requirements and to ensure that the requirements themselves were consistent with user maintenance objectives. The concept review emphasized completeness and soundness of the test design approach with respect to system and functional requirements, ATE capability, design specifications and schedules. It presented an opportunity to review the basic test requirements and to request specification changes and deviations for unrealizable requirements.

Execution time, probe time and operation intervention concerns were addressed. Many interpretation differences were resolved, especially in the areas of format of deliverable documentation and tests that could damage the unit under test.

The customer identified potential problems that might occur with the targeted equipment. This helped us address alternate methods of test rather than waste substantial effort writing code to force the system to do something it could not do.

Hardware necessary to interface the UUT (Unit Under Test) electrically and mechanically to the ATE was also identified at the Concept Review. The interface devices had to be optimized so that as many UUT's as practical could be tested on the same basic assembly. This reduced the number of ID's required, thereby reducing shop storage requirements.

Previous to the concept reviews, our plan had called for separate interface adaptors for each TPS. The concept review eliminated unnecessary design and build effort which had been anticipated and presented guidelines for a common adapter.

Coding commenced after the Concept Review cycle with recommendations/ suggestions in hand. A pseudo code was used in designing the test program to better document the program. Since the pseudo code is used to communicate the design during walk-throughs, as an aid in integration, and finally to maintain the Test Program Set after delivery, it had to be "religiously" updated throughout the TPS development. "Pseudo code" can be defined as "narrative documentation" constructed from simple imperative English sentences containing a single transition verb and a single, non-plural object. Simulations for software were then performed using a logic simulator and lab setups, thereby minimizing time and identifying potential problems in actual integration with the hardware. Upon completion of this phase of the test program development, another peer walk-through was conducted. The frequency of these walk-throughs assured compliance to the design criteria/objective established earlier.

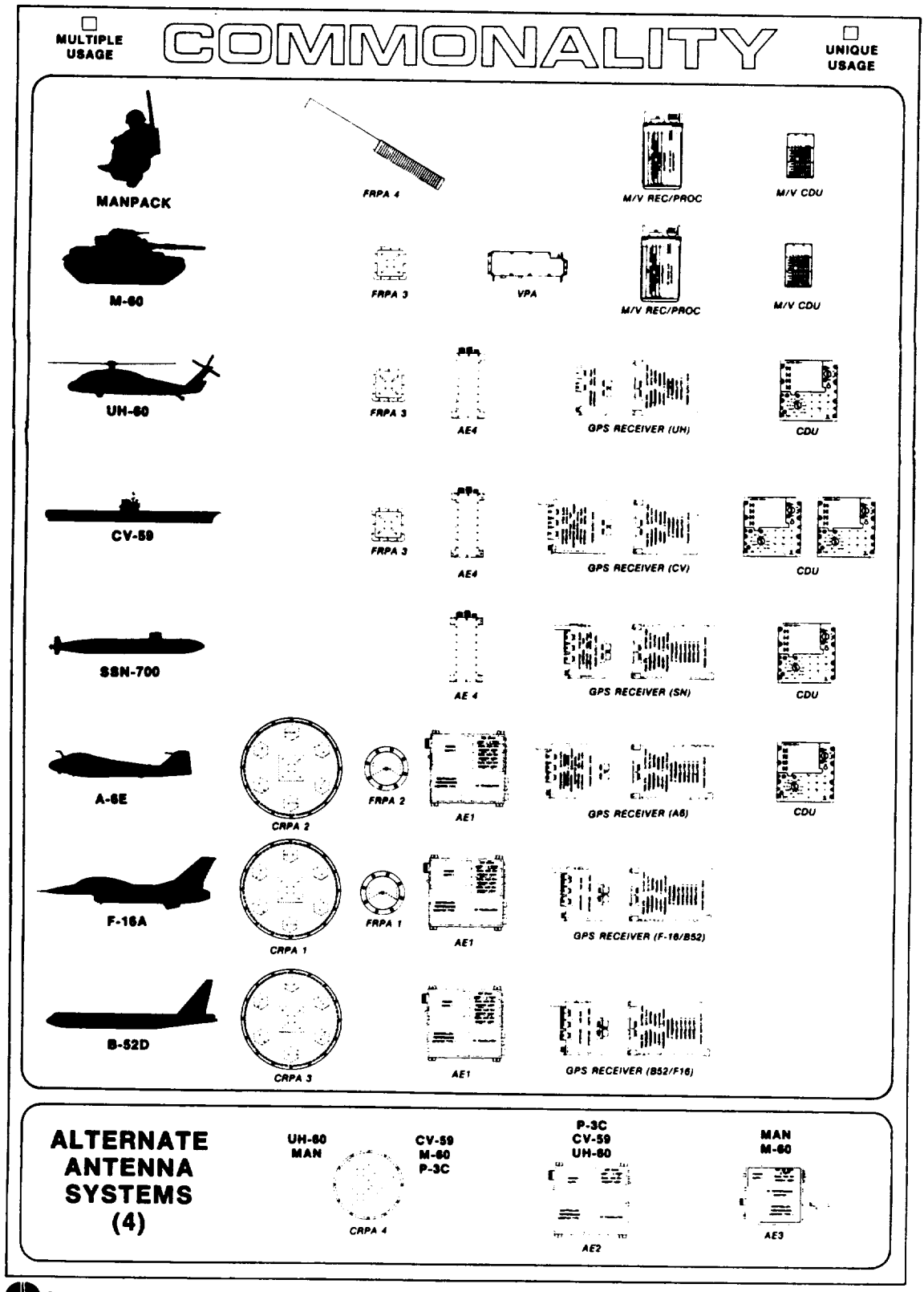
Integration of the software and hardware on the targeted systems was performed to validate those results seen on the simulator and in the lab environment. Problems were minimized by this prior action and became mostly differences associated with what the system expected to see or could perform.

Validation of the Test Program Set was accomplished by randomly selecting faults and inserting them into the UUT. The test program was then measured by its ability to fault-isolate to that fault. This effort was conducted previous to Final Acceptance Test for confirmation of any areas that may still require work. Final Acceptance can then be a formality as the confidence level of good Test Program Sets has been proven internally.

This Test Program Set Development demonstrates the interaction of the design engineer with other groups to verify the design objective of maintainable GPS hardware. In summary, key points to be remembered are:

- Maintain a close interaction between all working groups to identify all requirements;
- Establish a disciplined approach early on which addresses requirements, documentation, process used, and how to handle unknowns;
- Use simulation and test capability as much as possible.

This example is but a small part of the overall program but exemplifies one of the many areas that must be managed and implemented correctly in order to achieve the overall objective of providing reliable, flexible and affordable user equipment for the Navstar Global Positioning System.



Rockwell International

NAVSTAR GPS

LAUNCH OPERATIONS DESIGN ASSURANCE CONSIDERATIONS

Otto H. Fedor
Principal Engineer
NASA-Kennedy Space Center

Design Assurance for the ground segments of Space systems requires incorporation of reliability characteristics into the design of ground support equipment and facilities. Further, a credible and meaningful design validation technique must be employed. This paper will discuss some of the more successful methods used at KSC and will urge that additional efforts be made to improve the design certification and launch readiness review methodology for the operational era.

1. KSC Launch Operations. - If the objective for future launch operations is to reduce size, complexity, and cost of launch operations, it is necessary to understand today's practice and related problems. An examination of the activities at a launch operation center (figures 1 through 5) reveals that there are many heavily man-loaded activities with attendant high costs. The ground processing schedule is intensive and involves high risks and hazards wherein human judgment errors are catastrophic. The checkout, servicing, and launch tasks include payload processing, Orbiter processing, element mating, servicing, countdown, and a postlanding function.

2. Planning Goals to Improve Mission Assurance. - To reduce high operational costs, it is necessary to minimize real-time human involvement and to improve the design of human tasks to gain higher efficiency. This may involve design of launch site equipment for greater capacity of automation and reduce the man-machine operations. A major goal is to "streamline" the operating schedule to a more operational mode and reduce "idle time" (figures 6 and 7). Improvement of mission assurance for future operations will also require improvement in system integrity, maintenance, and logistics capabilities.

3. Human Factors. - We are concerned with human inactivity due to serial activities. We observe human factors delays due to extensive human activity and/or interaction with computers and other equipment. It is imperative to minimize human weariness in critical activities due to poor human factors planning.

4. An Approach to Aerospace Design. - To understand the problem of design assurance, it is necessary to realize that aerospace design differs from industrial design in many ways. There are ambitious space program goals that push the state-of-the-art, causing high technical risks. Space and launch hardware is exposed to severe environments with serious consequences in the event of failure. Then consider that hardware is usually made in limited production runs with severe schedule restraints. Another consideration is the need for major resources. Aerospace system assurance is characterized by the emphasis on providing visibility to higher management.

5. System Integrity. - An understanding of what it takes to obtain system integrity must be gained before reliable systems can be designed. A system assurance commitment must be made "up front." Tasks should be prioritized with establishment of hardware and functions criticalities. The reliability drivers of performance, cost, safety, and time should be traded off. This planning phase is followed by the establishment of a system to identify risks, unreliable elements, and hazards. We must ensure that hardware and software meet the design and system requirements. In fact, the major "ilities" role is to maintain this assurance rigor during the development of deployment cycles to preserve this design integrity. (Figure 8 shows some of the causes of unreliability; figure 9 illustrates some of the factors that lead to good reliability.) There is a sustaining reliability engineering effort that should be considered. A Problem Reporting and Corrective Action system is valuable. A management information system is required to involve higher management. Change

control is important to maintain configuration integrity. Incentive contracts should be provided for improvement in product quality. Figure 10 illustrates a technique to measure Reliability and Quality Assurance (R&QA) effectiveness.

6. KSC Design Philosophy. - Figure 11 shows the difference in development of flight hardware and support equipment. KSC makes a major effort to use only those R&QA requirements suited to operational needs and not to pay for more assurance than is needed. Proven hardware and design concepts are used wherever possible. Two documents have been particularly useful: The System Assurance Analysis Technique (figure 12) and KSC-DE-512-SM, "Guide For Design Engineering of Ground Support Equipment and Facilities For Use at Kennedy Space Center."

7. Safety as a Design Assurance Function. - While the goal for engineering safety is to have all the safety one can afford, a better approach is to insist on a balanced program with equal risk-taking for all elements of the design. This can be done by conducting a systems hazard analysis during the design phase to identify those conditions that result in damage to hardware or injury to personnel. Combining these results with an operational hazard analysis provides an aggregate assessment of all safety analytical efforts performed during the design and operational phases as well as identifying their physical and functional relationships. Figures 13, 14, and 15 suggest actions and techniques that improve safety during hazardous operations.

8. Limitations of Product Assurance. - The product assurance effort is usually limited by insufficient resources and time to perform both mandatory and highly desirable tasks. The control of subcontractor change control and process quality is limited and has presented problems in the past. Demonstration of component and parts reliability is difficult even with a large data base. Another problem is that support by top management and its involvement in the product assurance cycle varies during the various program phases.

9. Risk Acceptance. - The designers, testing agencies, and operators have tasks to support management in the readiness review cycle. Each should measure his product and report to management. This allows formation of an assessment and subjective analysis to permit the decision maker to be comfortable in his judgment. Figure 16 shows why the residual engineering and program risks must be presented in a concise and formal manner to the designated approval authority. They need a perspective to determine if they are dealing with a firecracker or a stick of dynamite. It is important that, having reached a decision, R&QA ensure that these risks have in fact been eliminated by design or procedural changes.

10. Lessons Learned. - Both civil servants and contractor personnel have learned that rules, regulations, and reports are not a substitute for sound engineering judgment. Twenty years of experience have produced these other observations:

a. Reliance on complex paper systems documenting the checks and balances functions should not obscure the need for reliability specialists' identifying the factors that truly need management attention.

b. Last minute improvisions can introduce hidden hazards. Beware of cost reduction and product improvements.

c. Fight for a consistent approach to redundancy design concepts throughout all systems to simplify the elements.

d. The "ilities" should be in the decision loop from the beginning.

e. Burn the Failure Modes and Effects Analyses (FMEA's), Hazard Analyses (HA's), and Single Failure Point Analyses (SFPA's) once the designer and managers understand their message unless you intend to keep the analyses current.

f. Never underestimate the power of any formal system that makes critical problems visible and understood by management. It will assure that action is taken to solve these problems.

g. The hallmark of a successful design assurance program is the participation of highly competent and motivated people. Carefully selected but small groups of individuals can contribute far beyond their numbers.

FIGURE 1. PAYLOAD PROCESSING

- WORK ACTIVITY
  - ANALYTICAL INTEGRATION
  - OPERATIONAL PLANNING
  - COMPLETION OF PAYLOAD ASSEMBLY
  - SUBSYSTEM CHECKOUT
  - INTEGRATED/MISSION TEST
  - UPPER STAGE/PAYLOAD MATING
  - SERVICING
  - VERIFICATION OF SHUTTLE INTERFACES
- AREAS AFFECTING MISSION READINESS
  - SOFTWARE GENERATION AND VERIFICATION
  - FAULT DETECTION AND ISOLATION

FIGURE 2. ORBITER PROCESSING

- WORK ACTIVITY
  - FLUIDS SYSTEMS DESERVICING
  - SUBSYSTEM CHECKOUT
  - THERMAL PROTECTION SYSTEM REFURBISHMENT
  - PAYLOAD INSTALLATION AND INTERFACE VERIFICATION
  - INTEGRATED MISSION TEST
- AREAS AFFECTING MISSION READINESS
  - FAULT DETECTION AND ISOLATION TECHNIQUES
  - LEAK DETECTION AND HAZARD MONITORING

FIGURE 3. SHUTTLE ELEMENT MATING

- WORK ACTIVITY
  - SOLID ROCKET BOOSTER (SRB) ASSEMBLY
  - SRB/EXTERNAL TANK (ET) ASSEMBLY
  - ET/ORBITER ASSEMBLY
  - FLUID AND ELECTRICAL UMBILICAL CONNECTIONS
  - INTERFACE VERIFICATION
- AREAS AFFECTING MISSION READINESS
  - ASSEMBLY TECHNIQUES
  - LEAK CHECKING
  - FAULT DETECTION AND ISOLATION
  - INTERFACE VERIFICATION

FIGURE 4. COUNTDOWN

- WORK ACTIVITY
  - CRYOGENIC PROPELLANT LOADING
  - FLIGHT CREW INGRESS
  - FINAL CHECKOUT OF SYSTEMS
  - "REDLINE" MONITORING
- AREAS AFFECTING MISSION READINESS
  - HAZARDOUS WARNING
  - DATA INTERPRETATION
  - FAULT DETECTION AND ISOLATION

FIGURE 5. ORBITER POSTLANDING

- WORK ACTIVITY
  - DETERMINATION THAT ORBITER IS SAFE
  - CONNECTION OF UMBILICALS - COOLING AND PURGES
  - FLIGHT CREW EGRESS
- AREAS AFFECTING MISSION SUCCESS
  - LEAK DETECTION
  - HAZARD MONITORING

FIGURE 6. PLANNING GOALS TO IMPROVE MISSION ASSURANCE

- REDUCE HIGH COSTS
- AID "HUMAN ELEMENT" OPERATIONAL MODE
- MAINTAIN/IMPROVE RELIABILITY AND QUALITY
- ACHIEVE BEST OPERATIONAL CAPABILITY POSSIBLE
- IMPROVE KSC OPERATIONAL TURNAROUND TIME

FIGURE 7. IMPROVE KSC OPERATIONAL TURNAROUND TIME

- TIME CONSUMING FUNCTIONS
  - MECHANICAL (ASSEMBLY, POSITIONING, TRANSPORT, CONNECT/DISCONNECT)
  - TESTING
  - MODIFICATIONS
  - DATA MANAGEMENT
  - SUPPORTING ACTIVITIES
  - INTEGRATED ACTIVITIES

FIGURE 8. MAJOR CAUSES OF MECHANICAL UNRELIABILITY

- UNCONTROLLABLE UNCERTAINTIES
  - INABILITY TO PREDICT WORKING ENVIRONMENT
  - DEFICIENT CUSTOMER REQUIREMENTS
  - INCOMPATIBILITY OF REQUIREMENTS WITH STATE-OF-THE-ART DESIGN
  - UNIT-TO-UNIT VARIATION IN ABILITY TO RESIST FAILURE WHEN ALL COMPONENTS ARE WITHIN SPEC LIMITS
- CONTROLLABLE UNCERTAINTIES
  - DESIGN ERRORS
  - MANUFACTURING ERRORS
  - MAINTENANCE ERRORS
  - HUMAN ERRORS

FIGURE 9. RELIABILITY AND MAINTAINABILITY

- RELIABILITY AND MAINTAINABILITY RESULT FROM:
  - SIMPLICITY
  - EASE OF OPERATION AND MAINTENANCE
  - BENIGN ENVIRONMENTS
  - LESSER LOADS AND STRESSES
  - CONTROL OF SUBCONTRACTOR COMPONENT QUALITY
  - DESIGN FOR REDUNDANCY BUT NOT DUPLICATIONS
  - AVOIDANCE OF PROGRESSIVE FAILURE DESIGN ARRANGEMENTS

FIGURE 10. MEASURES OF R&QA

- ENGINEERING CHANGES
- DEFECT LEVELS AT ACCEPTANCE DECISION POINTS
- DEFECT RATES AT SELECTED IN-PROCESS POINTS
- COMPLIANCE WITH DELIVERY SCHEDULES
- VARIATIONS FROM LIFE CYCLE COST PROJECTIONS
- DEFECTS FOUND AT CUSTOMER DELIVERY POINTS
- CUSTOMER COMPLAINTS DURING USAGE
- PROCESS CONTROL EFFECTIVENESS
- PERFORMANCE TESTING
- RECEIVING INSPECTION RESULTS
- AUDIT FINDINGS
- SCRAP AND REWORK COSTS
- MATERIAL REVIEW BOARD ACTIONS

FIGURE 11. RELATIVE DEVELOPMENT SCHEDULES - FLIGHT HARDWARE VERSUS SUPPORT EQUIPMENT

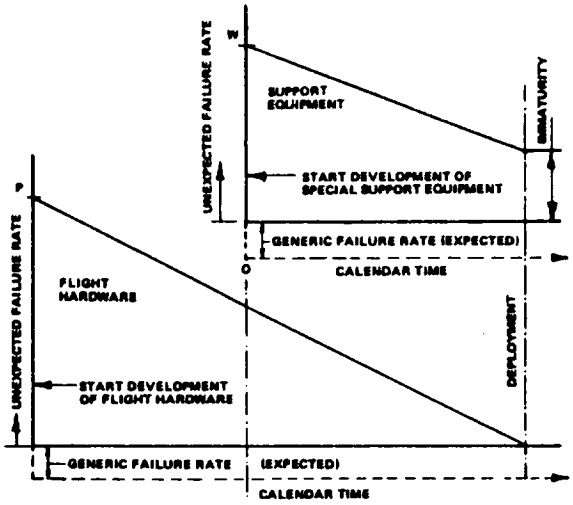


FIGURE 12. SYSTEM ASSURANCE ANALYSIS (SAA)

- <u>SAA IS A COMPREHENSIVE EVALUATION OF THE TOTAL SYSTEM, INCLUDING:</u>
  - SYSTEM IDENTIFICATION
  - SYSTEM FUNCTIONAL DESCRIPTION
  - SYSTEM CRITICALITY
  - FMEA
  - SFPA
  - CRITICAL ITEMS LIST
  - FORMAL COMPONENT QUALIFICATION STATUS
  - DESIGN HAZARD ANALYSIS
  - SNEAK CIRCUIT ANALYSIS
  - AREAS OF CONCERN
  - OPERATIONAL HAZARD ANALYSIS

FIGURE 13. MANAGEMENT SAFETY ACTIONS

- IMPROVE AREAS OF ORGANIZATION, MANAGEMENT, AND EMPLOYEE COMMUNICATIONS.
- IMPROVE REPORTING AND TRACKING OF MISHAPS, FAILURES, TRAINING, AND CERTIFICATION.
- ANALYZE DATA FOR TRENDS AND PROBLEMS SUCH AS DESIGN WEAKNESS, POOR SUPPLIERS, OPERATIONAL ERRORS, EXCESSIVE OVERRUNS, AND SCHEDULE SLIPPAGE.
- CONSOLIDATE RESPONSIBILITY TO BE CLOSE TO "HANDS-ON" OPERATIONS BY ASSIGNING EXPERIENCED ENGINEERS FOR FULL COVERAGE OF ALL SHIFT OPERATIONS

FIGURE 14. TECHNIQUES FOR REDUCING MISHAPS

- SPECIALIZED TRAINING AND CERTIFICATION PROGRAM
- COMPREHENSIVE OPERATIONAL AND MAINTENANCE PROCEDURE REVIEW
- OPERATIONAL DISCIPLINE
- CONTRACT INCENTIVES
- MOTIVATION PROGRAMS FOCUSED ON PROBLEM AREAS
- EFFECTIVE PROBLEM REPORTING AND CORRECTIVE ACTION SYSTEM
- USE OF PERSONNEL PERFORMANCE APPRAISAL SYSTEM TO ATTAIN SAFETY GOALS
- CONTROL OF DESIGN INTEGRITY

FIGURE 15. SENIOR MANAGEMENT PARTICIPATION AND CONTINUOUS SUPPORT

- MOTIVATION BY: CONTRACTS
  PERSONNEL APPRAISALS
- CONTROL BY: DESIGN INTEGRITY
  OPERATIONS & MAINTENANCE INSTRUCTION (OMI) REVIEWS
  TRAINING AND CERTIFICATION
  PROBLEM REPORTING SYSTEM

FIGURE 16. BASICS OF ENGINEERING RISK MANAGEMENT

- NASA CONCEPT

  THE BASIC APPROACH TO RISK MANAGEMENT IS:

  - KNOWLEDGE OF RISKS THROUGH ENGINEERING ANALYSIS AND TESTS
  - CONTROL OF RISKS THROUGH REDUNDANT OR FAIL-SAFE TECHNIQUES
  - DECISION BY MANAGEMENT TO ELIMINATE OR ACCEPT RISKS

- RESPONSIBILITY

  RISK MANAGEMENT IS AN ENGINEERING, MANAGEMENT, AND OPERATIONS RESPONSIBILITY

- COST EFFECTIVENESS

  BENEFITS ARE MORE THAN ENSURING SAFE AND RELIABLE FACILITY OPERATION:

  - FOCUS ATTENTION ON THE REAL RISK PROBLEMS
  - IMPROVED COMMUNICATIONS AND CREDIBILITY WITH INTERFACING REGULATORY AGENCIES OR ACCIDENT INVESTIGATION TEAMS
  - IMPROVES COMMUNITY RELATIONSHIPS
  - REDUCED INSURANCE AND OPERATING COSTS

A METHOD OF SOFTWARE SAFETY ANALYSIS

John Griggs, III
Martin Marietta Aerospace

INTRODUCTION

In 1955 only 10% of our weapons systems required computer software. Today more than 80% of the systems require computer software.

The increased use of software to control hardware functions has put both NASA and DOD in a position of controlling hazardous sequences with computers and associated software. Examples of this include the Minuteman remote command and control systems, and the Space Shuttle Ground Launch Systems as well as the Flight Control Systems. Attempts to ensure the safety of these computer-controlled systems has lagged behing in development. The Space Division of Air Force Systems Command has required, through the imposition of MIL-STD-1574A, that a Software Safety Analysis be performed on software used in spacecraft and missile systems developed for the Air Force. A similar requirement is being written into MIL-STD-882X.

Traditionally, in safety analyses, the computer subsystem has been considered a "black box" that gives a specific output based on a given input. This black box, and the software it contains, have traditionally not been considered in the safety analysis.

This paper, which considers this analysis concern, is subdivided into three sections. The management section discusses the usefulness and the limitations of the analysis technique, as well as correlating issues that must be coupled with the performance of a software safety analysis if the results are to be valid. The technical presentation of the analysis technique is given in Section 2, followed by a discussion of software checklist in Section 3.

DEFINITIONS

Firmware

A computer program (software) permanently burned into a logic circuit chip (hardware).

Software

Software, as used in this discussion, applies to a computer program, considered as it will reside in a computer memory.

The media (card deck, tape) are not at issue, and advanced forms of software usage (firmware, ROM, PROM, etc.) are not addressed because of the hardware connotation.

Software Safety Analysis

A study of a software program to ensure that the program, used in its intended operational environment, cannot cause or contribute to a category 1 (loss of life) or a category 2 (flight vehicle damage) hazard.

I. MANAGEMENT CONSIDERATIONS

The current attention given to safety and to software, because of its rapid growth, has prompted DOD to require software safety analyses on specific DOD programs. This leaves Program Management with a dilemma because the method of performing safety analyses on any software program is unclear; thus presenting the manager with both a technical performance uncertainty and a costing unknown.

Section II of this paper presents a technical approach for performing the analysis. Our studies to date indicate that only the category of software programs associated with command and control of hazardous hardware functions (test sequence software) merits this type of analysis. This is based on the fact that a physical hazard can only occur in hardware; hardware that can fail, malfunction, or cause an unwanted energy release due to improper control (procedural error, either manual or in the software).

Management should recognize that "associated with command and control" has a direct (the control program) and indirect (date base generation/manipulation programs) connotation. There is no advantage to performing this analysis on computational programs such as those used in analytical work unless the output is to be used in controlling hardware.

The analysis is structured in a manner similar to the hazard analysis technique; that is, the bulk of the effort is in the requirements phase. As a rule of thumb, spending 50% of the safety hours on requirement definition (DOD "B" Spec, and Phase I in the technical discussion); 30% on the code design and detailed flow (Phase II in the technical discussion); and the remaining 20% on the testing review and associated code change traffic. A further rule of thumb is to estimate 5% to 7% of the software development hours, excluding testing hours.

As described in the Technical Section, the Software Safety Analysis is an extension of the Hardware Preliminary Hazard Analysis. In effect, the objective is to ensure that any hazards identified against the hardware system (to be handled by procedural control) are built into the software if possible. This analysis includes an operating hazards analysis, as well as integrated analysis considerations. The overall objective then, is to integrate the hardware analysis with the command and control subsystem (including the operator interface) leading up to a complete operating hazard analysis.

I. MANAGEMENT CONSIDERATIONS (Continued)

There are a number of important side issues that management must consider in conjunction with performing a Software Safety Analysis. These areas have a parallel in the hardware world; the specific concern is different for the software world. The areas include:

1. What type of person can perform the software safety analysis The integration aspects of the software and hardware dictate that there must be a definite capability for a system view by the analyst. He must be capable of viewing and understanding the hardware and its functions, understanding the hazards and the preventative procedural requirements; and he must have a detailed understanding of software flow. He will be required to participate in defining the design of the software, and will review detailed flows to ensure that safety requirements are not inadvertently negated by the logic flow of the software. He does not have to be a programmer; he is a translator of hardware language into software language.

2. Configuration Control/Quality Control In brief, if there is not a strong configuration control/quality control program, the software programmer will be able to circumvent the controls. If this can happen, the safety program can be negated. The software issue is more complex than with hardware; one can inspect the physical configuration of hardware, but not the software program as loaded in the processor. One can lock up the hardware and account for all items, but he can never be sure he has all copies (or even the true master copy) of a software program without strong configuration control.

3. Human Factors The software being analyzed is the basic link between the hardware and the operator. We have addressed the hardware requirements upon the software; the interface between the software and the operator is an open area.

   The new MIL-STD-1472C is the first attempt at giving guidance to the software community. Issues of concern include the number of communications per unit time, type and clarity of displayed communications, required level of operator interaction, and the type of operator actions.

   Work in the area of defining the man/computer interface is currently being pursued by the Human Factors Society. A literature search will help the user in this area.

I. MANAGEMENT CONSIDERATIONS (Continued)

4. Reliability of Software By the classical definition of reliability, software is always 100% reliable, since by its very nature, software cannot fail. Software will contain errors, and because of errors, software may do the wrong thing; but software does not break. Attempts are being made to define reliability (and maintainability) of the software in terms of number of predicted errors, and time to recover and rerun the software. To date, no capability exists to segregate the number of predicted errors in the software by safety criticality or severity.

5. Software Checklists A checklist for designing software capable of controlling a unique hardware item can logically be divided into two checklist components. Part 1 includes the procedure controls for the type of hardware being controlled; Part 2 is a software design implementation checklist and is included in Section 3 of this paper.

II. TECHNICAL APPROACH

The analysis technique presented herein has been developed in compliance with the analysis requirement stated in paragraph 5.2.6 of MIL-STD-1574A, Systems Safety Program for Space and Missile Systems; a requirement which is now being applied to systems other than those governed by this standard.

For a hazard to occur, there must be some undesired/uncontrolled release of energy, energy normally contained by some hardware component(s). The failure, malfunction, or error in control of that hardware must cause or allow the hazard to occur; therefore, we are concerned with the software that controls the condition or state of the hardware components, or can monitor the state of the hardware components. A hazard can occur in the hardware because of a failure; because of an error in the software controlling the hardware; operator error, or because of a combination of hardware failures and software errors. The software error areas are treated by this paper. Two types of software fall into this narrow field of interest:

1. Command and control software - software that has direct control of the hardware, and which might cause a hazard (improper command), allow a hazard to propagate (does not detect the occurrence of the hazard), or allows a hazardous condition to go unnoticed by the operator (masking status). This category of software is termed "test sequence software."

2. Programs that generate the data base parameters upon which the test sequence software bases its decisions. Erroneous data can lead to erroneous decisions by error-free test sequence software or by competent human operators.

Implicit in the foregoing is that the software in the computer, and the I/O software are not subject to this type of analysis. These executive programs handle the internal operation of the processor, and are independent of the test sequence software. The computer self test software is normally contained in the systems or executive software. Safety should verify the capabilities of this self test and action taken if anomalies are detected. While errors that occur can affect the execution of the test sequence software, they are not directly traceable to a hazard. These programs need to undergo their own accuracy/repeatability studies. Qualification testing of the test sequence software should be designed in part to surface these errors.

II. TECHNICAL APPROACH (continued)

There are two basic differences between a software program, as loaded into a computer, and the logic circuitry of years past in which electronics performed the same sequence.

First, the software does not fail. Traditionally, the failure of the computer, and electronic components, have been considered in reliability and safety analyses. The electronics can be analyzed, and Mean Time Between Failure (MTFB) assigned. However, an MTBF cannot be applied to software, because the software does not fail. The "software failures" then, are actually software errors built into the program. This makes the issue of safety analysis both easier, because the failure modes need not be considered; and more difficult, because the errors are built into a difficult to visualize implementation scheme. Because software does not fail, a properly designed testing program can, in theory, detect all errors. In practice, however, this becomes virtually impossible due to the many possible combinations of sequences, and the timing of those internal sequences with respect to one another make the required testing exhaustive and time consuming. Further modern software testing include time compressions techniques which tend to make error windows so small that the window is not hit, and the error is undetected. An analysis technique that can prevent errors from being built into the code is more cost effective than an exhaustive testing program that hunts for the errors and in turn, requires changes to the code, followed by retest.

The second significant difference between software and hardwired logic is in the area of timing. Timing within a hardwired logic drawer is highly predictable. A set of stimuli given to a logic drawer will product highly predictable outputs, which are also highly repeatable. A computer, however, has only one set of processing registers, and can therefore only perform one logic function at a time. The computer scans from one input line to the next input line in a preplanned order. If a stimuli exists at a line (port), the processor will stop and perform the required function before continuing the scan. This makes the timing at a given input port unpredictable, although an outside limit can be set. Since the point in the scan is not known when a set of input stimuli are given to the computer, the starting point in the order of performance for the tasks is not known. The timing of a specific task, then, is not predictable or repeatable. This lack of timing predictability is what makes the software difficult to test and difficult to analyze.

II. TECHNICAL APPROACH (continued)

The analysis of the computer/software subsystem is best done hand-in-glove with the analysis of the hardware being controlled to ensure that hardware requirements are met.

There is, then, a very strong interaction between the Software Safety Analysis (SSA) and the Preliminary Hazard Analysis (PHA) of the hardware being controlled; the SSA can be considered as an extension of the PHA. The software is actually the data link in the command and control subsystem between the operator and the hardware system. Any hazard that occurs, or is allowed to propagate, occurs in the hardware, and as such, should have been identified in the PHA. The SSA then, is an extension of the PHA; in fact, the effort is to ensure that the software is designed to prevent hazards, and further, is designed so as not to contribute, but to detect hazards that do occur. For example, "valve XYZ fails to close" - translated into "software-eze" should be: "valve XYZ was not commanded closed" or "valve XYZ was not commanded closed at the proper time." The software analysis must confirm proper control and monitoring to meet the needs of the system operator.

Essentially, one needs to review the PHA and the functional capabilities of the hardware, and then search for ways that the software could cause, or allow, any particular problem to occur or to go unnoticed. This waterfall of the PHA into the SSA is in line with the logical design of the command and control link after the hardware system has reached some maturity, as in past Preliminary Design Reviews (PCR). The development timelines should overlap to allow command difficulties to be mitigated by hardware design change, as shown in Figure 1.

Software that links the operator to the hardware should be transparent to the system operation, much like the phone line and switching equipment are transparent to an individual making a phone call.

In reality, test sequence software is nothing more than an automated procedure. There is nothing done by the sequencing in the software that cannot be done with a simple series of switches and a step-by-step procedure, given time for the operator to react. The basic reason for using software is added capability, faster functioning, and the repeatability that lowers the possibility of human error.

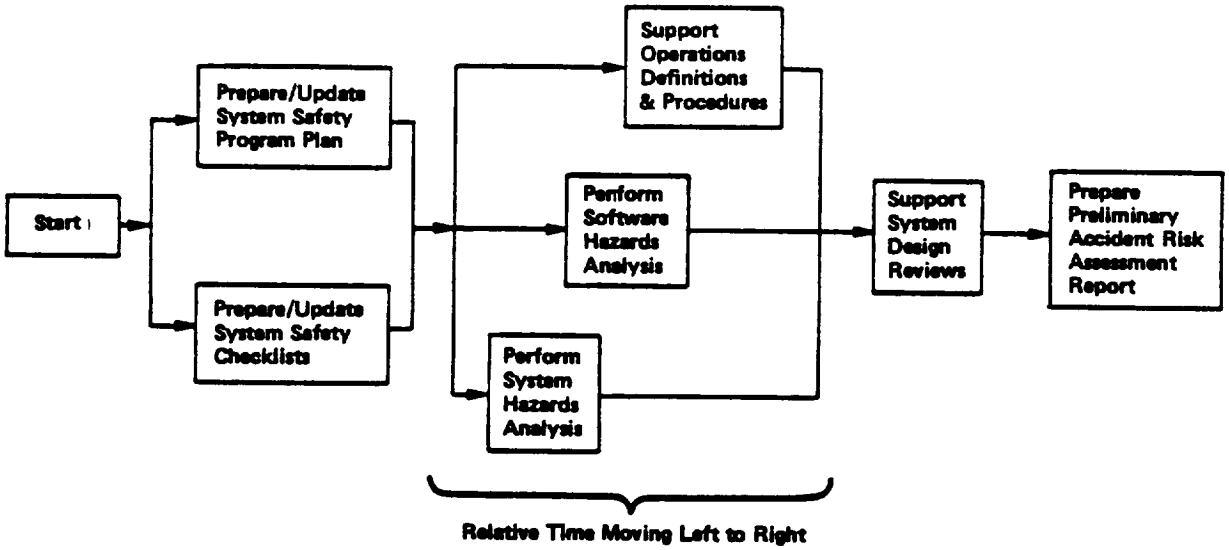


Figure 1. Safety Analysis Flow

II. TECHNICAL APPROACH (continued)

A large part of the analysis is in effect an Operating Hazards Analysis (OHA) against the automated procedure controlling a hardware system. The OHA must be extended to consider the processes, and often must be expanded to an integrated OHA, considering adjacent operating systems and mutual timelines.

The analysis itself is subdivided into two phases analogous to the differences between hardware and software. The timing relationships of the safety analysis to the software development as shown in Figure 2.

PHASE I: Analyze for the correct implementation of hardware functional requirements and interfaces. This consists of reviewing the functional and control requirements of the end-item hardware, along with the preliminary hazards analysis performed on the hardware; then participating in the software requirements definition and implementation so that appropriate safety requirements are included in the software requirements document.

During this phase, Part 1 of the Software Checklist (See Section 3) is used to help derive requirements associated with hardware procedural control requirements. It is also imperative that the analyst review all software requirements to ensure that none conflict with the safety requirements. Further, derived requirements can be expected to surface during this review.

For DOD Programs, these software requirements are published in a B-5 specification (as defined in MIL-STD-490). In developing the B-5 specification, the software analyst works just as a hardware system analyst. During the informal phase of the B-5 specification development safety requirements and concerns are worked informally; as the B-5 specification is prepared for internal management review, the software analyst should document open concerns on a Hazard Analysis form (See Figure 3a). The software hazard analysis (Phase I) should be documented and released to the Government along with the draft B-5 specification. For software program, the B-5 specification must be approved by the Government prior to holding the PDR. Government safety personnel should have the Phase I hazard analysis to review the B-5.

An often asked question, "Should all safety requirements be in one Section?" is answered this way. Many, if not most of the functional requirements affect safety and many safety requirements are functional. Therefore, I prefer to allow the functional

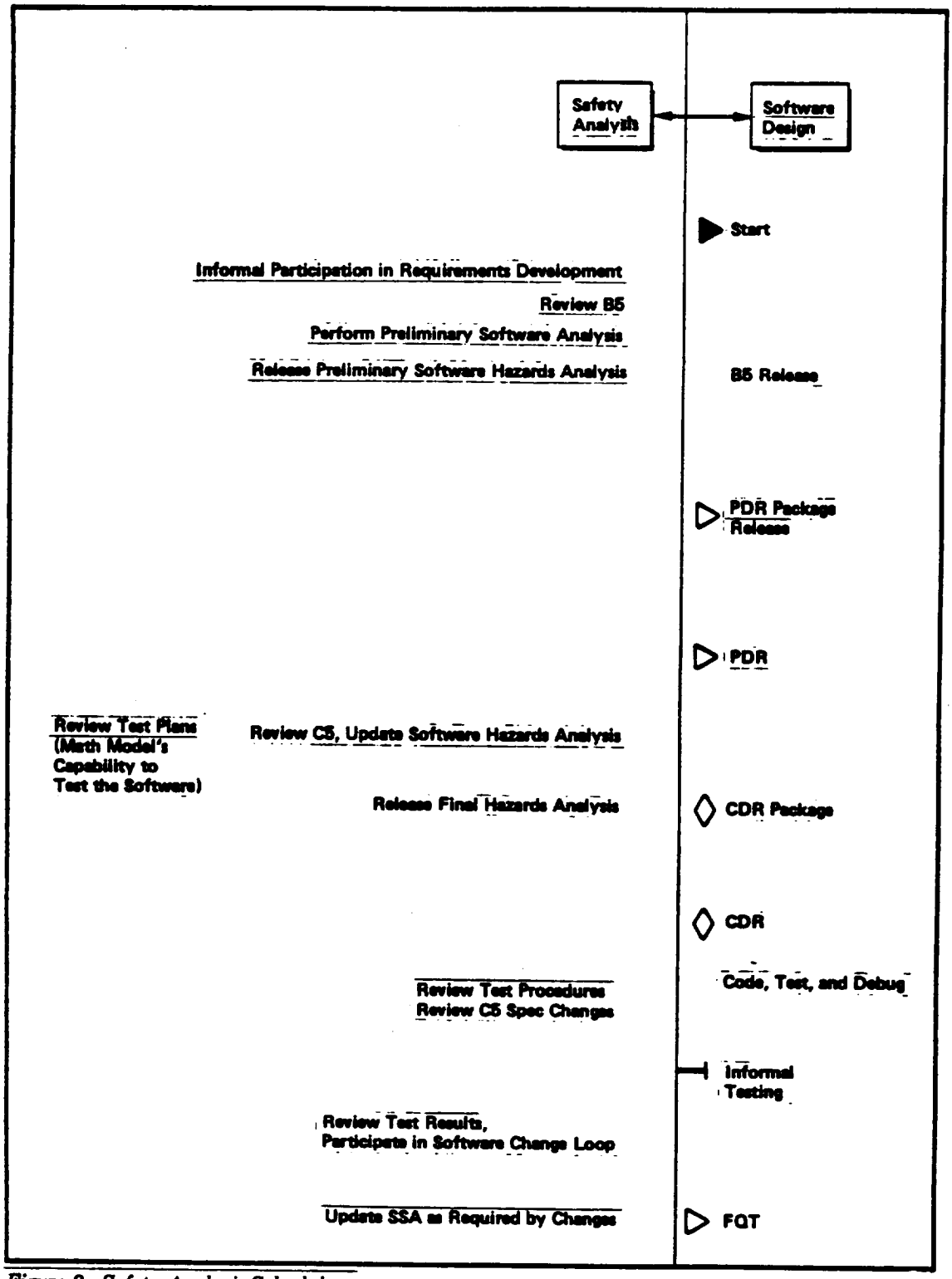


Figure 2. Safety Analysis Schedule

HAZARD ANALYSIS WORKSHEET

PROGRAM PHASE:	NO.:	REV.
AREA/STRUCTURE/SYSTEM:	PAGE: OF	
OPERATION/PHASE:	DATE:	
HAZARD GROUP:		
REFERENCES:		
HAZARD/NONCOMPLIANCE DESCRIPTION:		
POTENTIAL EFFECTS:		
ASSUMPTIONS/RATIONALE:		
HAZARD CONTROL REQUIREMENTS:		
DISPOSITION:		
HAZARD STATUS ☐ ELIMINATED ☐ CONTROLLED ☐ RESIDUAL	☐ CRITICAL ☐ CATASTROPHIC	☐ CONDITIONALLY CLOSED* ☐ NO HAZARD
PREPARED BY: (NAME/GROUP/PHONE)	HAZARD CLOSED (SIGNATURE/GROUP/DATE)	
REVIEWED (SIGNATURE/POSITION/DATE)	APPROVED (SIGNATURE/ORGANIZATION/DATE)	
* SEE:		

Figure 3a. Hazard Analysis Worksheet

SOFTWARE SAFETY ANALYSIS

Software function (change)	Function description Summary	System Hazard	Hazard Category	Safety Impact Discussion/ Conclusion	Recommended Requirements to Control Hazard	Remarks

Page ___ of___

Figure 3b. Software Safety Analysis Summary Sheet
(listing of several Hazard Analysis Worksheets)

requirements to fall with similar functional requirements; only very special safety requirements such as "terminal countdown entry check," critical anomaly detection, critical shutdowns, etc. are put in the Safety paragraph heading.

The final portion of Phase I is to delineate the test requirements used to verify that the code is in compliance with the safety requirements. The safety analyst should write the initial test requirements for safety requirements he submitted; further, he should review the test requirements document to ensure that safety related requirements are adequately tested.

In addition to hazards which can occur in the end-item hardware system, the Phase I SSA should also consider, to the extent possible, failure of the computer processor itself. Several items in the checklist deal with various possible hardware failures. Other methods include partitioning the memory to detect an instruction skip and top critical processing. Complete failures, which cannot be counteracted by the software design, must be considered by the end-item hardware design.

A further requirement on the software system can be made; if an illegal entry to a critical routine is detected, specific safing actions are to be taken, followed by removal of any command capability the computer has, since a hard failure of the processor can actually change the software. Special test or verification routines may be employed to periodically verify the integrity of the memory, such as a sumcheck.

Redundancy has not been successfully applied to software because the error is built into the program, and the software logic is deterministic. Therefore, the duplicate program, given identical stimuli, will make identical decisions. To achieve software redundancy, two different programs which are functionally equivalent are required.

PHASE II: <u>Analyze for correct implementation of the hardware functional requirements and interfaces</u>. This phase starts with design of the software architecture, selection of the host computer, and decisions such as dedicated computer versus a CPU sharing setup where more than one program is resident for execution. The objective of Phase II is to ensure that the software can execute the requirements, and will not be negated by other resident programs. The software architecture and CPU sharing issue has a significant

impact on how the safety requirements should be implemented. The use of priority interrupts, or frequency of cycle, etc., are examples of methods to ensure that the critical safety logic can be executed.

A passing word about modular, top-down structure code-in this code, each small function is contained in one subroutine, which does nothing else. There is exactly one entry and one exit point from each subroutine (or module) of code, and a main program or sequencer calls the modules. This makes the safety analysis, testing, and maintenance of the code much easier than coping with the many interlocking loops evident in some programs, where branching becomes more difficult to predict.

Phase II analysis deals with development of the detailed requirements in the C-specification (product specification), including detailed design of displays and interaction with the operator. In the interest of human factors, the safety analysis should review operator interface to ensure reporting of significant events and minimizing insignificant events or events over which the operator has no control. Display design should represent the hardware system to the operator. The code can also be designed to prevent operator error by screening status tables to ensure that proper conditions exist before an operator-initiated command is allowed to be sent to the hardware for execution.

During the Phase II implementation portion of the analysis, we are concerned with two parts of the program:

1. Analyze the implementation of these functions designated as safety critical.

2. Analyze functions that can influence the execution of safety-critical functions.

The detailed program design must be reviewed to ensure that each safety critical function is implemented and will execute correctly, but the detailed analysis can be restricted to code falling into those two areas. Much of the in-line code will not directly affect the safety of the operating hardware, and need not be studied in detail past the timing interface.

For those functions that must be analyzed in detail the sequence is:

II. TECHNICAL APPROACH (continued)

1. Detailed requirements, including application of the priority scheme.

2. The detailed program design, and in isolated cases, the code itself, or the logic design.

It is of prime interest to find possible errors in the timing of functions due to a) competition for the CPU, b) because of overlapping in the executions, or c) the use of data which is subject to change (or the use of incorrect data). Stated another way, we are interested in anomalies such as:

1. Software may fail to execute.

2. Software executes at the wrong time.

3. Software may execute based on incorrect data.

4. Software may execute differently in redundant processors (usually input timing, or changes of input stimuli within a single cycle).

During this analysis, the analyst can oversee the application of various coding techniques/restrictions imposed via Pat 2 of the checklist.

The Phase II analysis should be documented as the C-specification itself is documented, and transmitted to the customer for review prior to the critical design review (CDR), as shown on Figure 2.

The actual coding and debugging activities will commence after the CDR. A low level of safety effort is required during that time period to ensure that C-specification changes (of the detailed design requirement) are consistent with safety requirements.

Following the debugging activity, the software program should enter into informal (or internal) qualification testing, and should come under informal configuration control. This is necessary to ensure code integrity (from change) and prevent unnecessary re-testing when a change is required. During informal testing, the safety analyst should review test anomalies as well as the required C-specification and code changes for safety impact. This puts the analyst in the position of being able to recommend going ahead with Formal Qualification Test (FQT), and ultimately, makes the analyst able to sign at FQT with confidence.

III. DETAILED PROCEDURE

The first portion of the procedure is to gather and review the following data:

Hardware System PHA

Hardware System Functional Operation Requirements and Timeline Sequence Requirements

Hardware System FMEA

Hardware System Human Factor Analysis

As depicted in Figure 4, The analysis is performed as follows. For benefit of this paper, I have assumed that the system can be operator controlled or automatically controlled. The reader can restrict his analysis, if appropriate.

For each hazard which can be mitigated by some procedural control, take the steps of the flow in Figure 4,

1. Specify the procedural control (Block 3).

2. Specify the Software requirement as indicated by the Figure.

3. In Block 10, if there is a time constraint associated with the developing requirement, automatic control (Block 8) may be dictated because of the short response time.

4. In Blocks 6 or 7, be sure to annotate the HAW to insure that the written procedure is consistent with the software.

5. In Blocks 9 and 10, the determination of software priority associated with the requirement must be made. It is necessary to insure that the module of code can get access to the CPU, execute, and the output commands (or operator notification) acted on with the time constraint.

This basically concludes Phase I of the analysis. Any safety requirements generated, and not incorporated in the code requirements document (B-5 Spec) should be annotated on the open HAW, or a new HAW generated to document the safety concern.

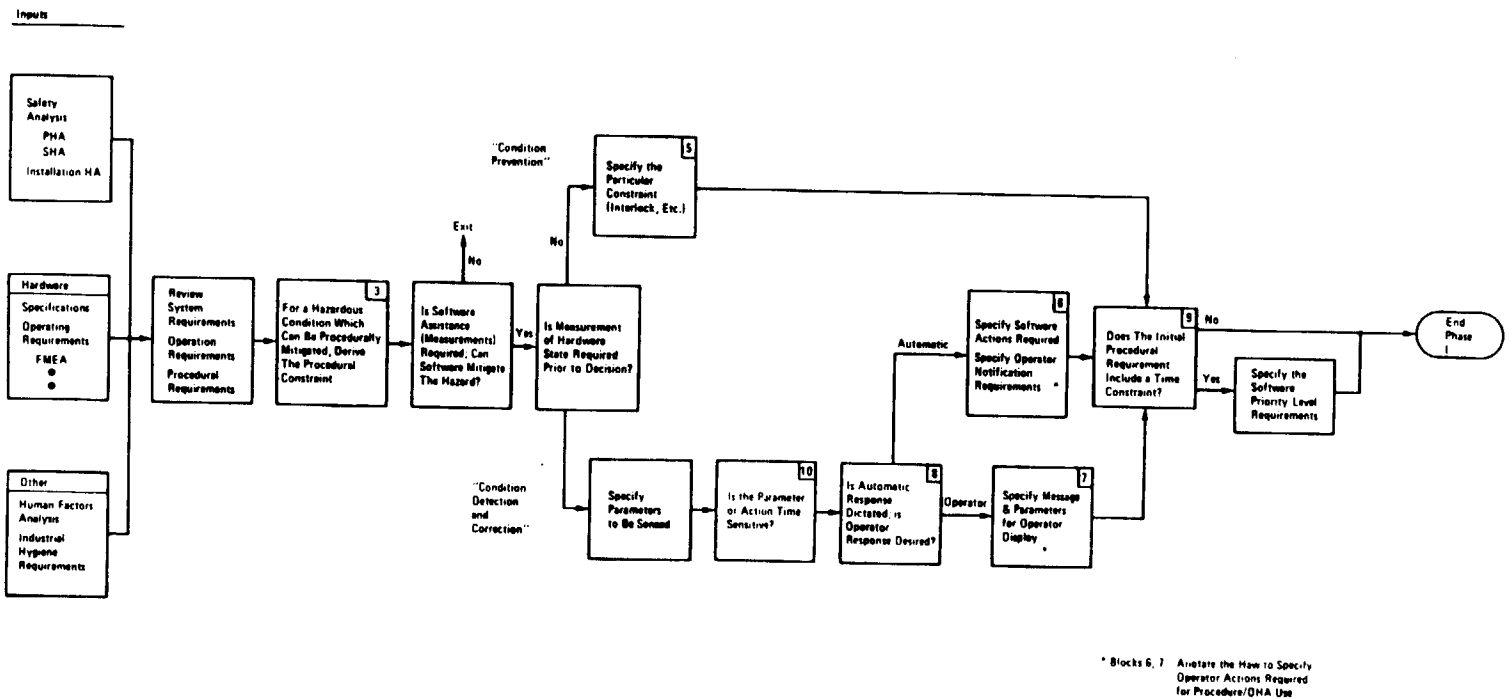


Figure 4 Phase I Software Requirement Determination

III. DETAILED PROCEDURE (Continued)

The Phase II portion of the effort is in some aspects more general, as in discussion on Software architecture, and then more specific, as in review. The detailed logic flows to insure implementation.

Parts of this effort are parallel to Phase I:

1. Discussion on architecture of the code to enhance safety and maintainability. In addition to the words in the technical approach, I can only admonish the Safety engineer to stay away from branches within loops, i.e., "spaghetti code."

2. Write any specific test requirements to insure the proper implementation of the Phase I requirements.

Parts of this effort follow acceptance of the design requirements:

1. Review the detailed design requirements for correct interpretation within written C-5 requirements, and detailed logic (preferably computer derived logic flows after initial coding) will follow acceptance of the B-5 Specification.

2. Review the logic flow for:

   a) Blocks of code controlling hazardous functions.

   b) Safety logic associated with anomaly detection for those hazardous operations.

   c) Any code which the architecture would allow to affect the execution of a) or b).

3. It is important to understand any timing quirks of the computing system the software program will execute on, and review safety critical timing requirements for any conflects.

Now you have given the analysis your best shot. Any safety concerns as a result of steps 1 through 5 should be documented on a HAW, either against an existing open worksheet or a newly documented concern.

The safety engineer must follow the informal qualification testing to insure,once again, proper execution of the safety critical code, as identified in 4. a) through c), above.

IV SOFTWARE CHECKLISTS

For test sequence software, which is effectively an automated procedure, the primary focal point will be the procedural requirements from safety source documentation. This information will be in Part 1 of the checklist (for use in Phase I of the analyses effort); and will be supplemented by software design standards that can be given to a programmer (congruent with Phase 2 of the analyses).

Part 1 Hardware Procedural Constraints

Procedural constraints from safety source material:

Air Force Design Handbook 1-6
Explosives Safety Manual
Etc.

Plus:

X. ANOMALY DETECTION

a. The software logic, upon detection of an anomaly, shall revert to a safe condition informing the operator of the following:

1) What anomaly was detected.

2) What action was taken.

3) Current system configuration.

Workaround procedures shall not be allowed (except contingencies).

b. Critical hardware situations shall be identified and safing features designed into the logic.

c. All functions identified as creating hazardous conditions for personnel or equipment shall have the appropriate logic designated and priority assigned.

d. Hardware anomalies requiring an action within (TBD) seconds shall be controlled by logic. A hardware safing panel is not considered to be a primary safing avenue.

e. Upon detection of a processor malfunction (by self test) appropriate safing commands shall be sent, critical (hazardous) commands shall be removed. Note that each output can look like a command to the hardware.

Y. OPERATOR INTERFACE

Operator I/O interactions shall be straightforward, unambiguous, and comprehensive. Thus,

a. Operator I/O interfaces shall provide detection of operator errors, error notification, recovery procedures, and failsafe features on critical commands or processes.

b. Defaults shall always be positively identified and prevailing conditions shall be given to operators.

c. The logic shall allow operator override. except for critical automatic safing. This shall be implemented with a dual selection "arm/fire" procedure. Any override of safety logic shall be identified to the test conductor via display on the test conductor's CRT. (This allows a 2-person agreement to the deviation.)

Part 2 Software Design Standards

1. GENERAL

a. Interrupt processing logic shall be specifically defined and documented.

b. Software baseline shall be tied to an approved hardware baseline. Floating baselines make safety requirements traceability difficult.

c. Concurrent programs being iteratively performed shall be in the same main and any time value used shall be from the Interval Timer.

d. When data/commands are passed from one program to another program, there must be a complete interchange (handshake) to ensure the correct exchange of data.

e. When sensing hardware configuration, equipment sensors shall be utilized (open/closed indicator). The presense of an open command output discrete shall not be used to indicate open state.

2. PROGRAM DESIGN

a. Functionally redundant processing shall be considered for critical functions.

b. Program design and code shall be structured to enhance visibility, readability, maintainability, and comprehension of program logic and program interfaces.

c. Program design and code shall be modular in an effort to reduce logic errors and improve logic error detection and correction functions.

d. Hazardous or time-sensitive processes shall be identified and considered for automation rather than operator execution (see 4.d. below).

e. Delay statements shall be evaluated to determine if interrupts will be adversely affected. (Interrupts are effective only after statement execution is completed).

f. Calls generated to other procedures shall require that conditions be positively specified, determined, and effected (interrupt processes).

g. Decision logic shall not be based on values of all "ones" or all "zeros." A specific pattern should be used to reduce the probability of malfunctioning hardware/software interfaces supplying false values that satisfy decision logic.

3. TIMING CONSIDERATIONS

a. I/O wait times, which occur when accessing disk or other memory for data, shall be considered in automatic/operator interactive decisions.

b. Interval Timer accuracy dependence on system loading to decrement once per second shall be considered in critical response time determinations.

c. Operations employing time limits shall consider the system processing load responses time. Time limits shall not be used to generate pulses of critical duration.

d. Operations employing time limits impacting the system safety shall have those time limits included in the logic. The safety-critical time limits in logic shall not be changeable by the operator from the console. I/O wait times (worst case) will be considered when timer logic is determined.

4. ANOMALY DETECTION

a. Upon detection of an anomaly, revert to a safe condition informing the operator of the following:

1) What anomaly was detected.

2) What action was taken.

3) Current system configuration.

Workaround procedures shall not be allowed (except as contingencies).

b. Critical hardware situations shall be identified and safing features designed into the logic.

c. All functions identified as creating hazardous conditions for personnel or equipment shall have the appropriate logic designated and priority assigned.

d. Hardware anomalies requiring an action within (TBD) second shall be controlled by logic. The requirements for safing (flow and revert) shall be contained within control logic. The safing panel is not considered to be a primary safing avenue.

5. OPERATOR INTERFACE

a. Operator I/O interactions shall be straightforward, unambiguous, and comprehensive.

b. Operator I/O interfaces shall provide detection of operator errors, error notification, recovery procedures and failsafe features on critical commands or process.

c. Defaults shall always be positively identified and prevailing conditions shall be given to operators.

d. The logic shall allow operator override. This shall be implemented with a dual selection "arm/fire" procedure. Any override of safety logic shall be identified to the test conductor via display on the test conductor's CRT.

SOFTWARE SAFETY ANALYSIS

WHAT IS IT?

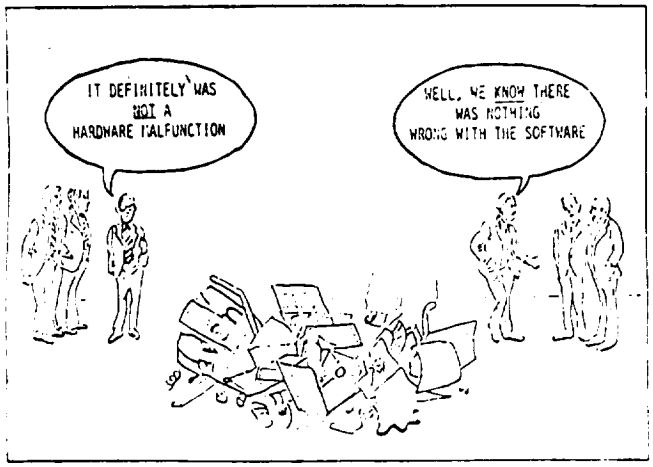
- O ANALYSIS OF REQUIREMENTS FOR, AND THEIR IMPLEMENTATION IN, SOFTWARE CONTROLLING END ITEM HARDWARE (WHICH HAS HAD A SAFETY ANALYSIS)
  - REQUIREMENTS DERIVE FROM HARDWARE FUNCTIONAL/SAFETY REQUIREMENTS
  - IMPLEMENTATION ANALYSIS IS TO INSURE "CORRECT" EXECUTION OF SOFTWARE AGAINST THE HARDWARE

SOFTWARE SAFETY ANALYSIS

OVERVIEW OF APPLICATION AND BENEFITS

- O APPLIES TO COMMAND AND CONTROL SOFTWARE (TEST SEQUENCE SOFTWARE) (DATA BASE)
  - EXTENSION OF SAFETY ANALYSIS PERFORMED ON END ITEM HARDWARE
  - COMPLETES AN ANALYSIS OF ENTIRE SYSTEM, INCLUDING C&C LINKS
    - o GIVES CONSISTENT LEVEL OF EXAMINATION TO ENTIRE SYSTEM
  - FILLS OUT THE BASIC MISSION ASSURANCE DISCIPLINE OF SAFETY

WHY ANALYZE SOFTWARE?



SOFTWARE SAFETY ANALYSIS

SOFTWARE SAFETY ANALYSIS IS PERFORMED IN 2 PHASES

PHASE 1

- O ANALYZE FOR CORRECT IMPLEMENTATION OF HARDWARE FUNCTIONAL REQUIREMENTS AND INTERFACES
  - REVIEW FUNCTIONAL AND CONTROL REQUIREMENTS OF THE EQUIPMENT BEING CONTROLLED AND THE HAZARDS ANALYSIS OF THAT EQUIPMENT
  - PARTICIPATE IN SOFTWARE REQUIREMENTS DEFINITION AND IMPLEMENTATION TO ENSURE THAT THE HARDWARE IS KEPT IN A SAFE MODE OR CONFIGURATION
  - BASICALLY AN OPERATING HAZARDS ANALYSIS, WORKING PROCEDURAL REQUIREMENTS ASSOCIATED WITH THE END ITEM HARDWARE
  - RESULTS IN A B-5 SPECIFICATION WHICH CONTAINS THE FUNCTIONAL AND SAFETY REQUIREMENTS, A SOFTWARE SAFETY ANALYSIS DOCUMENTING REMAINING SAFETY CONCERNS

SOFTWARE SAFETY ANALYSIS

SOFTWARE SAFETY ANALYSIS IS PERFORMED IN 2 PHASES (CONTINUED)

PHASE II - IMPLEMENTATION ANALYSIS

- O STRUCTURE/ARCHITECTURE/DESIGN OF SOFTWARE
  - ANALYZE/UNDERSTAND THE DESIGN OF THE SOFTWARE PROGRAM DURING ITS EVOLUTION, AND SAFETY IMPACT
    - o TOP DOWN STRUCTURE
    - o SHARING OF CPU WHERE CONCURRENT PROGRAMS ARE RESIDENT
- O REVIEW FOR IMPLEMENTATION ERRORS OR ANOMALIES
  - DETAILED REQUIREMENTS
  - DETAILED PROGRAM DESIGN
  - CODING (WHERE INDICATED)
  - PRIORITY SCHEMES (SAFETY CRITICAL COMMANDS/MEASUREMENTS MUST HAVE HIGH PRIORITY)
- O ANALYZE FOR INTERNAL ANOMALIES WHICH AFFECT THE EXECUTION OF THE SOFTWARE
  - IMPLEMENTATION OF THOSE FUNCTIONS DESIGNATED SAFETY CRITICAL
  - FUNCTIONS WHICH CAN AFFECT THE EXECUTION OF SAFETY CRITICAL FUNCTIONS

SOFTWARE SAFETY ANALYSIS

ENTIRE PAPER SUBMITTED FOR INCLUSION IN PROCEEDINGS

- THEORY
- DETAILED PROCEDURE
- SOFTWARE/PROCEDURE CHECKLISTS

DISCUSSION OF EVOLVING CAD/CAM USAGE

By R.G. Damskey, Hughes Aircraft Company

I believe CAD/CAM (Computer Aided Design/Manufacturing) to be a key tool in being a cost effective, contributing member of the high technology industry. Luckily, so does my management. As with any tool, there are great advantages in using it wisely and disadvantages in not. This discussion is to tell you what we've learned during the past 24 months utilizing CAD/CAM, what we hope to learn in the future, and to stimulate any comments from you.

BACKGROUND

First, a few words about our function and our current CAD system. We are a structures design/integration organization with two major support roles, mass properties/alignment and wire harness design. Additionally, we monitor the fabrication and integration of both the structure and wire harness. To aid these efforts, we have the Lockheed software (CADAM) on an IBM mainframe, Adage terminals, Fiberoptic transmission, plus 11 inch and 36 inch electrostatic plotters. Currently, we have 2-dimensional capability on the Basic System, 3-dimensional capability with the Northrop NCAD software, and real time rotational ability via a "joy stick" which is incorporated into the NCAD package. Prior to 1981, we had been using the Computervision (CV) system on a very limited basis and found it not to be user friendly enough for our general use. The complicated command sequences necessitates continuous use to maintain proficiency. This did not lend itself to our design organization where the design engineers are responsible for researching and coordinating their designs. Upon initiation of a project, a couple of weeks of research and liaison are not uncommon prior drawing the first line. As a result, a simple, user friendly system which would "key" the designers memory for learned manipulation commands was sought. The Lockheed package with its prompting menu was our choice.

Once the system was in-house, we started learning of it and on it. The first thing we learned was "DON'T THROW AWAY THE BOARDS"! CAD is not for everyone and everyone is not efficient on CAD. Current CAD systems have relatively small display tubes which means only a restrictive portion of a large drawing can be seen at one time to work on. The individual operator must be able to keep the rest of the drawing in his minds eye to insure any change made on one "seen" area of the drawing are incorporated later in all other affected views. Not all individuals are adept at doing this. Additionally, to be efficient at the "tube", the manipulation of the keyboard and light pen must be automatic to the operator. Some individuals cannot master the necessary techniques. As a result, we have found it advantageous to keep drawing boards for those individuals in the above categories. Plus, it is hard to hold informal design critiques around a terminal due the focus of the equipment on a small operator area and the fragmentation of drawing. A large print layed out on a drafting table with access all around it has been found most conducive to a meaningful exchange.

Our CAD terminals are adjacent to the conventional boards in the design areas Some thought has been given to the low level hum inherent to the CAD terminals. Research indicates that the hum is considered a beneficial "white" noise which drowns out the superfluous conversations allowing better concentration on and off the terminals.

Finally, design engineers in our organization spend approximately 40% of their time researching, conceptualizing, and doing liaison work. This leaves 60% of his/her time for designing on the tube. Obviously, having a dedicated terminal for each individual is an extravagant use of resources. Therefore, the terminals have been placed in groups of four (the maximum string for Adage) adjacent to the conventional boards to encourage a sense of common usage not individual ownership. Currently, this approach is working with terminal usage consistently above 100% based on an 8-hour day. (This is done via a flextime policy allowing 11 hours of possible usage per day).

CURRENT USAGE

Our efforts to present have concentrated on the CAD portion of CAD/CAM. We have only just begun working out our approach on CAM. Our two basic software packages, CADAM (2-Dimensional) and NCAD (3-Dimensional), are most useful in different design and support areas. CADAM is used primarily for production drawings (assembly, installation and detail), CAE (Computer Aided Engineering) which are the analysis functions, the engineering check function, and as a data base for the manufacturing tooling and parts inventory systems. NCAD is primarily used for preliminary design studies with view studies done on the "joy stick" terminals.

Design Assurance is enhanced in two areas via the use of CAD. First, the basic hardware design, its verification, and resulting tooling; second, the increased number of iterations possible with automated design.

Basic hardware design consists of layouts which are detailed to give the necessary drawings needed to manufacture and assemble piece parts. A layout on CAD can literally be "built up" of major subassemblies given a basic set of design parameters. This process does not save any time over the conventional manual approach, but the number of iterations possible in the same time frame is significant. In fact, there is a definite tendency to continue iterating a design after an efficient configuration has been found just because it has become so easy. After a layout is complete, a large time savings and increase of accuracy is found in the engineering check activity. Large dimension structures have been difficult to check for scale errors among the different views, the locations of critical points and tabulated dimensions not coinciding with their field of drawing counterpoints. With CAD the time to check the above has been cut from days to a few hours while the accuracy is as exact as the computer can give, in that it electronically measures between the points the checker indicates.

After the layout has been completed and looked at from all angles to insure the design envelope has not been violated, the detail drawings are made. Here a library of piece parts which has been built up is used extensively to detail the hardware being built. Gone are the days of drawing the nut and bolts over and over. Now they are electronically duplicated from the library in milliseconds. Additionally, the parts lists from these drawings are electronically transferred from the CAD into the Product Information Network System (PINS). This data system is used by Manufacturing to procure piece part material and hardware. Further, the PINS information is used to plan the fabrication of hardware plus positively control the material going into the fabrication process. It now takes a conscious

and concentrated effort to make erroneous hardware substitutions in that the only materials available for the particular item being fabricated are those listed in the PINS.

We have found no significant gain in using CAD to check detail drawings over the conventional means.

Next, assembly and installation drawings are electronically generated using the layouts as an overall guide and details as the building blocks. The time saved plus accuracy incurred by using the existing detail drawings data base to generate these drawings is significant. However, a real bonus comes with changes. An alteration to a drawing and analysis of its affect with subsequent changes to all other affected drawings can now be done in hours, not days. Plus, we have found a significant reduction in the number of changes needed, our first CAD program gave an order of magnitude less changes than the prior historical average.

On the manufacturing front, the same data base is used to design the tooling necessary to fabricate and assemble the subassemblies. The design of this tooling is also done on CADAM and enjoys all of the benefits of accuracy previously explained. As a result, we have experienced a phenomenal ease of hardware "go together". This has a very positive effect on both the fabrication and design personnel.

Additionally, the engineering check of insert locations from the installation drawings has dropped from one week to one day with a significant increase in accuracy. To a lesser degree, this is also true of the blanket and paint/marking drawings.

Finally, we are utilizing CAD in the electronic harness design. This is the one design that ties every system together and as such, could benefit the most from the use of CAD.

Presently, however, CAD is just used as drafting tool in this area. Some time is saved on each harness drawing in that there is a high degree of redundancy of wires and terminals, an ability that CAD is especially suited for. This is not where the real benefits lie, it is getting the electronic box and systems people on the same electronic data base network. When this happens, the communications between the people who design the system (systems engineers) and the people who implement that system (electronic and harness engineers) will be iterative, not sequencial, as is currently prevalent. Overall design time will go down and product quality will go up.

As seen, although we have been using CAD for a relatively short period of time, it has impacted virtually every area in engineering and manufacturing. We have learned much about how to use this tool effectively, but we can see many areas where increased future use will be key to our viability.

FUTURE USAGE

For the immediate future, CAE in the form of the analysis functions and CAM abilities of the system are being explored. The analysis field has two areas of development. The first is in Mass Properties. There is considerable analysis time savings and increase of accuracy feasible with a computer-aided analysis by having the computer obtain the weight, moments of inertia, C.G. locations directly off the CAD

drawings. The former reduces the number of personnel needed to do the task, which can be relatively substantial on a large structure program. This allows fewer resources to be highly efficient and accomplish the same tasks. The latter allows the weight margin needed to adjust for the inaccuracies inherent to existing analysis techniques to be traded directly for increased payload weight capacities. This is a direct increase of the overall efficiency of the end product.

Currently, the available software Mass Properties analysis packages require a redefinition of the detail/assembly in terms of spheres, pyramids and rectangles. This added step, though the actual analysis is very fast, is not cost effective. We are currently working on the procurement of a viable software package.

The second analysis area being explored is that of stress and dynamics. Presently, this type of analysis is done by hand or if more complicated, a special computer program is written/adapted for each item. We want to take the detail/assembly electronic drawing and have the computer analyze it directly, needing only design imput criteria from the analyst. The benefits gained in terms of time and accuracy are the same as for mass properties. Currently we are looking at Finite Element Modeling (FEM), however, state-of-the-art software still requires massive reidentification of a part by the analyst to allow computer analysis. This again is costly and currently not acceptable. Direct analysis of the electronic design is the goal.

Now we have reached a point of this discussion where the major engineering is finished and manufacturing begins. CAM could eliminate a large error source in the manufacturing, i.e., one man's interpretation of another man's drawing. With CAM, the electronic drawing is directly digitized onto a tape which activates the milling machine, lathe, etc. Man assumes passive control of the process, he does not have to actively interpret drawings. We have run some experimental CAM parts, but are finding the software available still not advanced enough to handle our current operating mode. An example is that all lines on a drawing must be closed out before it is digitized. This necessitates the designer to view every section of a drawing at a sufficient scale to insure every curve and intersection are complete. This greatly increases the time at the terminal per drawing. We are working on the software which will automatically project lines over small distances and check that the part is contigous. Once this is accomplished, the inherent accuracy of the basic system combined with the ability to check fit by electronically assemblying equipment before it is made will be a major breakthrough in "getting it right the first time" without having an excessive expenditure of resources.

Possibilities in the near distant future include major data management available at the design engineers fingertips. Currently, a designer spends 40% of his time doing research and liaison to insure this design meets the requirements. With the overall equipment complexity going

up, we see this percentage going up. A major reduction could be realized if the information sought were in a central data bank. This means that all design and support areas must be automated. It is feasible and practical, however, there is a significant capital cost. All involved personnel would have to be trained, equipment procured, communication lines multiplied (though fiber optics can accomplish this with minimal physical impact) and management dedicated. A formitable task, but the alternative is a work day filled with meetings on what is to be done, and no time to do it.

CONCLUSION

I have made a lot of statements on how I perceive the function of CAD/CAM in my working world. And, it all is centered around communications; communications with each other, communications between man and machine, and communications between machine and machine. It is a necessity to bring the "let your fingers do the walking" thought process into the engineering world. If we don't, our knowledge will soon outstrip our ability to use it. We will become the modern dinasour, unable to keep up with the evolving environment and eventually collapsing due to inefficiency.

CAD/CAM AT TRW

John L. Cairns
Manager CAD/CAM Project

SCOPE

This paper is an overview of the CAD/CAM project at TRW. The project was started in 1979 at Space Park in the area of Mechanical Design. The project is now spreading into Electronic Design, circuit board design being the focus at this time.

The first part of the paper covers a brief description of the TRW organization that manages the development of CAD/CAM within the corporation. The objectives of CAD/CAM at Space Park and a description of the current system configuration are also included. The second part of the paper describes the performance that is being experienced and the lessons that have been learned during the the past three years.

ORGANIZATION FOR INTEGRATION

TRW is made up of three business sectors, 1) Electronic and Defense, 2) Automotive World Wide, and 3) Industrial and Replacement. (see figure 1) Each sector has a a CAD/CAM Council made up of senior group managers who establish the policy and perform management reviews. Each of the sectors also has a technical committee made up of department managers who either have, or want CAD/CAM systems. The technical committee is charged with the planning of systems, the evaluation of equipment and software, the day to day operations and performance evaluations.

Key members of the sector council serve on the Corporate Science and Technology Council which establish the policy and evaluate the development for the corporation. The CAD/CAM panel at the corporate office is made up of key members of the sector technical committees. This crossed network of councils and technical committees provides a good flow of information from the bottom of the organization to upper management and provides an excellent path for communications between the "high technology" side of the company and the commercial side of the company.

SYSTEM OBJECTIVES

Six system characteristics were defined in the beginning and continue to be the guiding objectives today.

1) NO DEDICATED OPERATORS. CAD/CAM is a tool not a service at TRW. We are dedicated to the eradication of the design merry go round, where an engineer gives sketches to an operator who loads the data in a computer, plots the result, returns the drawings to the Engineer for review and then go around the loop each time a

PAGE 152 INTENTIONALLY BLANK

change is made. Our studies show that an Engineer spends a full 30% of his time telling lead designers and technicians what he wants and checking to see that he gets it. The same Engineer with a tool that is 3 to 4 times more productive than manual methods, can do the job in the same time that it takes to explain the job. That frees the engineers time to do more work, or put more effort into the design task at hand.

2) EASY TO LEARN. TRW wants all engineers to be users. Training programs of 100 hours of formal training and another 6 months of on-the-job training would be unthinkable for 7,000 engineers. The Engineer is also an occasional user of a design system (the TRW average is 500 hours per year) therefore the systems must be more than easy to learn it must also prompt the user and provide online help.

3) STANDARD HARDWARE AND SOFTWARE. The half life of computer hardware and software is about 9 months. We do not want to get stuck with some special equipment or programs that only run on a particular brand of hardware. Particularly devistating are the systems that require new hardware to run the latest version of the software you are committed to.

4) PRACTICAL DATA SECURITY . Users should not be bothered with remembering which disk pack or magnetic tape the data is on. A user should be able to call up an index of drawings and pick the one they want. If the user wants an archived drawing there should be one phone number to call to have it restored.

5) DATA BASE ACCESSIBILITY. Once information is captured in a computer it should stay in the computer. Many systems only produce a drawing that must be read by people and selected information copied into the next computer. Vendors have this thing about priority data structures, that's ok if they will just make the information that is stored in the structure available as information to other computers.

6) INTEGRATED SYSTEM. TRW will pass up the best system in the world if it will not integrate into the company system. The developers of Mechanical Design systems are learning this lesson. Most of the major systems in Mechanical Design and analysis can exchange major portions of their data bases with each other. The national standard IGES is making this exchange even easier. The electrical design and analysis folks however, are way behind in this integration effort. Particular consern at TRW is the integration of schematic capture systems with the circuit board design process. There are several good schematic capture systems on the market. All can extract a "string list" and a "parts list" but none will provide the capability to transfer the picture along with the symbols and information content of the schematic to another system.

SYSTEM DEFINITION - MECHANICAL DESIGN

The Space Park system for mechanical design presently consists of:

(a) An IBM 3081 computer at a central site with communications to user work areas up to 10 miles away from the computer.

(b) CADAM -(105 Stations in 27 Locations) CADAM is a software package developed by Lockheed and marketed by IBM. About 35% of the mechanical design at Space Park is done on this system. That share will grow to 70% by the beginning of 1985.

(c) NASTRAN, SAP, SINDA are the primary structural and thermal analysis packages that have been integrated with the CADAM system.

(d) CADAM-APT and McAUTO UNI-GRAPHICS are the primary tools of the N/C tool preparation. CADAM is used for conventinal tool design, while the McAUTO system is used for N/C tool preparation.

(e) COMPUTER OUTPUT MICROFILM (COM) is a direct output of the CADAM and is the method we use for the documention of engineering drawings for configuration control.

The Space Park system for Electronic design presently consists of:

(a) CBDS -(8 stations in 4 locations) the Circuit Board Design System is a software package developed by the Bell Northern Electric company of Canada and marketed by IBM . CBDS is a complete system that includes schematic capture, logic simulation, parts placement, placement optimization, routing, artwork generation, engineering documentation, fault test pattern generation for automatic test and fabrication tooling. This system will be expanded to 21 stations in 12 locations by the middle of next year.

(b) Silvar Lisco, Design Aids, and a few others are being used for schematic capture.

(c) SPICE, DVS, TEGAS are used by the electronic engineers for circuit simulation and analysis.

(d) Gerber photo plotters are used for circuit board and substrait artwork generation.

(e) Excellon n/c drills and FACT test machines are used in the fabrication and test of circuit boards.

PERFORMANCE

Performance by the traditional ratios of time to do a job are as follows:

Conceptual Design	2 : 1
Structural Modeling	7 : 1
Mechanical Design	3 : 1
Facilities Design	4 : 1

Tasks that require thinking time do not benefit as much as those that are more mechanical. These numbers are interesting, but what does it mean at the bottom line? The computer costs at TRW are charged direct to project job numbers. This means that the cost of a person working with a computer is approximately 80% more expensive than a person working without a computer. A person that is two times more productive with a computer is only approximately 20% more productive on cost.

An example of a real bottom line can be seen in table 1. The following example is from one design department.

TABLE 1

	Head count (Average)			
YEAR	DIRECT	CONTRACT	TOTAL	DRAWING COUNT
1981	80	54	134	1608
1982	77	12	89	1968
1983	85	5	90	2880

The work load since 1980 has been steadily increasing while during the introduction of the CADAM system the manpower in the department has been dropping, and the productivity of the department has been increasing. This department does not have enough terminals for all of the designers. The productivity ratios of 3:1 or 4:1 do not account for all of the increase. The real productivity increase comes from the integration of the analysis with the design function the elimination of some checking functions the compatibility of the work with the manufacturing systems, and the reduced number of errors is the design that used to 'rob' the productivity of the department. This department put the best people in the organization on the system thus getting more mileage out of the people that are already highly productive.

The same thing can be seen in the performance on the first 9 circuit boards that were designed with the new CBDS system. (see table 2) Manual design process produces a multilayer circuit board in an average of 19 weeks, for a cost of $28,000. Over 40% of the labor in the cost of the PCB is in checking.

The results of the first 9 boards designed with CBDS show an average of 12 weeks were required to present the artwork to

manufacturing at an average cost of $21,000. These boards were produced by designers during their training period with beta test software, that changed configurations 3 times during the production phase. We expect the schedule for circuit board design to drop to 8 weeks and costs to drop to approximately $10,000 by the end of the year. This will be a 70% drop in both labor and cost along with a 57% drop in span time. Our expectations might be high but they are in the right ball park.

Table 2 - Circuit Board Design Performance

	BENCH	CHECK	WEEKS	TOTAL COST
MANUAL METHODS	359	260	19	$28K
PRE-PRODUCTION (BETA TEST)	291	161	13	$20K
EXPECTED	146	36	8	$ 8K

LESSONS LEARNED

The lessons taht we learned are the same ones that we learn over and over again. For some reason we always think new systems bring new problems but in reallity they just are the same problems in a new format. The 8 key lessons are:

SYSTEM LESSONS

1) First and foremost, have a long range plan for development, review it, and stick to it. A real plan is easy to explain to management and customers alike. The plan will help to resist the neat new products that you are bombarded with by virtually every vendor in the business. The plan will need attention every 3 to 6 months as new requirements bubble up and new products become available.

2) We believe that the integration of automation systems into an overall company tool is more valuable than the original automation of functions. There is now such thing as a turnkey system. Nothing really stands alone, everything requires at least software to integrate the outputs of the turnkey's with the company system. Automation is available from everyone, but integration is very dificult to buy. The selection of an automation system will have a major impact on the ammount of work that must be done in the integration task.

LESSONS ABOUT MANAGING CAD/CAM

3) Traditional ratios of direct to indirect costs are obsolete. Financial people constantly compare this years performance to last year through a series of ratios. When the design function is automated the "bench " labor goes down faster than the overhead. Overhead functions that depend on a percentage of billable labor will have to be re-examined. In some cases the profit of the company is one of those items that are computed as a ratio to the basic labor. Our customers as well as our contracts people , who negotiate overhead rates and profit margins, are taking these things into account. The idea is to reduce the cost to the customer and to increase the profits of the company. These are not necessarily mutually exclusive things.

4) Middle managers need training not pressure. Middle managers are the people that make systems work. They are the people that are directly responsible for producing a good product, within budget and on schedule. They have seen a lot of systems come and go, and are naturally skeptical. CAD/CAM is an easy concept to sell to senior managers but there are a lot of snaggs that can keep a system from working up to its potential. When a middle manager wants a system to work these snaggs all have solutions, when the system is viewed with suspicion then these snaggs become major obstacles.

There is a tendency on the part of policy makers to give these folks 'heat' if they 'resist' progress. If the system is good it will stand up to their skepticism.

LESSONS ABOUT PEOPLE

5) CAD/CAM will not create engineers out of draftsmen. If a draftsman does not know analytical geometry when he sits down to the computer those functions are just not usable by him. You can't use F.E.M. modeling if you don't understand Finite Element Analysis. Managers have a tendency to think they can put the lower level people on these systems and up grade their whole function, where in reality the best people in the department should be on the system first. This is hard to do because the best people are never "between jobs". It always hurts to loose these people even for a week or two.

6) People need recognition. Recognition does not have to be money. As long some people are using the automatic tools and others are not, there will be a class distinction. So what do you do for a designer that is 4 times more productive that his counterpart on a drafting board. You can not pay him 4 times as much. But you can provide a better working environment, (carpet, piped music, bright colors etc) invent some perks if you have to. You can give them some attention in the company paper. Take tours through the area. All of these things constitute recognition. Many of these things won't be necessary after the "new" systems are established but during transition everything helps.

LESSONS ABOUT APPLICATIONS

7) Have clear responsibility assignments for data. When the product of a drafting room was a piece of paper you could always tell who was responsible for the 'vellum' it was the person holding it. Who's holding the vellum for an electronically released drawing? Each designer had a drawer of templates to assist with the drawing. Now there is a standard library of symbols, these symbols have attributes that convey information to a variety of functions that the designer is not even aware of. Who is responsible for the accuracy and completeness of the symbols? We have had to modify the Policies and Proceedures Manual, the Configuration Managment Manual and the Drafting Room Manual to clarify the responsibilities. We have also created a new type of job (which we are having a hard time finding a proper name for) that is responsible for libraries of standard symbols, and data elements.

8) Training never ends. Hardware and software are being introduce at a rapid pace, and users are finding new ways to use the systems. This kind of activity requires a constant and formal program for training. A combination of class work and seminars, round tables, news letters are required to keep the training levels high.

TECHNICAL PROBLEMS

We are all faced by the same technical problems. The industry is finding solutions but not fast enough.

1) Communications. 9600 baud is not fast enough when you want to ship a FEM model that has 100,000 vectors between computers. This has a solution if you work close to the computer, but if you work 10 miles away, and at a different building every year, the problem is severe.

2) Data compatibility between systems that are written by different vendors and run on different computers is a major problem. National standards like IGES are making a difference, but a lot more work is required particularly in the electronic data field.

3) Hardware compatibility. This is a serious problem as new generations of machines are integrated into the systems.

4) The rapid obsolescence of hardware is faster than the IRS allows depreciation. The problem is to select tools that will useful for 3 to 6 years without holding back the development of the system.

5) Another problem that is finally softening is the dependency of software on particular computers. The widespread use of FORTRAN 77, UNIX, etc is making software available to a variety of computers.

WHEN WILL THE SYSTEM BE COMPLETE ?

The single most asked question by financial managers. We at TRW are about 25% complete on the system that we have defined for our self. We can get the next 50% done in about 3 years. The last 25 % is not well defined at this time and depends on directions that the computer industry takes in the next 3 years.

DESIGN ASSURANCE TECHNIQUES CAD/CAM

CADDS and Industry Effectiveness

by

Richard A. Sharp

HUGHES

CADDS and Industry Effectiveness

HUGHES

- APPLICATION
- IMPLEMENTATION
- CONCLUSIONS

CURRENT CADDS EFFECTIVENESS COVERS THE FULL RANGE FROM GOOD TO BAD TO INDIFFERENT.

FOR THE PURPOSE OF THIS DISCUSSION, I WILL DEFINE "EFFECTIVENESS FACTOR" AS THE RATIO OF VALUES COMPARING CADDS AND ITS EQUIPMENT TO MANUAL EQUIVALENT.

EVEN THOUGH IT CAN BE UNLIMITED, LET'S KEEP THE RANGE BOUNDED BETWEEN ONE AND TEN.

HUGHES

Ratio of values

$$\frac{\text{CADDS and equipment cost}}{\text{Manual equivalent}}$$

Range

ONE TO TEN

EFFECTIVENESS FACTOR

Defined

THIS CHART DEMONSTRATES HOW CADDS MAY BE JUSTIFIED BY ANTICIPATING HIGH OUTPUT FOR UNIT COST. IT IS THE "BACKBONE" OF CADDS ACQUISITION.

IT IS ONLY NATURAL TO USE MANUAL COSTS AND OUTPUT AS THE COMPARISON LINE.

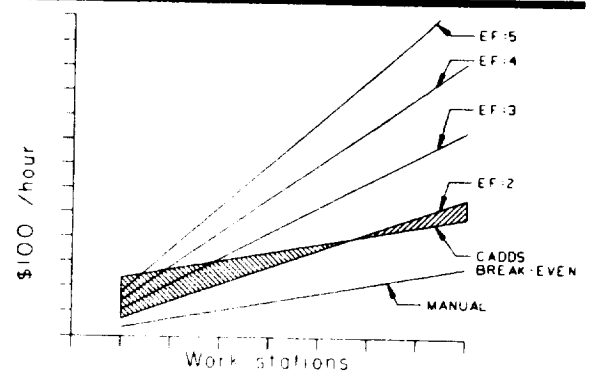
FOR A VALID COMPARISON, IT IS NECESSARY TO MAKE ADJUSTMENTS TO THE MANUAL LINE FOR BOTH THE CADDS SYSTEM AND OPERATIONAL COSTS.

THE JUSTIFICATION FOR ACQUIRING CADDS IS THE DRAMATIC INCREASE IN PROFIT BASED ON VARIOUS EFFECTIVENESS FACTORS.

THIS CHART IS TRUE, BUT IT IS NOT AUTOMATIC!

EFFECTIVENESS Anticipated

HUGHES



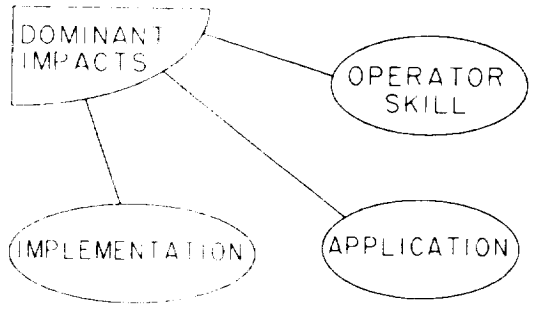
EFFECTIVENESS IS THE PRIME JUSTIFICATION FOR ACQUIRING CADDS!

IT IS MY EXPERIENCE THAT "APPLICATION" AND "IMPLEMENTATION" ARE THE TWO MOST CRITICAL FACTORS REGULATING "EFFECTIVENESS."

"OPERATOR SKILL" IS ALSO AN IMPORTANT FACTOR! SO LET'S ADDRESS THIS SUBJECT!

EFFECTIVENESS Justification

HUGHES



AS PERCEIVED BY SUPERVISION, "OPERATOR SKILL" IS ACTUALLY DEPENDENT ON "APPLICATION" AND "IMPLEMENTATION."

AN OPERATOR WHO IS WELL TRAINED ON CADDS DEVELOPS "OPERATOR EFFECTIVENESS" BY DOING REAL WORK!

IN REGARD TO APPLICATION --

HUGHES

PERCEIVED

Dependent on application and implementation

DOING WORK

Develops effectiveness

EFFECTIVENESS
Operator Skill

THERE ARE TWO PARTS TO APPLICATION:
- -THE PRODUCT
- -OPERATOR EXPERIENCE IN THE PRODUCT AREA

CADDS SYSTEM CAPABILITY NEEDS TO MATCH THE PRODUCT
- -TOO MUCH SYSTEM DRIVES UP OPERATING COST
- -TOO LITTLE SYSTEM AND THE JOB CAN'T BE DONE

IT IS CRITICAL TO HAVE "PRODUCT EXPERIENCED" LEAD OPERATORS. OTHERWISE, TIME WILL BE WASTED ON NON-PROBLEMS -- AND REAL PRODUCT PROBLEMS WILL NOT BE ADDRESSED.

"IMPLEMENTATION" CAN ALSO LIMIT "OPERATOR EFFECTIVENESS"
- -RESTRICTED TO MAKING RELEASE DRAWINGS COPIED FROM BOARD LAYOUTS. THE "EFFECTIVENESS FACTOR" CAN BE ONE OR LESS!

EFFECTIVENESS
Operator Skill

HUGHES

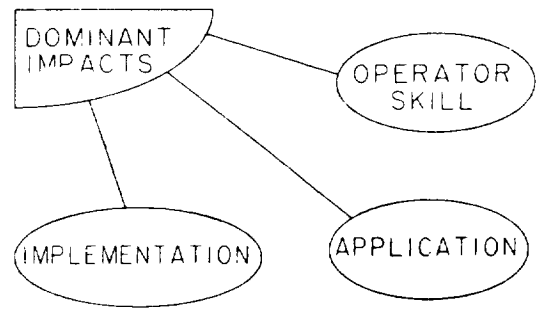
TWIN APPLICATION FACES

- Product used on
- Operator experience in the product area

NOW, LET'S DISCUSS CRITICAL GOVERNING FACTORS. LET'S BEGIN WITH "APPLICATION."

EFFECTIVENESS
Justification

HUGHES



DISCIPLINING CADDS FOR THE STRUCTURAL/MECHANICAL AREA OF THE AEROSPACE INDUSTRY HAS BEEN EVOLVING FOR QUITE A FEW YEARS.

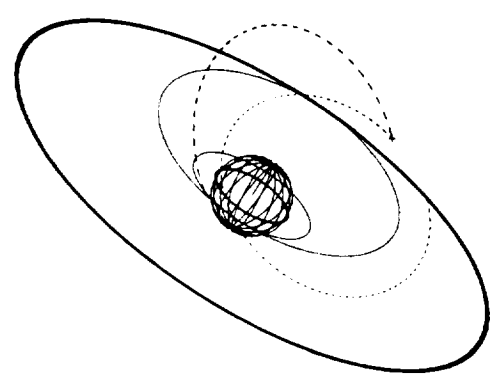
ALTHOUGH IT IS TWO DIMENSIONAL, THE PRINTED CIRCUIT AND INTEGRATED CHIP FIELD IS QUITE MATURE IN USING CADDS.

THE AUTOMOTIVE FIELD IS ALSO MATURING RAPIDLY.

ORBITAL MECHANICS IS A GOOD "TARGET" FOR CADDS APPLICATION.

APPLICATION
Orbital Mechanics

HUGHES

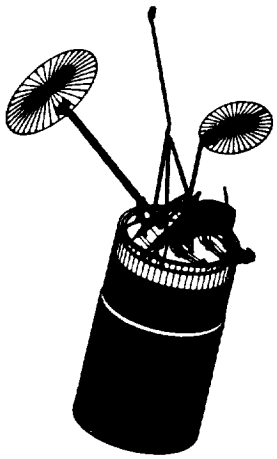


THE SPACE AND COMMUICATIONS GROUP AT HUGHES HAS BEEN IMPLEMENTING CADDS FOR THE COMPLEX AND COMPOUND GEOMETRY OF COMMUNICATIONS SPACECRAFT.

INTELSAT VI, SHOWN HERE IN ITS DEPLOYED CONFIGURATION, WILL BE USED FOR THE EXAMPLES SHOWN IN THIS PRESENTATION.

ONCE IT IS IN GEOSYNCHRONOUS ORBIT, THIS SPACECRAFT IS DESIGNED TO HANDLE 50,000 SIMULTANEOUS INTERNATIONAL PHONE CONVERSATIONS -- OR 4 TELEVISION CHANNELS AND 33,000 PHONE CONVERSATIONS.

IT HAS GLOBAL ANTENNAS FOR SATELLITE RELAY AND TWO MANEUVERING SPOT ANTENNAS FOR SELECTABLE COVERAGE.



HUGHES

APPLICATION

Communication Spacecraft

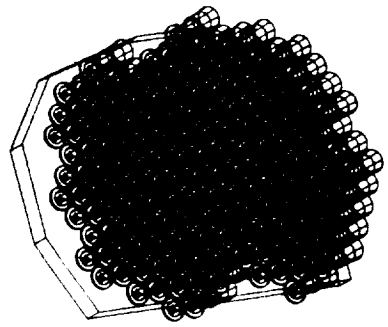
THE ANTENNA FEED SYSTEM IS A SELECTED MAJOR COMPONENT. IT IS RESPONSIBLE FOR MAKING A LARGE NUMBER OF CALLS POSSIBLE.

THIS IS THE TRANSMIT ANTENNA FEED SYSTEM WHICH IS ABOUT 4 BY 5 FEET AND CONTAINS 146 ANTENNA HORNS.

THE RECEIVE ANTENNA FEED SYSTEM IS SIMILAR, BUT ONLY 2/3 THE SIZE. IT ALSO CONTAINS 146 ANTENNA HORNS.

BEHIND THE HORNS IS THE MICROWAVE TRANSMIT NETWORK WHICH IS A VERY PRODUCTIVE AREA FOR 2 DIMENSIONAL CADDS -- BOTH IN COMPLEX AND COMPOUND GEOMETRY.

HUGHES



APPLICATION

Antenna Feed System

THIS IS THE OUTLINE OF ONE LAYER OF THE MICROWAVE TRANSMISSION NETWORK SETTING ON THE BASE PLATE. DUE TO THE LARGE DATA BASE SIZE, THE NETWORK IS SPLIT INTO TWO PARTS.

THERE ARE A NUMBER OF NETWORK LAYERS THAT ARE "ONE ON TOP OF THE OTHER." AGAIN, THE RECEIVE NETWORK IS SIMILAR.

IF THE WAVE GUIDES WERE USED IN LEIU OF THIS TYPE OF DESIGN, THE ENTIRE SPACECRAFT VOLUME MIGHT BE CONSUMED BY ONLY WAVE GUIDES.

LET'S DEFINE COMPLEX GEOMETRY AS THAT WHICH REQUIRES SIMULTANEOUS SOLUTION OF EQUATIONS
-OR-
MORE THAN ONE PARAMETER IS A DEPENDENT FUNCTION.

LET'S DEFINE COMPOUND GEOMETRY AS A LOT OF LINES OR CIRCLES, ETC.

HUGHES

COMPLEX

Has more than one parameter as dependent function

COMPOUND

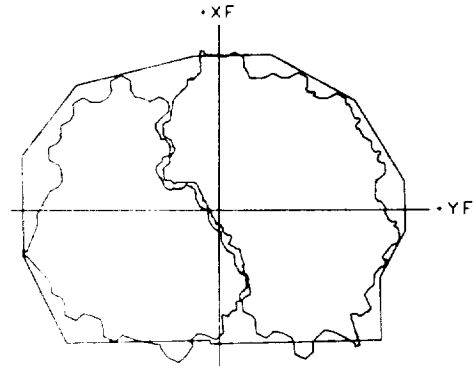
Consists of a multitude of lines

APPLICATION

Geometry Defined

APPLICATION
Microwave Transmitting Network

HUGHES



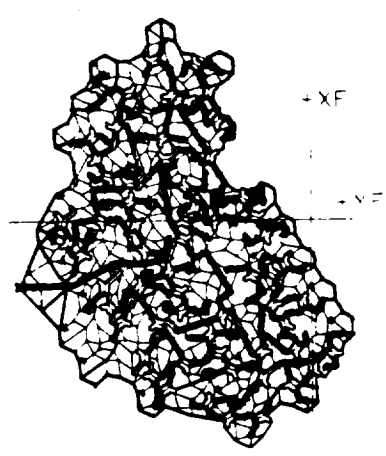
WE'RE NOW LOOKING AT ONE PART OF THE NETWORK ... AND JUST ITS OUTLINE...WITH THE NETWORK CENTERLINE SHOWING.

DESIGN CRITERIA MEANS THAT EACH CENTERLINE LENGTH MUST MATCH A SPECIFIC ELECTRICAL LENGTH. IT MUST ALSO CONNECT REQUIRED POINTS. BUT, ONE MUST NOT INTERFERE WITH THE OTHER -- OR WITH "STAY OUT" AREAS.

AUTOMATED SUB-ROUTINES WERE CREATED TO ADJUST EACH CENTERLINE LENGTH TO A "REQUIRED" LENGTH BY CALCULATING NEW LINE ANGLES...AND SEGMENT LENGTHS ...WITHOUT CHANGING END POINTS.

WITH THE ABILITY TO QUICKLY REVISE THE DATA BASE, THE PRODUCTION BEGAN PRIOR TO THE COMPLETION OF THE ELECTRICAL DESIGN.

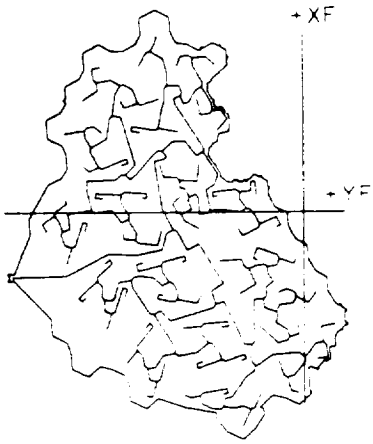
BASED ON SIMPLER NETWORKS THAT ARE REQUIRED TO WORK MANUALLY, THIS NETWORK WAS CONSIDERED BEYOND FEASIBILITY FOR MANUAL METHODS DUE TO COST AND SCHEDULE.



HUGHES

APPLICATION

Microwave Compound Geometry



HUGHES

APPLICATION

Microwave Complex Geometry

A LOCAL CLOSE-UP OF THE SAME PART WILL SHOW THE REQUIRED MACHINING DETAIL.

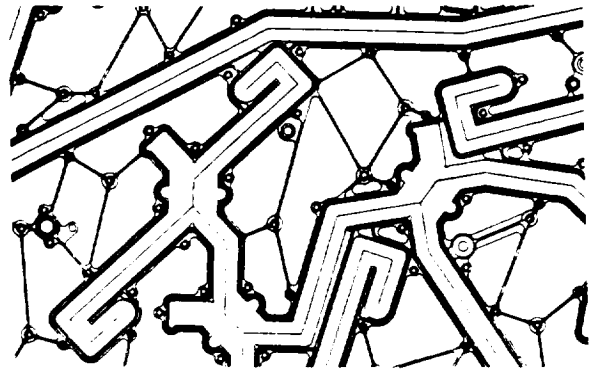
THE ELECTRONIC DATA BASE WAS USED DIRECTLY FOR NUMERICAL CONTROL CUTTER PATH PROGRAMMING, AND THE PARTS ARE ALL NC.

SPEAKING OF ELECTRONIC DATA BASE, WE DO CONVERT FROM CV CADDS 3 TO CV CADDS 4 TO GERBER OR TO NCAD OR CADAM.

WE ALSO DO SOME REVERSE FLOW.

APPLICATION
Data Base for Numerical Control

HUGHES



THIS IS THE SAME PART -- WITH THE MACHINING DETAIL ADDED.

COMPOUND GEOMETRY -- YES! THIS IS ONE OF MANY PARTS REQUIRED FOR A COMPLETE MICROWAVE TRANSMIT NETWORK.

AUTOMATED SUB-ROUTINES WERE ALSO CREATED FOR INSERTING MUCH OF THE MACHINING DETAIL.

THE MANUAL EQUIVALENT REQUIRED FOR THIS JOB WOULD ALSO BE UNFEASIBLE. TRY TO IMAGINE YOUR HAVING TO MANUALLY DEMENSION THIS ONE DRAWING! AND CONSIDER THE SCOPE OF WORK REQUIRED TO CONVERT A DIMENSIONED DRAWING INTO A FINISHED PART!

THE ELECTRONIC DATA BASE ENTITY COUNT OF THESE PARTS USUALLY RUNS ABOUT 30,000.

I HAVE HEARD THIS IS A DEVELOPING CAPABILITY AND IT SHOULD BE VERY COST EFFECTIVE.

THE PHYSICAL NC PART MUST AGAIN BE COMPARED TO THE INPUT INFORMATION AND ALL DISCREPANCIES MUST BE NOTED FOR EVALUATION.

NOW, LET'S LOOK AT 3 DIMENSIONAL APPLICATIONS.

TO PRESENT A NUMBER OF 3D CADDS APPLICATIONS, I WILL HIGHLIGHT A PRODUCT DEVELOPMENT CYCLE AND APPLY CADDS.

FIRST, THE CONCEPT IS ESTABLISHED
- PHASE 0 LAYS OUT WHAT IS WANTED
- PHASE 1 IS EXPECTED TO DO IT

APPLICATION Needed

HUGHES

We have CAD and
We have CAM
What about CAI ?
Computer Aided Inspection

APPLICATION Concept

HUGHES

PHASE 0

Performance criteria !

Performance boundaries !

PHASE 1

How can it do this ?

ITERATION IS THE MOST IMPORTANT CAPABILITY FOR PRODUCT CONFIGURATION.

ITERATION CAPABILITY BECOMES EVEN MORE IMPORTANT WHEN MANY DISCIPLINES HAVE A MAJOR IMPACT ON A PRODUCT.

IT IS WELL KNOWN THAT A REFINED DESIGN IS THE MOST VIABLE -- AND THAT IT IS DEPENDENT ON ITERATION!

THE MOST EFFECTIVE TIME TO EMPLOY CADDS IS AT THE BEGINNING OF A PROGRAM.

IT IS ALSO THE MOST CRITICAL TIME TO USE "PRODUCT EXPERIENCED" CADDS OPERATORS ON THE PROGRAM.

DIRECTIONS ESTABLISHED AT THE BEGINNING RUN THROUGHOUT THE LIFE OF A PROJECT.

HUGHES

Iteration process

Crosses disciplines

Refinement dependent on

ITERATION → STRONGEST COMPETITOR

APPLICATION

Configuration

APPLICATION Configuration

HUGHES

PHASE 1

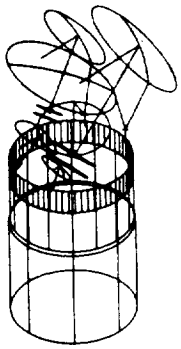
USE CADDS NOW

First stages of geometry layout

THIS IS AN EARLY INTELSAT VI CONFIGURATION ATTEMPT.

ONLY FUNDAMENTAL GEOMETRY IS WORKED WITH AT THIS STAGE.

A MANUAL LAYOUT OF ONE ITERATION, LIKE THE ONE SHOWN IN THIS EXAMPLE, CAN BE EFFECTIVE WHEN COMPARED TO CADDS. THIS IS TRUE EVEN WHEN WE WORK USING ORTHOGRAPHIC 3D.



HUGHES

APPLICATION

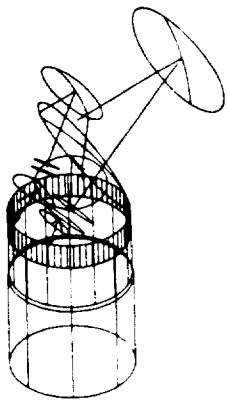
Iteration

Opt 1

THE SECOND MANUAL ITERATION MAY AS TIME CONSUMING AS THE FIRST.

HOWEVER, CADDS BECOMES QUITE FAST.

THE EXAMPLE SHOWN BELOW, IS ANOTHER CONFIGURATION ATTEMPT



HUGHES

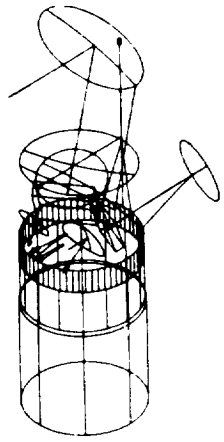
APPLICATION

Iteration

Opt 3

THEREAFTER, EACH ITERATION MAGNAFIES CADDS' EFFECTIVENESS -- WHEN IT IS USED AT THE BEGINNING OF A PROGRAM.

THE EXAMPLE SHOWN BELOW, IS ANOTHER ATTEMPT AT CONFIGURATION



HUGHES

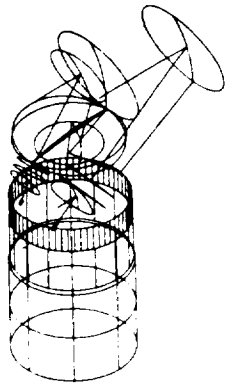
APPLICATION

Iteration

Opt 8

THE EXAMPLE SHOWN BELOW, IS YET ANOTHER ATTEMPT AT CONFIGURATION.

THE PROBLEM CONCERNS THE ABILITY TO STOW EVERYTHING.



HUGHES

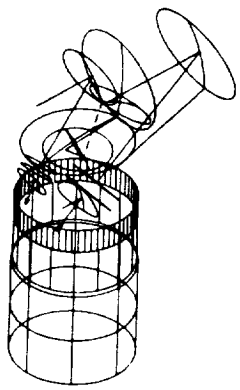
APPLICATION

Iteration

Opt 9

ATTEMPTING TO STOW THE ANTENNA FARM, USING NORMAL ORTHOGRAPHI TECHNIQUES AVAILABLE TO THE MANUAL WORLD, IS NOT ACCOMPLISHIN THE JOB.

THE EXAMPLE BELOW, IS ANOTHER ATTEMPT.



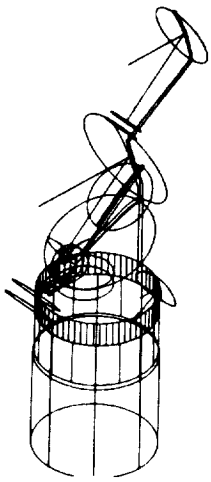
HUGHES

APPLICATION

Iteration
Opt 12

EVEN THOUGH WE TRY MANY DIFFERENT APPROACHES, IT IS STILL NOT WORKING.

THE EXAMPLE BELOW, IS ANOTHER EXAMPLE.



HUGHES

APPLICATION

Iteration
Opt 15

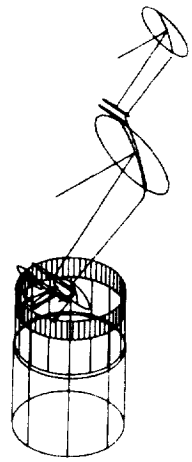
WHAT ABOUT USING A BIFOLD ARRANGEMENT HAVING SLEWED HINGE LINES?

MANUALLY, THIS WOULD BE QUITE A JOB! IN FACT, IT MIGHT REQUIRE THE DEVELOPMENT OF A MOCK-UP!

BY USING CADDS, THIS PROCESS BECAME RATHER SIMPLE -- ONCE THE TECHNIQUE WAS WORKED OUT.

MANY VARIATIONS OF BIFOLD/SLEWED HINGE LINES WERE GENERATED -- EACH REQUIRED ONLY A FEW HOURS.

NOW, LET'S TAKE A SHORT EXCURSION INTO GENERATING BIFOLD DEPLOYMENT USING SLEWED HINGE LINES.



HUGHES

APPLICATION

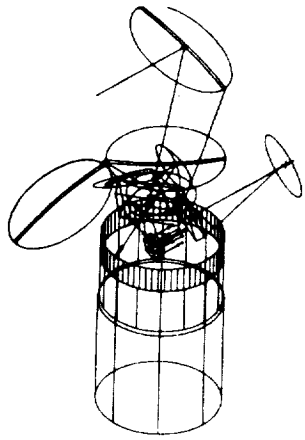
Iteration
Opt 18

NOW, FOR THE SECOND HALF OF THE PROBLEM: "THE UPPER HINGE"

FOUR CONSTRUCTION LINES ARE ADDED -- WHICH ALLOW TWO PLANE SURFACES TO BE DEFINED.

THEN, THE SECOND HINGE LINE IS AT THE INTERSECTION OF THE TWO PLANE SURFACES.

RESULTS OF THE BIFOLD SLEWED HINGE LINE ANTENNA ARE COMPLETE AND WORKABLE!



HUGHES

APPLICATION

Iteration
Opt 10

THE PROBLEM IS SET UP:

- -SELECT A DEPLOYED POSITION.
- -SELECT A TENTATIVE STOWED POSITION IGNORING CLOCKING.

CONSTRUCTION INFORMATION IS ADDED:

- -SELECT ANY UPPER AND LOWER HINGE POINT.
- -ADD BOOM C.L. BETWEEN POINTS AND SWEEP IN A SURFACE OF REVOLUTION.

SELECT A CLEARANCE DIAMETER -- AND PROJECT A "T" CYLINDER.

CUT THE SURFACES TO GENERATE AN INTERCEPT LINE OF POSSIBLE POINTS.

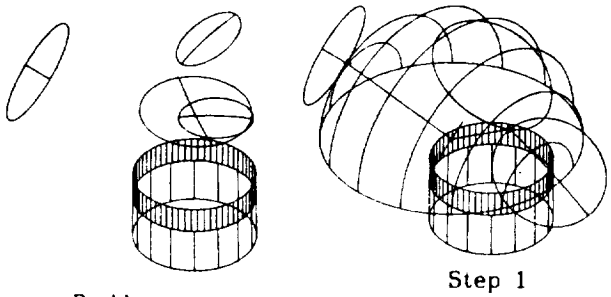
THE BIFOLD HINGE POINT, AT THE SEMI-DEPLOYED POSITION, CAN BE ANYWHERE ON THE INTERCEPT LINE.

SELECT ANY POINT ON THE INTERCEPT LINE -- AND THE LOWER HINGE LINE IS DEFINED BY VIRTUE OF THREE POINTS IN SPACE THAT DEFINE A PLANE.

THE LOWER HINGE LINE IS PERPENDICULAR TO THAT PLANE -- AND PASSES THROUGH THE SELECTED LOWER HINGE POINT.

APPLICATION
Descriptive Geometry-Free Space

HUGHES

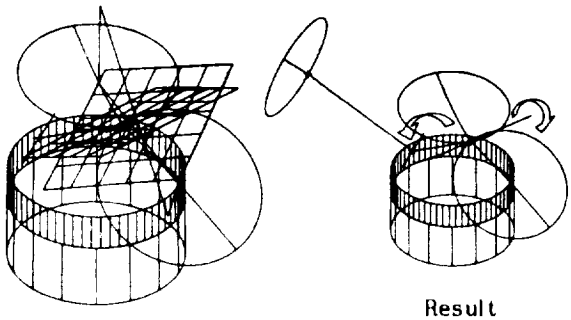


Problem

Step 1

APPLICATION
Descriptive Geometry-Free Space

HUGHES



Step 2

Result

CADDS IS MORE THAN JUST AN EXPENSIVE DRAFTING MACHINE! IT IS ALSO A FULL SCALE HIGH FIDELITY 3D MOCK-UP MACHINE.

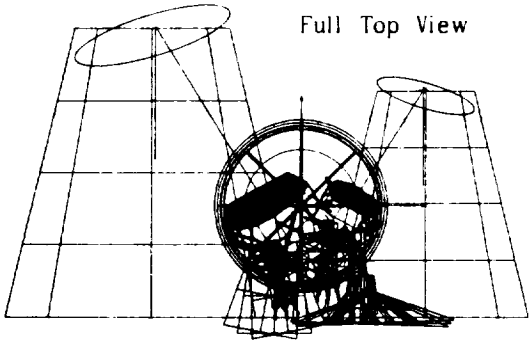
FOR EXAMPLE, THE PROGRAM DEVELOPMENT CYCLE IS CLOSING-IN ON A BASELINE DESIGN.

THE EXAMPLE BELOW IS A TOP VIEW OF THE DATA BASE.

APPLICATION
Mock-Up

HUGHES

Full Top View



INVISIBLE FEATURES SUCH AS A RADIO FREQUENCY CONE AND ANGLE PATTERNS CAN ALSO BE INCORPORATED INTO THE MOCK-UP.

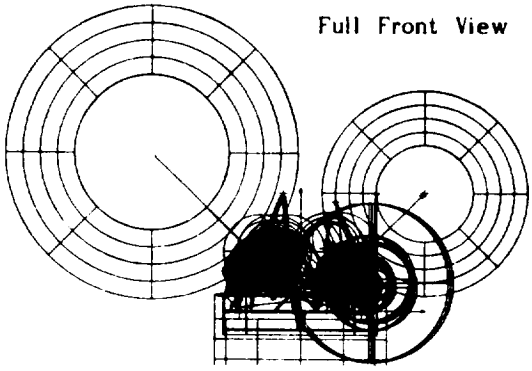
LOCATING THE SPOT BEAMS AND GLOBAL HORNS IS THE REMAINING MAJOR PROBLEM. IT IS A PROBLEM BECAUSE IT "HOLDS UP" THE PROCESS OF DESIGNING A BASELINE.

THE EXAMPLE BELOW, IS A FRONT VIEW OF THE DATA BASE.

APPLICATION
Mock-Up

HUGHES

Full Front View



THE PROBLEM WAS POSITIONING THE TWO SPOT BEAM ANTENNAS.

THE CRITERIA:

- -THEY MUST STOW.
- -THEY MUST DEPLOY.
- -THEY MUST HAVE REQUIRED "LOOK ANGLES" WITH ALL RF CONES CLEAR
- -THEY MUST HAVE NO STRUCTURAL INTERFERENCE THROUGHOUT THE DEPLOYMENT SEQUENCE.

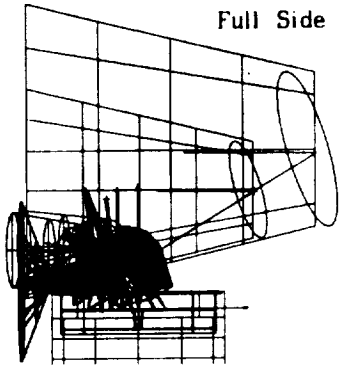
IT BECAME NECESSARY TO DEVELOP A DEPLOYMENT SEQUENCE THAT PARTIALLY DEPLOYED ONE, SO THE OTHER COULD DEPLOY -- AND THEN -- COMPLETE THE FIRST.

THE EXAMPLE BELOW IS A SIDE VIEW OF THE DATA BASE.

APPLICATION
Mock-Up

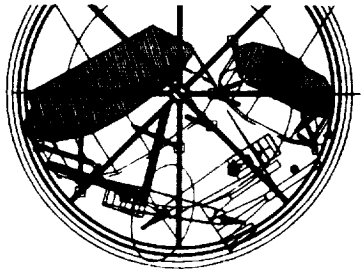
HUGHES

Full Side View



THE EXAMPLE BELOW IS THE SAME SOLUTION. HOWEVER, IT IS IN A STOWED POSITION.

HUGHES



SOLUTION

APPLICATION
Mock-Up

Top View
Stowed

THE EXAMPLE BELOW IS AN ISOMETRIC VIEW OF THE SOLUTION IN THE STOWED POSITION.

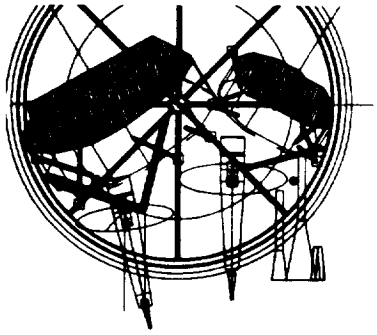
ALL OF THIS INFORMATION IS CONTAINED IN ONE MODEL'S DATA BASE. ANY NUMBER OF VIEWS -- AT ANY SCALE -- CAN BE SELECTED.

ANY COMBINATION OF LAYERS MAY BE VIEWED IN ANY NUMBER OF WAYS.

THE EXAMPLE BELOW, IS A TOP VIEW OF THE SOLUTION. IT IS DEPLOYED.

IT IS THE RESULT OF NUMEROUS ITERATION ATTEMPTS -- YET, IT WAS ACCOMPLISHED IN JUST A FEW "MAN" WEEKS.

HUGHES

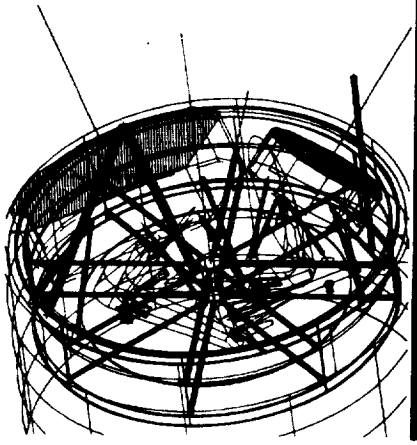


SOLUTION

APPLICATION
Mock-Up

Top View
Deployed

HUGHES

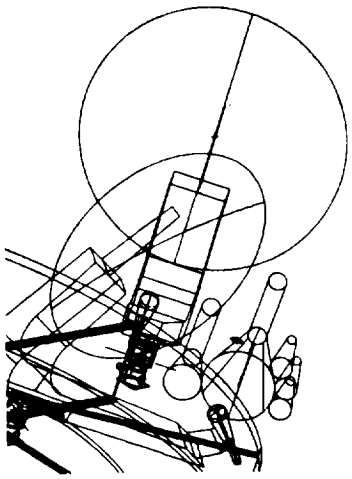


APPLICATION
Mock-Up

Isometric
View
Stowed

THE EXAMPLE BELOW IS A CLOSE-UP DETAIL CLEARANCE VIEW. IT SHOWS THAT GLOBAL HORNS ARE CLEAR OF THE EAST SPOT BEAM.

THIS IS TYPICAL OF NUMEROUS VIEWS TAKEN TO PROVE THE ARRANGEMENT IS WORKABLE.



HUGHES

APPLICATION
Mock-Up

Isometric
Stowed
Detail
View

BY "ZOOMING-IN" WE FIND THAT THE INDICATED INTERFERENCE IS CROSS HATCHED.

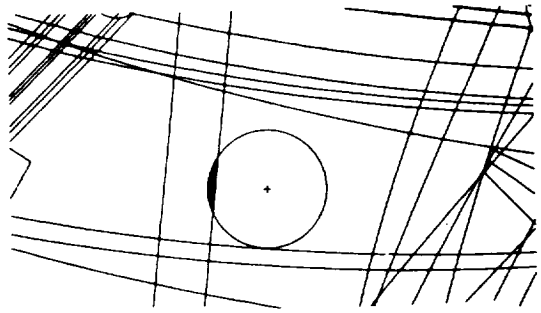
THIS IS TYPICAL OF THE NUMEROUS ITERATIONS REQUIRED PRIOR TO FINDING A SOLUTION.

ACTUALLY, THE LINE OF INTERFERENCE IS A CONSTRUCTION LINE -- AND NO TRUE PROBLEM EXISTS!

ONCE THE BASE LINE IS ESTABLISHED, THE OTHER TASKS BECOME IMPORTANT TO THE PROGRAM.

APPLICATION
Identify Problems

HUGHES



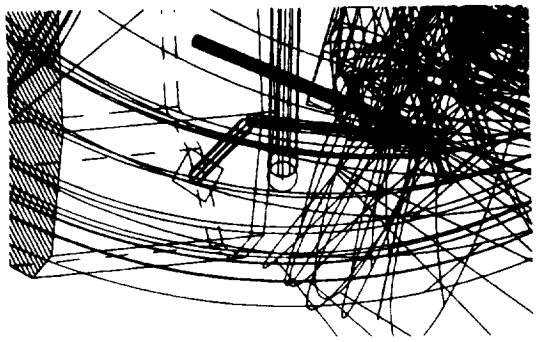
THE EXAMPLE BELOW IS ANOTHER CLOSE-UP OF THE DETAIL CLEARANCE VIEW. BUT A CLOSER LOOK WILL REVEAL THAT THERE APPEARS TO BE INTERFERENCE.

THIS IS A TRUE END VIEW OF ONE BIPOD TUBE.

TAKE A CLOSE LOOK!

APPLICATION
Identify Problems

HUGHES



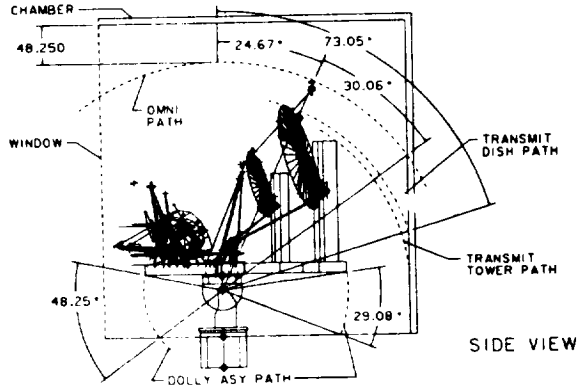
A MAJOR COST FACTOR OF A PROGRAM LIKE INTELSAT VI IS THE ABILITY TO USE EITHER EXISTING TEST OR MANUFACTURING FACILITIES -- OR TO IDENTIFY THE NEED FOR NEW ONES EARLY IN THE PROGRAM.

WITHIN ONLY A FEW DAYS, CADDS INSTALLED THE DEPLOYED BASELINE SPACECRAFT ON ITS TEST FIXTURES -- AND INTO AN EXISTING TEST CHAMBER.

THE EXAMPLE BELOW IS A "PLAN VIEW" OF THAT INSTALLATION.

APPLICATION
Impact Studies

HUGHES



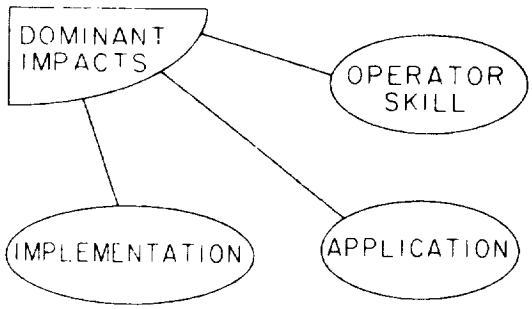
IMPLEMENTATION AND APPLICATION OVERLAP EACH OTHER

THERE IS A DISTINGUISHING DIFFERENCE
-IMPLEMENTATION REQUIRES ADVANCE COMMITMENT AND PLANNING TO ENABLE THE APPLICATION

I WILL NOW DISCUSS THE OTHER CRITICAL EFFECTIVENESS FACTOR: IMPLEMENTATION.

EFFECTIVENESS Justification

HUGHES



IMPLEMENTATION Comments

HUGHES

- Define
  - NEEDS
  - EXPECTATIONS

"The failure rate of interactive CAD/CAM systems is over 30% of installations. Much of this failure is due to insufficient planning and improper expectations."

NCGA 1983 Conference & Exposition brochure

IT IS NICE TO KNOW THAT THIS PROBLEM IS CLEARLY IDENTIFIED.

NOTE THAT THE STATED CAUSE OF FAILURE FALLS UNDER THE CATEGORY OF IMPLEMENTATION.

TO AVOID THIS FAILURE:
-DEFINE YOUR NEEDS AND EXPECTATIONS.
-SATISFY YOURSELF THAT YOUR DEFINITIONS FIT REALITY.

AN OPERATOR MUST DEVELOP A MENTAL COMMAND SEQUENCE APPROACH THAT WILL REQUIRE 5 TO 15 MINUTES TO EXECUTE.

INTERRUPTIONS DEFEAT THE CONCENTRATION THAT IS NEEDED TO BE EFFECTIVE.

OPERATORS HAVE BEEN PUSHING TO PROVE THEMSELVES -- AND THEIR SYSTEMS -- FOR SOME TIME. AS THEIR CAPABILITY IS ACCEPTED, EVEN GREATER DEMANDS ARE PLACED ON THEIR EFFECTIVENESS -- AND THEY TRY TO COMPLY!

A COMBINATION OF LONG-TERM "HEAVY EFFECTIVENESS" DEMANDS -- AND EXECESSIVE INTERRUPTIONS CAUSE OPERATORS TO LOSE THEIR CONCENTRATION -- AND THAT REDUCES THEIR EFFECTIVENESS!

THE RESULT IS FRUSTRATION AND BURN-OUT!

Concentration REQUIRED

HUGHES

If too much is expected of the operator -

AS THE NORM

Expect

BURN-OUT

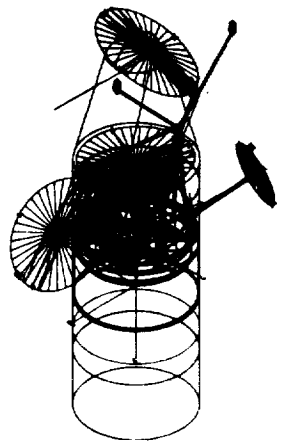
IMPLEMENTATION

Comment

THIS IS THE MASTER DATA BASE FOR THE INTELSAT VI ANTENNA FARM.

MASTER CONFIGURATION CONTROL IS A STRONG CADDS CAPABILITY. IF IT IS TO BE WORTH ITS EFFORT, THIS PROCESS REQUIRES ADVANCE PLANNING AND ANTICIPATION OF POSSIBLE DETAIL PROBLEMS. DETAIL PROCEDURE AND OPERATOR CONVENTIONS NEED TO BE ESTABLISHED.

A COMMITMENT TO MAINTAIN THE MASTER FILE IS ALSO REQUIRED.



HUGHES

IMPLEMENTATION

Master Configuration Control

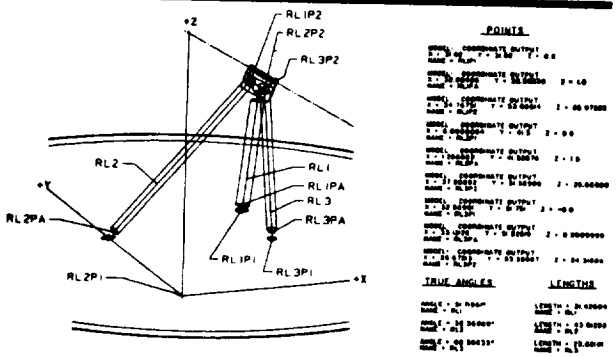
AMONG THE DISCIPLINES, DISTRIBUTING THE BASE LINE MASTER BECOMES A MAJOR TASK IN A PROGRAM LIKE INTELSAT VI.

A LITTLE ADVANCE PLANNING AND CADDS CAN PROVIDE VIRTUALLY ANY DESIRED INFORMATION.

SHOWN IS AN EXAMPLE OF ONE TYPE OF INFORMATION FORM. THIS ONE REQUIRES TAG NAME PLANNING.

IMPLEMENTATION
Coordination Information

HUGHES



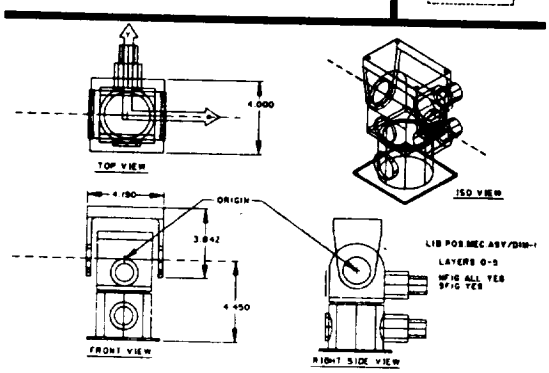
THIS IS A COMPONENT LIBRARY PART IN A VISIBLE FORMAT.

LIBRARY PARTS TAKE A LOT OF PLANNING -- INCLUDING NAMING AND LAYERING CONVENTIONS -- DISK MANAGEMENT AND TAPE BACK UP -- AXIS AND ORIGIN POSITION -- AND DRAWING INFORMATION.

THE EFFECTIVE BENEFIT FOR THE NEXT PROGRAM IS GREAT.
THE BENEFIT TO THE CURRENT PROGRAM IS ADEQUATE TO JUSTIFY THE TIME.

IMPLEMENTATION
Part Library-Visible

HUGHES



SUB-ASSEMBLY LIBRARY PARTS, LIKE THIS ONE, CAN BE VERY EFFECTIVE ON THE CURRENT PROGRAM.

INTELSAT VI TAKES TWO OF THESE. BUT THEY ARE BOTH DIFFERENT -- EXCEPT THE SUPPORT IS COMMON.

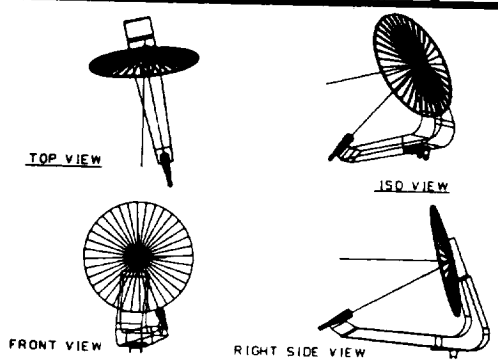
THESE SUB-ASSEMBLIES ARE SET-UP IN SUCH A WAY THAT ANY CHANGE TO THE COMMON STRUCTURE IS AUTOMATICALLY UPDATED IN BOTH SUB-ASSEMBLIES.

AT THIS POINT, REMEMBER THE MOCK-UP SLIDES.
THIS SUB-ASSEMBLY IS A SPOT BEAM ANTENNA AND WAS USED IN NUMEROUS POSITIONS.

ALL THOSE POSITIONS SHOWN IN THE FULL DATA BASE WERE ACCOMPLISHED IN ONLY A FEW MINUTES BY MANIPULATING ONLY ONE DATA BASE ENTITY -- THIS PART.

IMPLEMENTATION
Sub-Assembly

HUGHES



INTELSAT VI HAS A FAIRLY COMPLEX DEPLOYMENT SEQUENCE.

THE SMALL AMOUNT OF CADDS ANIMATION WE HAVE DONE LOOKS LIKE KINEMATIC EFFECTIVENESS WILL BE WELL WORTH THE EFFORT!

AT THIS POINT, SETTING IT UP SEEMS A BIT TEDIOUS. BUT COMPARED TO MANUAL KINEMATIC LAYOUTS -- IT'S A "BREEZE!"

LOOKING BACK, SOFTWARE DEVELOPMENT, OVER THE PAST 5 YEARS -- THIS CAPABILITY IS BOUND TO IMPROVE AS USE DEMANDS IT!

IMPLEMENTATION
Kinematics

HUGHES

VIDEO

Animation

STRESS ANALYSIS IS A "HIGH IMPACT" DISCIPLINE ON DESIGN -- FROM CONFIGURATION -- TO -- DETAIL DESIGN. A PART MUST BE DESIGNED FOR A LOAD SET.

GOOD DESIGN LOADS ARE NEEDED EARLY IN THE DESIGN CYCLE -- BUT USUALLY TRAIL BECAUSE LOADS ARE ALSO BASED ON THE DESIGN CONFIGURATION.

MAIN FRAME COMPUTER ANALYSIS OF LOADS SUCH AS NASTRAN ARE USED FOR THESE COMPUTATIONS. A FINITE ELEMENT MODEL OF THE CONFIGURATION IS FIRST REQUIRED -- AND ITS CREATION IS WHERE THE TIME LAG OCCURS.

CADDS FEM CAN PROVIDE THIS "UP FRONT" SOLUTION -- IF IT IS IMPLEMENTED.

IMPLEMENTATION
Finite Element Modeling

HUGHES

This topic worthy of a paper

- TIMELY DESIGN LOADS NEEDED
- CADDS FEM provides solution

CADDS OPENS THE DOOR TO NEW PRODUCTS -- PRODUCTS THAT ARE NOT OTHERWISE FEASABLE.

NEW PRODUCTS DO CONVERT TO NEW JOBS -- JOBS THAT WOULD OTHERWISE NOT EXIST!

BECAUSE CADDS CAN DO SO MUCH -- AT SUCH A RAPID PACE -- IT SHOULD NOT BE COMPARED TO MANUAL. YET, CADDS AND MANUAL CAN BE MUTUALLY SUPPORTIVE.

BY USING SOME OF CADDS EXISTING CAPABILITY, ITS EFFECTIVENESS FACTOR EXCEEDS A MEANINGFUL VALUE!

CONSIDER THE EFFECTIVENESS FACTOR AS A MULTIPLIER
-3 "TOP PRODUCT" PERSONNEL USING CADDS AT A FACTOR OF 5, ARE NOW 15 "TOP PRODUCT" PERSONNEL!

THE NEED TO REDUCE THE CADDS OPERATOR WORK FORCE IS REDUCED BY THE COMPENSATOR.

ONE CAN STILL BE EQUAL TO ONE.

OUR ONLY REAL WORRY SHOULD BE THE DRAGON OF OBSOLESCENCE!!

CONCLUSIONS

HUGHES

- Multiplied productivity

  1 = 5

- Stabilize personnel no. Peak and Valley

  1 CAN ALSO = 1

- The OBSOLETE will not tame

  THE DRAGON

I LIKE "TAME" DRAGONS -- AND ONLY TAME ONES!

SO, I WILL DO MY BEST TO PREVENT MY COMPANY FROM BEING .....NEXT!!!!

CONCLUSIONS

HUGHES

- NEW products possible using CADDS
- NEW jobs generated for all skills
- MAJOR effectiveness factor is

  BEYOND MANUAL COMPARISON

HUGHES

SOMETIMES THE DRAGON WINS

IMPROVED PRODUCTS THROUGH INTEGRATION OF DESIGN AND ANALYSIS: FINITE ELEMENT MODELING AS AN EXAMPLE

Henry Jacobson
MCAD/CAM Applications Development
Boeing Aerospace Company

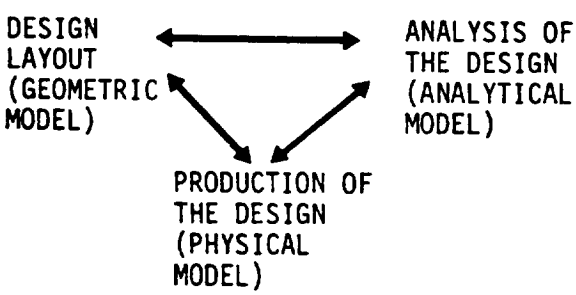
and

Philip H. Ness
Manager, MCAD/CAM Applications
Boeing Aerospace Company

OVERVIEW

The design and manufacturing process involves the continuous flow of product definition data through a series of phases. In each phase, actions are performed based on the product definition data. Final product quality depends on the accuracy of both the data and the actions.

Computer Aided Engineering/Computer Aided Manufacturing (CAD/CAM) seeks to improve the accuracy and timeliness of the information flow, in order to facilitate the correct performance of the many actions that are required to create a product. Within the total process, there are constant iterations between the following:



To support the iterations between these functional activities with computer systems requires the following capabilities:

- Interactive Computer Graphics
- Engineering Analysis Computing
- Automated Manufacturing
- Integrated Data Bases
- Networking

Application capability must exist to progressively capture data so that each step in the process builds to the next. The functions illustrated above require careful analysis of the levels of application integration and the required data sharing between disciplines and organizations.

Four levels of integration are identified:

1. Integration between engineering design and engineering analysis disciplines.

2. Integration between engineering design and production.

3. Integration of standard process specifications, material, tooling data, properties data for computer simulations/emulations of hardware.

4. Integration of computer based planning and control information with other three levels to provide configuration control from engineering release through fabrication, test and delivery.

The productivity and quality improvements obtainable with integrated computer systems are:

1. More accurate designs.

2. Reduction in the effort to create and maintain product definition.

3. Reduction in the effort required to gather data for analysis.

4. Earlier identification and elimination of design errors before they are propagated downstream to Manufacturing.

5. Reduction in manual bookkeeping.

6. Reduction in design and manufacturing flow times.

7. Leveraging of previous designs, reducing the required skill levels.

The remainder of this paper is confined to a description of the factors involved in achieving integration between engineering design and engineering analysis disciplines.

TRADITIONAL ENGINEERING PROCESS FLOW

In order to derive the improved quality and productivity benefits from the integration of engineering design and analysis, it is necessary to examine the factors in the current process that require improvement. Figure 1 illustrates the current design process from the point at which a design specification is complete. The engineering design functions are depicted by rectangles and the engineering analysis functions are depicted by hexagons in Figure 1. Design engineers prepare preliminary layouts; design analysts then review the geometry for size and shape compliance with the design specification. When the layout is complete, detail design proceeds; the detail design is then analyzed for structural integrity and validated against the design specification. Formal drawings are then released to Manufacturing. The use of computers to improve the upstream functions of conceptual design and configuration sizing is beyond the scope of this article.

The design process is serial and iterative. This is to be expected because a design has to be created before it can be analyzed; the analysis may then require modifications to the design. In medium to large scale engineering projects, the design and analysis functions are purposely separated to provide a "check and balance" system. The purpose of this "check and balance" system is to yield a more satisfactory product than is possible without this separation of functions. Unfortunately, in practice, several factors often prevent this division of labor from yielding the desired result of a more reliable, higher quality product.

In the traditional design process, time lags occur between layout or design completion and analysis completion. These time lags cause great difficulty in coordinating a particular design version and balancing the workloads and manpower between the design and analysis organizations. Even when computer systems are used locally in each of the design and analysis organizations, respectively, maintaining coordination of the current configuration between the design organizations and the analysis organizations is difficult at best.

As project costs and schedules impinge on this inefficient process, the situation becomes worse. Careful and thorough design analyses are often slighted, as is the incorporation of required design changes. The consequences are that errors and changes must be corrected, at much greater cost, in the downstream functions when the physical model, or the final physical product, is constructed. Such "retrofit" corrections downstream inevitably yield lower quality and less reliable products than products engineered properly in the first place.

DATA SHARING

The sharing of computer based data between the engineering functions and disciplines is clearly the most effective way to correct the deficiencies described in the traditional engineering process flow, above. Data continuity must be provided across the engineering

functions of preliminary design, detail design, and design analysis. Data continuity must also be provided across the engineering design analysis disciplines, such as structures, systems, weights, propulsion, etc.

The selection of computer system applications that provide data continuity between engineering functions and across engineering disciplines then becomes of paramount importance to obtain the productivity and quality benefits that can be obtained with computer data sharing. A good application is charactefized by its capability to integrate the design using selected parameters to create both geometry and analytic models from the geometry.

Specifically, the analytical model will be representative of the hardware and will be easily interfaced to the geometric model. The geometric model will be a complete 3-D representation that is usable for the fabrication of hardware. The benefits will be to greatly reduce the time and effort required to validate the design within each engineering discipline.

Integrated applications are also characterized by providing for product design iterations, product derivatives, and customer mandated changes, based on configuration controlled data models. Some examples of applications that meet the characteristic requirements are:

1. Finite Element Modeling.

2. Kinematics analysis of mechanical linkage systems.

3. Clearance and interference checking.

4. Cable, wire, tube routing.

5. Assembly and installation fit analysis.

The implementation of an integrated Finite Element Modeling and Analysis (FEMA) capability at the Boeing Aerospace Company will be used as an example to illustrate the specific factors involved in evaluating an integrated application.

FINITE ELEMENT MODELING AND ANALYSIS

Finite Element Modeling and Analysis (FEMA) has a wide scope of potential applications in the engineering disciplines. The method may be used to analyze a wide variety of problems in structural mechanics, thermodynamics, hydraulics, aerodynamics, etc. The method provides a convenient and effective way to share data between functions and disciplines to build analytical models. Figure 2 illustrates the integrated design process. System level models may be constructed by the analytical staff during the conceptual design phase to establish performance parameters. Designers may then refine the models to meet functional situations during the preliminary and detail design phases. Finally, the analytical staff may validate the design model against the design specification by checking critical areas of design. In the event that the analytical staff determines deficiencies in the design, they may build their own detailed models from the shared geometry database.

Finite Element Analysis is especially effective for designing better quality products in the following situations:

1. New designs, not typical of previous generations.

2. New material being considered.

3. Significantly complicated part shape.

4. Part size minimization.

5. Severe design loads.

6. Deformation that is crucial to function of the part.

7. Severe stress gradients expected in critical area(s).

8. Industry requirements; e.g., nuclear.

FINITE ELEMENT MODELING PROCEDURE

The procedure illustrated in Figure 3 will relieve the design engineers and analysts of the time-consuming laborious tasks that have been associated with building finite element models in the past. The capability to use geometry models resident on interactive graphic systems with automatic "mesh" generation software provides a tremendous productivity improvement potential in model construction. Model construction time can be reduced from weeks to hours. Model accuracy can be markedly improved with the immediate visualization provided by interactive computer graphics. The assembly of the complete input data is facilitated by the computer generation of accurate executive control data that is automatically merged with the model data. The Finite Element Analysis is performed on the model, and the analysis results of geometry displacements and stress contours may be output to graphic display stations and/or plotted on hardcopy.

INTEGRATING FINITE ELEMENT MODELING AND ANALYSIS

The current Finite Element Modeling systems lack adequate capabilities in several areas. User interfaces are not friendly; designers and analysts who are system operators must be expert not only in their disciplines, but also in the details, and often times workarounds of the interactive computer graphics system commands. In several Finite Element Modeling systems, the basic geometry database is not structured to be shared between Engineering and Manufacturing.

The current FEMA system setup and data flow is illustrated in Figure 4. The deficiencies are as described above. The data transfer of the model for analysis is accomplished by magnetic tape which is, at best, inconvenient. The executive control data must be input manually and merged with the model data after data transmission to the system where the analysis will be performed. The Finite Element Modeling commands operation on the interactive graphics systems requires the user to have in-depth understanding of the command's idiosyncrasies. Analysis results are output on the analysis system printers/plotters, which typically are geographically removed from the user graphics workstations.

The new FEM system, illustrated on the right side of Figure 4, corrects the deficiencies cited above. Users guides in the form of step-by-step "recipes" are developed that relieve the designer and/or analyst user from the requirement to have in-depth FEM commands operation knowledge.

Model data transmission is accomplished by directly coupling the interactive graphics system to the FEMA system via communication links. Executive control data is created and merged with the model data before transmission to the analysis system. Results from the analysis are transmitted via the communications link back to the graphics system. Displacements and deformed geometry are then displayed to the user on the interactive graphics system display terminal. The user may then modify the geometry, as required, at the same workstation on which both the geometry and the FEM model are created.

When the analysis indicates that the model meets performance requirements, the geometry will be transmitted to Manufacturing.

FEM EXAMPLES

Two models are presented. The first model is of a valve (Figure 5). The objective of the FEM analysis is to determine the ratio of the stress in the valve body to the valve nozzle. This ratio is a constant and is a function of the valve body diameter. The FEM model for the valve, without grid points (there are 870 grid points) is shown in Figure 6.

The second model is a strut fitting on a propellant delivery tank (Figures 7 and 8). The strut is subjected to 2000 lbs tensile force. The strut fitting FEM model is shown in Figure 9 and the "rod" end is shown in Figure 10. Note that the solid elements are used to model the fitting, while 2D quadrilateral elements are used for the rod end which has constant thickness. The geometry displacement results from the model analysis of the strut fitting are shown in Figure 11.

LEADS - an Integrated CAE/CAD/CAM/CAT System

Roy W. Roberts
Manager, Electronic Development Department

Presented at

1983 Conference on Mission Assurance

Hyatt House, Los Angeles
7, 8, and 9 June 1983

Lockheed Electronic Analysis and Design System—LEADS

The Lockheed Electronic Analysis and Design System (LEADS) is an integrated approach to computer-aided engineering (CAE), design (CAD), manufacture (CAM), and test (CAT). Although LEADS is not intended for use in manufacturing management, process planning, scheduling, or inventory control, it will interface with those systems as they are developed and put online.

Anticipated benefits from such a system are not only improvements in cost and schedule, but, most importantly, the reduction of errors that occur in manual systems.

The LEADS information flow begins with the project engineer, who defines the system in block-diagram form (see Fig. 1). Each block is further broken down into detailed diagrams of interconnecting information. This information, updated by the computer, will subsequently be used during system integration and checkout. With a top-down design (Fig. 2), LEADS aids the project engineer in delegating portions of the design activity to individual designers while maintaining an accurate and updated definition of the electrical interfaces between different parts of the overall system.

Fig. 1 Electrical Designer at CAE Terminal

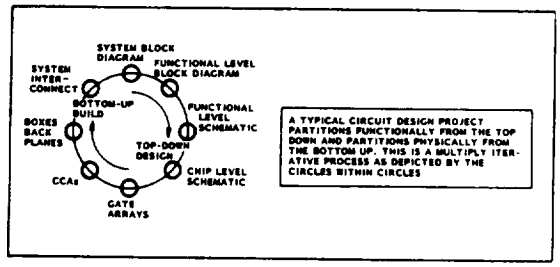


Fig. 2 Top-Down Functional Design—Bottom-Up Physical Design

The individual designer creates the detailed design by drawing the schematic directly on the computer (Fig. 3). The designer may choose at this point to implement a portion of his design with custom chips such as gate arrays (Fig. 4). As he proceeds, he can test the design (including gate arrays) by simulation on the LEADS computer. He can test for timing errors, both nominal and worst case (Fig. 5). When the electrical designer is satisfied the design will function correctly, the product designer

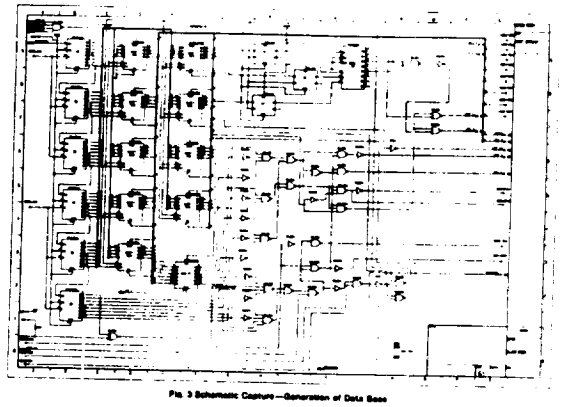


Fig. 3 Schematic Capture—Generation of Data Base

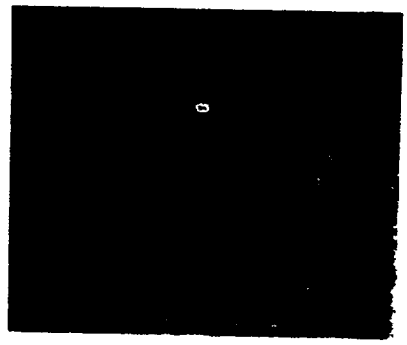


Fig. 4 Gate Array Schematic and Chip

PAGE 180 INTENTIONALLY BLANK

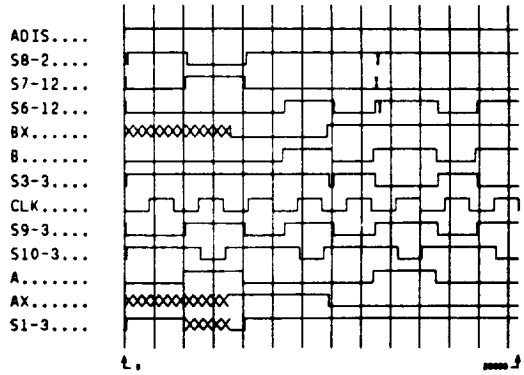


Fig. 5 Checking for Timing Errors

steps in and, operating on the same data base (and thus minimizing errors), proceeds with the design of the printed circuit boards and produces the manufacturing tooling in the form of photonegatives and NC drill tapes for manufacturing boards. The product designer also designs electrical interconnects in the form of wiring harnesses and backplanes (mother boards), again from the same data base. He produces manufacturing tooling in the form of magnetic tape for automatically wire-wrapping the backplane, for continuity testing of backplanes and harnesses, and for computer-controlled assembly aids (see Figs. 6, 7, and 8). LEADS will soon be able to add computer-aided tests of both bare printed circuit board and fully assembled printed circuit board, also from the same design data base.

Fig. 6 Wire-Wrapped Backplane Built from Data Base

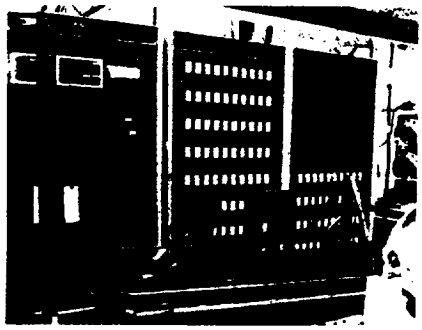


Fig. 7 Continuity Testing

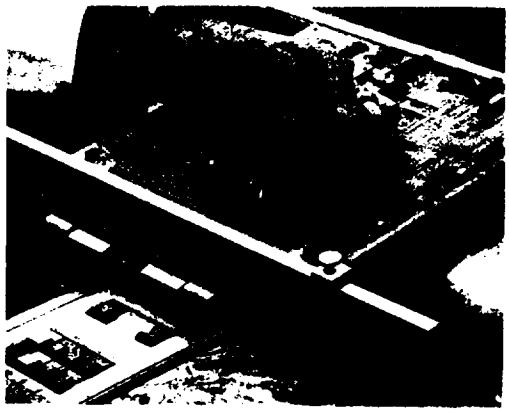


Fig. 8 Avionics Assembly Aid—Computer Controlled from Data Base

System Description

LEADS has grown out of a study plan initiated by the Space Systems Division of Lockheed Missiles & Space Company, Inc., Sunnyvale, California, in late 1981. The principal requirement of the system was that it be end-to-end.

The availability of turnkey hardware and software CAE and CAD systems in 1981 was limited and, after a survey of needs and requirements at the LMSC facility, it was determined to set up an interim system. There was already a large installation of

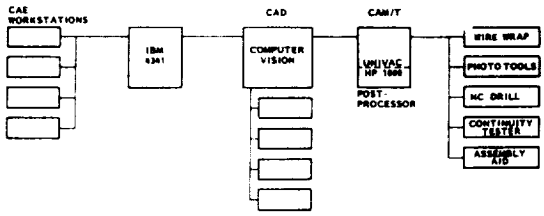


Fig. 9 Interim LEADS Installation

CADAM at LMSC, primarily used for mechanical design, and a large installation of Computer-Vision, primarily used for printed circuit board design and layout. Because of this, it was decided to use CV as the principal and vital link between CAE (engineering) and CAM/T (manufacturing and test). For the CAE function, a software package designed by Bell Northern Research and marketed by IBM was obtained on a trial basis. This software was installed on an existing IBM 4341 mainframe computer and operated on a time-sharing basis under the VM/CMS system. In this manner, an interim solution was developed at minimum investment cost. This system is shown in Fig. 9.

During the latter half of 1982 and early 1983, effort was made to establish links and procedures required for end-to-end operation. In December 1982, a wire-wrapped board was produced successfully which included schematic capture, simulation, timing analysis, layout and routing, and manufacturing reports and magnetic tapes.

Early 1983 efforts were directed toward writing a user manual. Such a manual will ease the training burden and will be used to obtain approval from various operating groups, which is very important because the system must have user support and credibility at the supervisory level as well as middle-level management.

Future Plans

The other major thrust in early 1983 was to plan for the next implementation phase. Key problems in CAE are to provide more workstations and more software capability. In CAM/T, the key problem is to define the engineering-manufacturing interface and to provide for growth.

Figure 10 shows the new CAE portion to be added in 1983. It consists of a dedicated VAX 11/780 with a 2-Gbyte Winchester disk, a SUN 150 Fileserver, six SUN 100 workstations, and associated peripherals such as a Versatec plotter, Imagen printer, and an 84-Mbyte Winchester. An important feature of the system long-term plan is that the VAX will have the Berkeley version of UNIX for an operating system. This versatility will provide three major software programs:

SCALDS - for circuit design
COMPACT - for microwave integrated circuits
CAESAR - for VLSI design

In conjunction with the latter two software packages, there will be three Tektronix 4041 terminals for microwave design and three AED 512 terminals for VLSI design, for a total of 12 new terminals on the VAX.

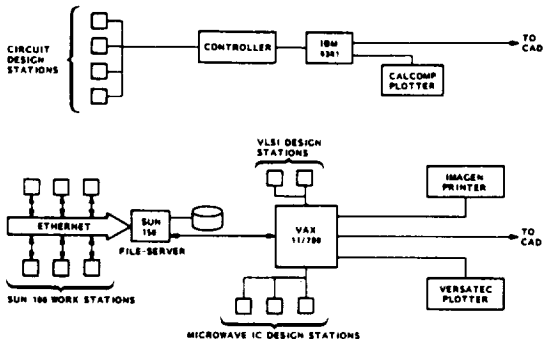


Fig. 10 Phase II LEADS Installation

Conclusion

LEADS should not be viewed as a single computer system, but rather as an integrated approach to engineering, manufacturing, and test. Its underlying concept is a single data base carried through the process. As LEADS grows, the additional advantages will result in improved productivity with fewer schedule surprises.

AN ELECTRONIC DESIGN MANAGEMENT SYSTEM

Ray McCann
Prime Computer, Incorporated

ABSTRACT

EDMS™, the Electronic Design Management System, is an advanced CAE/CAD/CAM system developed to aid engineers and engineering management in the design of electronic products. It provides design engineers with the tools needed to translate their ideas into hardware while giving management the tools to control and monitor the design process. EDMS has a consistent, interactive environment that can increase engineering productivity of capturing, analyzing, physically implementing, and controlling electronic designs. It has been proven in a production environment and runs on a computer which was developed using EDMS.

INTRODUCTION

With electronic products becoming more complex, and having increasingly shorter life cycles, the demand for engineering manpower is increasing faster than the supply. The goal of producing a reliable product on time and within budget is in danger of becoming unattainable. As a consequence, engineering management must stay well informed and in full control of the product development cycle, while the engineers must become more productive. The solution is an integrated CAE/CAD/CAM system that provides program managers with the necessary controls and the engineers with the tools needed to progress from design capture to manufacturing. One such system is Prime's EDMS (Electronic Design Management System).

SYSTEM PHILOSOPHY

The goal of EDMS is to provide engineering and its management with a totally integrated computer aided system that can support the entire electronic design cycle. Some of the key attributes of the system are:

- management controls
- easy-to-use, interactive command environment
- design analysis and error checking tools
- centralized library control
- networking
- integrated interfaces to key third-party software
- system expandability
- technology independence
- standardized documentation (especially for the DoD)
- revision/release control

Unlike other systems that support only one or two parts of the design cycle, EDMS, in an effort to decrease product development cost, has integrated in one system all the tools necessary to produce electronic products.

PAGE 184 INTENTIONALLY BLANK

EDMS SOFTWARE CONFIGURATION

EDMS consists of two modules, the Design Management System (DMS) and the Library Management System (LMS). Based on Prime's DBMS, DMS (see Figure 1) provides the interactive design environment, third party interfaces, and management controls necessary for electronic product development. LMS (see Figure 2) provides the interactive environment needed to define electronic components for use in the design. LMS includes the ability to automatically generate schematic symbols conforming to corporate or military standards directly from the component definition, produces documentation in data book format (see Figure 3), and provides the ability to add data for simulation, routing, and loading and power analysis.

MANAGEMENT CONTROLS

Many management controls are used throughout the entire design cycle, most of which are transparent to the user. When a new development is undertaken, a project leader is established. This project leader controls who has the privilege to access the design data and the level of access: DBA (data base administrator), edit, or read only. The DBA privilege allows the user to act as a project leader, i.e., to set or change project parameters, change the component libraries used by the project, lock or unlock a design, edit, and read the design data base. Read privilege, as the name implies, allows only viewing of the data base. Those not given a specific privilege have no access at all.

Controls are also provided for management of the component libraries. As mentioned, project leaders alone determine which libraries (DMS can search multiple libraries) their designers will be using, and they can change these according to the dictates of project requirements. For example, at the start of a project, using a prototype library may be suitable; but as the project nears the production stage, only libraries of fully qualified parts are appropriate. Since all DMS users access the same central files of library data, all users of a given library are guaranteed to be getting the same information. For those sites developing their own libraries, the LMS program provides management controls similar to those provided by DMS. The library administrators can control who has what kind of access (i.e., data base administrator, edit or read) for each library they administer.

Another management feature is revision control of the design. When a design is ready for release to production, DMS will set the revision level and lock the data so that it cannot be modified. If changes are required on the released design, DMS will copy the design data and automatically lable it with a new revision. Editing can then proceed on the new revision and the updated design can be rereleased at the new revision.

Designed into EDMS is a history of all changes and an audit trail of all design activity. Whenever a design is modified, a record is automatically stored recording a brief description of the change, who made the change, and the time and data. A user can easily find out what changes have been made to the design by using one of the query commands. To provide further control, the history of changes cannot be altered in any way. In addition, whenever DMS is invoked, a command log is opened. This command log will capture the entire DMS session and store

it in a format that can be loaded directly into DMS. This log can be very useful for automating engineering change orders or to recreate any work previously performed, and in fact forms a complete record of all work done on the schematic data base.

DESIGN CYCLE

To provide a better understanding of EDMS, this section discusses the stages of a typical design cycle using EDMS (see Figure 4).

Design Capture

DMS provides two modes of interactive design capture, graphical and non-graphical. Both methods allow the user to define, edit, and query the design data base. The graphics editor is a specialized interface designed to capture electronic design data on a schematic page. Using a menu of commands, the engineer can add components, symbols, text, interconnecting lines, signal names, and labels in a fast, consistent manner. Any group of these items on the schematic can be copied, moved, deleted, modified, mirrored, or rotated to give the designer maximum flexibility in an easy to use interactive environment. The graphics editor also features rubber banding, optional orthogonality, horizontal or vertical test, selectable text size and justification, solid or dashed lines, multiple grid sizes, and line management to produce high quality schematics. While in graphics, the user can query information which resides in either the design data base or the library. In addition, the graphical data can be saved and used to resume the process at a later date. By requiring that this data be verified before it can be stored in the data base, EDMS guards against drafting errors being stored in the data base. The possible errors that the verification process checks for include: four-way junctions, unnamed signal paths, conflicting signal names on a node, unnamed components, and multiple components with the same name. (Note that if the user chooses, component and signal names can be automatically generated.) If an error is detected, it will be highlighted on the graphics screen so that the engineer can pinpoint and correct it easily. This process ensures the integrity of the schematic before it is stored in the data base. Once all errors have been corrected, a command may be issued to store the current state of the schematic in the data base. (Note that in the event of a computer crash or program failure, DBMS guarantees the integrity of this data base.)

As stated earlier, design capture may also be performed nongraphically. All of the functionality available in the graphics editor is available from an alphanumeric terminal. Using the commands provided by the interactive environment, an engineer can completely design his circuit from a low cost alphanumeric terminal in his office (or home). Later, if the engineer chooses, he can enter the graphics editor and will be prompted to place the schematic symbols on a page until all the components have been added.

Design Analysis

After the design is stored in the data base, several design analysis functions are available to help verify the correctness of the design. Reports easily generated include net list errors, loading and fanout, and total current usage and power dissipation.

Once the connectivity errors identified on the net list error reports have been corrected, it is frequently desirable to simulate the logical behavior of the design. To facilitate the simulation task, EDMS provides interfaces to two very powerful simulators.

Generally, the first phase in design analysis is the check for net list errors. The net list errors detected include single pin nets, missing load and source pins, illegal output pins interconnected, open collector signals missing pullup resistors, and signals shorted to power or ground.

Loading analysis reports and loading error lists can be easily extracted from the design data base. The loading analysis report shows the complet loading information for a specific signal or for all signals. The loading error list shows all signals that contain too many loads. Total power dissipation and current can be calculated for either a page or for an entire schematic. A fanout report for each signal can also be obtained using one simple command. For each signal, the report itemizes the number of loads, the sources component type, and the load component types.

For simulation, EDMS interfaces to Prime's own hierarchical, interactive, mixed-mode simulator. EDMS also provides a complete interface to TEGAS™ (available from COMSAT General Integrated Systems), one of the most widely used simulators. To use the interface, the engineer simply types a command at his terminal and the topological data needed for a TEGAS simulation is automatically extracted from the design data base. The simulation can be run on either all or part of a design. Since TEGAS and EDMS run on the same CPU, the TEGAS simulation can be performed as a background task. Alternatively, the TEGAS TDL file may be transferred to another system to run the simulation.

Hardware Implementation

DMS supports three types of physical design: wrapped-wire, printed circuit board, or integrated circuit, either directly or through interfaces to third-party software. For prototype development, DMS has incorporated a wrapped-wire module with special processing to handle ECL logic. Having the ability to store assignment and placement information, DMS has incorporated commands to assign gates to packages and place the packages on the board. As each gate is assigned to a package, the corresponding annotation is added to the schematic diagram. For example, the pin numbers tht indicate the position of the gates within the package will be automatically added to the schematic. In addition, the name of the gate will be changed to reflect the package name. The wrapped-wire features include automatic generation of power and ground etch cuts and/or jumpers, minimization of the total wire length and automatic selection of termination resistors if the board contains ECL logic. During the debug mode, the engineer will likely change, delete, or add signals or gates. The wrapped-wire routine includes a rewire command that will update only those signals or gates requiring changes. Complete consistency with the schematic is maintained throughout the debug/correction process (i.e., any changes to the signals or gates will automatically be reflected in the schematic).

For printed circuit board (PCB) placement and routing, EDMS interfaces to two programs, SCICARDS™ (from Scientific Calculations, Inc.) and PRANCE™

(from Automated Systems, Inc.). All design data needed to drive these programs is easily exttracted using one simple command. The SCICARDS user need not maintain a SCICARDS component library, since EDMS will supply all of the component library information from the EDMS library. This alone will save time, money, and the frustration trying to synchronize two libraries.

At the completion of the placement and routing process, the data base of schematic pages can be back-annotated to reflect logic changes, gate and pin swaps, and assignment and placement changes. No manual intervention is needed; and, therefore, a consistent and reliable design data base is maintained.

EDMS provides interfaces to both gate array and standard cell routers. For gate array routing, MERLYN-G™ (provided by VR Information Systems) is used. With one caommand, all of the design data required by the MERLYN-G preprocessor is generated in proper format. MERLYN-G can then be executed on the same CPU as EDMS, or the input data could be transmitted to another system for processing.

The MP2D (Multi-Port Two-Dimensional) Placement and Routing program, available from the U.S. Army, will perform placement and routing of a standard cell LSI chip. Using DMS to capture the chip design permits easy extraction of the design data needed for input to the MP2D program. Commands to add row, order, and weight parameters to the design have been implemented. In keeping with the concept of a totally integrated system, MP2D will run on the same system as EDMS.

Drafting/Documentation

In the design cycle described above, it is apparent that the efforts of the drafting department can be minimized. Using a military standard symbol library means the schematics produced by DMS are of high quality and meet DoD specifications. Since the graphical information is automatically updated as changes are made to the design, iterative loops through the drafting department are not longer necessary.

EDMS supports a wide range of report and plotting capabilities. All reports may be listed to either an alphanumerical terminal, a file, or a printer. The reports include bills of material, net lists, gate equivalence counts, library data, and board geometry data. EDMS also supports user-supplied formats allowing certain reports to be tailored to meet specific company needs. Plots may be extracted from the data base, using either an alphanumeric or graphics terminal. Plotting schematics may be done on a per page basis or for several pages at once. Page cross references can be added automatically, if desired. Also available are assembly plots which have proven to be quite valuable. For example, if any parts are placed in a forbidden zone of the board or on top of another, viewing the plots will make this apparent.

EDMS IN PRODUCTION

At Prime, EDMS has been used in production to produce its next generation computers. In fact, with the announcement of Prime's newest computer, the 2250, it is now possible to run EDMS on a computer designed using EDMS.

As an EDMS Beta site, ITT/Avionics (see Figure 5 for ITT's system configuration) began at ground zero and reached their present production mode in about nine months. In the last six months they have released 24 printed circuit boards of about 100 ICs per board complexity and by the end of the year expect to release 35 more. There are presently 100 active designs in the system, three of which are gate arrays (4K-6K gates per chip) and twelve thick-film hybrids (which will expand to 40 by the end of the year).

At Hughes Aircraft, EDMS is being used to produce high density printed circuit boards (see Figure 6 for Hughes' system configuration). The PCB's design uses less than 1/2 square inch per DIP. For example, they typically fit 55 ICs on a single 5 x 5 board using two signal layers. Notice, in Figure 6, that EDMS is running on a Prime 850 while networked to a Prime 750 running SCICARDS, MERLYN, and other internal CAD software.

Auragen Systems Corp. in Fort Lee, New Jersey, uses EDMS to produce both wrapped wire and printed circuit boards. The boards have up to 320 ICs and over 2500 signal nets. Though a new user, they already have used EDMS to cut their product design cycle time from 18 to 12 months.

Ford Aerospace and Communications Co. has the most wide-spread system configuration (see Figure 7) of any of our customer sites. Using Prime's networking capability, Ford performs PCB routing on a Prime 550, corporate library maintenance and electronic design on a Prime 750 in Newport Beach, California, and design work on a second Prime 750 in Palo Alto, California. Ford also has two satellite design centers in Houston, Texas and Colorado Springs using CPUs in California.

Conclusions

EDMS provides a consistent, technology independent design environment to aid engineering and management in the production of reliable electronic products in a timely manner. It has improved the accuracy of design data by eliminating inconsistencies between logic diagrams and the design data base. Numerous complex products can be designed, analyzed, physically implemented, and controlled using one or a multiple of systems. EDMS's management controls, history records, revision controls, and central library make it more suitable for control of design data than a system of independent satellite processors, where each satellite has to manage its own library and project management is done manually across several satellites. Lastly, EDMS has a proven record in the development of electronic products, ranging from aerospace applications to the design of general-purpose computers.

Footnotes:

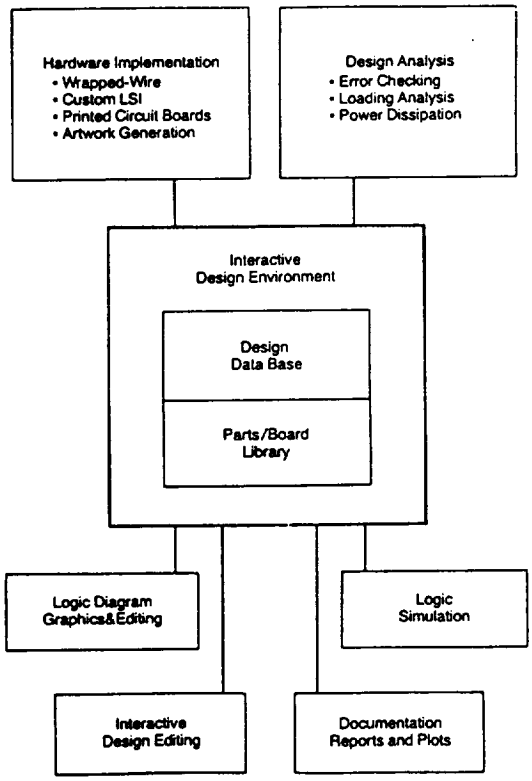
Electronic Design Management System and EDMS are trademarks of Prime Computer, Inc.

MERLYN-G is a trademark of VR Information Systems Corporation.

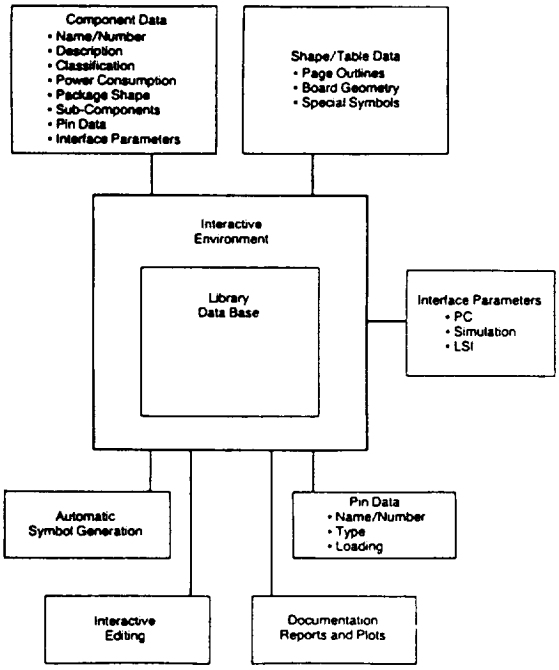
PRANCE is a trademark of Automated Systems, Inc.

SCICARDS is a trademark of Scientific Calcuations, Inc.

TEGAS-5 is a trademark of Comsat General Integrated Systems.



DMS Common Data Base
Figure 1



LMS Common Data Base
Figure 2

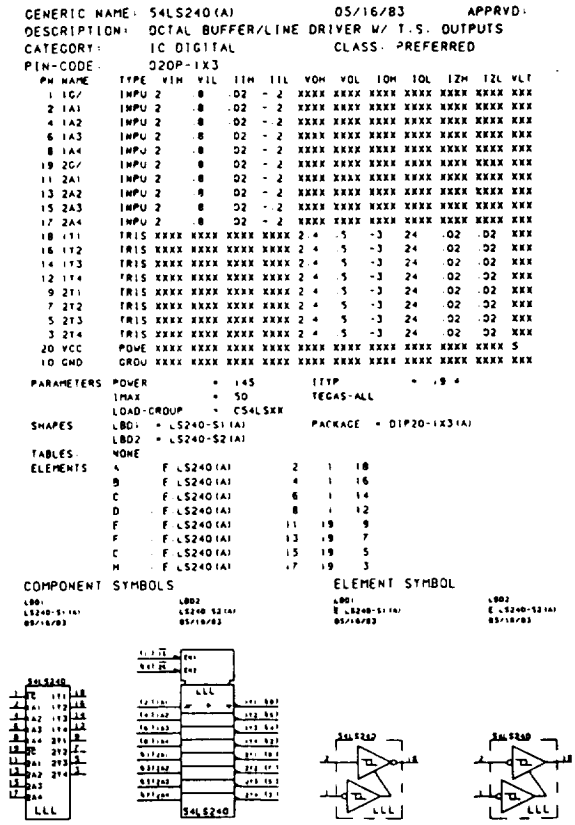


Figure 3
Component Documentation

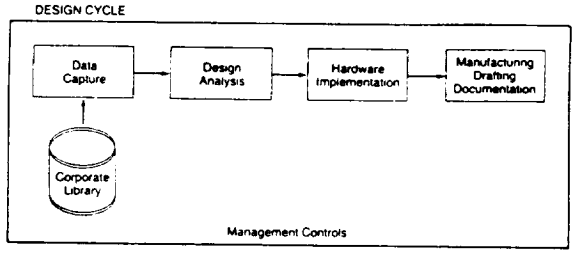


Figure 4

ITT/AVIONICS
ELECTRONICS DESIGN MANAGEMENT SYSTEM™ — EDMS™

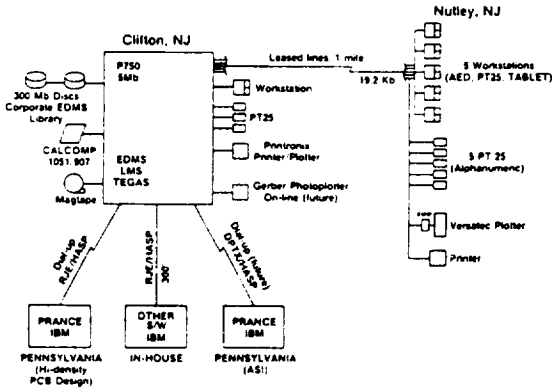


Figure 5

HUGHES AIRCRAFT
ELECTRONICS DESIGN MANAGEMENT SYSTEM™ — EDMS™

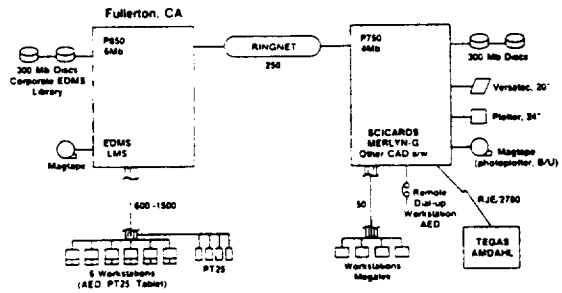


Figure 6

FORD AEROSPACE & COMMUNICATIONS CO.
ELECTRONICS DESIGN MANAGEMENT SYSTEM™ — EDMS™

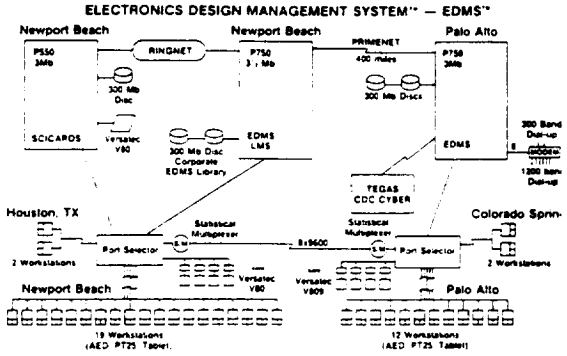


Figure 7

DESIGN ASSURANCE

Ken Hughey
Hughes Aircraft

DESIGN ASSURANCE

- FIVE WORKSHOP GROUPS
  - DESIGN REVIEW
  - LONG LIFE COMMSATS
  - AUTOMATED DESIGN AND ANALYSIS
  - DESIGN ASSURANCE FOR GROUND SEGMENTS
  - FAULT TOLERANT DESIGN AND TEST

DESIGN REVIEW WORKSHOP GROUP

- DESIGN REVIEWS ARE NECESSARY AND PRODUCTIVE
- REVIEWS DO NOT UNCOVER DESIGN DEFICIENCIES
- 95% OF PURPOSE IS ACCOMPLISHED IN PREPARATION
- TEST EQUIPMENT AND TEST SOFTWARE SHOULD BE INCLUDED
- SRR IS MOST IMPORTANT REVIEW
  - END WITH MINIMUM SET OF REQUIREMENTS
  - WEED OUT REQUIREMENTS WHICH
    - HAVE UNKNOWN ORIGINS
    - ARE NICE BUT NOT NECESSARY
    - HAVE MYTHOLOGICAL ORIGINS
    - ARE WISH LIST ITEMS
    - CANNOT BE MET

LONG LIFE COMMSATS WORKING GROUP

COMPARISON BETWEEN GOVERNMENT AND COMMERCIAL
- SIMILARITIES
  - DESIGN, TEST AND ANALYSIS DETAIL
  - PARTS AND SYSTEMS EFFECTIVENESS REQUIREMENTS
  - MANAGEMENT SYSTEM AND PROGRAM CONTROLS
- DIFFERENCES
  - GOVERNMENT ALLOWS LESS LATITUDE FOR CONTRACTOR DECISIONS
  - GOVERNMENT REQUIRES GREATER DOCUMENTATION DETAIL PLUS VOLUME
  - GOVERNMENT CUSTOMER INVOLVEMENT IS MUCH MORE INTENSE
  - COMMERCIAL PROGRAMS ARE MORE HEAVILY INCENTIVIZED

DATA SHARING ON MAJOR SYSTEM COMPONENTS NEEDED
- TWTAS
- COMPUTERS

GROUND SEGMENT WORKSHOP GROUP

- IMPORTANCE OF EARLY DEFINITION OF I/F'S NOT STRESSED
  - ADDRESS AT SRRs, PDRs, CDRs
  - DEFINE NEW REQUIREMENTS SPEC CALLED I/F SPEC
  - ADD I/F SPEC TO MIL-STDS
- SIMULATORS ARE FREQUENTLY OMITTED DUE TO COST, NOT NEED
  - IMPORTANT TO I/T, AND TRAINING
  - USE DURING OPS TO RETAIN PROFICIENCY
- MAN-MACHINE I/F FREQUENTLY IGNORES HUMAN FACTORS
  - MAN IS HYPNOTIZED, NOT ALERT WHEN ANOMALIES OCCUR
  - MAN IS OFTEN TOO SLOW IN RESPONSE TIME
- ACQUISITION PROCESS
  - EMPHASIZE DESIGN-TO-BUDGET, NOT LOWEST BID
  - USE INCENTIVES FOR P^3I (PRE-PLANNED PRODUCT IMPROVEMENT)
- INDUSTRY ENGINEERS
  - NEED SYSTEMS ENGINEER (KNOW HARDWARE AND SOFTWARE)
  - SHORTAGE - MAKE MORE PRODUCTIVE WITH AUTOMATED TOOLS AND HIGHER PAY

AUTOMATED DESIGN AND ANALYSIS WORKSHOP GROUP

- FORMAT
  - INFORMATION EXCHANGE
  - PAPERS COVERED MOST APPLICATIONS OF DESIGN, MANUFACTURING AND MANAGEMENT
- CONCLUSIONS
  - DESIGN RELIABILITY INCREASES WHILE DESIGN TIME DECREASES
  - MOCK-UP ROLE CHANGE
    - SIMULATION MINIMIZES NEED FOR ELECTRICAL BREADBOARDS AND MECHANICAL MODELS
  - AUTOMATED REAL TIME DATA AND FAILURE REPORTING IMPROVE PRODUCT ASSURANCE

FAULT TOLERANT DESIGN AND TEST WORKSHOP GROUP

- PERFORMANCE AND RELIABILITY PARAMETERS NEED TO BE RELATED MORE CLOSELY
- DESIGN FAILURES AS THEY RELATE TO FAULT TOLERANCE ARE NOT WELL UNDERSTOOD
- LACK OF COMMON LANGUAGE DEFINING FAULT TOLERANT DESIGN, TEST AND RELIABILITY PREDICTIONS
- NEED TO IMPROVE UNDERSTANDING OF "SINGLE POINT" FAILURE IMPLICATION IN FAULT-TOLERANT SYSTEMS

TEST FAILURE TREND MANAGEMENT

an approach for identifying and preventing recurring flight equipment failures and potential reliability risks

Arthur W. Ackerman, JR.
Ford Aerospace and Communications Corporation

INTRODUCTION

To minimize on-orbit failures or potential reliability risks in complex space programs, it is critical to identify failure trends (repetitive failures), resolve them and prevent their recurrence during ground testing. This task is difficult due to the complexity and diversity of manufacturing and test operations for satellites and the complexity of satellites themselves. In facilities where concurrent spacecraft programs are in progress, failure communication must be timely, and cross program lines. This is especially true when similar or identical equipment is involved. Ultimately, the unforgiving reality of a 22,500 mile orbit dictates that all solutions to problems are in place long before final count down begins. Presented here is one approach which has been developed to deal with this task.

PREREQUISITES

The cornerstone of Test Failure Trend Management is an aggressive, closed loop failure reporting system. It is frivolous to assume potential test failure trends can be identified and corrected if all of the pieces of the puzzle are not available for analysis. To accomplish such closed loop failure reporting, a detailed system is required wherein test failure reporting occurs religiously in every prescribed situation, and test failure reports are reviewed in a timely manner.

At Ford Aerospace, on all space programs, formal failure reporting begins at the first powered test of a module, the lowest level of powered circuit testing, and continues through all levels of testing. The Quality Assurance Organization is responsible for the preparation and control of Test Failure Reports (TFRs) and associated Material Review Board (MRB) actions. Reliability Assurance has Failure Review Board (FRB) responsibility. In support of the FRB function, TFRs initiated by the MRB are continuously forwarded to a central Reliability Assurance group which reviews the MRB actions both while ongoing and as they are completed.

Each TFR contains as a minimum four essential elements: problem specific information and a three-part narrative section consisting of a detailed description or statement of the test problem, the cause of the problem, and the corrective action taken.

- Problem specific information is identifying data. It includes such data items as the part number/serial number of the item under test, the failure date, the test environment, the level of testing, related TFR numbers, failure analysis report numbers, and the part and serial numbers of parts or equipment which have failed or are suspect.

PAGE 194 INTENTIONALLY BLANK

Included in the problem specific information are two failure cause codes which will subsequently be discussed in detail.

- The statement of the test problem includes the identification of the applicable test procedure and a complete description of all nonconformances noted during the conduct of the test.

- The cause of the test problem parallels the statement of the problem and provides a specific cause for each nonconformance noted.

- The corrective action portion of the TFR details the corrective action for each of the problems identified and provides the ultimate disposition of the item under test and any further actions required concerning parts or equipment which have been removed and replaced. If equipment is spun off on a separate TFR for further evaluation or rework, that action and the appropriate TFR number(s) are noted. If failure analysis of parts is required, such direction and the Failure Analysis Report number are also noted.

FAILURE REVIEW AND ANALYSIS

The Reliability/Failure Review Board function associated with powered test failures is twofold. First, Reliability enters the TFR data in a computerized Failure Summary Report (FSR) in the four-element format identified above. Each TFR reported is segregated by space program, assigned a Failure Review Board problem number, and summarized as part of a discrete space program Failure Summary Report. Failure Review Boards are then convened monthly to review completed and ongoing MRB actions. At Ford Aerospace, the Failure Review Board is chaired by the Reliability Organization and consists of members from Quality Assurance, Engineering, Manufacturing, Parts Engineering, the Customer, and the applicable Program Office. Designated manufacturing, design and test engineers also participate on an as-required basis. On a problem-by-problem basis, the function of the Failure Review Board is to ensure that the individual test failure is fully described, that failure causes have been identified in detail, and that corrective action is adequate to correct the failure and prevent its recurrence on the same or similar equipment. The Failure Review Board routinely conducts additional inquiries and assigns action items in situations when further information is required to resolve all issues prior to closure of a Failure Review Board problem.

The second Reliability/Failure Review Board function is the identification of potential failure trends. The Failure Review Board Failure Summary Reports provide the data base which makes this function possible. To put this task in perspective, on several active space programs involving many satellites, it is not unusual to maintain a data base of thousands of Failure Review Board Problems. Considering problem specific information only, this results in a significant number of data entries relevant to powered test failures. The narrative portions of Failure Review Board Problems in such a data base, with supporting documents and reports, comprise tens of thousands of pages of highly technical and somewhat cryptic reading material. As a result of the volume of information available, the management of data becomes paramount. As effective failure trend analysis requires the availability of every key element of data in a retrievable condition for review, this voluminous data must be manipulated to separate essential information required to identify failure trends.

MANAGING TEST FAILURE DATA TO ACCOMPLISH FAILURE TREND ANALYSIS

The problem specific information associated with each reported test failure is in most cases a listing of hard data elements which are not subject to interpretation. The part numbers, dates, report numbers, environments, tests, and equipment levels are fixed factors which only require accurate description. As such, these data items are ideal for computer application and permit unlimited comparison, correlation and sorting. These data are typically listed on the TFR and can be entered directly from reported information. The narrative portions of the Failure Summary Report, however, are nonstructured in a computer sense and require review, analysis and interpretation. An example of a Failure Review Board Problem is shown in Table 1.

To maximize existing computer capabilities and minimize labor-intensive problem searches, a simple test failure cause code system was developed. This system summarizes the cause of each test failure into two descriptive cause codes which are written in English. These codes provide a dedicated description of the nature of the failure cause, understandable without prior reference to either the narrative portion of the Failure Review Board Problem or to a cryptic code book. The codes consist typically of an adjective and a noun. The codes, in conjunction with selected problem specific information, provide data inputs which can be queried to produce reports containing abbreviated summaries of information pertinent to potential failure trends. The reports contain sufficient references to permit further analysis as required.

To minimize the number of variables listed as outputs on a Failure Trend Analysis Report, software was designed to control variables during input so that multiple factors or conditions do not require printout. These factors are assumed to be present, consistent with the structure of the inquiry to the computer. Table 2 is a listing of all the problem specific data points which can be interactive in the computerized Failure Trend Analysis.

Shown in Table 3 are the variables which are used in the actual inquiry process. The inquiry is conducted via a remote video computer terminal, linked to a central computer which stores all the problem specific data on all space programs. Inquiries can be performed on a program specific or a cross program basis. The system uses a match code logic on an equals/less than/greater than basis. For example, inquiries concerning failures which occur at the satellite level during thermal vacuum testing receive an inquiry such as: FL (failure level) = Satellite and EN (environment) = T/V. Such a simple inquiry identifies and lists all failures in that specific combination of categories. The less than/greater function is typically used to specify time periods.

This same inquiry for failures occurring in 1982 only is entered as FL = Satellite, and EN = T/V, and FD (failure date) > 811231, and FD < 830101. A further refinement to limit the output of failures concerning a single part number is queried as FL = satellite, and EN = T/V, and FD > 811231, and FD < 830101, and PN (part number) = 523100. Additional parameters common to all inquiries can be continued to be added using the "and" function and the desired codes for failure level, failure source, program code, etc. Inquiries concerning the same parameters but for different part number or other variables are queried by using an "or" function. This permits entering up to a total of 99

potential parameters in a single inquiry.

The output is available in four separate sort configurations or any combination of the four. The four sort parameters are: FRB problem number sequence, part number sequence, fail date sequence, and failure cause code sequence. The inquiry and sort parameters can be manipulated to provide both limited statistical analysis or a preferred output format. This feature aids in the identification of the failure trend if it exists.

EXAMPLE PROBLEM SOLVING

As an example, consider a hypothetical case which involves the latent failure of a thick film ceramic substrate at various levels of test. Furthermore, assume that this failure is caused by cracking of the substrate in various locations resulting in an electrical anomaly or failure during test. Additionally, assume that the design and application of this substrate is similar for five other separate substrates used in five different RF carriers. Finally assume that failures of all six substrates have occurred sporadically in time over various environments, levels of testing and space programs, resulting in a failure of the next higher assembly of the substrate.

As previously mentioned, a normal function of Reliability is to conduct failure trend analysis as an ongoing action. Unusual failures typically receive increased attention. For example, it would be unusual to detect a receiver failure due to a cracked substrate in a lower level RF carrier during satellite level thermal vacuum testing. Both the carrier and the entire receiver complete several iterations of powered monitored vibration and extended periods of powered monitored temperature and thermal vacuum cycles at lower levels of testing.

For this hypothetical example, the start point for a failure trend inquiry is to input the part number of the failed substrate in the computer to determine if other failures of that substrate have occurred. This information is available at the video terminal in a matter of minutes and is displayed both quantitatively and with a descriptive format as shown in Table 4. Note that "cracked substrate" is displayed as the failure cause. In addition, pertinent information, such as the failed date, the next higher assembly or assemblies and reference document numbers, is also shown.

Once it has been established that previous failures have occurred, the next phase of the inquiry is to determine at what levels of hardware these failures occurred and under what environments. Inquiry inputs can be entered in several ways; however, typically the computer is instructed to identify failures occurring at the satellite, system, component, tray or module level. This inquiry redistributes the same data quantitatively as shown in Table 5. The descriptive portion of the report, not shown, remains the same as shown in Table 4. Information shown in Table 5 shows that this failure occurred twice during satellite level testing, once at the communications module (system) level of testing, and three times at the receiver (component) level of testing. No failures are recorded at the tray or module level of testing.

Such data suggests that this is a failure trend requiring further attention. An additional inquiry, not shown, distributes these failures over various environments to determine what effect if any environmental testing plays in the cause or discovery of the failure. To simplify the remaining processes, which involve further investigation and coordination, assume

that the final outcome of this problem was the determination that the substrate mounting strategy was inadequate. Further investigation revealed that the failure was induced during powered vibration at the receiver and satellite levels. In all cases this failure was not detected during that test and sometimes was not discovered until subsequent thermal vacuum testing. Such a finding would result in a design change to the substrate mounting strategy.

Concurrent with the first failure trend analysis, it would be typical to investigate cracked substrates as a generic problem. A Failure Trend Analysis Report, sorted by failed part number, would disclose repetitive failures of different part number substrates experiencing similar failures individually or collectively. This inquiry is initiated by inquiring by the cause code (cracked substrate) and failure source code (substrate). Substrates identified as repetitive failures would be investigated in a similar manner. This would identify the five other hypothetical substrates similar in design to the first example. These design defects would also be corrected. Another possible scenario which is typical is that the potential defect in all six substrates would be addressed as a logical offshoot during the ensuing engineering investigation resulting from the failure of the first example substrate.

CONCLUSIONS

The extensive data base contained in the Failure Summary Report Problem Specific Information, coupled with the consistent and concise cause codes assigned to each Failure Review Board Problem, permit rapid identification of potential failure trends and greatly assist failure trend identification and resolution of conditions which may cause ground test failures, potential reliability concerns, or on-orbit failures. As the Failure Summary Report data base is continuously updated via on-line video computer terminals linked to a central computer, the Failure Trend Analysis System remains as current as the data which is entered. Arduous searches through data which took days and even weeks with earlier manual systems are possible in a few minutes from the time the inquiry is entered. Human errors and omissions are minimized due to the increased accuracy achieved by use of the computer. Data inputs to the Failure Summary Report are carefully reviewed for accuracy during normal Failure Review Board functions. Obviously, the accuracy of the data entries is essential to successful operation of the computerized Failure Trend Analysis System.

Periodic reports of failures by cause code permit previously undetected failure trends to become more visible. Routine inquiries concerning particular environmental test failures or other parameters may often disclose an unforeseen failure trend. The ability to manipulate inquiry parameters has proved to be extremely effective in both the identification of the failure trend as well as the determination of test parameters which contribute toward the identification of the failure. Ultimately, the Failure Trend Analysis System provides an aid to the identification and correction of potential failures and reliability risks while someone can still reach out and touch them.

TABLE 1

EXAMPLE FAILURE REVIEW BOARD PROBLEM

PROBLEM NUMBER	XX-04-0352	
ASSEMBLY NUMBER	281200-04-0004	
ASSEMBLY NAME	6 GHZ RECEIVER	
FAILED P/N	282610-OA	0066
FAILURE LVL/SOURCE	SATELLITE	SUBSTRATE
FAIL DATE	05-01-83	
TFR NUMBER	28251	28242
SUB ASSY P/N	281210-01	0016
CAUSE CODE 1	CRACKED	
CAUSE CODE 2	SUBSTRATE	
TEST LVL/OPEN DAYS	A/T 015	
OPEN DATE	05-03-83	
PROB CLOSE DATE	05-27-83	
ENVIRONMENT	T/V	
TFR CLOSE DATE	05-15-83	

STATEMENT OF PROBLEM

COMM, 6 GHZ RECEIVER FLT UNIT
SB-307222/6.5 (THERMAL VACUUM). RECEIVER REMOVED FROM SATELLITE ON TFR 28242, XX-01-0410. OUTPUT IS ERRATIC AT COLD TEMPERATURE.

CAUSE OF PROBLEM

TROUBLESHOOTING REVEALED THAT THE A1A3 CARRIER WAS INTERMITTENT DURING TEMPERATURE CYCLING. VISUAL EXAMINATION DISCLOSED THAT THE A1A3 SUBSTRATE IS CRACKED.

A FAILURE TREND INVESTIGATION WAS INITIATED AND DETERMINED THAT FIVE PREVIOUS FAILURES OF THIS SUBSTRATE HAVE OCCURRED. ALL WERE DUE TO CRACKS DETECTED DURING VIBRATION TESTING OR SUBSEQUENT THERMAL VACUUM TESTING. INVESTIGATION BY ENGINEERING INDICATED THAT THE MOUNTING SYSTEM OF THE SUBSTRATE WAS THE CAUSE OF THE PROBLEM.

CORRECTIVE ACTION

ECO G5326 WAS ISSUED TO REVISE THE MOUNTING SYSTEM OF THE A1A3 SUBSTRATE AS WELL AS FOR THE FOLLOWING SUBSTRATES WHICH ARE SIMILAR IN DESIGN AND APPLICATION: 282611-OA, 282612-OA, 282613-OA, 282614-OA AND 282615-OA. SPECIAL TESTING WAS SUCCESSFULLY CONDUCTED AT AND ABOVE QUAL LEVEL VIBRATION LOADING. ECO EFFECTIVITY IS FOR ALL UNITS.

THE A1A3 SUBSTRATE WAS REMOVED AND REPLACED USING THE REVISED MOUNTING DESIGN. THE RECEIVER SUCCESSFULLY COMPLETED ALL COMPONENT LEVEL TESTING AND WAS RETURNED TO THE SATELLITE.

TABLE 2

INTERACTIVE FAILURE TREND DATA

PROGRAM CODE - APPLICABLE SPACE PROGRAM CODE		
ASSEMBLY PART #/SERIAL #		(SEE NOTE 1)
SUBASSEMBLY PART #'S/SERIAL #'S		(SEE NOTE 1)
FAILED PART #'S/SERIAL #'S		(SEE NOTE 1)
FAIL DATE	-	(DATE FAILURE OCCURRED)
TEST ENVIRONMENT	-	(VIBRATION, AMBIENT, THERMAL VACUUM, TEMPERATURE CYCLE, ETC.)
TEST LEVEL	-	(I.E. IN PROCESS, INTEGRATION, QUALIFICATION, ACCEPTANCE TESTING)
LEVEL OF FAILURE	-	(LEVEL OF TESTING AT WHICH FAILURE IS DETECTED)
LEVEL OF FAILURE SOURCE	-	(LEVEL OF EQUIPMENT WHERE CAUSE OF FAILURE IS FOUND)
# OF DAYS TFR IS "OPEN"	-	(ELAPSED TIME IN DAYS FROM THE DATE OF FAILURE UNTIL MRB ACTIONS ARE COMPLETED)
FAILURE CAUSE CODE 1	-	(I.E. CRACKED, REVERSED, MISWIRED, ETC.)
FAILURE CAUSE CODE 2	-	(I.E. SUBSTRATE, TRANSISTOR, DIODE, ETC.)

NOTE 1:

Any part number typically has a base part number, an option number and a serial number. Entry of the base part number only will identify all problems for that base number, all of its available options and all serial numbers. Entry of the base & option number will find only that option number and all serial numbers of that option. Entry of of base, option, and a serial number will identify all failures of that particular unit.

TABLE 3

DATA IDENTIFIER CODES (DIC)

CAUSE CODE	CC	PART NUMBER	PN
CAUSE CODE 1	C1	PROGRAM CODE	PC
CAUSE CODE 2	C2	SPARE 1	S1
DOWNGRADE	DG	SPARE 2	S2
ENVIRONMENT	EN	SPARE 3	S3
FAILURE DATE	FD*	SPARE 4	S4
FAILURE LEVEL	FL	SPARE 5	S5
FAILURE SOURCE	FS	TEST LEVEL	TL
OPEN TFR DAYS	OD	(* YYMMDD)	

ENTER RETURN WHEN COMPLETE.

ENTER EACH PARAMETER INDIVIDUALLY.
FORMAT: DIC, MC, VALUE, REL RETURN

NOTE:

Spares have been included in the software for future use when additional parameters useful to failure trend analysis are identified.

MATCH CODES (MC)

LESS THAN	<
EQUAL TO	=
GREATER THAN	>

RELATIONSHIPS (REL)

OR
AND

TABLE 4

DESIGN SUPPORT RELIABILITY TREND ANALYSIS REPORT DATE 05-05-83 PAGE 1

```
PROBLEM NBR  TFR NBR   TP PART NBR        CAUSE CODE 1 CAUSE CODE 2 YR FAR NBR

*****************************************************************************
*                                                                           *
*                THIS IS A SUMMARY REPORT OF FRS09 FOR 05/05/83             *
*                                                                           *
*                                                                           *
*       OF        644  OPEN PROBLEMS AND                                    *
*                8056  HISTORICAL PROBLEMS, THERE WERE                      *
*                   6  PROBLEMS SELECTED, (  0.07%), AND                    *
*                  15  DETAIL RECORDS WERE PRINTED.                         *
*                   2  PARAMETER RECORDS WERE SUBMITTED:                    *
* NO PARAMETER CHECKED                  SEL-MTCH %-SELECTED TOT-MTCH  %-TOTAL *
* 01  PN = 282610-QA                        6      (100%)       6   (0.07%) *
* 02  SORT  CC                                                              *
*                                                                           *
*****************************************************************************

                      THIS IS THE DETAIL FOR 05/05/83

XX010057    04985    AP 280000-02 0002    CRACKED    SUBSTRATE    81
            11610    FP 282610-QA 0014
                     SP 281200-02 0005

XX010215    11610    AP 280000-03 0003    CRACKED    SUBSTRATE    82
            04985    FP 282610-QA 0097
                     SP 281200-02 0070

XX030410    28242    AP 28100-04  0004    CRACKED    SUBSTRATE    83
                     FP 282610-QA 0066
                     SP 281200-02 0174

XX040148    19415    AP 281200-02 0017    CRACKED    SUBSTRATE    80  L9327
                     FP 282610-QA 0033

XX040350    28044    AP 281200-02 0055    CRACKED    SUBSTRATE    82
            35369    FP 282610-QA 0045

XX050119    36235    AP 281300-01 0013    CRACKED    SUBSTRATE    82
                     FP 282610-QA 0059
```

TABLE 5

DESIGN SUPPORT RELIABILITY TREND ANALYSIS REPORT DATE 05-05-83 PAGE 1

PROBLEM NBR TFR NBR TP PART NBR CAUSE CODE 1 CAUSE CODE 2 YR FAR NBR

```
*****************************************************************************
*                                                                           *
*                THIS IS A SUMMARY REPORT OF FRS09 FOR 05/05/83             *
*                                                                           *
*        OF          644  OPEN PROBLEMS AND                                 *
*                   8056  HISTORICAL PROBLEMS, THERE WERE                   *
*                      6  PROBLEMS SELECTED, (  0.07%), AND                 *
*                     15  DETAIL RECORDS WERE PRINTED                       *
*                      9  PARAMETER RECORDS WERE SUBMITTED:                 *
*  NO PARAMETER CHECKED                    SEL-MTCH %-SELECTED TOT-MTCH  %-TOTAL  *
*  01  PN = 282610-QA                 AND,     6     ( 100%)       6  (  0.07%) *
*  02  CC = CRACKED       SUBSTRATE   AND,     6     ( 100%)       6  (  0.07%) *
*  03  FS = SUBSTRATE                 AND,     6     ( 100%)       6  (  0.07%) *
*  04  FL = SATELLITE                 OR ,     2     (   0%)     166  (  1.91%) *
*  05  FL = SYSTEM                    OR ,     1     (   0%)      42  (  0.48%) *
*  06  FL = COMPONENT                 OR ,     3     (   0%)     713  (  8.20%) *
*  07  FL = TRAY                      OR ,     0     (   0%)      76  (  0.87%) *
*  08  FL = MODULE                       ,     0     (   0%)    1703  ( 19.58%) *
*  09  SORT                                                                   *
*                                                                             *
*****************************************************************************
```

PANEL DISCUSSION

CONCLUSIONS

This working group was conducted as a free-form discussion with the panelists interacting both among themselves and the audience. The proceedings began with a description of PDRs and CDRs. The remainder of the four-hour discussions centered around these reviews, their usefulness, conduct, preparation, audience, cost effectiveness, format, etc. The major findings of this group can be summarized as follows:

1. Design Reviews (SRR, PDR, CDR) as defined by MIL-STD-1521 and AFSD directives are necessary and productive.

2. These reviews are management tools and do not work well in uncovering design deficiencies.

3. 95% of the efficiency of Reviews is accomplished in the preparation--i.e., internal audits, dry runs, and "peer" reviews. These are the effective tools in uncovering design deficiences.

4. The test equipment to test unit interface requires FEMA analysis to establish test equipment failure modes which might damage flight hardware under test. Test equipment design reviews should be required to eliminate such failure modes.

COMPUTER-AIDED RELIABILITY
ESTIMATION TECHNIQUES
FOR FAULT-TOLERANT SYSTEMS

COMBINED ANALYTIC SIMULATIVE
TECHNIQUE

Salvatore J. Bavuso
Fault Tolerant Systems Branch
NASA Langley Research Center
Hampton, Virginia

OBJECTIVE: DEVELOP A CAPABILITY TO ASSESS THE RELIABILITY OF ANY FAULT TOLERANT DIGITAL COMPUTER-BASED SYSTEM, INCLUDING THE SYSTEM EFFECTS OF SOFTWARE

The technique for performing reliability assessments is based on the utilization of an analytic model which accounts for the long time constants of hardware and/or software failures and a simulative model that tracks the short time constants of system fault-handling mechanisms. Together these models, which are embodied in computer programs, make possible the reliability assessment of large, practical fault-tolerant systems.

CARE III

COMPUTER-AIDED RELIABILITY ESTIMATION

AN ANALYTIC CAPABILITY

The analytic capability is the CARE III computer program that was codeveloped by the Raytheon Company and the Langley Research Center.

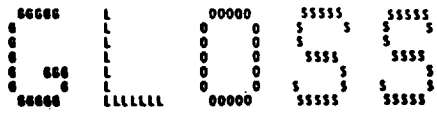
- Stiffler, J. J., et al.: CARE III Final Report Phase I, Vols. 1 and 2. Raytheon Co., Sudbury, Mass. NASA CR-159122 and NASA CR-159123, 1979.

- Stiffler, J. J., et al.: CARE III Phase II Report - Mathematical Description. Raytheon Co., Sudbury, Mass. NASA CR-3566, 1982.

- Trivedi, K. S., et al.: A Tutorial on the CARE III Approach to Reliability Modeling, Duke University, Durham, N.C. NASA CR-3488, 1981.

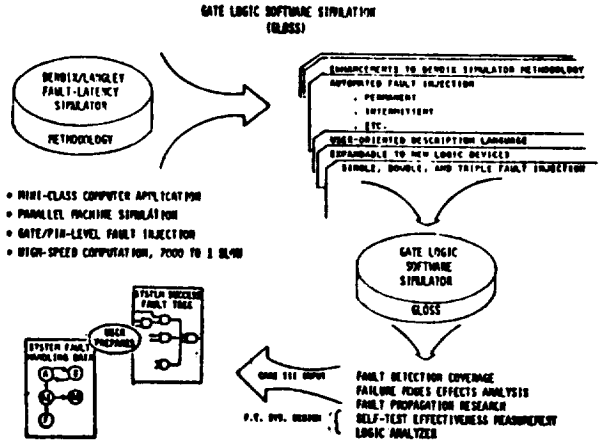
 PAGE 206 INTENTIONALLY BLANK

- Bavuso, S. J.: Advanced Reliability Modeling of Fault-Tolerant Computer-Based Systems, presented at NATO Advanced Study Institute on Electronic Systems Effectiveness and Life Cycle Testing, University of East Anglia, Norwich, Norfolk, UK, 1982.



GATE LOGIC SOFTWARE SIMULATION

A SIMULATIVE CAPABILITY



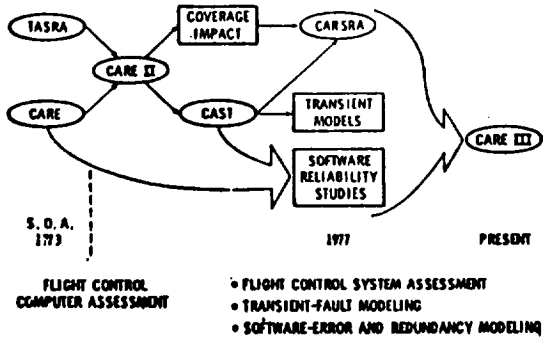
The GLOSS is the simulative capability which provides CARE III with stochastic fault-handling data. The GLOSS concept was demonstrated by application to the CPU of an avionic processor. A generalized GLOSS that provides a user-friendly hardware description language interface is currently being developed. The GLOSS was codeveloped by the Bendix Corporation, Teterboro, New Jersey, and the Langley Research Center.

- McGough, J. G., et al.: Measurement of Fault Latency in a Digital Avionic Miniprocessor, Bendix Co., Teterboro, N.J. NASA CR-3462, 1981.

- McGough, J. G., et al.: Measurement of Fault Latency in a Digital Avionic Processor Part II, Bendix Corp., Teterboro, N.J. NASA CR-145371, 1983.

- Nagel, P.: Modeling of a Latent Fault Detector in a Digital System. NASA CR-145371, 1978.

- Bavuso, S. J., et al.: Latent Fault Modeling and Measurement Methodology for Application to Digital Flight Controls. Advanced Flight Control Symposium, USAF Academy, Colorado Springs, Colo., August 1981.



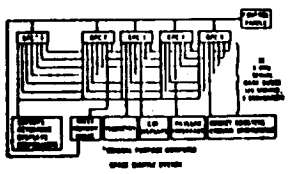
Our quest for achieving our goal began circa 1973. State-of-the-art reliability evaluators were typical of CARE, a Jet Propulsion Laboratory computer program and TASRA, Tabular System Reliability Analysis due to Battelle Memorial Laboratories, Columbus, Ohio. The Raytheon Company at Sudbury, Massachusetts, and Langley developed the CARE II which provided a superset CARE model with an extensive fault-handling model. Langley was also involved in the development of CAST, Combined Analytic Simulative Technique which provides the current Langley modeling concept. CAST was an Ultra Systems, Inc., Newport Beach, California, product; and CARSRA, Computer-Aided Redundant System Reliability Analysis, a spin-off from the Boeing ARCS (Advanced Reconfigurable Computer System) study. Langley has also been involved in numerous technology development studies, some of which are depicted by the boxes. The result of this long-term involvement has culminated in the development of the CARE III.

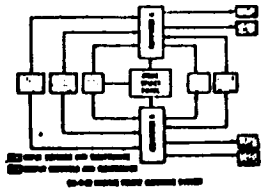
PROBLEMS:

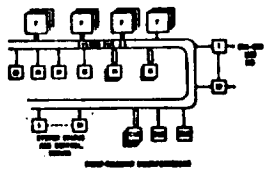
1. EVERYTHING IMPORTANT WHEN $P_F < 10^{-9}$

2. PROGRAM VERSATILITY VS. CONVENIENCE AND EFFICIENCY

On our way toward developing the specifications for the CARE III, we found that for ultrareliable systems certain factors that previously were of little interest to the reliability analyst, now potentially have a significant effect. An example of this observation is the latent (undetected) fault. We also realized that even complex assessment capabilities must be user friendly - always a difficult task for complex capabilities.

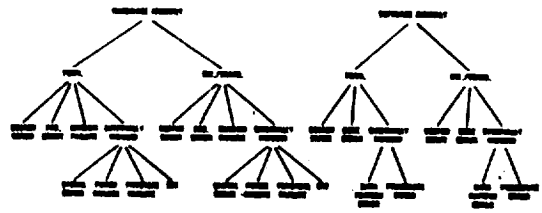






In ultrareliable fault-tolerant systems, the inability of a system to achieve perfect fault/error-handling is often the dominant system failure cause. The major contributor of diminished fault/error-handling is often attributed to the latent fault/error. The long-term (latent) accumulation of faults/errors poses a severe threat to the system's ability to detect and mask-out anomalies. The modeling of fault/error-handling adds a tremendous amount of additional complexity to the reliability assessment task.

The class of fault-tolerant systems of most interest nowadays utilizes off-the-shelf processors or computers. These systems rely heavily on the ability of the processors to detect system faults/errors, to identify the fault/error to the smallest reconfigurable unit, and to effect recovery.



THE PREDOMINATE CAUSE OF FAILURE IN ULTRARELIABLE DIGITAL SYSTEMS HAS BEEN SHOWN TO BE ATTRIBUTED TO FACTORS OTHER THAN HARDWARE SPARES DEPLETION.

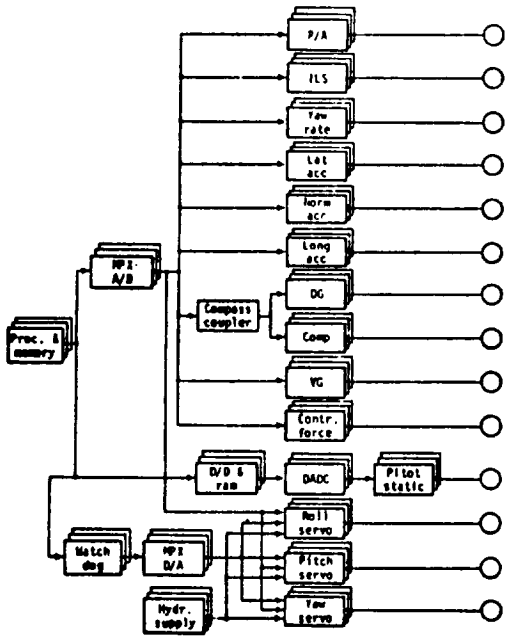
COVERAGE - MEASURE OF SYSTEM'S ABILITY TO HANDLE FAULTS ⇒ SYSTEM

- FAULT DETECTION
- FAULT ISOLATION
- RECONFIGURATION AND RECOVERY

UNDETECTED FAULT - LATENT FAULT

The increased complexity is indicated by the number of additional fault/error models that must be considered. Each ach in the above trees resents a fault/error model. Faults are hardware caused while errors are caused by fault or by software design anomalies. Either

may be permanent or appear transient or intermittent. The common piece-parts reliability analysis is shown as a permanent random hardware failure.



Ultrareliable fault-tolerant systems increase the system reliability by employing redundancy, further compounding the modeling task. A typical proposed advanced reconfigurable flight control system would utilize triple voting of units shown above for the sensors, processor-memories, and actuator electronics. In this example, the number of units increased from 22 for a nonredundant system to 64 for the fault-tolerant architecture.

POSSIBLE STOCHASTIC MODELING APPROACHES

- **MARKOV (CAST, ARIES, CARSRA, SURF)**
- **COMBINATORIAL (CARE, CARE II)**
- **KOLMOGOROV (CARE III)**

Until recently, the reliability analyst was forced to compromise his analysis of such large systems by either modeling sections of the problem at a time and/or he had to make simplifying assumptions to keep the size of the reliability model tractable. The difficulty in this approach is that it is time consuming and difficult, and perhaps more importantly, it is prone to error and often unreproducible. Reliability models for the advanced reconfigurable system example that included the detail previously discussed would require on the order of millions of states in the Markov modeling sense. For each state there exists an ordinary differential equation. Thus, a Markov model for this system would require the solution of millions of differential equations - a task that is expensive, if not impossible.

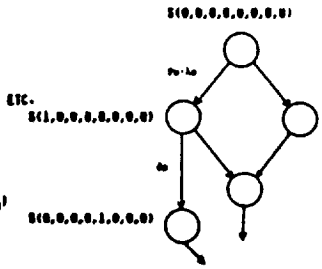
MARKOV MODELS

SYSTEM STATES TYPICALLY CHARACTERIZED BY NUMBERS OF FAULTS OF EACH MODULE TYPE AND FAULT-HANDLING STATUS

- TYPE: RECONFIGURATION MODULE e.g., PROCESSOR (P), MEMORY (M), ETC.
- STATUS: DETECTED, UNDETECTED PERMANENT, UNDETECTED ACTIVE INTERMITTENT, ETC.

EX. $S(P_u, M_u, B_u, G_u, P_D, M_D, B_D, G_D)$

u ⇒ UNDETECTED D ⇒ DETECTED



TRANSITION RATES ARE USUALLY TREATED AS CONSTANTS (EXPONENTIAL TRANSITION PROBABILITIES)

The explosion in states can be seen from a brief analysis of a simple Markov model. Each circle is a system state and is defined by an n-tuple called S. Each n-tuple contains component values that describe a system state and contains information about the number of faults of each module type (P_u, M_u, B_u, G_u) and, in this example, only detection status (P_D, M_D, B_D, G_D). The S n-tuple is described by $S(P_u, M_u, B_u, G_u, P_D, M_D, B_D, G_D)$, an 8-tuple. Transitions from state to state are shown by the arrows and are determined by the module failure and detection rates, and the number of operational modules at a given state.

In practical systems, the number of different module types would easily approach 22 as for the advanced reconfigurable flight control system previously discussed. For each module, a set of fault-handling components is included in the n-tuple for each type of failure, e.g., transient, permanent, and intermittent. The total number of n-tuple components becomes very large. For example, if the 22 stage system was coupled so that failures across stages could cause system failure and two types of faults, permanent and transient were considered where each requires 3 states to describe its stage fault-handling and each stage could tolerate at most one failure, the number of Markov states is greater than 10^{17}. By contrast, CARE III currently supports 70 stages, 5 fault types, 6 states/fault-handling model, and the number of failures a stage can tolerate is a user defined parameter. A test version of CARE III was expanded to 200 stages.

POSSIBLE STOCHASTIC MODELING APPROACHES

- MARKOV (CAST, ARIES, CARSRA, SURF)
- → COMBINATORIAL (CARE, CARE II)
- → KOLMOGOROV (CARE III)

Aside from using the popular Markov technique, two other approaches come to mind: The combinatorial method is the

traditional piece-parts technique. In applying this technique to a fault-tolerant system with a reasonable degree of complexity, one soon learns, as in CARE II, that the computational aspects become unmanageable involving nested integrals four or more deep. The Kolmogorov method in conjunction with a state aggregation technique overcomes the computational difficulties of both the Markov and combinatorial techniques.

REQUIREMENTS SUMMARY

- STRUCTURAL MODELING
  - LARGE STATE SIZE ~O 10^6 (MARKOV SENSE)
  - NON-CONSTANT HAZARD RATES
  - GENERAL FAULT REDUNDANCY MANAGEMENT CAPABILITY AND SYSTEMS SUCCESS CRITERIA
  - A FUNCTION OF COVERAGE
  - SENSITIVITY ANALYSIS
  - FAULT TOLERANT SOFTWARE
- COVERAGE MODELING
  - TIME DEPENDENT COVERAGE
  - COVER MOST FAULT CLASSES AND ENHANCEMENT MECHANISMS (DETECTORS, ISOLATORS AND RECOVERY SCHEMES)
  - COVERAGE MODEL DEFINED IN TERMS OF AVAILABLE OR MEASUREABLE INPUT DATA
  - LATENT FAULT MODELS
  - INTERMITTENT/TRANSIENT DETECTION AND RECOVERY MODELS
- HAZARD RATE DATA
  - INTERMITTENT/TRANSIENT FAULT MODEL
  - SOFTWARE ERROR MODEL
  - LATENT FAULT MODEL

Our long-term involvement and study of reliability assessment methodologies brought us to conclude with the above requirements. I'll touch on the not so obvious ones. The requirement for nonconstant hazard rates is in recognition that some electronic devices exhibit such failure rates and devices with wear-out mechanisms common to mechanical and hydraulic systems also have nonconstant failure rates. It's too early to tell if software will follow the common constant failure rate model; however, a distribution such as Weibull should have enough generality to cover it in the event it is not constant. The word "coverage" is synonymous with fault-handling.

- GENERAL-PURPOSE RELIABILITY ANALYSIS AND DESIGN TOOL FOR FAULT-TOLERANT SYSTEMS
- LARGE REDUCTION OF STATE SIZE
- FAULT-HANDLING MODEL BASED ON PROBABILISTIC DESCRIPTION OF DETECTION, ISOLATION, AND RECOVERY MECHANISMS
- VARIETY OF FAULT AND ERROR MODELS (STATIONARY AND NON-STATIONARY)
  - PERMANENT
  - TRANSIENT
  - INTERMITTENT
  - DESIGN FAULTS
  - SOFTWARE ERRORS
  - LATENT FAULTS
- USER-ORIENTED LANGUAGE FOR DESCRIBING COMPLEX SYSTEM CONFIGURATIONS AND SUCCESS CRITERIA (FAULT TREE)

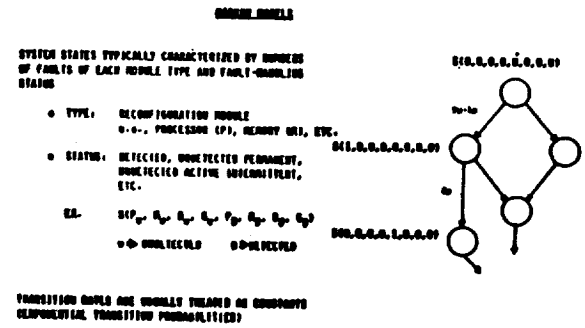
CARE III was designed to meet all of the previously stated goals; however, the current version does not have an automated sensitivity analysis capability. Sensitivity studies can be conducted with multiple runs. Although CARE III contains the framework to include software modeling, a software error model has not been selected for CARE III. The user must fit his software error models to the Weibull error/fault occurrence model. The flexibility of the Weibull model should cover most applications.

Time dependent coverage alludes to the capability of modeling the system reaction to time varying fault-handling processes such as accounting for an intermittent failure manifesting its benign state ("healed" failure) while errors generated during its active state (failed manifestation) propagate through the system posing a potential failure situation.

CARE III APPROACH

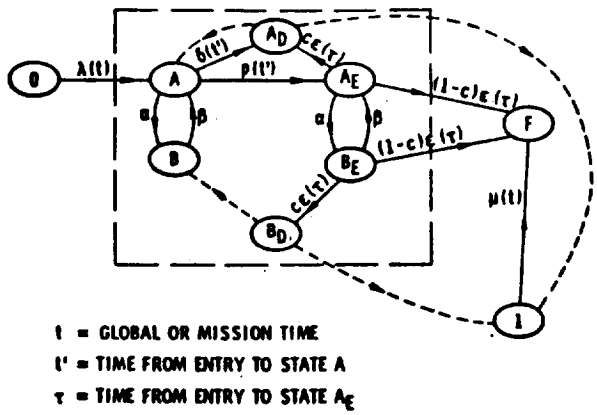
- DEFINE SYSTEM STATE ONLY IN TERMS OF NUMBER OF EXISTING FAULTS
- INDEPENDENTLY EVALUATE TRANSITION PARAMETERS AS A FUNCTION OF DISTRIBUTION OF POSSIBLE FAULT TYPES AND STATES
- DETERMINE RELIABILITY USING KOLMOGOROV'S FORWARD DIFFERENTIAL EQUATIONS
- NUMBER OF STATES DRASTICALLY REDUCED; TRANSITION RATES NECESSARILY TIME-DEPENDENT

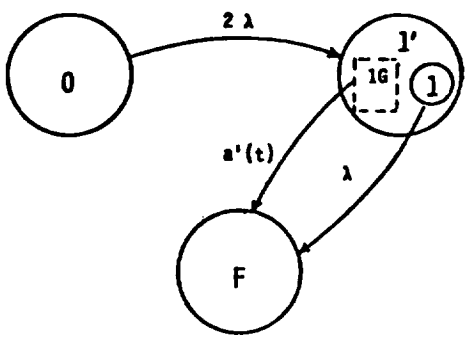
The ability of CARE III to provide extensive fault-occurrence and fault-handling models is largely attributed to its ability to cope with large state spaces and is made possible by the observation that the time constants associated with fault-occurrence are on the order of 10^4 hours while the time constants of the fault-handling model are on the order of 10^{-5} hour. This wide time separation allows the fault-occurrence model to be treated as independent of the fault-handling model. Thus, the fault-handling model is evaluated without regard to failure occurrences. The results of the fault-handling model evaluation are then combined with the fault-occurrence model to produce the desired reliability outputs. The fault-occurrence model is solved using Kolmogorov's forward differential equations. The Kolmogorov technique is used because the state reduction process discussed above necessarily requires the solution of a nonhomogenous (time dependent failure rates) Markov process.



As a point of clarification, in the popular Markov method, each state contains information about failure occurrences and fault-handling, fault detection in this example. CARE III removes the fault-handling information from the state and defines a state only in terms of the number of failed units. In this chart, all the

components with subscript "D" vanish - clearly reducing the state size.





AGGREGATED RELIABILITY MODEL
WITH $\lambda(t) = 2\lambda$, $\mu(t) = \lambda$

An illustration of how this works can be seen by observing the reliability model of a two-unit system. States 0, 1, and F are the fault-occurrence states. The states enclosed in the dashed lines are the fault-handling states. The two-unit model is a mixture of a nonhomogenous and a semi-Markov model which is the type of model CARE III was designed to evaluate. The model CARE III actually evaluates is the aggregated reliability model shown in the lower chart. The aggregated model is a nonhomogenous Markov model. CARE III approximates the mixed process with a nonhomogenous Markov process and can do so because of the wide separation in time constants in the fault-occurrence and fault-handling models. In the aggregate model, the states are strictly fault-occurrence states (defines number of failed units). The fault-handling model information contained in the dashed box of the two-unit system is mapped into the time varying transition rate, a'(t). The nonhomogenous aggregated Markov model is solved using the Kolmogorov solution technique to produce time varying probabilities of being in states 0, 1', and F, the failure state, over the desired mission time.

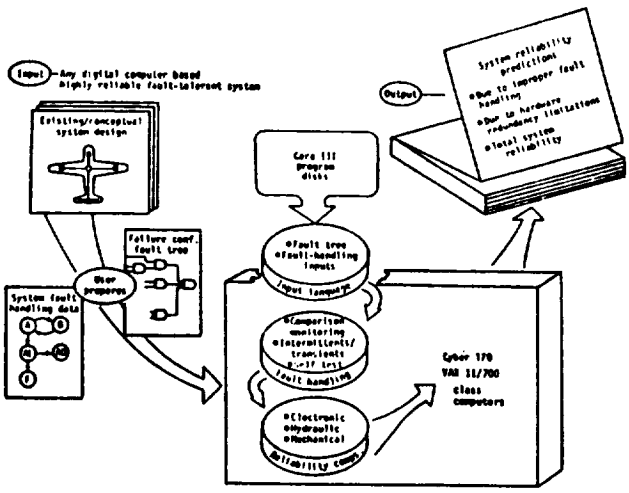
Although the state reduction wasn't too dramatic for this simple example, in practical assessments, state reductions of 6 orders of magnitude have been estimated. CARE III supports a single and a double fault model. The double fault model accounts for critically coupled coexisting failures which are user defined. The critically coupled failures, when they exist, are defined by certain combinations of pairs of states in the single fault model, e.g., two critically

coupled units each in state A will cause system failure. A description of the single fault model can be grasped by referring to the figure of the reliability model of a two-unit system.

Initially, the system is in state 0 and has experienced no failures. When a failure occurs, the system enters state A, the active latent state, given by the arrival density, $\lambda(t)$. Depending upon the nature of the failure, i.e., permanent transient, intermittent, etc., the fault-handling model will be defined differently. For example, if the failure is intermittent, $\lambda(t)$ would be the probability density function (pdf) for the arrival of an intermittent, and states A and B define the intermittent model where α and β are constant transition rates into and out of state B. When the system is in state B, the benign state, the failed unit appears to have healed itself; i.e., the manifestation of the failure, a fault, vanishes. However, when the failed manifestation is once again resumed (the fault reappears), the system enters state A where the failure looks like a permanent failure. It could be detected by a self-test program with pdf $\delta(t')$, and the system would enter state A_D, the active detected state. Given that a spare exists, the system will purge the faulty unit and switch in the spare (dashed arc to state 1). Or while in the active state, the fault could generate errors with pdf $\rho(t')$. The system then will enter the A_E, active error state. The intermittent failure could manifest its intermittent state again so the system would then enter state B_E, the benign error state. Although the failure is benign, the error may not be benign and may cause system failure which is denoted by the B_E to F transition $[(1-c)\varepsilon(\tau)]$. The error detection density is $\varepsilon(\tau)$, and 1-C is the proportion of errors from which the system is unable to recover. While in state B_E, the error could be detected and corrected. In this event, the system enters state B_D (benign detected) by transition $c\varepsilon(\tau)$. At this point, the system may choose to do nothing further with the detected and corrected error and so move to the benign state, or the system may choose to reconfigure out the module containing the error and, therefore, move to state 1. The dashed arcs are instantaneous transitions. The other transition out of state A_E is to state F, the single point failure transition $[(1-c)\varepsilon(\tau)]$. This transition is similar to the B_E to F transition. In a well-designed fault-tolerant system, $(1-c)\varepsilon(\tau)$ should be near zero. If $\lambda(t)$ is the pdf for the arrival of a transient, α would be set to a value greater than zero and β would be equal to zero. The pdf $\lambda(t)$ for the arrival of a permanent failure would be defined so that $\alpha = \beta = 0$. The dashed arc going from state A_D to A enables the analyst to include the effects of the system decision that the detected fault which took the system from state A to A_D was, in fact, a transient. In this regard, the system would not reconfigure out a non-failed module. A judicious choice of values for the single-fault model affords the

analyst a wide range of models. A different fault model may be assigned to each stage, or several models may be assigned to a given stage to cover the effects of different failure mechanisms such as transients, intermittents, and permanent failures.

The reader will note that the reliability model has three measures of time associated with it which necessarily makes the model a semi-Markov process. This added complexity is required because the behavior of the system is dependent on the onset of the various fault-behavior events.



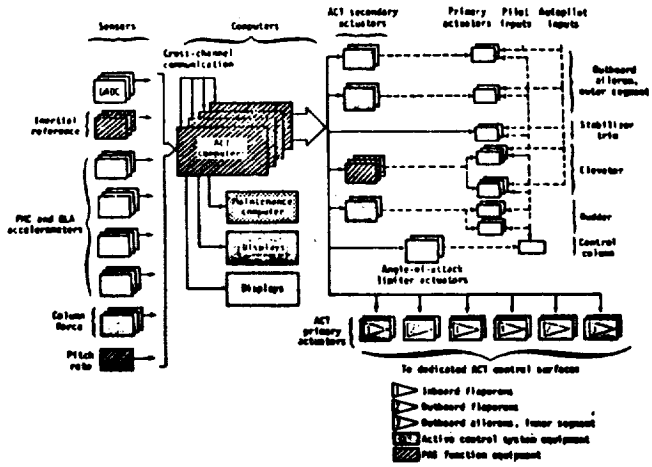
CARE III was designed to model large ultrareliable replicated systems incorporating digital electronics. The CARE III assessment process is depicted and begins with an architectural description of an ultrareliable system. If that description is based on a conceptual model of the system, then CARE III is used as a design tool; or, if the system is well-defined, CARE III is utilized as an analysis tool. In either case, the analyst generates a set of failure rates and probability density functions for the various failure and error models he wishes to include in the analysis. Inclusion of any of these models will necessarily lower the system reliability estimate. The need to include a model is, of course, a function of the architectural structure, its fault-handling mechanisms, and the magnitudes of the parameters in the model. The large choice of failure and error models is provided to increase the realism and credibility of the analysis. The models are user options in CARE III and may be omitted at the discretion of the analyst. Usually he will omit certain models after determining that they have a minimal effect. Some of this modeling information is used to define the system fault-handling model(s) required as a user input, indicated by the state diagram. The remainder of the failure and error modeling data is entered as Fortran NAMELIST statements in the CDC Cyber 173 version or via a prompting user-friendly interface in the VAX 11/780 version.

Another important step in setting up CARE III input is the generation of the system configuration and success criteria. Fault-tolerant systems are usually designed to have many hardware and functional combinations that enable proper system operation. CARE III uses the powerful fault tree

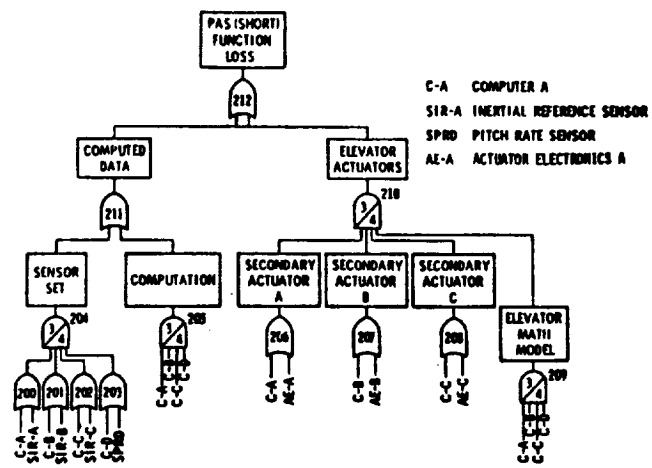
language to describe system failure configurations. In large systems, the number of success combinations can be very large, and for this reason CARE III uses the "unsuccess" or failure combinations instead. In a properly designed system, the number of failure combinations should be considerabley less than the number of system success combinations, thus easing the computational task. The fault tree language provides an excellent medium for delineating the system failure combinations.

The user-prepared data are initially processed by the CARE III input subprogram, CAREIN, shown as the upper disk. It is essentially composed of the fault tree language program. The second CARE III subprogram, COVRGE, processes the fault-handling model data and puts them into the form required by the third subprogram, CARE3, which performs the reliability computations. CARE III is written entirely in Fortran IV language and currently executes on the Control Data Corporation CYBER 170 series computers. The VAX version is written in VAX-11 Fortran and runs under the VMS operating system.

The CARE III output data take two forms, graphical or tabular. In either case, the outputs of most interest are the total system reliability or system survival as a function of time and two vital components: probability of system failure due to hardware redundancy limitations (exhaustion of spares) and probability of system failure due to improper fault handling. In ultrareliable systems, the latter factor is the predominant cause of system failure.

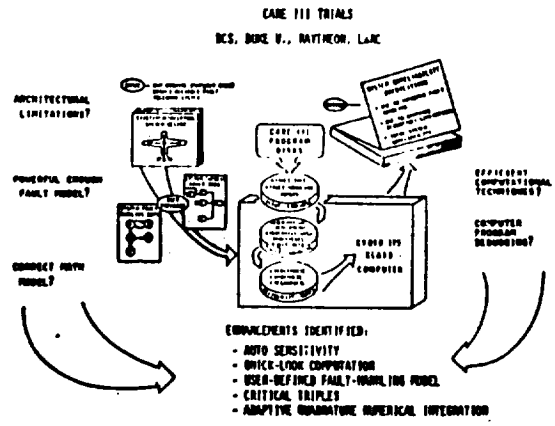


The description of the fault-occurrence model for a complex system can often be laborious and error prone. To alleviate this difficulty, CARE III utilizes the fault tree language to describe this model. An example of the use of this capability is illustrated for the pitch augmented stability function (PAS).



The system fault tree for this function is presented. This tree illustrates that not only hardware redundancy can be represented but functional redundancy as well. The elevator math model is functionally redundant to the secondary actuators. The melding of hardware and functional redundancy is a common practice in aircraft design. The proper entry of this fault tree into CARE III with the necessary failure rate and fault-handling data would yield a prediction of the probability of loss of PAS function as a function of mission time. For the uninitiated, the figure is read as follows. An output from logic OR gate 212 constitutes loss of PAS function which can occur if an output from OR gate 211 occurs, or if an output from gate 210 occurs, or both. Gate 210 yields an output if at least 3 out of 4 secondary actuators or actuator function (elevator math model) fail. Secondary actuator A will fail if computer A fails, or actuator A fails, or both. A similar description can be used to delineate failures due to loss of computation or loss of sensors.

It should also be observed at this point that since the fault-occurrence model and the fault-handling model are treated independently by CARE III, the fault-handling model needs to be evaluated only once if the fault-handling parameters remain unchanged. Thus, if the user is interested in evaluating other functions in the example fault-tolerant flight control system, only the fault tree must be modified and executed by CARE III.



As a result of the novelty of the CARE III modeling technique, CARE III has received considerable conceptual scrutiny and is currently undergoing testing and verification of its computational implementation. Some verification activities have been the rederivation of the mathematical model by an independent investigator using available documentation, a derivation of the math model by analysis of the CARE III code, and numerous test runs comparing CARE III results against Markov model results (where possible), and manually constructed mixed nonhomogenous and semi-Markov models.

The fundamental principles on which CARE III is based appear to have withstood the intensive examination to date. Weaknesses have been uncovered that appear to be attributed to computational

inefficiencies and in some cases, computational modeling errors. These anomalies are currently being corrected. With these corrections, CARE III will be ready for public release, most likely by the year's end. Identified enhancements are planned for a subsequent release.

SELF-CHECKING COMPUTER MODULES (SCCM)

Daniel E. Erickson
Dwight A. Geer
Jet Propulsion Laboratory

The Self-Checking Computer Module (SCCM) approach to fault-tolerant computing has been under development at JPL under NASA (OAST) and Navy (NOSC) sponsorship. Through a discussion of this development, this paper illuminates some of the issues and techniques involved with verifying the reliability of such a system. Since the SCCM is designed to be used in systems where graceful degradation is desirable, the concept of performability is used in expressing the requirement. The variety of fault-tolerance techniques used in the SCCM makes performability analysis difficult.

1. SCCM Design Goals - The SCCM concept was developed to support long-life space applications of moderate cost, moderate performance distributed computing. For deep space missions, low power and long unattended life are necessary. For space station application, repair is possible, but costly. For some satellite applications, and some deep space missions, radiation hardness is desired. The SCCM is a modular concept, compatible with VLSI implementation, and adaptable to a wide variety of missions.

2. Fault Tolerance Approach - The key to system fault-tolerance of the SCCM approach is the fault detection and isolation built into each self-checking computer module. Each SCCM has a high probability of recovering from a transient fault and safely shutting down in the presence of a permanent fault. Access to critical data in the memory of an SCCM may be maintained even though the SCCM is shut down. Because of the modular architecture, spare resources can be selected to achieve the desired performance and reliability for each function.

Within each SCCM, fault detection is provided in the hardware by the fault-tolerant building blocks which interface standard microprocessors and memory chips to the internal bus. For added flexibility, the software controls the fault recovery process. It may roll back processing to before the fault occurrence and may switch in spare building blocks and memory bit planes. The spare switching may also be directed via the external bus to recover a shut-down SCCM.

The building blocks themselves use a variety of techniques to detect and isolate faults. To detect faults, physical redundancy, informational redundancy and watchdog processors are used. Morphic dual-rail logic is used to create self-checking comparitors and parity generators. If a non-correctable fault is detected by any active building block, the microprocessors are reset and memory writes and all outputs are inhibited. If only one fault has occurred since the last software clear sequence, the reset is released and the software performs recovery actions including a clear sequence when it is satisfied that its SCCM is operating properly.

Spares are employed throughout the system to extend its life or its maintenance interval. Besides spare SCCMs and spare busses, spare

bit planes, spare microprocessors, and spare building blocks may be used.

3. Performance Levels - In order to determine whether a given configuration of SCCMs will meet the requirements of a particular mission, we must define the performance levels of the system for that mission. Then we must find the probability of each time profile of performance for the mission. For example, assume that a fully furnished data system can give enhanced service and that, in some degraded states, it can still give basic service. The mission requirement might be 0.95 probability of enhanced service through 5 years, 0.99 probability of basic service through 5 years, 0.999 probability of at least safely shutting down through the first 30 days, and 0.90 probability of basic service through 10 years.

The system performance is a function of the performance of the operational strategies used to maintain its capabilities as well as the performances of its parts. The operational strategies will be implemented manually or in hardware or software. The parts of the data system include the external bus, various I/O devices, and the SCCMs. For each such part we can define performance levels which will support the system performability analysis. We can continue this hierarchical refinement until we get down to units for which fault rates can be predicted such as semiconductor chips.

Even if, as in this case, the mission requirements are not known, this kind of analysis with hypothetical mission requirements leads us to the definition of appropriate performance levels for modular units such as the SCCM and its component building blocks.

4. Performability Analysis - A performability analysis has begun on the Memory Interface Building Block (MIBB). The approach that has been chosen is to perform a theoretical analysis of the MIBB response to stuck-at-one and stuck-at-zero faults and to validate the theoretical analysis with empirical tests. Since the theoretical analysis depends on certain groundrules and assumptions, it was felt that confirmation of its validity was advisable. In fact, the tests showed several discrepancies in the analysis which will be corrected when the analysis is repeated.

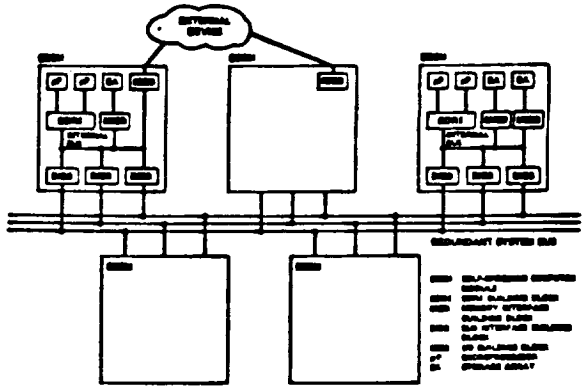
Besides yielding an estimate of the performability of the MIBB, the analysis revealed several design oversights. Some of these can be corrected in the breadboard. Others can be avoided in future designs. The tests also revealed a few implementation flaws such as cross-talk between signal lines which would not be evident from analysis alone.

One weak area of the analysis is the semiconductor fault rate model. The model which was adopted for use (MIL-HDBK-217D) is based on averages from a wide variety of fabrication processes. These data are appropriate for early estimation based on standard parts. For custom parts, new parts or gate arrays, the model may not be accurate. Parts quality assurance is recommended for obtaining final figures for parts from the fabrication line to be used.

5. Observations and Recommendations - The methodology loosely described above appears to be a feasible way

to obtain confidence in the performability of a new data system. It will be greatly aided if performability assurance needs are recognized throughout the development, in the definition of requirements, the selection of the fault-tolerance approach, and the detail design of system level performability models, supported by lower level performability analyses which allow cost-performability trade-offs. Lower level tools will aid the analyses.

DISTRIBUTED COMPUTER (SCCM) ARCHITECTURE



SCCM DESIGN GOALS

- MODULAR ARCHITECTURE FOR DIVERSE APPLICATIONS
- RELIABLE FOR LONG-LIFE APPLICATIONS
- MODERATE COST AND PERFORMANCE
- LOW POWER
- RADIATION HARDENABLE
- COMPATIBLE WITH VLSI IMPLEMENTATION

SCCM RELIABILITY DRIVERS

- DEEP SPACE MISSIONS
  - 5 - 10 YEAR DURATION — NO REPAIR
  - .99 / .95 BASIC/ENHANCED SUCCESS
- SPACE STATION
  - 10 - 20 YEAR LIFE -- INFREQUENT REPAIR
  - LIFE/MISSION CRITICAL
- AIR FORCE SATELLITE — AUTONOMOUS REDUNDANCY AND MAINTENANCE MANAGEMENT SUBSYSTEM (AMMS)
  - 6 MONTHS UNATTENDED OPERATION AFTER 5 YEARS ON ORBIT

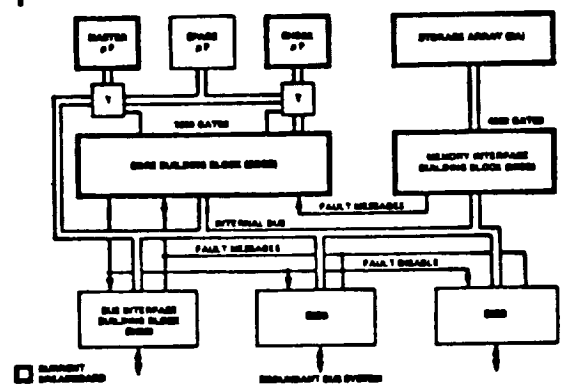
SYSTEM FAULT APPROACH

- SPARING AT COMPUTER MODULE LEVEL
- MODULES RECOVER FROM TRANSIENT FAULTS
- MODULES RECOVER FROM OR SHUT DOWN SAFELY FROM PERMANENT FAULTS
- ALLOW ALTERNATE ARCHITECTURES TO MEET RELIABILITY/PERFORMANCE REQUIREMENTS
- MAINTAIN ACCESS TO UNCORRUPTED CRITICAL DATA IN FAILED MODULES

SCCM FAULT APPROACH

- STANDARD MICROPROCESSOR AND MEMORY CHIPS
- HARDWARE FAULT DETECTION (BUILDING BLOCKS)
  - CORE — PROCESSOR INTERFACE
  - MIBB — MEMORY INTERFACE
  - BIBB — EXTERNAL BUS INTERFACE
  - IOBB — INPUT/OUTPUT
- SOFTWARE CONTROLLED RECOVERY
  - ROLLBACK — FOR TRANSIENT FAULTS
  - SPARE SWITCHING — FOR PERMANENT FAULTS

SELF CHECKING COMPUTER MODULE (SCCM)



HARDWARE FAULT DETECTION

- PHYSICAL REDUNDANCY
  - DUPLICATED CIRCUITS
  - SELF-CHECKING COMPARITORS
- INFORMATIONAL REDUNDANCY
  - PARITY ON INTERNAL BUS
  - MODIFIED HAMMING CODED MEMORY SEC/DED
- WATCHDOG PROCESSORS
  - TIMEOUT COUNTER
  - SEQUENCE CHECKER

HARDWARE FAULT ISOLATION

- TRI-STATE DRIVERS ON INTERNAL BUS
- MICROPROCESSOR RESET
- FAULT COUNTER (0 - 1 - MANY)
  - RESET BY SOFTWARE
  - SET TO 1 BY POWER UP
  - INCREMENTED BY FAULT
  - "MANY" CAUSES A SHUT-DOWN

SPARING

- SPARE SCCMs
- SPARE EXTERNAL BUSSES
- SPARE MIBBs AND BIBBs
- SPARE MEMORY BIT PLANES
- SPARE MICROPROCESSORS

PERFORMABILITY

- PERFORMANCE -- DEALS WITH SYSTEM BEHAVIOR WITH NO FAULTS
- RELIABILITY -- DEALS WITH PROBABILITY OF SYSTEM SUCCESS OR FAILURE
- PERFORMABILITY AS DEFINED BY DR. JOHN MEYER (U. OF M.)
  - -- DEALS WITH THE PROBABILITY OF THE SYSTEM OPERATING AT EACH PERFORMANCE LEVEL AS A FUNCTION OF TIME. THUS, IT DEALS WITH SUCH CONCEPTS AS GRACEFUL DEGRADATION.

SYSTEM PERFORMANCE LEVELS

- FULLY FUNCTIONAL (ENHANCED MISSION)
- PERFORMING BASIC FUNCTIONS ONLY
- PERFORMING KEEP-ALIVE, SAFING ONLY
- CRASHED
- SYSTEM PERFORMANCE DEPENDS ON
  - EXTERNAL BUS AND I/O PERFORMANCE
  - EACH SCCM's PERFORMANCE
  - OPERATIONAL STRATEGIES

SCCM PERFORMANCE LEVELS

- FULLY FUNCTIONAL
- PARTIAL SHUT DOWN (N/MO LIMITED POLLUTION)
  - I/O, EXTERNAL BUS, MEMORY
- SCCM SHUT DOWN (N/MO LIMITED POLLUTION)
  - N/MO MEMORY ACCESSIBLE VIA EXTERNAL BUS
- ACTIVE FAILURE (N/MO CONTINUING POLLUTION)
- SCCM PERFORMANCE DEPENDS ON
  - INTERNAL BUS PERFORMANCE
  - EACH BUILDING BLOCK's PERFORMANCE
  - SOFTWARE PERFORMANCE

MIBB PERFORMANCE LEVELS

- FULLY FUNCTIONAL
- CORRECTING SINGLE BIT ERRORS
- DIAGNOSABILITY IMPAIRED
- DETECTING FAULTS -- 2 BIT, CIRCUIT, INTERNAL BUS
  - N/MO LIMITED MEMORY POLLUTION
  - N/MO LOCKING UP SCCM
- UNDETECTED ERRORS
- MIBB PERFORMANCE DEPENDS ON
  - PART RELIABILITY
  - FAULT COVERAGE, LATENCY
  - FAULT HANDLING STRATEGY

MIBB PERFORMABILITY ANALYSIS
APPROACH

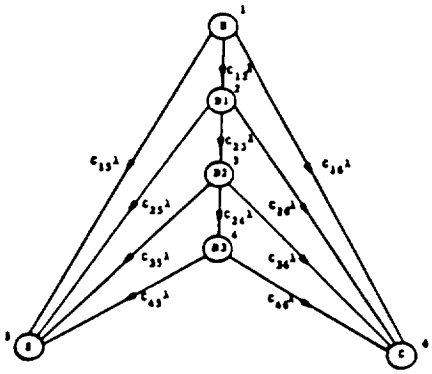
- PERFORM ANALYTICAL ANALYSIS
- PERFORM EMPIRICAL TEST ANALYSIS
- EVALUATE CONSISTENCY OF ANALYSIS
- REDO FMEA (AS NECESSARY)

MIBB PERFORMABILITY ANALYSIS
ANALYTICAL

- IDENTIFY OPERATIONAL STATES -- MODEL
- IDENTIFY MEASURES FOR PERFORMABILITY
- DETERMINE PARAMETRIC SOLUTIONS OF MEASURES
- PERFORM FAILURE MODES AND EFFECTS ANALYSIS (FMEA)
- CALCULATE PARAMETER VALUES
  - FMEA RESULTS
  - FAULT RATE MODELS (eg. MIL-HDBK-217B)
- CALCULATE PERFORMABILITY MEASURES

RIBB OPERATIONAL STATES

MARKOV MODEL



RIBB PARAMETRIC SOLUTION

PROBABILITY OF RUNNING = ≤ 3 DIT-PLANES FAILED

$$P_{10}(t) = P_{11}(t) + P_{12}(t) + P_{13}(t) + P_{14}(t)$$

$$= e^{-\lambda t}$$

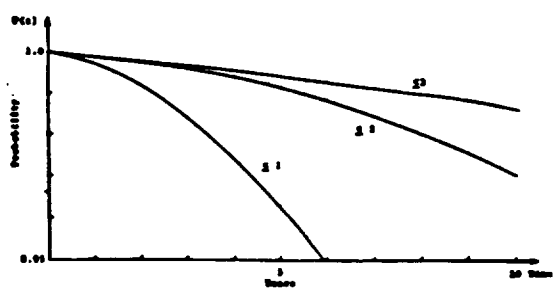
$$+ c_{12}\lambda t e^{-\lambda t}$$

$$+ \frac{1}{2} c_{12} c_{23} \lambda^2 t^2 e^{-\lambda t}$$

$$+ \frac{1}{6} c_{12} c_{23} c_{34} \lambda^3 t^3 e^{-\lambda t}$$

RIBB PERFORMABILITY MEASURE

(NUMBER OF DIT-PLANES FAILED DURING OPERATION)



RIBB PERFORMABILITY ANALYSIS

EMPIRICAL

- RANDOMLY SELECT 250 (OF 4800) SIGNAL TEST POINTS
- TEST WITH TRANSIENT SHORTS TO GROUND
- SPECIAL TEST SOFTWARE
  - EXERCISE MEMORY AND RIBB HARDWARE
  - TEST FOR ERRORS UNDETECTED BY HARDWARE
  - COLLECT/DISPLAY TEST RESULTS

RIBB PERFORMABILITY ANALYSIS

CONSISTENCY EVALUATION

- EMPIRICAL TEST CRASHES >> ANALYTICAL (FMEA) PREDICTION

  CRASHES INCLUDE: MEMORY POLLUTION
  INCORRECT PROGRAM EXECUTION

- MANY EMPIRICAL TEST CRASH POINTS DID NOT MAP TO FMEA CRASH STATES
- COMPARISON REVEALED CORRECTABLE FLAWS IN FMEA METHODOLOGY

RIBB PERFORMABILITY CONCLUSIONS

- THE PROBABILITY OF CRASHES MUST BE REDUCED (DESIGN FIXES HAVE BEEN IDENTIFIED)
- THE RIBB APPEARS TO BE ABLE TO MEET PERFORMABILITY GOALS
- MOST RIBB FAULTS WILL BE DUE TO MEMORY FAULTS
- SEMI-CONDUCTOR FAULT RATE MODEL NEEDS CONFIRMATION FOR ACTUAL PARTS (QUALITY ASSURANCE)

OBSERVATIONS ON ANALYSIS METHODOLOGY

- GROUND RULES ARE IMPORTANT
- TOOLS COULD REDUCE TIME AND ERRORS
- DESIGN APPROACHES STRESSING REGULARITY AND MODULARITY WILL SIMPLIFY ANALYSIS
- EMPIRICAL CONFIRMATION IS CRUCIAL

RECOMMENDATIONS

- DEVELOP SYSTEM PERFORMABILITY MODELS
  - TO MAKE TRADE-OFFS
  - TO DISCOVER APPROPRIATE PERFORMABILITY GOALS/MEASURES FOR DETAILED DESIGN
- CONSIDER TESTABILITY UP FRONT
  - FAULT SIMULATION/INSERTION
  - PERFORMANCE OBSERVATION
- DESIGN FOR VERIFICATION OF PERFORMABILITY
  - MODULARITY
  - TIMING -- SYNCHRONIZATION
  - BUILT IN TEST
- ASSURE QUALITY OF PARTS

MULTIPLE MICROPROCESSOR FLIGHT CONTROL ARCHITECTURES

Gary Hartmann
and
Kevin Driscoll
Honeywell Systems and Research Center

Honeywell is well known for its work in digital avionics control systems, from the first digital flight control system in the Saab Viggen to the high performance inlet and flight control system for the SR-71 and the main engine controller for the space shuttle. Honeywell's continuing research in the area of ultra-reliable real-time control systems has developed several new concepts that have been blended into the Multiple Micro-processor Flight Control System (MMFCS) and its variant architectures. The orginal MMFCS studies were undertaken to determine how the developments in integrated circuit technology would impact digital avionic systems architectures. Further, these studies were to determine how the design of an ultra-reliable digital flight control system could best exploit these technology developments.

RELIABILITY

When examining an ultra-reliable system it is important to consider several reliability issues. These issues include fault tolerance, fault avoidance, fool-proof implementation/application, provability, requirements and protection.

Fault tolerance is the ability of a system to continue operation even though a failure has occurred in one of its components. Fault tolerance is required in ultra-reliable systems. This is because components cannot be made reliable enough to meet the requirements for these systems. Even though integrated circuit technology has increased overall systems reliability, fault tolerance will be needed in ultra-reliable systems for the foreseeable future and may always be required.

More important than fault tolerance, however, is fault avoidance. Fault avoidance is the technique of designing a system such that failures will not occur. Classical techniques for fault avoidance include the use of high quality components, high quality workmanship, and conservative design. Conservative design generally means designing the system to withstand the sum of all possible worst case conditions. It has been proposed that conservative design should also mean avoiding the state-of-the-art; technologies that have a proven track record should be used instead. There is more to fault avoidance than the classical techniques. Some designs are inherently more dangerous than others. These dangerous designs are most often caused by the inclusion of single points of failure and/or fault propagation modes for which the statistics are unknown and cannot be determined.

A good ultra-reliable design should consider that the design elements for reliability can be nullified by the system's implementor and/or the system's end user. Great care should be taken in anticipating the possible implementations and applications for the system. The reliability techniques that can be inadvertantly or easily nullified should be avoided. The design should explicitly state which of its elements are required for reliability and what actions may adversely affect them. The design should also explicitly and clearly state which implementation techniques and what end uses are compatible with this design. The implementation and application rules are just as much a part of the design as the component interconnections.

The failure rates of ultra-reliable systems are so small that the systems cannot be exhaustively tested to determine what these failure rates are. It is important, therefore, that the

PAGE 226 INTENTIONALLY BLANK

system be composed of a small set of simple modules with simple interfaces. These simple modules and simple interfaces can then be exhaustively tested to determine their failure rates. The overall system reliability then can be inferred from a process similar to proof by mathematical induction.

The failure rate requirements for ultra-reliable systems are extremely small. It is common for these requirements to indicate that the probability of failure will be on the order of one in a billion. The most bizarre occurrences imaginable (and more importantly, unimaginable) can happen with this kind of probability. What do these numbers really mean? What is the best way to precisely specify this high level of reliability?

An ultra-reliable system should not only be able to continue proper functioning after one of its components has failed, but it should also make sure that this failure does not affect any of the systems in its environment. And conversely, a failure in any of the systems that surround an ultra-reliable system should not be able to adversely affect the the ultra-reliable system. This means an ultra-reliable system must embody the concept of protection.

PHILOSOPHY

The design philosophy for the MMFCS includes the tailoring of the system to ultra-reliable real-time requirements, the exploitation of cheap microelectronics, and the use of a systems approach.

Ultra-realiable Real-Time Requirements - It was determined very early in the MMFCS program that the system should be specifically tailored to ultra-reliable real-time control. This meant the system would be designed with all these requirements in mind simultaneously. We did not want to try to solve a general class of computing problems. We also felt that a system optimized for any one requirement, for example reliability, could not efficiently solve the entire problem.

Exploitation of Microelectronics - We determined the best way to exploit the developments in integrated circuit technology was to use this new cheap microelectronics to replace the increasingly expensive software. Software cost has been increasing not only because it is a labor intensive product but also because the software is taking up a larger proportion of the entire system and the software itself is becoming more complex and therefore more costly per line of code. The ratio of the cost of software to hardware seems to increase as the systems reliability requirements increase. This is due to the higher cost of validation and verification (V&V) of software. There appears to be an incompatibility with system reliability requirements on the order of one in a billion when accepted software ability is on the order of one error per one-thousand lines of code. This is somewhat of an apples and oranges comparison, but it does give some indication of the problem. As software complexity increases so will the chance for errors. There are many reasons for the increasing complexity of software. One of these reasons is that, in systems that do not have a high reliability requirement, software ware can provide more functionality at a lower cost. When the cost of V&V is included, however, this is not true for ultra-reliable systems. Another reason for the increasing complexity of software is the fact that in most systems the hardware design and build is completed before the software is written and tested. This creates a tendency for the hardware designers to design hardware that meets the minimum requirements. When design decisions are difficult or when the requirements are not well specified, hardware designers tend to "leave it to the software". We decided to avoid this trap by trying to solve as many of the complex problems as possible with hardware.

Some of the most complex parts of a system design have to do with its reliability. We noticed that two-thirds of the software for many of the systems that we studied had to do with redundancy management and the reliability of the system. There have even been systems where 95% of the software was concerned with reliability. These reliability areas appear to be the most fruitful place where we can replace software with hardware.

Another influence of the shrinking microelectronics is caused by the fact that the central processing unit (CPU) part of a computer has shrunk faster than the other parts. Because the CPUs are smaller, draw less power and are cheaper, it is now feasible to include a large of CPUs in a system. This larger number of CPUs can be used to both speed up the system for real-time requirements and to increase the systems reliability.

Systems Approach - We decided to use a tops-down systems approach with the MMFCS. This seems to be a motherhood statement and the obvious approach, but it is not. Many endeavors in the area of designing ultra-reliable systems have seemed to equate the designing of a fault tolerant computer with the design of an ultra-reliable system. A fault tolerant computer is not the total system solution.

We can view an ultra-reliable system as containing a protected environment. The system is designed to continue correct operation even though failures have occurred within the protected environment. A system that contains a central fault tolerant computer has a protected environment that includes only the computer. The systems approach is to try to expand this protected environment to encompass as much of the system as is possible. This leads to a system that is distributed physically, electrically, and logically.

INFORMATION TRANSFER SYSTEM (ITS)

The distributed environment that was dictated by our systems approach has led us to the view that communication rather computation is the heart of a system's reliability. We can imagine a group of very reliable processors trying to cooperate on a given task by communicating over an unreliable communications system. This system will not be reliable. On the other hand, a group of unreliable processors cooperating over a reliable communications system can be made reliable. Note that this is analogous to a reliable device being built out of unreliable components.

We have therefore concentrated our efforts in providing reliable communications. This is accomplished through an Information Transfer System (ITS). The ITS is the "glue" that binds all the system's resources together. The ITS includes busses, buss interface units, protocols, information management, and applications interfaces. The ITS must be able to provide communications to the degree of reliability that is required by the most reliable subsystem that it is connected to. It must do this while at the same time servicing subsystems that do not have the same high degree of reliability. This means that the ITS must be able to protect itself against subsystems of lesser reliability.

The ITS is the only interface between the subsystems. It must be capable of transferring all information that is shared between these subsystems. Therefore, the ITS not only communicates data but also communicates timing and syncronization information as well.

FAULT TOLERANT CONSIDERATIONS

It is well known that redundancy is required to make a system fault tolerant. In addition, the redundant

elements of a system must be made independent from each other. If these redundant elements are not made independent, then a failure in one of these components may propagate to all other components, thereby making the redundancy useless. This concept is called fault containment. It is not possible, however, to make the redundant components totally independent. They must somehow share inputs and cooperate on outputs. To enable this cooperation, we must share something between the redundant elements. Whenever we share something between the redundant elements, we are building bridges between the areas of fault containment. These bridges, then, can become ways that faults can propagate from one redundant element to another. Wherever one of these bridges exists between fault containment areas, we must place guards at both ends of the bridge to ensure that faults do not propagate over the bridge. This is called fault isolation.

The concept of fault containment areas connected by bridges can be applied recursively. This leads to a hierachy of fault containment. This hierachy allows optimization in the cost/containment tradeoff.

The design of a fault tolerant system, can be reviewed as composed of three steps. First, designing adequate fault containment barriers. Second, identifying all possible bridges between containment areas. Third, the placing of adequate guards at both ends of all of these bridges.

An important aspect, that is very often forgotten, is that guards must be placed at both ends of the bridges. This is because the guard at the end of the bridge, that is in the containment area which has experienced a fault, cannot be trusted. Whatever the fault was, it may affect the guard.

SYSTEM DESCRIPTION

Self Checking Microprocessor Pairs (SCMP) - All the components in the MMFCS architectures are pairs. This means that every component has one duplicate. The duplicate halves of a pair receive identical input and (with no failures) produce identical output. The outputs of both halves are checked to make sure that they are identical. When they are not identical, the pair removes itself from the system.

The overall system architecture looks like an N-modular redundant system. Each channel is composed of a buss pair that supports nodes that are pairs. The nodes on any given buss pair have the right to transmit and receive on that buss. In addition, nodes have the right to receive data from any buss in the system.

Each half of a node is composed of an applications processor and a buss interface unit processor. The applications processor (AP) performs the algorithms to exercise the control for which the system was designed. The buss interface unit (BIU) is used only to perform the communications between APs of the various nodes. These BIUs and the busses make up the ITS for the MMFCS architectures.

A typical transfer from one node to another would start with the two APs in a given node performing some calculation and producing an output. This output will be sent from the AP to its associated BIU. The BIU converts this data into a serial data stream. The data streams from the two BIUs are compared, bit for bit, to ensure that they are identical. If they are identical, the data is transmitted over the buss to all the nodes that are listening to that buss. When the data arrives at any of the other nodes, data is again checked to make sure that both copies are identical. Note that the data is checked when it is transmitted on the buss and when it is received from the buss. This

corresponds to the output and input guards on the bridge.

If a node should have an internal failure it will cause the two copies of the data to be not identical. When it is time to transmit this data on the buss, the output checking will determine that the two halves are not identical. This will cause the node to disconnect itself from the buss. This action is taken by both halves of the pair. The half, that does not have a failure, will be the force that actually effects the disconnect.

The functions of any node in the system is replicated by nodes on the other busses. Therefore, when one node disconnects itself from its buss, there are other nodes on other busses that are performing the same function. No system functionality is lost due to this disconnect.

These self-checking nodes are the simple modules in the MMFCS architectures. The buss forms the simple interface.

Consistency - Any architecture that uses exact comparisons to detect failures must ensure that inputs to the redundant components are identical. If the inputs are not identical, then the outputs probably will not be identical. The output checker will then assume that one of the redundant components has failed. This will be an erroneous deduction. Hence, the importance of insuring that the inputs are identical. This problem has been called interactive consistency or source congruency by others in the literature.

This is not as easy a problem to solve as it first appears. A simple wire connecting the inputs of redundant components together does not ensure identical inputs will be perceived by these redundant units. A borderline signal on the wire can be perceived as different signals by the varying tolerances within the redundant components. The only method of ensuring what each redundant unit received as input is for that unit to retransmit the data that it received. This turns out to be a complex procedure requiring a very high communications bandwidth, particularly when the number of redundant units becomes large. The self-checking pair approach of the MMFCS architectures minimizes the complexity and bandwidth required to perform this function.

Table Driven Proportional Access (TDPA) - There is a problem with the ITS use of communication busses. This is the same problem that must be faced by all bussing schemes. The problem is to allocate sufficient buss bandwidth to each of the buss connections without mutual interference. The MMFCS ITS does this through a technique called Table Driven Proportional Access (TDPA).

In determining how to control the message traffic over the buss we considered the classes of data that this communication scheme must support. One criteria for classifying data is the timeliness required. We could develop three categories for timeliness: routine, no time requirements; fast, arriving not to late; real-time, arriving not to late and not to early. We can also classify the data as to whether it is critical for correct operation or not. The most stringent requirements are on the real-time critical class of data. By designing a buss bandwidth allocation scheme that can handle the real-time critical data we would have a scheme that would probably handle any of the other classes of data.

The scheme chosen for the MMFCS, TDPA, can be thought of as a cross between virtual token passing and time division multiplex. The scheme is based on having an access table that is replicated within each node. These tables contain entries that indicate which node has permission to transmit on the buss at any given time. Each node maintains an internal pointer to the entry in the table that is current-

ly governing the buss. When the current message is finished, each node advances its pointer to the next entry in the table. Because each of these tables is identical, the new entry in the tables will tell each node who will have control of the buss for the next time segment. By this means, each node can be guaranteed an exclusive use of a particular time on the buss.

Because this scheme is to be used in ultra-reliable systems, it must work even though there are faults in the system. The TDPA does this by allocating a specific amount of time for each entry in the table. If a node should fail to transmit during its time slot, all the other nodes would advance their pointers at the end of the slot time anyway. This allows the TDPA scheme to continue correct operation even though all but one node has failed. The use of self-checking BIUs insure that the only buss control failure will be the failure to transmit. Proprietary schemes for the start up of the buss and the maintenance of the individual pointers have been developed.

Synchronization - There is a contining "great debate" on the issue of the synchronization of fault tolerant systems. Most arguments tend to treat this question as a black or white issue. But the degree of synchronous or asynchronous operation is actually continuum. The range can be from the very tightly synchronized, where the processor clocks are fed by the same oscillator, to the very asynchronous in which the synchronism is supplied only by the external closure of the real-world physics. Furthermore, all components of the system do not have the same degree of synchronism. The arguments on synchronism can then be reduced to picking the best degree of synchronism to balance the system performance against a set of criteria for each component within the system. For example, the MMFCS coprocessors within a node are synchronized to the microcycle level, the halves of a node are synchronized to the serial bit level (when they are transmitting), nodes on a buss are synchronized to the serial message level, and other components of the system are made to be synchronized to the level that is required.

Executive - The executive software for the MMFCS is very simple. There are no interrupts and there is no multitasking. Each process is allocated to one processor. This greatly simplifies the software V&V.

The task loading on each processor and the message loading on each buss is known a priori. There are no variable task cues, no pending interrupts, no variable buss message cues. The system is very deterministic.

The control of the execution of tasks within the distributed processing nodes is done by the action of the buss itself. This eliminates the need for interrupts, multi-tasking, rate trees, etc. This can be shown to be an optimum method for controlling the execution of distributed processes, both in buss bandwidth and fault containment.

BREADBOARD

Honeywell has built and demonstrated a breadboard of these MMFCS concepts. We have demonstrated that the system works correctly without faults. We have also shown that the system operates correctly when faults are induced into the system.

IUS DESIGN FOR FAULT TOLERANCE

J. H. KEENEY
SYSTEM ASSURANCE MANAGER

F. E. PERRY
HARDWARE DESIGN MANAGER

IUS PROGRAM
BOEING AEROSPACE COMPANY
KENT, WASHINGTON

INTRODUCTION - The design challenge for the IUS program was to produce an upper stage vehicle that would provide mission reliability well beyond that of previous vehicles. Mission reliability was to be carried as far as possible or to the point that additional improvements were no longer cost effective. Since the payloads the IUS was to carry are much more expensive than the IUS itself and there was no flight test program of the vehicle, fault tolerance and fault elimination were paramount design considerations. This paper tracks IUS design activity from the point in time where subsystem trades had matured to the state that individual components were identified. Five topics have been selected that identify the salient considerations in the IUS Fault Tolerant Design (Chart 1) as follows; (1) Cost Effective Redundancy, (2) Design for Autonomy, (3) Significant Margins, (4) Fault Tolerant Features, and (5) Test Feedback. Each of these will be discussed in the following paragraphs.

1. COST EFFECTIVE REDUNDANCY - Once the system design for IUS had progressed to the point that the subsystems and components could be identified , the resulting mission reliability fell short of that desired by the Air Force. A mission reliability requirement of .96 with a goal of .98 (Chart 2) had been established and a single thread reliability was estimated at .815. The question of how much reliability is needed and how to obtain it was answered by trading cost vs. redundancy. Reliability could be improved by redundancy and other fault tolerant approaches but at the same time costs would go up not only from the additional direct hardware costs but also from the counter productive effects of additional weight, volume and power requirements. It was believed that from the relationship of program cost of failures vs. the cost of generating fewer failures, an optimization cost figure of merit could be developed that would answer the question, how much is enough?

An IUS cost impact model was developed which included the salient cost elements along with the results of performance analyses conducted that detailed the IUS/Shuttle missions needed to place the spacecraft identified in the mission model into their respective orbits. Tandem IUS flights were groundruled out although multiple spacecraft on a single IUS were permissible. These studies resulted in IUS/Shuttle launches involving 14 different IUS payload configurations of from one to four spacecraft each.

A model (Chart 3) was constructed based on a Markov chain analysis of the mission accomplishment, abort, and mission failure logic. The model evaluates 10 different success paths with replacement launches for each IUS mission failure. For a given IUS payload configuration, the program cost for one IUS mission is evaluated and multiplied by the total number of IUS missions of that payload configuration. The process is repeated for all 14 IUS payload configurations. The resultant cost plus the IUS development cost is the IUS program cost.

As each new concept of reliability improvement was envisioned, its attendant cost on IUS recurring and non-recurring costs was estimated and updated in the model. The cost of the spacecraft and shuttle launch costs were provided by the government. However, for each IUS vehicle configuration, the abort and mission failure probabilities and the attendant IUS development cost and production costs were determined. The development and production costs were determined for the entire mission model quantity.

As shown (Chart 4) an optimum did exist and any additional improvement in IUS reliability through major configuration changes would not be cost effective. The elimination of the second sring TT&C and the third computer were difficult decisions but the Air Force agreed and stood by their original cost effective consideration for decision making.

Sensitivity studies were performed and showed that IUS program costs are most sensitive to variations in spacecraft unit costs. Therefore, IUS development costs that might lower spacecraft costs were good investments. A 25 percent increase in the unit costs for all spacecraft will increase the total IUS program cost by 14 percent. Variations in IUS development costs have the least effect. A 50 percent increase in development costs will increase the total IUS program cost by only 0.3 percent. This small impact was expected because the IUS development cost is nonrecurring and constitutes a small portion of the overall total program cost. For comparison, the cost of a replacement mission for one IUS failure could vary from 0.3 to 0.8 percent of the total program cost, depending on the payload. This comparison does show the cost effectiveness of money spent during development to ensure an adequate design.

The IUS program cost model was also used to evaluate the cost effectiveness of the IUS baseline reliability for different unit and development cost variations. The worst-case condition considered included a 25 percent increase in the cost of all spacecraft, a 50 percent increase in IUS unit cost, and a 50 percent increase in IUS development costs. For these conditions, the IUS optimum mission reliability of 0.983 was still the most cost effective value.

The resulting system architecture (Chart 5) shows in simplified form the resulting redundancy and cross strapping for the vehicle. This architecture was established during the validation phase of the program and has not been altered since.

2. DESIGN FOR AUTONOMY - The IUS hardware/software is purposely designed to perform the entire mission with no intervention. This is completely true for

the Titan configuration which doesn't have an uplink capability and also true for the Shuttle Configuration Vehicle after shuttle deployment. Thus, navigation, guidance, staging, and event sequencing are all under software control which is loaded prior to the mission. Each computer has a 64K memory and approximately 19% of that is allocated to redundancy management, checkout and test. Computer failure modes and interrupts used for redundancy management are as shown (Chart 6).

The Shuttle configuration does require some astronaut intervention to perform the deployment functions. The crew has a control panel inside the crew compartment which allows performance of the deployment task and monitoring of system operation. Basically, the crew elevates the IUS to 29° for checkout and then to 58° for deployment. Actual deployment is accomplished by the mission specialist who fires the ordnance device that separates the IUS and its payload from the Airborne Support Equipment. Six springs then gently push the IUS away from the orbiter at a 0.4 FPS rate. From this point on, the entire mission is autonomous. The orbiter vehicle will then back away from the IUS, attitude control begins in ten minutes and motor ignition will occur some 55 minutes after deployment unless the crew sends a master safe command.

Vehicle Uplink Commands are possible with the shuttle version which provides additional interrogation and status monitoring. This uplink also allows some control alteration if problems develop, that can be worked around in this manner. Uplink Commands are emergency situations and are pre-planned and practiced events for the most part. These are in turn simulated many times prior to the actual mission.

In summary, the vehicle needs no ground link to perform its mission and responses to faults are built into the software. However, if emergency situations do arise, the ground has the capability to workaround for the shuttle missions. This provides for both a fault tolerant and fault responsive design.

3. SIGNIFICANT MARGINS - Design margins are another aspect of designing for fault elimination and fault tolerance. Ample margins can make designs less prone to fail and at the same time more tolerant to failures if they should occur. Stress is the root of failure and this section discusses both electrical stress and environmental stress.

Piece part electrical stress (e.g., current, voltage, power or fanout) can be tolerated or not, depending on temperature. Failures can be reduced or tolerated based on where the electrical component is operated within the stress versus temperature profile. The prescribed operating limits are known as derating. Derating is generally arrived at by reducing the manufacturer's (suppliers) maximum allowable data by some safety factor (Chart 7), generally in the range of 30° to 40° C. This in turn establishes the worst case limit for design. On IUS this was taken a step further and an additional uncertainty factor of 11°C bounded the upper end of the electrical stress vs. temperature profile. Examples of derating for several part types are shown (Chart 8). The specification of, testing and screening of parts to preclude failure is a subject not covered in this discussion. Suffice it to say that parts control on the IUS program were the most stringent requirements issued by government and accepted by industry in the 1978-1980 time frame.

Derating is a design discipline and was policed by reviewing individual piece part stress analysis. If a part crept into the disallowed region, it was subject to an immediate redesign. If redesign was not practical, individual derating exceptions were only granted by close scrutiny of the government/industry parts, materials, and processes control board. Some parts were approved on an interim basis until such time as the supplier could redesign and deliver a more tolerant part, and then all prior parts were purged from stores.

Environmental stresses as a design consideration were carried through part, component and system levels and encompassed; vibration, shock, thermal, pressure, humidity, acoustic and EMI. In general the design and test concept on IUS was to predict the worst case mission environment through analysis and/or prototype testing and then add an uncertainty margin to that before establishing the qualification margins. This process followed the requirement of MIL-STD 1540A.

Three environments are illustrated (Chart 9) that show this concept. Random vibration acceptance requirements were established by determining the composite of the various individual responses and then drawing a curve that envelopes the composite. A six db margin was added to that for qualification. A similar process was followed for the shock environment. The thermal environment was established for the worst case mission and then 11°C added to that for uncertainty to establish the acceptance envrionment. Later 10°C was added to that to establish the qualification limits. The curve shown is that for thermal cycling; however, thermal vacuum testing was accomplished in a similar manner.

Additional design margins were established for structures design with safety factors as follows:

Pressure Vessels -	4:1 MEOP
Limit Load -	1.0
Ultimate Load -	1.4

Also, dynamic loads included a 1.5 uncertainty factor.

4. FAULT TOLERANT FEATURES - As seen from the previous discussion, IUS early life-cycle cost studies indicated that extensive redundancy in the avionics and attitude control subsystems would be cost effective. Redundancy was not considered necessary for structural elements, because of high predicted reliability, and was not feasible for the solid rocket motors. In addition to redundancy, certain other features were included in the avionics to enhance reliability. This section addresses the overall IUS system architecture, the failure detection and isolation methods used, and redundancy management for the major subsystem elements.

The IUS avionics system architecture in simplified form (Chart 10) is basically a two channel system with the exception of the Redundant Inertial Measurement Unit (RIMU). Here, a strapped-down configuration, with redundancy at the instrument level is used. The avionics system in more detail at the Line Replaceable Unit (LRU) level is as shown (Chart 11). The following sections discuss each of the major LRUs in terms of features that enhance reliability and on-board automatic redundancy management.

Redundant Inertial Measurement Unit - The Hamilton-Standard RIMU design utilizes five Kearfott 2401 single-axis linear accelerometers and 5 Hamilton Standard RI-1010 single-degree-of-freedom rate integrating gyros. The sensor input axes are arranged symmetrically in a skewed conical array as shown (Chart 12) in the photograph. Locations relative to the vehicle axes are illustrated on the next chart (Chart 13). In the event of a sensor failure the navigation function can be performed by a trio or quartet of instruments.

Secondary power supplies and sensors are arranged in three independent channels (Chart 14). Loss of any channel thus permits the system to continue to function. Data from each channel is sent simultaneously to both computers.

Hardware Built In Test Equipment (BITE) and computer software are used to detect component or subassembly failures. Critical voltages, temperatures, loop closure discretes, gyro spin motor rotation, and data transmission errors are monitored by BITE.

The greatest challenge is in detecting "soft" failures. Small errors, if propagated throughout a mission, could reduce injection accuracy. The system uses both real time data and longer-term filtered data, sampled at three different computational rates, to effect a suitable compromise between detection probability and false alarm rate.

The failure detection and isolation (FDI) algorithm makes use of the known orientation of the sensor axes. This permits linear dependence equations to be written for sensor outputs. These define equivalence, or parity, of the sensor outputs. A failed sensor will cause a deviation from the expected parity. Threshold checks can then be used to isolate a faulty sensor. Sensor data is transmitted to the computers in serial data form. Transmission errors are prevented by parity checks, the use of pattern words and check sums. Operational reliability is further enhanced by software reasonableness checks that limit the magnitude of a RIMU update to preclude false inputs.

Computers - System trades during the IUS validation phase compared the use of three computers, in a "pair-and-spare" configuration to dual computers using self-check as a means of failure isolation. In the "pair-and-spare" configuration, outputs of the computer pair are compared, and in the event of a miscompare, system control is switched to a "hot" -spare computer. Essentially 100% error detection is possible. For the expected range of semiconductor operating temperatures, however, two computers were found to have a life cycle cost advantage (Chart 15) in addition to weight and space advantages.

Memory protection is included in the computer design by the use of Hamming codes. Any single bit error is automatically corrected, while two bit errors in the same memory word are used to set a failure flag which causes redundancy management to select the other computer. Non-memory elements of the computer utilize a series of instructions for a self check that would ideally check all states of all gates. Practicality dictates a compromise between self check coverage and the number of instructions used. For the IUS avionics, reliability goals were met with 2,500 instructions, providing a self-check coverage of non-memory elements in excess of 70 percent. A complete self check is performed every 120 milliseconds. Built in test equipment, plus the self test program, permit computer failure detection and isolation.

Star Scanner - The Ball Aerospace star scanner used for IUS is internally redundant except for the optics. The optics use only passive devices, and the expected reliability does not warrant the use of redundancy. The remainder of the unit, including the detectors, is dual-channel.

The star scanner updates RIMU attitude data in three inertial reference axes. This is done by scanning a star pair approximately 90 degrees apart. Coincidence pulses are used as interrupts to the computers as the detectors scan past a star.

Redundancy management makes use of a health check performed on the unit prior to a star scan. Two light-emitting diodes inside the unit are energized briefly to simulate stars and the magnitude of the output pulse is measured. Both detectors view both diodes, which are energized one second apart.

Signal Conditioner Units (SCUs) - The signal conditioner units perform a variety of tasks, including some input-output functions normally performed in computers. Functions include command decoding, analog-to-digital conversion, data formatting, digital-to-analog conversions, relay drivers, and reaction engine valve drivers. Switching of other system elements for redundancy management purposes is controlled by logic circuits within the SCUs, based on inputs from the computers. Commands to the SCUs from the computers are in serial digital form. Each command consists of a 32 bits plus parity. To avoid transmission errors, the message is echoed to the computer where a comparison is made. The computer then sends a "message valid" pulse to the SCU. Each computer also sends an OK or not-OK flag to the SCUs every 10 milliseconds, based on computer BITE and program self check results previously discussed.

Malfunctions in reaction control valve drivers are detected by wrap-around tests, wherein the signal to the valve is sampled by the SCU multiplexer, converted from analog-to-digital form, and routed to the computer where it is compared with the command. A similar test is performed for certain relay drivers. In this case, however, operation of the relays themselves is checked by sensing the closing of a set of contacts on the same relay. It was not necessary to check all relays and motor-driven switches in this manner, when failure to operate one would initiate other events that would cause system reconfiguration.

Reaction Control Subsystem - The reaction control subsystem controls vehicle attitude about all three axes during the coast phase, and about the roll axis during solid rocket motor burn. Tolerance for two types of faults is necessary: (1) A reaction engine valve failed "open," which would deplete the fuel supply, and (2) A valve or engine failed "off," which would cause loss of control. To avoid valve open failures, two series valves are provided for each engine; thus the system tolerates any single failure, and multiple failures not associated with the same engine. To preclude mission loss from a single engine or valve failed "off," redundant engines are installed for each axis, for a total of 12 as illustrated (Chart 16). Redundancy management is passive, in that no reconfiguration is required if an engine fails. The two engines for each direction of each axis are pulsed alternately. As illustrated, if one fails, the attitude control system senses that the dead zone limit has been exceeded, and the other engine of the pair will be pulsed on in the next 40-millisecond computation cycle.

Each of the 24 valves in the system can be driven by either SCU-A or SCU-B through diode "or" circuits.

Thrust Vector Control (TVC) Subsystem - The thrust vector control system steers the vehicle during solid rocket motor burn by moving the nozzle in the pitch and yaw planes. The servo system consists of an electronic controller and ball-screw actuator, with position potentiometers as feedback elements.

Fault tolerance is achieved by redundant electronic controllers for each axis and dual motors on a single shaft in the actuator assembly (Chart 17). Dual solenoid-operated locking gears prevent ball screw rotation when the system is off during coasting flight. Energizing either solenoid will unlock the actuator.

The performance of the TVC subsystem with channel A in control, is monitored by the computer in that channel. Actuator position is compared to that commanded with actuator rate and acceleration checked against pre-set thresholds. Failure to meet these criteria results in transfer of control to channel B.

Telemetry, Tracking and Command (TT&C) - The IUS vehicle was designed and qualified using a redundant TT&C subsystem, which is available as a factory option. No user to date has chosen that option since the subsystem is not mission-critical. Redundancy management for the dual system is accomplished by ground command for all failures except reduced R. F. output power. Automatic system reconfiguration would occur if the nominally 20-watt R. F. output dropped below 12 watts.

General - The IUS avionics is basically a two-channel subsystem with minimal cross strapping of subsystem elements. Obviously, many more success paths through the system would be possible if cross-strapping was used. That system architecture was not selected because the reliability of the non-cross-strappped system was the most cost effective approach because system complexity is greatly reduced. Further cost avoidance results from elimination of the multitude of test combinations that would be necessary with cross-strapped elements.

The system is initialized at power-up with channel A in control. Upon detection of a single failure, control is transferred to channel B. In the event channel B fails, no provision is made for returning to channel A, as it is assumed that a hard failure exists in that channel. Except for certain computer errors, faults are checked two or more times before switchover to minimize response to false alarms.

5. **TEST FEEDBACK** - Since the IUS was to have no flight test program it became more important to structure a ground test program to prove the design and its ability for fault tolerance. The test program was conducted in accordance with MIL-STD-1540A which requires that qualification hardware successfully pass acceptance testing prior to formal qualification testing. This discussion will concentrate on the qualification program and how the design responded to those rigors. Specifically, the questions of how well our redundancy management system performed upon failure and what failures had to be resolved by design modification will be addressed.

Component level acceptance and qualification tests followed the flow as shown (Chart 18). The pyro shock/vibration screening test (sometimes referred to as an alligator test) includes pyroshock in the most sensitive axis followed by 3 axis vibration with power on. The screening vibration duration was 5 minutes sustained and then 20 bursts of 10 seconds on and 10 seconds off, hence the nickname "alligator." Since the qualification units were subjected to thermal cycling at both acceptance and qualification levels, burn-in testing was not performed.

Every failure or significant non-conformance during the qualification program was subjected to a formal Failure Review Board consisting of the chief engineer as chairman and various other managers as members. The Failure Review Board reviewed every problem and concurred on the cause and corrective action prior to submitting a Failure Summary Report to the Air Force and NASA for concurrence. This process assured that the appropriate corrective action was taken, and the authority of the chief engineer made it happen. As will be shown from the statistics, our chief engineer was not timid about recommending design fixes, or other changes where this was the appropriate action.

System level qualification tests were conducted both at the vehicle level and the ASE level and then combined for EMC/Tempest and acoustic environments (Chart 19). The Shuttle configuration vehicle was selected as the qualification test vehicle because it also used the ASE (Airborne Support Equipment). Any vehicle level configuration differences between the Shuttle and Titan versions were qualified by component level, subsystem level testing or analyses.

Results of the qualification program in terms of failures and significant non-conformances are as shown (Chart 20). As expected, the preponderance of the problems occurred in the Avionics equipment and it should also be pointed out that although 10% of the problems were in the propulsion area, there has never been a development or qualification failure of a solid rocket motor. Most of these propulsion

problems were in the reaction control system. The number of problems occuring at the system level qualification significantly dropped over that experienced during component level (approximately 16%) as expected. However, it should also be pointed out that this could have been reduced even further had there been program schedule slack which would have permitted final design changes implemented in the system test components that were identified during component level testing, prior to performing system level qualification testing.

This data (Chart 20) also shows that although there were a significant number of qualification problems, only 13 percent of these were mission critical, i.e., would have caused mission failure. This was because the redundancy already designed into the system merely switches to an alternate success path. Major failures refers to the scenario that it now takes a second failure of the same item to cause a mission failure.

Another point, also shown on this same data, is the magnitude of corrective actions implemented by changes. Better than 80 percent of all qualification failures resulted in a formal design change either to the hardware, process specifications or procedures. These changes, of course, ranged from complete redesign of a mechanical assembly to the addition of staking compound under a piece part. Each, however, was reviewed and approved by the Failure Review Board. The other 20 percent were human error, workmanship and piece part failures and are not addressed here.

SUMMARY - This paper presented only a summary of the IUS design for fault tolerance. All facets of the design and testing for fault elimination and mission assurance are much too vast for one paper. The authors believe that the five topics presented are the salient subjects when addressing design features. One might ask, what about mission results of the two flights thus far and to that the following is offered.

The first IUS flight took place on October 30, 1982 on a Titan launch vehicle. The mission was a complete success and the IUS placed two spacecraft in geosynchronous orbit with the following orbit injections errors; radius 6 nmi, drift rate 0.0567 degrees per day and inclination, 0.0175 degrees. This mission experienced telemetry downlink problems; however, the autonomous design functioned perfectly.

The second IUS flight took place on April 4, 1983 on a shuttle (STS-6). This mission was not the success that the first was but did prove correct some of the design decisions discussed in this paper. As of this writing necessary energy changes are slowly being made to place the spacecraft in a geosynchronous orbit.

- **Cost Effective Redundancy**
- **Design for Autonomy**
- **Significant Margins**
- **Fault Tolerant Features**
- **Test Feedback**

Chart 1. IUS Fault Tolerant Design

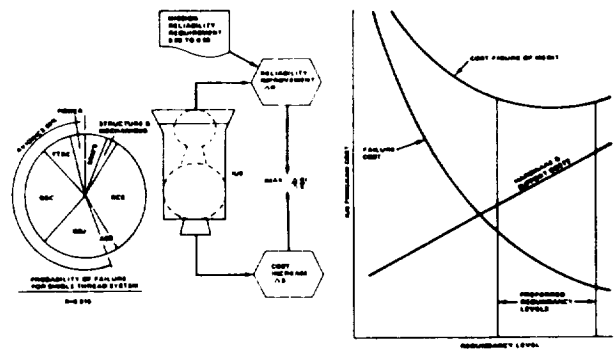


Chart 2. Redundancy Selection by Cost Minimization

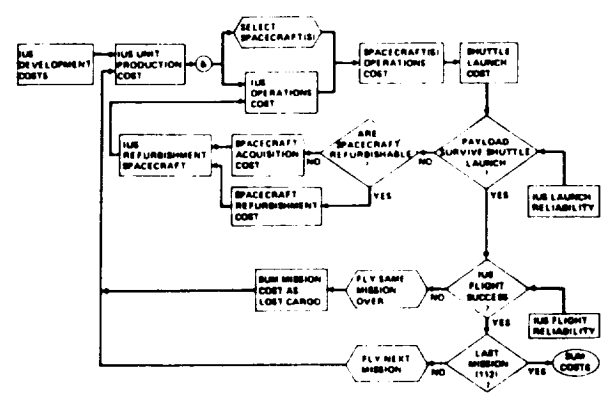


Chart 3. IUS Program Cost Model

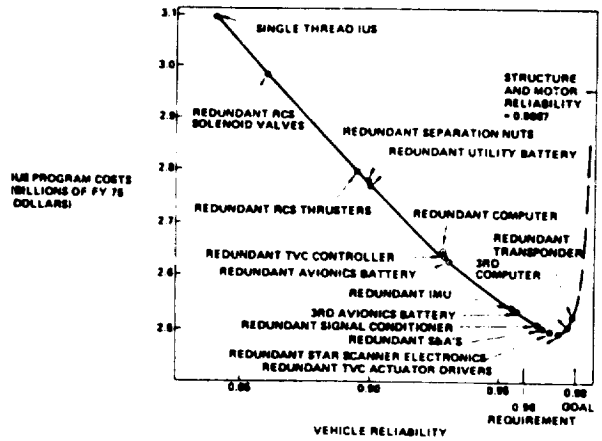


Chart 4. Cost-Optimized System Architecture

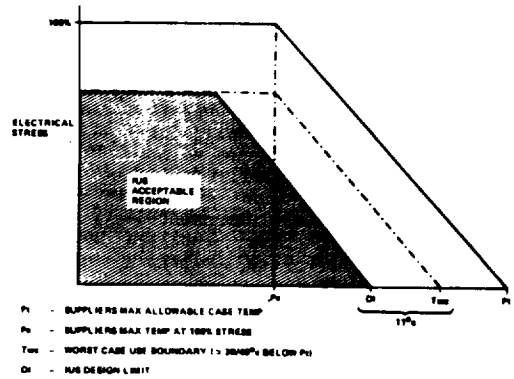


Chart 7. IUS Part Derating Methodology

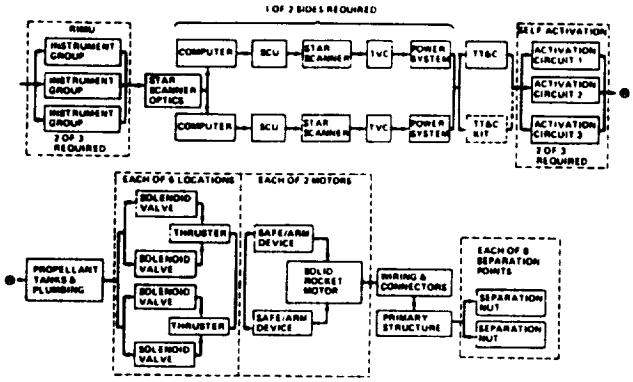


Chart 5. IUS System Architecture

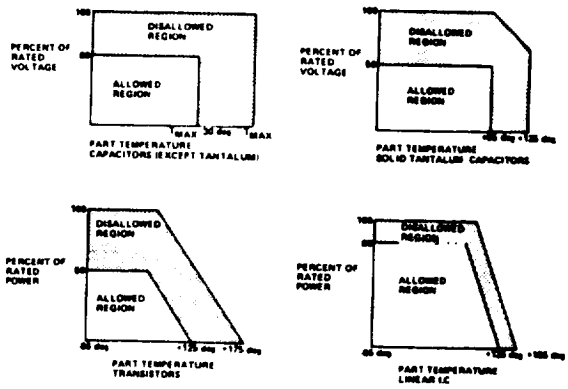


Chart 8. Conservative Design Derating Process

Failure Mode	Computer Interrupt
Memory Error	Hardware
Illegal Memory Store	Hardware and Software
System Synchronization Failure	Hardware
Computer I/O Failures	Hardware and Software
GMT Input Circuit Failure	Software
Interrupt Processing Failure	Software
Measurement Data Failure	Hardware and Software
Control Status Failure	Software

Chart 6. Computer Failure Modes and Interrupts Used for Redundancy Management

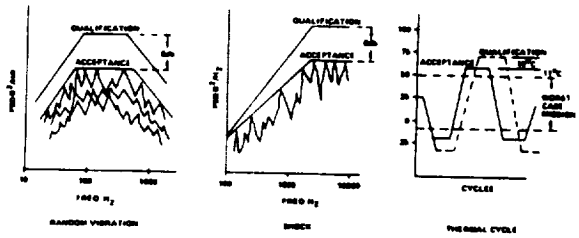


Chart 9. Component Qualification Margins

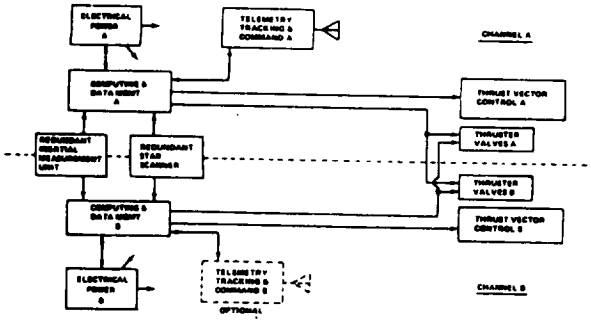


Chart 10. IUS Avionics Redundancy

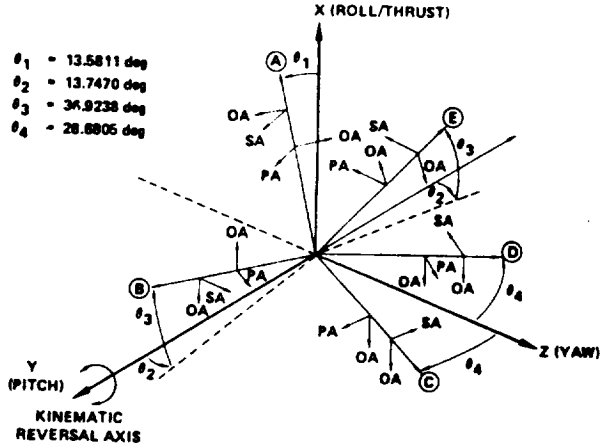


Chart 13. Skewed Conical Sensor Axes Orientations

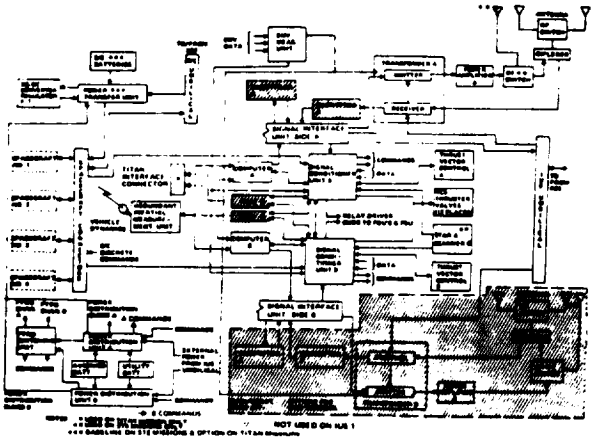


Chart 11. IUS Avionics Redundancy System Block Diagram

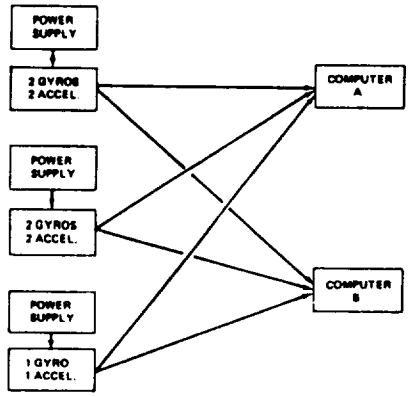


Chart 14. Inertial Measurement Unit Instrument Redundancy Configuration

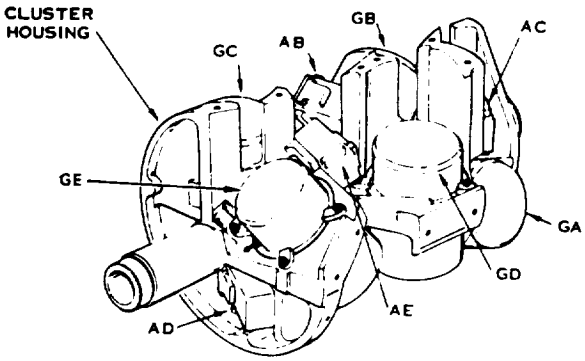


Chart 12. Redundancy Inertial Measurement Unit Instrument Cluster Arrangement

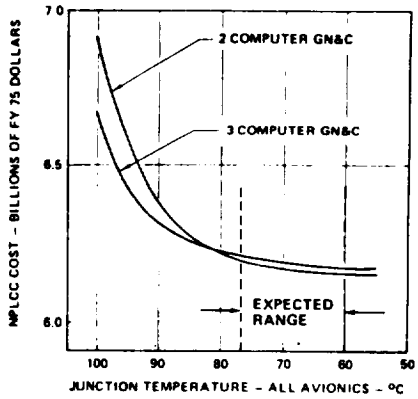


Chart 15. Impact of Two Versus Three Computers on National Program Life Cycle Costs

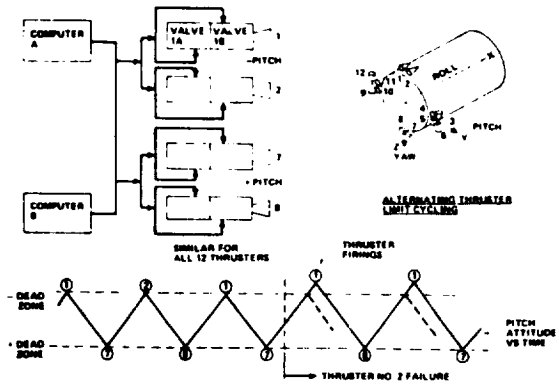


Chart 16. Coast Attitude Control System Redundancy

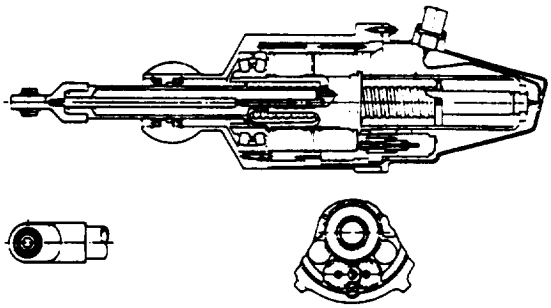


Chart 17. Actuator Assembly - TVC

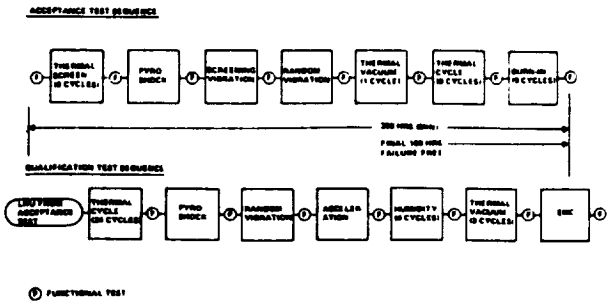


Chart 18. Component Level Qualification Tests

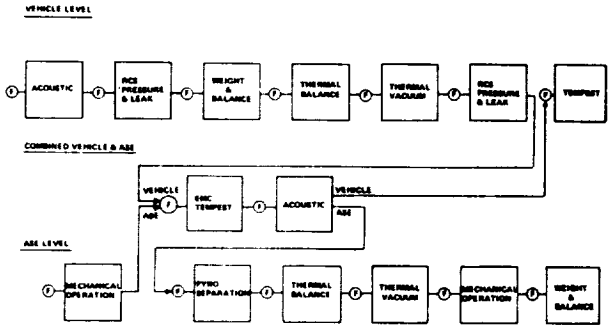


Chart 19. System Level Qualification Tests

SYSTEM	COMPONENT LEVEL TESTING	SYSTEM TESTING VEHICLE	SYSTEM TESTING ASE	TOTAL
AVIONICS	212	17	2	231
ELECTRICAL	62	14	2	78
STRUCTURAL MECHANICAL	34	1	30	65
PROPULSION	41	1	-	42
TOTAL	349	33	34	406

70% OF FAILURES WERE ELECT/ELECTRONIC

NO SOLID MOTOR FAILURES

FAILURE EFFECT:

MISSION CRITICAL	45	7	1	53
MAJOR	123	10	8	141
MINOR	181	16	15	212

ONLY 13% OF FAILURES

FAILURE RESOLUTION:

HARDWARE CHANGE	176	10	11	197
PROCESS CHANGE	103	11	7	121
PROCEDURE CHANGE	3	4	2	9

81% OF FAILURES RESOLVED BY DESIGN CHANGE

Chart 20. Qualification Program Summary
(Failures and Significant Non-Conformances)

GALILEO ORBITER FAULT PROTECTION

Mission Assurance Through Spacecraft Autonomous Action

Peter M. Kobele
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California 91103

Summary

Mission imposed requirements for reliable operation for long duration deep-space missions, the existence of mission critical activities for which any required ground-based intervention is over an hour distant, and operational constraints which result in completely unattended spacecraft operations for periods of up to a week have dictated the implementation of autonomous, generally software intensive fault detection and correction processes in the design of recent JPL spacecraft. Planned for launch by the Shuttle/Centaur combination in the late spring of 1986 as the logical follow-on to the Voyager flybys of Jupiter in order to perform an intense and comprehensive exploration of the Jovian system, the Galileo orbiter is the latest, and in some respects, the most sophisticated of these spacecraft.

This presentation will provide an overview description of the Galileo orbiter and its fault protection system, as well as some discussion about the rationale which drove some of the fault protection design features. It is hoped that these observations regarding some of the difficulties that were encountered during the fault protection design process as well as the challenges identified for future missions will prove instructive.

FAULT PROTECTION SYSTEM - GENERAL

PURPOSE : MAINTAIN CAPABILITY TO ACCOMPLISH MAJOR MISSION OBJECTIVES IN THE PRESENCE OF MOST FAULTS

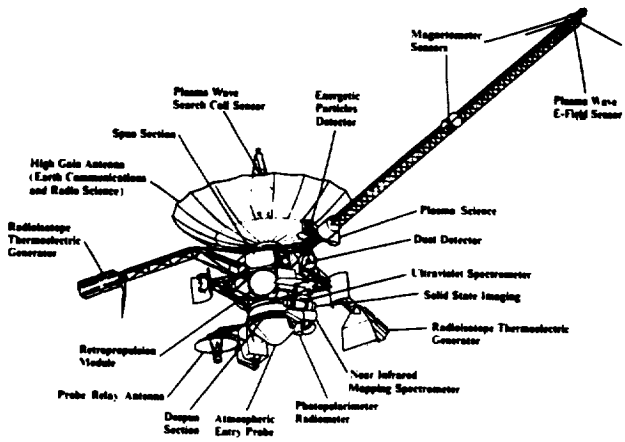
CONSTITUENTS :
- SPACECRAFT SYSTEM AND CONFIGURATION DESIGN
- MISSION OPERATIONS SYSTEM
- ON-BOARD FAULT DETECTION AND CORRECTION ELEMENTS
  - o SENSORS
  - o HARDWARE LOGIC CIRCUITS
  - o DEDICATED FAULT PROTECTION SOFTWARE

GALILEO ORBITER SUMMARY

MISSION : OBJECTIVES
- o EXPLORATION OF THE JOVIAN SYSTEM WITH EMPHASIS ON THE GALILEAN SATELLITES
- o SUPPORT PROBE OBSERVATIONS OF UPPER ATMOSPHERE

PHASES
- o LAUNCH IN SPRING 1986 ON SHUTTLE/CENTAUR
- o INTERPLANETARY CRUISE
- o JUPITER ARRIVAL - AUG 88
- o ORBITAL SATELLITE TOUR

SPACECRAFT :
- o SPIN-STABILIZED, DUAL SPIN WITH DEPLOYABLE APPENDAGES
- o TEN SCIENCE INVESTIGATIONS
- o S- AND X-BAND DOWNLINK (40 bps TO 134 kbps)
- o RETROPROPULSION MODULE WITH 10N AND 400N ENGINES USES MMH AND N_2O_4 WITH He PRESSURANT
- o TWO RADIOISOTOPE THERMOELECTRIC GENERATORS YIELD 486W AT END OF MISSION
- o TOTAL SPACECRAFT INJECTED MASS : 2550 KG
  ORBITER BURNOUT MASS : 1138 KG
  SCIENCE PAYLOAD MASS : 100 KG



SYSTEM DESIGN REQUIREMENTS/CONSTRAINTS

- o RESOURCE LIMITS
  - - $, SCHEDULE, MASS, POWER, ETC.
- o INHERITANCE/EXPERIENCE UTILIZATION
- o OPERATIONS
  - - LONG MISSION DURATIONS
  - - NO REAL-TIME SUPPORT
  - - UNATTENDED PERIODS
- o SINGLE-POINT FAILURES
- o FAULT PROTECTION SYSTEM OBJECTIVES
  - - STS SAFETY
  - - MISSION CRITICAL EVENTS
  - - S/C INTEGRITY AND COMMANDABILITY
  - - CONSUMABLE PROTECTION
  - - DOWNLINK

FAULT TOLERANT DESIGN FEATURES

REDUNDANCY

- o FUNCTIONAL REDUNDANCY
  - - HARDWARE
    - o THRUSTERS
    - o ATTITUDE SENSORS
  - - PROCESSES
    - o FLIGHT SOFTWARE
    - o OPERATIONS ACTIVITIES
- o BLOCK REDUNDANCY
  - - ACTIVE
  - - STANDBY/INACTIVE
- o CROSS-STRAPPING/REDUNDANCY LEVELS

FAULT TOLERANT DESIGN FEATURES (-2)

FAULT DETECTION

- o DIRECT SENSING TECHNIQUES
  - - VOLTAGE COMPARATORS
  - - TEMPERATURE TRANSDUCERS
  - - PRESSURE TRANSDUCERS
  - - PARITY/CHECKSUM IMPLEMENTATIONS
- o INDIRECT SENSING MECHANISMS
  - - PERFORMANCE MONITORING
  - - REASONABLENESS CHECKS
  - - COMPARISON TECHNIQUES
  - - WATCHDOG TIMERS

FAULT PROTECTION SYSTEM DESIGN APPROACH

- o IDENTIFY MAJOR FUNCTIONS WHICH ARE CANDIDATES FOR PROTECTION
  - - UPLINK
  - - DOWNLINK
  - - POWER AVAILABILITY
  - - ATTITUDE CONTROL FUNCTIONS
  - - SEQUENCE EXECUTION CAPABILITY
  - - OTHER IMPORTANT FUNCTIONS
    - - CONSUMABLE PROTECTION
    - - TEMPERATURE CONTROL
    - - SUBSYSTEM INTEGRITY
- o ASSESS PROTECTION REQUIREMENTS
- o DETERMINE PROTECTION IMPLEMENTATION APPROACH

FAULT PROTECTION IMPLEMENTATION CONSIDERATIONS

- DETERMINATION OF SPACECRAFT OR GROUND ALLOCATION

PRIORITY FOR S/C ALLOCATION	FAULT RESPONSE CRITERION
1	EARTH-BASED RESPONSE NOT FEASIBLE
2	IMMEDIATE INTERVENTION REQUIRED
3	REQUIRED RESPONSE TIME LESS THAN 7 DAYS
4	GROUND VISIBILITY COMPROMISED
5	SCIENCE ACQUISITION/OPERABILITY COMPROMISED
6	NO SYSTEM COMPROMISE FOR DELAYED RESPONSE

- RESULTING SPACECRAFT RESIDENT FAULT PROTECTION
  - SYSTEM UNDERVOLTAGE RECOVERY
  - COMMAND LOSS RECOVERY
  - RPM OVERPRESSURE
  - AACS & CDS SUBSYSTEM AND INTERFACE PROTECTION
  - RPS LOW POWER OUTPUT RECOVERY
  - ACTIVE TEMPERATURE CONTROL
  - SCIENCE INSTRUMENT MONITOR

TECHNICAL DIFFICULTIES ENCOUNTERED

- FAULT INDICATOR RELIABILITY
  - DESIRE NO ADVERSE IMPACT FOR ERRONEOUS FAULT SIGNAL
  - IF NOT FEASIBLE, CORROBORATION REQUIRED
    - COMPLEXITY
    - DANGER OF NO RESPONSE
- TEMPERATURE CONTROL
  - SINGLE MEASUREMENT AVAILABLE PER CDS HALF
  - MIDRANGE TRANSDUCER FAILURE
  - LIGHT SWITCH RELAYS
- INTER-STRING CO-ORDINATION WITHOUT COMPROMISE
  - DEFINITION OF "SINGLE FAULT"
  - CRITICAL SEQUENCES BEING WORKED

CHALLENGES FOR FUTURE MISSIONS

- FAULT PROTECTION RESOURCE PLANNING AND MANAGEMENT
- "FAULT TOLERANT" COMPUTER IMPLEMENTATION
  - ALLOW MORE REASONABLE MAJORITY VOTE PROCESSES
  - PROBLEM APPEARS TO BE IN COMPUTER INTERFACES, NOT COMPUTER ITSELF
- RE-DEFINE "SINGLE POINT FAILURE" REQUIREMENT
- RE-ASSESS REQUIREMENTS FOR QUANTITATIVE RELIABILITY ESTIMATES
- LOOP CLOSURE IN ALL COMMAND PROCESSES
- ADDRESS ISSUE OF COMPLETENESS VS. FEASIBILITY OF VERIFICATION PROCESS

The Need For Design Verification in Fault-Tolerant Systems[1]

Karl N. Levitt
Computer Science Laboratory
SRI International
Menlo Park CA 94025

This paper is concerned with the Formal Verification of computer systems. In the course of carrying out the work reported herein we have developed a number of methodologies for verifying systems, developed computer-based tools that assist users in verifying their systems, and have applied these tools to verifying in part the SIFT ultrareliable aircraft computer. It is clear that with sufficient hardware redundancy, computers can continue operating despite the occurrence of hardware failures. However, hardware redundancy does not guarantee the absence of software failures, particularly in the software that manages the redundancy. We show that the technique of formal verification can, in principle, be used to guarantee that the software is sufficiently reliable so that the high reliability promised by the use of hardware redundancy is actually achieved.

1. Objectives

By *formal verification* we mean showing by mathematical reasoning that a system satisfies its requirement. By a *system* we mean the computer hardware and the collection of programs that run on the hardware. A *requirement* is a description of the function to be carried out by the system. The requirement indicates the system's response to all sequences of inputs that could be applied to the system. If the verification is successfully carried out, the system is, in principle, guaranteed to be *correct*; no further validation (e.g., testing) should be required.[2] However, it should be noted that the system might still contain some errors that require conventional testing to uncover e.g. due to: errors in the requirement, ommisions in the requirement, errors in portions of the system that are not verified. Thus one should view formal verification as a systematic approach to analyzing a system that when combined with standard methods, is potentially capable of reducing significantly the number of errors in delivered systems.

How can it be assured that requirements are free of errors? The most obvious answer is to produce requirements that are short enough and simple enough to be carefully reviewed. We have found that even for very complex systems, requirements can be written that indicate only what is essential to understand *what* the system is supposed to be doing. Details of the system's implementation need not be part of the requirement. It is our conclusion, then, that short requirements statements can be produced. In order to

---

[1]The work reported here was supported by the NASA Langley Research Center under Contract NAS1-155528

[2]To understand the implications of verification, the reader might find it helpful to make analogy to what a proof in, say, plane geometry accomplishes. Reasoning similar to what we describe in this paper is used to prove the Pythagorean Theorem. Once verified, a theorem can be freely used in any circumstance where it applies -- the assumptions underlying the theorem (there is a right triangle with sides a, b and hypotenuse c.) are satisfied.

PAGE 241 INTENTIONALLY BLANK

be processible by the verification tools, the requirement must be expressed in a formal language. We will be referring to three different languages for stating requirements (STP, Boyer-Moore theory and SPECIAL). Those familiar with mathematical logic will have no trouble reading and understanding the requirements we present in this paper. Those who have not been exposed to mathematical logic will be able to understand the requirements through the English comments that accompany the formal logic statements.

We wish to contrast our approach to system verification to the traditional approach, in particular stressing why our approach leads to simpler and more believable requirements, and why it makes the process of verifying large systems feasible. The *traditional* approach, which we call *code verification*, is concerned with verifying algorithms expressed in a programming language. For example a *sort* algorithm is shown to satisfy the requirement that the output is ordered and a permutation of the input. (The requirement for such algorithms is commonly called the *specification*.)

Code verification in itself is inadequate since it does not lead to requirements that are short and simple. For example, an operating system consists of many subprograms. Each of these subprograms can be given a specification: The specification for the *scheduler* will indicate what it means to schedule; the specification for the *directory* manager will indicate what it means for directory entries to handled properly. However, these specifications in toto fail to clearly indicate the overall purpose of the operating system which, in the case of SIFT, is to assure that aircraft control programs are processed correctly. It is the interaction of these programs that determines if the overall goals of the system are to be satisfied.

What is needed, then, is a requirement statement that addresses higher-level issues. To achieve this, we suggest that a system requirement be expressed as a *model*, which we take to mean a collection of higher-level functions together with properties (expressed as axioms on the functions). This model must be shown to be consistent with the specifications for the subprograms that comprise the system; we call this process *design verification*. (Code verification is then employed to show the consistency of the subprograms' specifications with their implementation.)

Often, as was the case in SIFT, the jump from the model to the specifications is too large to be carried out in one step. Hence we introduce additional models, the collection of models forming a hierarchy. Design verification, then, consists of proving that each model is consistent with the one directly below it in the hierarchy. It should be noted that in a well-conceived hierarchy, each model will introduce a particular element of design; the SIFT hierarchy nicely illustrates this concept.

In support of the steps of design verification and code verification, we have developed a collection of interactive tools. The heart of the design verification tools is the STP theorem prover, the language which it supports being used to define the models. Besides the theorem prover, the tool set contains various support packages, including ones that manage the overall verification process and that output a final proof in a form that is reasonably readable.

The code verification system supports the verification of Pascal programs whose specifications are expressed in the SPECIAL specification language. The methodology underlying the tools, (called the Hierarchical Development Methodology [HDM]), allows the code itself to be decomposed in a hierarchical layering of levels. This process greatly simplifies the verification of large programs. Another significant feature of the code verification system is that it can be easily tailored to handle any particular programming

language. This independence of programming language is achieved through a tool called the *meta- verification condition generator*, which accepts the syntax and (axiomatic) semantics of a programming to produce a code verification system unique to that language.

As indicated above, the scope of this project is aircraft electronics systems. Our primary accomplishment was the development of the verification systems (for design and code verification) and the application of the design verification system to SIFT. It should be noted that design and code verification, as carried out in this project, do not cover all parts of an aircraft electronics system. Missing from our verification (besides the parts of SIFT we did not verify -- see below) are the following: Assembly-level programs, Hardware logic, and Application programs

Consideration of these areas begins to complete a full hierarchy for an aircraft electronics system, the components of which are:

- Application programs -- in particular, flight control programs
- Design for a fault-tolerant aircraft computer -- SIFT
- Higher order code (Pascal) for SIFT software
- Assembly level code, which in the case of SIFT is in the Bendix BDX930 instruction set
- Hardware logic -- implementation of the BDX930 instruction set

In each of these areas, we developed techniques that give some promise of being suitable for verifying real systems.

2. Significant Accomplishments of Project

The significant accomplishments of the project are the following:

1. The development of experimental verification tools (for design and code verification). It should be noted that versions of these tools have been used to verify security-related properties of operating systems; this work was carried out on other SRI projects.

2. The application of the design verification tools tools to the verification (in part) of the SIFT operating system. As we indicate later, SIFT achieves reliability by having tasks execute on 3 or more processors, the results being voted on after completion of each task. When an error can be pinpointed to a processor, it is logically removed and replaced by another processor. The requirement for SIFT (informally stated) is that the probability of producing an incorrect result shall be less than 10^{-10} per hour over a 10 hour mission. The major property addressed by the design verification exercise and expressed in a model is that all aircraft *tasks* managed by the SIFT system will yield *correct* results within their prescribed *deadlines*, as long as the

system is in a *safe state*. Here *tasks* are programs that implement the various aircraft functions (flight control, navigation, etc.); *correct* means that the tasks will always get the right inputs and deliver the output as would be produced by a working processor; within the *deadline* means that the result is produced according to some preassigned schedule. *Safe state*, is not given a definition at the highest model; when defined in a lower model, it means that the number of good processors exceeds the number of failed processors in a configuration -- voting works. We believe that this model expresses exactly the significant functional properties of SIFT. Moreover, it should be noted that the design verification turned up a significant design error, that previously escaped our attention. Our verification of the SIFT design is currently incomplete, failing to prove the following

- **Reconfiguration**: When a processor is found to be faulty, the reconfiguration design will logically remove from the configuration of processors. Current work is considering this verification.

- **Quantitative reliability**, the failure probability for SIFT is 10^-10/hour for a 10 hour mission. Our design hierarchy does include a model, called the *reliability model* that expresses, in terms of a Markov model, the concept of system failure. However, we did not formally relate the reliability model to the other models; this connection might be carried out in current work. This connection by itself, however, will not lead immediately to a verification of quantitative reliability since the rates of processor failure and reconfiguration must be derived; these rates can only be derived from significant experimentation with SIFT, as currently being carried out by NASA-Langley.

3. The application of the code verification tools to the verification (in part) to *a* Pascal implementation of SIFT. As in the design verification exercise, our concern was just with the safety-related properties; we did not carry out the verification of the SIFT code concerned with *reconfiguration*. Among other current deficiencies of the code verification effort are:

   - Although the code we did verify would successfully run SIFT, it is not the code that in the SIFT system delivered to NASA-Langley. The delivered code is a combination of Pascal and BDX930 assembly code. Moreover, the Pascal portion is written in a version of Pascal tailored to an efficient real-time processing; for example, it permits the specification of absolute addresses, and accomplishes the transfer of data among processors by a special assignment statement. Our code verification system, supporting standard Pascal, does not handle such features.

   - The program that assures the clocks of the SIFT processors are in synchronization was not verified. The specifications for the clock

synchronization program were, however, used in the design proof.

- The program that handles *interactive consistency*, i.e., the transfer of single source data among processors.

4. An initial approach to verifying assembly language programs. This approach, if mechanized, would be used to verify that portion of a system not expressible in a high-level language. We attempted, but did not complete, the verification of a scheduler for a real time system. A byproduct of this effort was a formal definition of the BDX930 instruction set in the Boyer-Moore theory.

5. An approach to verifying the precision of numerical algorithms, e.g., navigation programs. This approach is suitable only for programs where the *correctness* property (ignoring precision, the algorithm computes a certain function), and the precision property (the error introduced through round-off and other error-introducing operations is bounded by a specified value) can be handled separately.

6. An approach to verifying control applications. Again, separating design verification from code verification proved to be extremely useful. The code verification exercise proves that the program correctly implements a particular filter function. The design verification shows that a configuration of filters achieves a particular control law. We used the Boyer-Moore theory to express the control law for a simple application -- the control of a vehicle subject to bounded disturbances in one dimension.

7. An approach to verifying hardware logic that demonstrates the consistency between a hardware logic circuit and a specification for the functional behavior of that circuit. Using our method, we successfully verified a hardware frequency comparator. A more ambitious undertaking, not yet attempted, would be to verify the implementation of the instruction set for a computer.

3. Conclusions and Recommendations

Our general conclusions are the following:

1. This effort and others (University of Texas, Stanford University, University of Southern California - Information Sciences Institute, Compion Company, and the System Development Corporation) show that the mechanized verification of complex systems is now feasible, although still difficult.

2. We are quite pleased with the progress in verification from 5 years ago when most of the effort was on verifying algorithms (albeit complex ones).

3. We are also quite pleased that comparatively simple requirements statements can be formulated for complex statements; when one removes the implementation detail from a description of a system what is left can be quite simple.

4. In the verification of SIFT, most of the effort was absorbed in creating the models; once created, the verification of the models was not difficult using the STP Theorem Prover.

5. The creation of models is not a routine activity. It is likely that inexperienced users or those not skilled in mathematical logic will have to go through many iterations of model creation/proof before succeeding in a verification of a complex system. Frustration might force such users to give up before achieving success. Several improvements below should help alleviate this problem, including a more intuitive specification language, a library of previously created models, and tools that are more helpful in the design/verification process.

6. The tools we developed, although still relatively primitive and experimental, were useful; indeed, a completely manual proof of SIFT would have been too tedious and error-prone.

7. The techniques for verification of numerical algorithms and hardware are still very tentative. A major effort, perhaps of the magnitude we carried out for SIFT, is required to fully understand the problems and to develop the tools and techniques to make the verification process feasible.

Our recommendations for future investigation are the following:

1. The verification of SIFT should be completed, including the reconfiguration design (and code), the synchronization and interactive consistency programs, and the actual running code of SIFT. By having the running code verified, it will be possible for demonstrate that a verified system can withstand the careful testing that skeptics of verification would wish to carry out. Under a continuation of the effort reported here, SRI is moving towards a complete verification of SIFT.

2. A single specification language should be developed; in this project we suffered through a succession of three specification languages, each with its advantages for particular situations, but none ideal. Under contract to the DoD Computer Security Center, we are currently developing such a language, the novel features of which are:

    - Semantics that can be completely defined in terms of the STP Theorem Prover logic

- Support for the creation and use of new types

- *Parameterized* types, to allow specifications to apply in many situations

- Specifications that can apply to a single operation or to a sequence of operations

- Specifications of temporal behavior. Current techniques require time to directly expressed as functions. A more implicit approach, as provided by the newly emerging Temporal and Interval Logics, give promise of leading to more concise specifications of real time systems.

- Incorporation of Hoare logic directly in the language. This feature will allow the user to direct the verification of programs without having to confront verification conditions. It has been our experience that errors in programs are difficult to analyze when presented with reference to verification conditions. Moreover, it is difficult for a user to create lemmas that would help verify verification conditions; such lemmas are more easily formulated with reference to the code and specifications -- as facilitated by the inclusion of Hoare logic in the specification language.

3. Better engineered tools are essential. These tools should support an incremental approach to verification, since that is the only way a complex system can be verified. In this incremental paradigm, the user should be able to compose and verify incomplete programs, models and specifications; to change and reverify components -- the reverification being only for those components impacted by the change; and to ask for the status of the verification at any time.

4. The verification of other *real* systems should be attempted; only by gaining experience in verifying such systems can the verification community fully understand what is required for particular applications and the user community appreciate the utility of the verification technique There is considerable effort being devoted to the design and verification of operating systems that are secure, where *security* means that the operating system prevents data flows in violation of various security models. Despite some limitations, the most popular security model is that concerned with *multilevel security* -- the allowed information flows are governed by the security level (unclassified, confidential, etc.) of the system users and their documents. Other applications worthy of verification are those typically associated with life-critical functions or where errors can have a severe cost penalty, e.g., power plant control systems, mass transit control systems, flight-critical aircraft control systems, or electronic funds transfer systems.

5. As experience is accumulated in real system verification, specifications and models should be published and made available -- perhaps over the Arpanet. Having such documentation should make it easier for the less experienced users to verify their systems.

4. Further Reading

The following publications are recommended for more details. The STP theorem prover is described in [4]. The overall approach to design verification and a detailed description of the models of the SIFT hierarchy are presented in [2]. A new logic, called *Interval Logic*, that we believe will lead to readable specifications for describing temporal behavior is presented in [3]. The Hierarchical Development Methodology (HDM) approach to system specification and verification is discussed in detail in [1].

5. Acknowledgments

P. M. Melliar-Smith and Richard Schwartz, using the STP Theorem Prover developed by Rob Shostak, carried out the design verification of SIFT. Dwight Hare, using the code verification system he developed, worked on the verification of the SIFT Pascal code. Bob Boyer and J Moore, besides developing the theorem prover that was used in the initial stages of this project, worked on the problem of verifying the SIFT assembly language code and, with Mike Gren, on the verification of flight control systems.

Others at SRI who have also been involved are John Wensley (the SIFT concept and its detailed design), Jack Goldberg (leader of the final 2 years of the SIFT implementation project) and Chuck Weinstock (chief implementer of SIFT). Larry Robinson was the principal developer of the HDM proof methodology.

We are grateful to a number of individuals of the NASA-Langley Research Center for their assistance on many aspects of this project. Nick Murray was the initial monitor of the work. He provided constant encouragement, technical feedback on the aspects of this work related to SIFT, and helped guide our initial attempts to verify flight control programs. Rick Butler, the current monitor, has carefully read our reports, and has provided us with valuable criticism and suggestions on a wide range of technical areas related to the project. In addition, he did much of the work on the mechanized verification of a simple voting system. Rick has also helped out considerably in the creation of a plan for producing a demonstrable verification system that could be used for a wide variety of applications.

Billy Dove (at the start of this effort) and, currently, Milt Holt are responsible for NASA-Langley's program in reliable aircraft computers. We also wish to thank the various NASA-Langley professionals who participated in the oral reviews we had them suffer through. These individuals (Brian Lupton, Earl Migneault, Larry Spencer, Sal Bavuso, Dan Palumbo -- among others), although often skeptical of the verification approach, provided us with gentle and helpful criticism.

Earl Boebert (Honeywell -- Systems and Research Division) suggested the problem we used to illustrate our technique for flight control verification.

We are also grateful to the DoD Computer Security Center for their current support of SRI's verification effort and to particular individuals (Ann Marmor-Squires, Bret Hartman, Marv Schaefer, and Roger Schell) for their ongoing technical advice.

References

[1]
K. Levitt, L. Robinson, and B. Silverberg. *The HDM Handbook, vols. I,II,III.* Technical Report, SRI International, Computer Science Lab, 1979.

[2]
P. M. Melliar-Smith and R. L. Schwartz. Formal Specification and Mechanical Verification of SIFT: a Fault-Tolerant Flight Control System. *IEEE Transactions on Computers* :616-630, July, 1982.

[3]
R. L. Schwartz, P. M. Melliar-Smith, F.H. Vogt. *An Interval Logic for Higher-Level Temporal Reasoning.* Tech Report CSL-138, SRI International, Computer Science Laboratory, February, 1983.

[4]
R. E. Shostak, R. L. Schwartz, P. M. Melliar-Smith. STP: A Mechanized Logic for Specification and Verification. In *Proceedings of the Sixth Conference on Automated Deduction*, pages 11-20. Springer Verlag, June, 1982.

OUTLINE

What is Verification

Why it is Important

Can it Eliminate the Need for Testing

How Verification Process is Enhanced by Computer-Based Tools

How we Applied Verification to a "Real" System -- SIFT Ultrareliable Aircraft Computer

What are the Prospects for "Commercial" Use of Verification

TOWARDS ATTAINMENT OF RELIABLE SYSTEMS

Hardware Redundancy Techniques Well-Known

Reliability Modeling Techniques (e.g., CARE) Can Determine if Redundancy is Adequate

Conventional Techniques to Software Validation (e.g., Testing) are Inadequate

The Reliability Requirement

Rate of Failure: less than 10^{-10} for a 10 hour flight

(equivalent to a MTBF of 1 million years of operation)

Rates of failure this small cannot be measured

Reliability Estimation

from a Reliability Model

It is possible to estimate the reliability by use of a probabilistic reliability model.

It is also possible to get highly optimistic reliability estimates by:

- states or transitions do not match real system,
- transition rates do not match real system,
- invalid assumptions of independence,
- use of negative exponential distributions

BUGS HEARD 'ROUND THE WORLD

Space Shuttle Inability to Synchronize Backup Computer

Mariner 18 Abort Due to Missing NOT

F16 Flipped Over Crossing Equator in Simulation

F18 crash due to Missing Exception Condition

Project Mercury Simulation

ARPAnet Collapse 27 October 1980

FAA ATC Outages

SF Muni Metro Ghost Train

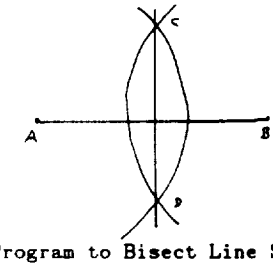
Wizards Penetrating "Secure" systems

WHAT IS FORMAL VERIFICATION?

```
------------------
| Computers      |
| Programs       |          ------------
|   Compilers    |          | "System  |
| Sensors        |<=====>   | is Safe" |
|   Actuators    |          ------------
| Special logic  |
------------------
  System                     Requirement
```

Note: Verification Technolgy Cannot Handle All Components of System -- Hence Testing Still Needed, But Can Focus on Particular Parts of System

VERIFICATION IN PLANE GEOMETRY



Program to Bisect Line Segment AB

(1) Draw Arc Centered at A

(2) Draw Arc Centered at B

(3) Connect Intersections of Arcs

CLAIM: CD Bisects AB

How Might We Verify This "Program"

```
          |System        |Requirement
          |              |
          |              |
          V              V
       -----------------------
       |                     |
Help!  |     Mechanical      |
----<--|                     |
|      |    Verification     |
User   |                     |
|      |      System         |
------>|                     |
Lemmas |                     |
       -----------------------
          |              |
          |              |
          V              V
       Proved         Unproved
```

Note: SYSTEM, REQUIREMENT and LEMMAS must be in a FORMAL (Machine Understandible) Language

STEPS IN VERIFICATION OF SYSTEMS

```
Requirements
     |
     |              Design
     |              Verification
     V
Specifications
     |
     |              Code
     |              Verification
     V
Programs
     |
     |              Hardware
     |              Verification
     V
Hardware
```

The various specifications are described in
Predicate Calculus
with Quantification and Set Theory

Each of the Specifications is defined by a Set of Axioms

At each level we must

- define a mapping
  to the primitives of that specification
  from the more concrete specification below it
- then show that
  each of the axioms of the specification
  can be proved
  as a theorem of the specification below

SIFT GOALS

FLIGHT CONTROL FOR ADVANCED AIR TRANSPORTS

- 10^{-9} PROBABILITY OF FAILURE FOR 10-HOUR FLIGHT
- 1/2 MIP INSTRUCTION RATE

SIFT APPROACH

TOLERANCE OF HARDWARE FAULTS

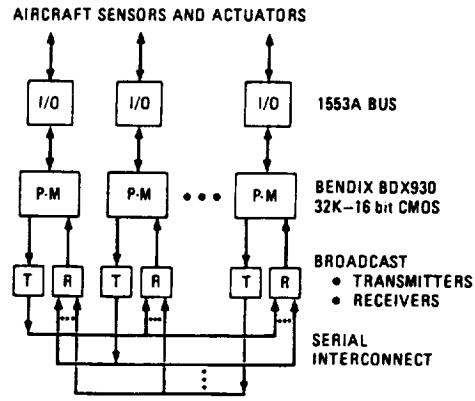
- MULTIPLE COMPUTERS WITH FAULT ISOLATION
- VARIABLE-REDUNDANCY PROGRAM EXECUTION
- FAULT MASKING AND DETECTION (MAJORITY VOTING)
- AUTOMATIC RECONFIGURATION

SIFT DESIGN

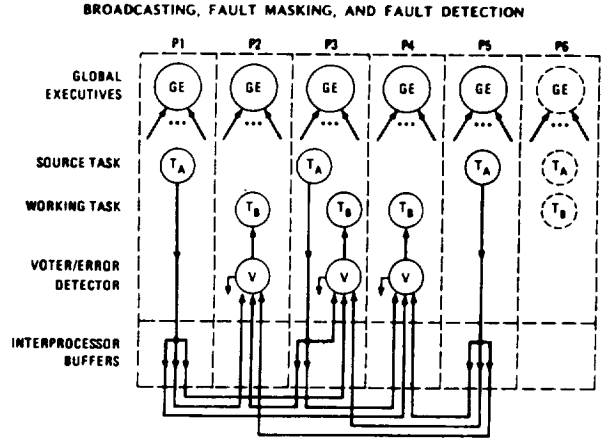
HARDWARE AND SOFTWARE SIMPLICITY FOR EASE OF PROOF

- STANDARD AVIONIC COMPUTERS (6–8), STAR-CONNECTED
- FAULT TOLERANCE ALGORITHMS IN SOFTWARE
- MODULAR, HIERARCHICAL EXECUTIVE SYSTEM
- LOGICALLY SYNCHRONIZED, DISTRIBUTED CLOCKS
- LOOSE SYNCHRONIZATION OF PROCESSORS

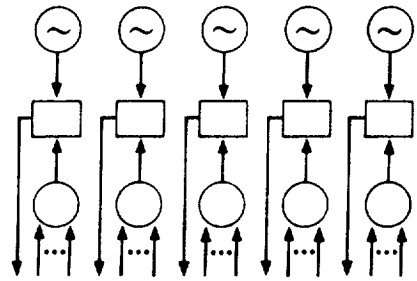
SIFT HARDWARE CONFIGURATION



BROADCASTING, FAULT MASKING, AND FAULT DETECTION



CLOCK SYNCHRONIZATION



FAULT TYPES: OUT-OF-RANGE TIME
INCONSISTENT BROADCASTS OF LOCAL TIME

PERFORMANCE: 3f + 1 CLOCKS FOR f FAULTS
50 μs MAXIMUM SKEW FOR 10ppm OSCILLATOR DRIFT

Replication Specification

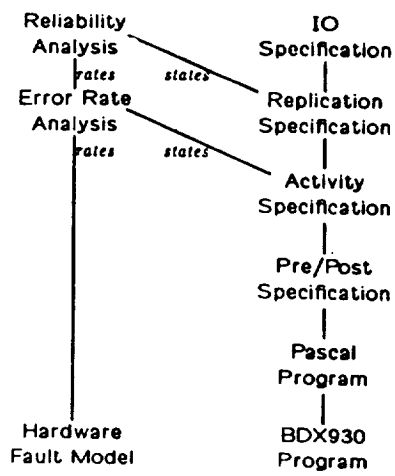
Augments the I/O Specification with the concepts of
- processors
- allocation of tasks to processors
- voting and error reporting
- reconfiguration

All processors are assumed to be exactly synchronized

No concept of
- resource requirements
- schedules
- buffer use
- broadcasting mechanism

At this level, we can prove that

- Majority voting suffices to mask errors



Broadcast Specification

Each processor has its own clock

- hence can prove synchronization

Scheduling of tasks

Time to execute tasks, vote results, overhead

- hence can validate schedule

Space for broadcast results

At this level, we can prove that

- an asynchronous distributed system remain consistent, even in the presence of errors
- the individual programs of SIFT can be regarded as sequential programs that do not interfere with each other

Input/Output Specification

so long as **system safe** remains true, specifies that
- tasks are executed
- at the right time
- using the right inputs
- generating the right outputs

The Specification has no concept of
- processors
- replication of tasks
- voting
- reconfiguration

This Specification contains everything that a flight control engineer needs to know

Pre/Post Specification

Describes sequential programs on a single processor

Provides Preconditions and Postconditions

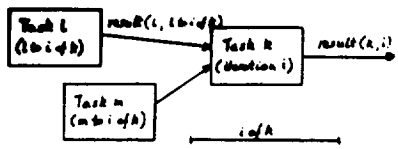
- for each task
- for each iteration of the scheduler

These specifications are very close to the Pascal code they define

"The programs do what they do"

I/O Specification

In every **system safe** configuration,
the result of a task is
the effect of applying its designated function ,
to the results of a set of designated input tasks,
and will be obtained within a real time constraint.




```
/*
MAIN EXECUTE AXIOM: If a task K is 'on' and 'safe' during
 iteration I, then output set Result(K,I) will be the set
 resulting from applying the proper mathematical function
 to the input set V.INPUTS.A2, composed according to the
 previous set abstraction axiom
*/

IO.A2: axiom
ON.DURING(K, I)
∧ TASK.SAFE(K, I)
∧ (∀ L: L ∈ INPUTS(K)  ⊃  CARD(RESULT(L, TO.OF(L, I, K))) = 1)
    ⊃
SINGLETON(RESULT(K, I), APPLY(FUNCTION(K), V.INPUTS.A2(I, K)))
```

VERIFICATION AND SOFTWARE DEVELOPMENT

```
 --->   Requirements
|          |
|          |
|          V
 <-->  1st Design
|          |
|          |          Prototypes
|          V
 <-->  2nd Design
|          |
|          |          Production
|          |          Systems
 <--  Implementation
```

SECURITY: Another Application of Verification

Organization	System Verified -- partial or complete
Univ. of Texas	Security Guard
SRI	PSOS
Ford Aerospace	KSOS-11
Honeywell	SCOMP
Sytek	SACDIN
Merdan	Secure msg system
Compion	Security Kernel

CONCLUSIONS

VERIFICATION Is Now Feasible, But:

- Tools are Clumsy and Fragile
- It is Difficult to Formulate Requirements
- It is Expensive

Next Steps:

- Build Highly Interactive Tools
- Verify Different Types of Systems
- Build up Library of Verified Models

AUTONOMOUS REDUNDANCY AND MAINTENANCE MANAGEMENT

Michael G. Marcucci
Jet Propulsion Laboratory

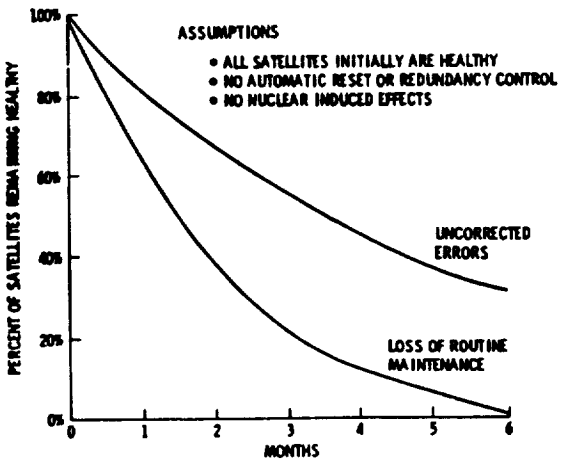
AUTONOMOUS REDUNDANCY AND MAINTENANCE MANAGEMENT (ARMMS I)

- DEFINITION
- NEED
- TERMINOLOGY
- CONCEPT
- PROJECT
- CHARACTERISTICS
- DEMONSTRATION
- FUTURE

AUTONOMOUS REDUNDANCY AND MAINTENANCE MANAGEMENT (ARMMS I)

- WHAT IS IT?
- WHAT'S IT FOR?

PROBABILITY OF SATELLITES REMAINING HEALTHY IF GROUND CONTROL IS LOST



AUTONOMY LEVEL DEFINITIONS

LEVEL 0 - NO AUTONOMY, NO REDUNDANCY

LEVEL 1 - BLOCK COMMAND/FUNCTIONAL REDUNDANCY

LEVEL 2 - BLOCK REDUNDANCY CONTROLLED BY GROUND

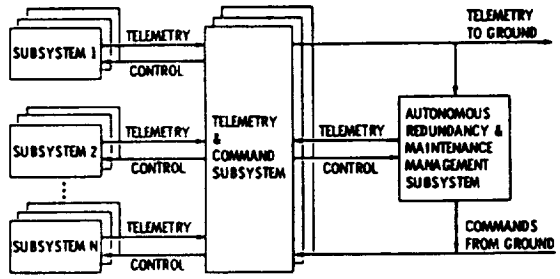
LEVEL 3 - AUTONOMOUS SAFE-HOLD MODES

LEVEL 4 - TIMED SEQUENCE EXECUTION/COMMAND VALIDATION

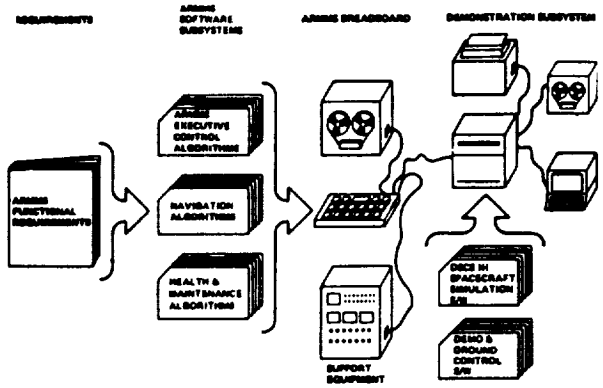
*LEVEL 5 - AUTONOMOUS FAULT TOLERANCE

*REQUIRED FOR MOST FUNCTIONS FOR 6 MONTH UNATTENDED OPERATION

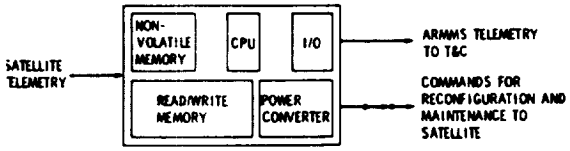
THE ARMMS CONCEPT



ARMMS DEMONSTRATION SYSTEM



AUTONOMOUS REDUNDANCY AND MAINTENANCE MANAGEMENT (ARMMS I)

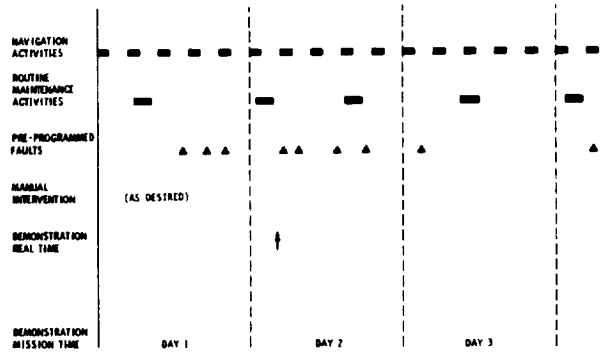


CHARACTERISTICS:
- CURRENT TECHNOLOGY
- INTERFACES WITH SATELLITE COMPOSITE TELEMETRY STREAM
- ARMMS FAULTS NOT PROPAGATED INTO ANY OTHER SUBSYSTEM
- INTERNALLY FAULT TOLERANT
- SENDS COMMANDS TO SATELLITE T&C FOR EXECUTION
- EASY INCORPORATION INTO EXISTING DEFENSE SATELLITE PROGRAMS
- DETECTS FAULTS -- CALLS FOR REDUNDANT UNIT
- PERFORMS ROUTINE MAINTENANCE -- BATTERIES, WHEELS, SENSORS, ETC.
- ALGORITHMS DEVELOPED CAN BE USED IN ANY AUTONOMY IMPLEMENTATION CONCEPT: IN INDIVIDUAL SUBSYSTEMS, IN CENTRAL COMPUTER, OR IN ADD-ON PACKAGE SUCH AS ARMMS
- USE OF ARMMS FUNCTION OPTIONAL

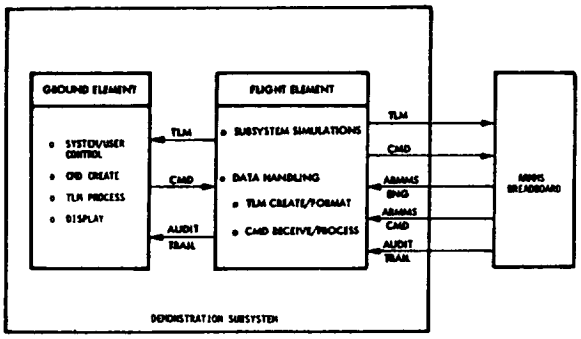
ARMMS CPU CHARACTERISTICS

- TI 9989 -- I^2L
- 5 x 16K RAM -- 32K ADDRESSABLE BY CPU
  -- 16K SWITCHABLE BLOCKS
  -- 2K RESERVED FOR MEMORY MAPPED I/O
- 2 MH_z INTERNAL CLOCK -- MAIN, MASTER FRAME SYNCHRONIZATION
- 4 x 64K NVM MEMORY -- AUDIT TRAIL
  -- STORE BACKUP PROGRAM DATA
- ERROR PROTECTION -- PARITY CHECKS
  -- HAMMING CODE
  -- SPARE BIT PLANE REPLACEMENTS
  -- RESETS, ROLLBACKS, & CHECKPOINT
  -- SELF TESTS

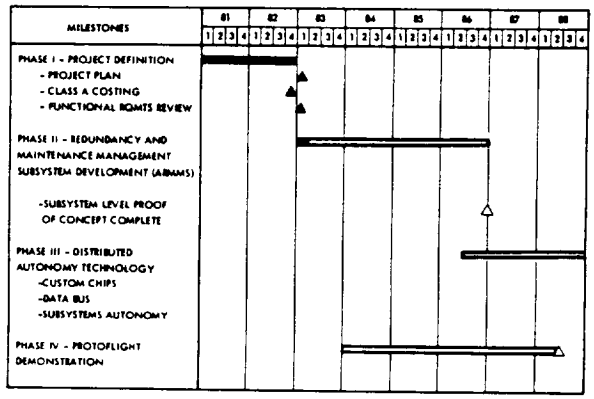
EXAMPLE TIMELINE



SUBSYSTEM FUNCTIONAL BLOCK DIAGRAM



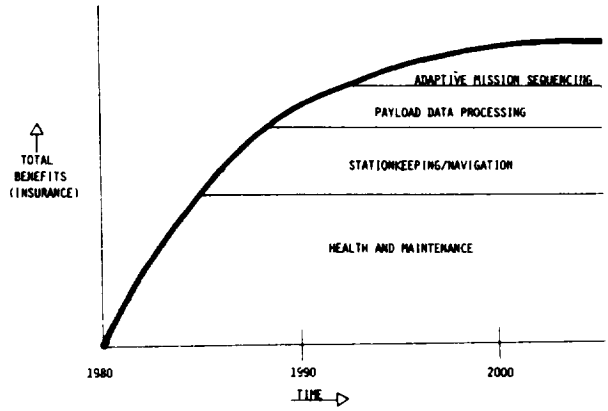
TIMETABLE FOR FUTURE



DEMONSTRATION TYPES

- BRIEF
  - APPROXIMATELY 1/2 HOUR
  - INTRODUCTION
  - SELECTED KEY FEATURES
- STANDARD
  - APPROXIMATELY 4 HOURS
  - COMPREHENSIVE
  - ALL REQUIREMENTS
- MISSION TIME
  - APPROXIMATELY 72 HOURS
  - NO SPEEDUP OR SLOWDOWN
  - NORMAL OPERATIONS

AUTONOMY BENEFITS



SUMMARY OF ISSUES AND RECOMMENDATIONS

SUMMARY OF ISSUES AND RECOMMENDATIONS

1. Design Assurance Techniques

Issue

Lack of quantitative TWT and TWTA data on failure rate life and failure modes is inhibiting the efficient design and definition of redundancy approaches aimed at meeting military and commercial satellite reliability goals.

There is a reluctance on the part of the satellite suppliers, users, and the TWTA manufacturers to collect and/or discuss the data due to competitive and security reasons.

Recommendations

A cooperative effort by NASA, AF, SD, and the commercial industry is required to:

a) Define criterion for "failure," life, and performance on TWT and TWTA.

b) Organize a data collection activity to identify and tabulate ground test and in-orbit performance of TWT and TWTA; (both failures and successful operations). All data (manufacturers, users, spacecraft, etc.) to be coded.

c) Analyze collected data and publish failure rates, achieved life time, life indicators, failure modes, etc.

Suggested OPR

SD, NASA (PRC Corp.)

2. Design Assurance for Ground Segments

Issue

Difficulties occur when (spacecraft, ground system, etc.) controllers do not perform control or problem correction activities as required due to lack of current and continuous training and skills maintenance. The controller may have performed correctly using system (or partial system) simulators before full-up system operation but in certain operational situations will not because of a lack of continuing training and fault simulation.

Recommendations

Use system simulators during operational periods as a training tool to keep controllers efficient. Simulators give the alternatives that may exist and methods to utilize before that situation occurs. This can be defined also as a monitoring system on the controller for deviations that creep into the operation. This simulation capability may also create other tasks to be accomplished during operations, e.g., monitoring of other operations in addition to main controller. Simulators should be co-located with the control centers, not at remote sites.

Suggested OPR

AF/SD, NASA

3. Design Assurance for Ground Segments, ICDs

Issue

ICDs are not in the formal MIL-STD structure, although they contain real requirements: between h/w and h/w, h/w and s/w, s/w and s/w. Although subsystems are often well defined, systems levels are not.

Recommendations

- Modify appropriate MIL-STDs (490, 1521, 483, etc.) to define an Interface Control Specification, detailing s/w, peripheral, electrical i/f requirements, and require delivery at SSR, PDR, CDR, etc.
- Require ICD between flight and ground systems be formal CDRL item.
- Generate a specific DIC to document ICD via MIL-STD-490.

Suggested OPR

NASA Level II
AF SPO Engineering

4. Man/Machine Interface

Issue

We are not alert to "Human Weariness" in man/machine interface design (man is hypnotized and not alert when anomalies occur). Also, human inactivity (waiting) during serial activities can produce "sleep." Human factors delays occur due to extensive human activity and/or interaction with computers and other equipment.

Recommendations

Improve MIL-STD-1472 requirement for human factors planning; establish requirements to prevent "Human Weariness" and "sleep."

Suggested OPR

SD
MIL-STD-1472 OPR

5. Simulation

Issue

This issue is stated from a contractor's point of view.

The customer (USAF) does not seem to be aware of the advantages of the use of simulation as a development tool. Specifically in the DSM project,

simulation is a very limited training only capability. But the STC will continue systems development indefinitely and a sophisticated systems simulation capability would prove invaluable. The current cost effective simulation approach to determine simulation needs is short-sighted and not really cost effective.

Recommendations

At the next Mission Assurance Workshop, have a simulation workshop on use a development and T&I tool. The Houston (JSC) Shuttle simulation, including the RTCC simulation, would be very enlightening. Attempt to get customer representation and participation. Stress cost effective use for development and T&I.

Suggested OPR

NSIA (Next Mission Assurance Workshop)

6. The Design Review Process

Issue

In-depth Design Reviews and Design/Fault Analysis on test equipment and test unit interface hardware is not required on many contracts. Flight hardware is exposed to potential risk if a failure of the test equipment or handling fixtures occurs.

Recommendations

Revise MIL-STD-785 and other appropriate specs and standards, re.: Design Review requirements for test and handling equipment for flight hardware. Require emphasis on FMECA, Design Analysis, Single Point Critical Failures (for handling fixtures), etc.

Suggested OPR

AF/SD, NASA, MIL-SPECs OPRs.

7. Fault Tolerant Equipment Design, Test, and Reliability Predictions

Issues

- Lack of common language defining fault tolerant design, test, and reliability predictions.
- Need for a better understanding of "single point" failure implication in fault tolerant systems.
- Design failures as they relate to fault tolerant are not well understood.
- Performance and reliability parameters need to be related more closely.

Recommendations

Work toward common language (definitions), design guidelines, etc. to aid in developing improved tools for fault tolerant design, test, and reliability prediction.

Revision of MIL-STDs (such as MIL-STD-217, etc.) on reliability prediction, definitions, and design requirements is recommended to include specific requirements and techniques for fault tolerant equipment. A FTE "Design Guidelines" handbook is also recommended and should be based on, and include details of design errors to be avoided.

Suggested OPR

AF/SD, NASA

WORKSHOP G

MATERIALS AND PROCESSING INTEGRITY WORKSHOP

Chairmen:

Norm Nielson
Manager, Quality Assurance
ALCOA

Hank Lange
Director, Quality Assurance
MDAC

Rudy Molina
Director, Quality Assurance
DCASR

Coordinators:

Bruce Baldwin
DCASR

Mike Butler
MDAC

SESSION 1 — INSPECTION CONTROLS, HOW EFFECTIVE?

Workshop Keynote Speech: Quality — Illusion or Reality in U.S. Industry	RADM Frank C. Collins, Jr., Defense Logistics Agency
Essential Qualities of a Source Inspector	Ben Pagenkoph, NASA Rocketdyne
Zone Defense in Relation to Source Inspection	Jack Medina, LeFiell Mfg. Co.
Supplier and Source Inspection	Arthur Welch, Martin Marietta
Process Integrity	Joseph Kozma, Hemet Castings
Uniform Interpretation of Quality Specifications	R. E. Fear, DCASMA

SESSION 2 — ALUMINUM INDUSTRY RESPONSE TO PAST PROBLEMS

Introduction	Normal W. Nielsen, ALCOA
Aluminum Task Force on Plate Soft Spots	Paul V. Mara, Aluminum Association
Forging Soft Spots — The Problem and Its Resolution	Dennis P. Masterson, ALCOA
A.I.A.-A.A. Joint Task Force on Extrusions for Revision of MIL-H-6088F	Roger Carter, Boeing
Automated Conductivity Testing System	James E. Durate, Reynolds

WORKSHOP G

MATERIALS AND PROCESSING INTEGRITY WORKSHOP

SESSION 3 — ADVANCED COMPOSITE MATERIALS

Opening Remarks	Hank Lange, McDonnell Douglas
Resin Systems	Dr. Jerry Patterson, Marshall Space Flight Center
Standardized Composite Materials	Ray Juergens, McDonnell Douglas
Differing User Specifications Result in Custom Formulations	Dr. Jim Allen, Fiberite Corporation
Variations Among User Test Specifications Require Additional Supplier Testing	Steve Russell, Hercules Incorporated

QUALITY - ILLUSION OR REALITY IN U.S. INDUSTRY?

RADM Frank C. Collins, Jr.
Defense Logistics Agency

It is an honor for me to address such a distinguished audience and a pleasure to speak on my favorite subject which is of such strategic importance to U.S. industry and government alike.

Today, I'd like to explore with you the issue of quality. Is the renewed emphasis by industry through the media a real reawakening to the indisputable facts of life or is it all just a mirage?

In my travels, I've discovered a wide disparity of perception about what quality is and how its achieved. First off, I've found to my amazement, that many in top management fail to use scrap and rework costs as a barometer for the effectiveness of their quality program; further, I've queried many top executives about who is responsible for quality, only to have them point to their QC manager or director.

Its disturbed me to note companies pouring millions of dollars into the purchasing of precision equipment - while at the same time maintaining legions of inspectors on the payroll conducting multiplicitous inspections on the product those machines produce.

I've also noted companies more interested in putting progress payments into adding new corporate acquisitions than in upgrading and automating equipment to improve productivity and quality.

Why is this? Can you answer yourself honestly and say that the quality of the product is your greatest concern?

I think few will disagree with me that following World War II, the U.S. was perhaps the only country to emerge with our industrial complex intact. Consequently:

1. We found it unnecessary to complete with the rest of the world.

2. We put quantity above quality.

3. We became complacent - world markets were ours.

Japanese industrial managers, on the other hand, adopted the improvement of quality as a national strategy developing an export economy. Before you turn me off as just another extoiler of Japanese management, please hear me out.

These Japanese managers turned to two American quality experts to teach them the principles for assuring and improving quality: Dr. W. Edwards Deming - taught them the proper methods of statistical quality control; Dr. Joseph M. Juran - taught them management's role in quality improvement. The Japanese feel that quality and not price would win the world's markets - and obviously they've been right!

Consider the fields today in which the Japanese set the quality standards: automobiles, televisions, computer memory chips, etc. These are the reasons:

A. New U.S.-made cars incur two to six times as many problems. Henry Ford II admitted on a public broadcasting station that during the 1970's, Ford, as well as the other major car manufacturers, produced many poor quality automobiles.

B. American T.V. sets fail two to four times as often. So why buy American?

C. U.S.-made computer memory chips fail three times as often (50% scrap with 15% considered no concern).

As a result, the Japanese have successfully penetrated a large market in cars, sporting goods, microwave ovens, CB radios, motorcycles, and nearly 100% in video cassette recorders. They are likewise capturing large shares of the calculator, T.V., camera, stereo components, digital watch, radial tire, and many other markets in which America either pioneered or were one-time leaders.

We, who were largely responsible for the development of sophisticated quality control principles are now faced with rapidly diminishing markets due largely to our failure to capitalize on those principles. We have come into shallow waters and must set a course which will lead us back into a position of respect and competition in the world market place.

I believe that American industry must first accept its responsibility for and then make a national commitment to quality. Inherent in such a commitment is the understanding and implementation of the steps I am about to outline.

The first of these steps is realization and acceptance of the fact that quality begins at the top.

Too many believe that so long as they have a QC organization, all is well. They give lip service to "support" of their quality organization, we don't need support of top management, we need their participation.

For example, I visited a high technological equipment firm in Dallas - these people have a real quality problem. I was briefed by the QC manager who, incidently, reported to production. When I commented on the less than optimum situation that was represented, I was told, "That's company policy."

Earlier, I had visited another well known leader in the communication field whose plant was in the Phoenix area. In that plant of over 3,000 employees, the division vice president, who was also the plant manager, personally gave the quality brief which was a comprehensive coverage of what management was doing to promote quality.

Need I comment on the quality of their output? It was extremely high in a field that is now experiencing severe quality problems.

My reaction to management is - don't tell me how much you support quality - show me what you are doing to achieve it - which brings me to step two.

This entails provision of some training, in the basic principles of quality, for our entire industrial workforce - every worker, supervisor, and manager in industry and government.

A. The first group to be provided the training, as was the case in Japan, should be the executives and top managers. Again, quality begins at the top.

B. The bulk of the industrial workforce in both industry and government can be trained by local company/government quality specialists.

C. Design and production engineers must receive extensive training in how quality impacts productivity.

A third important step is for management to change its attitudes towards its workers. Management must begin to treat employees as members of the team who important ideas that can:

A. Improve quality and productivity.

B. Reduce costs.

Robert Hayes, Professor of Business Administration at the Harvard School of Business, made some interesting observations in an article he wrote for the Harvard Business Review entitled, "why Japanese factories work." "In Japan," he wrote, "The worker is king. In the U.S. the stockholder is king." Management can institute techniques such as quality circles, remembering that these will only be as effective as management's participation, acceptance of feedback, and willingness to implement suggestions and solutions permit them to be and treat people as kings.

Management must not blame poor quality primarily on the worker. They must realize that they set the standards to which workers respond. If the manager pushes for meeting schedules and reducing short-term costs at the sacrifice of quality and attendant reduction of long-term (life cycle) costs, the workers will provide the same. Naive as it may sound, I believe most workers want to turn out a good quality product. They look to management to provide them the tools and training to enable them to perform.

Step four is for U.S. managers to move away from the pure scientific management approach to product design. You can produce almost anything under laboratory conditions, but is it "DOABLE" on a production line which is regulated by various skills and motivational levels, a delivery schedule, and a profit motive?

Quality and manufacturing professionals should review preliminary and detailed designs and use the PITRM screen, i.e., producibility, inspectability, testability, reliability, and maintainability. Lessons learned should be documented in design handbooks and programmed into computer-aided design software.

A fifth very important step is the detection of flaws in the company's manufacturing system, i.e., procedures, policies, and implementation of continuously monitored process control systems. To those who might argue that they already have such process controls well entrenched and in operation, I would suggest that they take another look.

Finally, top management in industry must elevate quality and manufacturing professionals to the executive officer level and provide them with the authority to halt final designs and production lines if quality is

being sacrificed for short-term reductions or schedules.

So far, I have been primarily talking about steps that management in industry must take to achieve a national commitment to quality. Now, I would like to address the role of management in government in achieving a national commitment to quality.

Top managers in OSD, the military services, and DLA, as well as program managers, contracting officers, and buyers in subordinate commands, must impress our contractors with the fact that quality is important to us, as important as cost and schedule. Speaking today for DLA, may I stress this! Managers in DOD must insist that contractors design the required quality into the product. We need to back up our words by:

A. Demanding strict compliance to all legitimate technical requirements.

B. Withholding progress payments or terminating contracts for default, when necessary.

C. Making it more difficult for contractors to obtain waivers and deviations to legitimate technical requirements.

D. Insist that contracting officers support government in-plant QA personnel in their efforts to obtain corrective action by contractors who are not meeting contractual QA systems and technical requirements.

As for the Defense Logistics Agency, the DOD common user buying organization, we are in the process of shifting from a PQA approach which relies heavily on production verification inspection (PVI) to one which is the key elements of the systems approach. Those of you familiar with our PQA program may say that the five elements have not changed. That is true. We are not changing the basic structure of our program; we are merely redirecting our efforts to place increased emphasis on the more efficient systems evaluation elements. Note, there is no plan to ignore the product as many perceive, for production verification has a place - but PVI must be judiciously used with procedures review and evaluation. Bear in mind, the finest procedures your production/quality staff can produce are valueless if not implemented on the production floor. A procedures review, compliance-poor system is as disastrous to our mutual goals as no procedures at all.

At DLA, we have taken the essential steps to make the systems approach a reality. The time for preaching the need to improve quality is over. Now is the time to implement those actions necessary to improve the quality of products. Contrary to popular belief in this country, improved quality is not a cost-driver, and I am always incensed when a quality manager or CEO talks about the cost of quality. Improved quality reduces the costs and labor hours associated with:

A. Repairing defective products.

B. Repairing failed components.

C. Writing off scrap.

D. Producing extra inventories to cushion against defects.

E. Handling customer complaints.

F. Processing warranty claims.

G. And most important, the good name and reputation of the firm.

Managers are going to have to learn to think in terms of longer range strategy and less in terms of short-term bottom line. We are going to have to overcome the resistance that grows out of the myth that moving from adequate to superlative quality can only be achieved at great costs.

For some years, the emphasis seems to have been on advertising or marketing rather than quality. American industry must face the fact that if quality merchandise is produced, that very quality is your best long-term marketing factor.

Dr. Juran made an ominous observation which appeared in the February 1979 issue of quality, "In my observation," he said, "No other nation is so completely unified on the importance of good quality achievement, so eager to discover and adopt the best practices being followed in other countries, so avid in training all company levels and functions in modern methods of controlling quality, so vigilant in regulating the quality of exported goods. To be sure, there is progress along these fronts in all countries, but nowhere else is there the broadbased sense of devotion and especially the sense of urgency which is so evident among the Japanese. The Japanese are headed for world quality leadership and will attain it in the next two decades because no one else is moving there at the same pace." What progress, for a country whose product association is "made in Japan" once stamped into defacto, inferior in most American consumers minds.

Ladies and gentlemen, the result is quality for it brings not only satisfaction to the customer, it increases productivity, enhances company reputation, and ensures competitiveness. All of these mean profit and survivability. An excellent reputation, and ensures competitiveness. All of these mean profit and survivability. An excellent reputation built over the years can be destroyed in a short period of time by lapse in concern for quality. This, I fear, has happened to many of our once proud leaders of quality. We hear much about quality from the media - let us hope it is in reality, a reawakening, not a market place illusion.

Thank you.

ESSENTIAL QUALITIES OF A SOURCE INSPECTOR

Ben L. Pagenkopf
Senior Representative
Safety, Reliability, and Quality Assurance
Space Shuttle Main Engine
NASA-Geo. C. Marshall Space Flight Center

The framework of the following remarks is a paragraph from a speech made by one of the world's strongest proponents for quality - a speech made 20 years ago - but just as appropriate today - a man not involved in space flight but in underwater warfare - the man, Admiral Hyman G. Rickover. The paragraph reads as follows:

"To prevent poor workmanship, quality must be considered as embracing all factors which contribute to reliable and safe operation. What is needed is an atmosphere, a subtle attitude, an uncompromising insistence on excellence, as well as a healthy pessimism in technical matters, a pessimism which offsets the normal human tendency to expect that everything will come out right and that no accident can be foreseen--and forestalled--before it happens."

There has been a lot of news lately about the sad state of our school system and there is a trend developing to get back to the basics in education. It's good for any profession to stay close to some basics. Admiral Rickover's remarks contained some very basic principles which can be applied to source inspectors.

1. An Atmosphere

"Atmosphere" is defined by Webster as "a surrounding influence or condition."

In one sense, this is a factor that the source inspector has little control over because, in the final analysis, it must be built into the management and procurement process. The supplier must be made to understand that the buying agency is going to accept only high quality products. The source inspector is only one link in that entire chain. It matters little how "tough" the source inspector is with the vendor if his management is willing to accept inferior quality. In another sense, however, the source inspector creates his own atmosphere. Over time, he acquires a reputation because of the type of person he is, how he conducts himself, and the depth to which he goes in performing his job. For example, a certain college instructor was recently respectfully referred to as - "he's the toughest one here." He had a reputation of making hard assignments and expecting much of his students. He had an atmosphere surrounding him.

A good source inspector will have an atmosphere surrounding him - a reputation of being tough but fair - not expecting more than the vendor has contracted for but demanding every bit of what he has contracted for. If your source inspectors have that atmosphere surrounding them, your vendors are going to take greater care in what they present for acceptance.

2. Subtle Attitude

"Subtle" is defined by Webster as "given to or characterized by very fine mental distinctions, sharpness of perception and analysis - shrewd." Some synonyms from Roget's Thesaurus are: "sly, artful, crafty, wily" (dishonesty not to be implied).

In a sense you don't have to act like a source inspector to be a good one. I have a friend who is retired from Bell Telephone. His favorite saying used to be "we might be the only telephone company in town but we try not to act like it." The source inspector must not fall into a stereotyped mold. There are more tools to the trade than height gages and micrometers. He must look for those subtle errors which are not obvious on the surface. A good source inspector has to be a digger and digging for information, just like digging for gold, can be misleading without careful examination. Sometimes the gold is really quicksilver. For example, a supplier assembled a 120-blade turbine stator. It looked O.K. to the supplier's manufacturing people and inspectors, the prime contractor's source inspector, and the prime's turbopump assembly mechanics and inspectors, but it didn't look O.K. to the rocket engine on which it was installed. The engine failed to function because the blades were upside down. Fortunately, no engine damage resulted but heavy losses of time and money did result, including several precious days of test schedule. Now I'm not placing the entire blame on the source inspector. This was a very subtle condition but could have been caught with a subtle attitude. The upside down condition looks almost identical to the right side up condition. A subtle attitude was lacking on the part of several people and probably most clearly the design engineer who failed to incorporate a goof-proof feature. It would have been nice though if someone

PAGE 6 INTENTIONALLY BLANK

along the way, the source inspector perhaps, would have displayed the characteristics of one with a subtle attitude - "very fine mental distinction, sharpness of perception and analysis - shrewd" - and discovered the similarity earlier. By the way, this stator was in a lot of eight and they were all assembled wrong. Result: approximately $1M worth of scrap.

3. An Uncompromising Insistence on Excellence

"Uncompromising" is defined by Webster as "unyielding, inflexible."

I have the highest regard for our Astronauts. I have a son who is a Navy jet pilot and I know something about the Navy's pilot training program. That program has an uncompromising insistence upon excellence in selecting and training its personnel. Many of our Astronauts came out of that program or similar programs. In order for a mission to be assured, the machine must be of the same level of excellence as the man. The source inspector plays a vital role in this by his uncompromising (unyielding) insistence on excellence.

Involved in the insistence on excellence is the ability to be a critic. During the investigation of the near disaster of Apollo 13 a few years ago national newspapers ran a UPI story headlined "critics needed in NASA." Members of Congress pointed out the need for the NASA to get some hard-nosed people who keep asking tough questions. "You really need some people who would not be very popular in NASA, who would not be invited to social events, but who have a critical bent and are not afraid to ask questions." The Drill Sergeant who trains Navy Cadets is critical of their performance. If it doesn't measure up to the required standard of excellence they are out of the program. The source inspector must be critical of the supplier's performance. If it doesn't measure up his hardware should not be in the vehicle that the excellently trained pilot is to fly.

4. A Healthy Pessimism in Technical Matters, A Pessimism Which Offsets The Normal Human Tendency To Expect That Everything Will Come Out Right And That No Accident Can Be Foreseen--And Forestalled--Before It Happens

"Pessimism" is defined by Webster as "a tendency to take the least hopeful view of events; an inclination toward gloominess about the future. An example of a healthy pessimism in technical matters is the pilot ejection system in most military aircraft. The aircraft are not designed to fail but--just in case--it's nice to be prepared. There are no ejection seats in the Space Shuttle anymore. That means that an extra measure of pessimism must be built into the vehicle itself to bridge that gap. The source inspector can be the final necessary plank in that bridge.

Recently, an inspector was reviewing the assembly records for an overhauled subsystem ready to be tested on a rocket engine. He noticed that a dimensional discrepancy had been accepted on the basis that it had previously been accepted O.K.-As-Is on a previous material review action during the original assembly operation several months ago. He went to the trouble to get a copy of the material review action, compared figures, and lo and behold, the present discrepancy was greater than the previously accepted condition indicating a potentially dangerous condition. The scheduled test was scrubbed, possibly saving a $30M engine. The point is - it took a healthy pessimism about a previously accepted condition to prevent a severe problem.

In summary, I think we would all agree that the above noted qualities; an atmosphere, a subtle attitude, an uncompromising insistence on excellence, and a healthy pessimism in technical matters, are desirable in a source inspector. We can all cite examples from our own experience to support this. How to develop such qualities in a person is another problem, however, and gets to the heart of some of the questions to be addressed by this inspection controls panel. Obviously, these are difficult questions or else there would be no need for them to be addressed. In closing, let me briefly touch on some of them.

Training. I know of no single training course that would turn out a source inspector with the qualities I have just described. Further, it probably is not possible for a person to fully develop all these traits through experience. A person possessing them is most likely a product of a varied and complex mix of training, experience, and personality. To a large extent, these qualities are dependent upon the

basic, relatively fixed personality of the individual. Certain personalities simply would not mesh with the qualities listed above and there is no point in trying. Psychological testing and screening should therefore be considered in the selection of source inspectors. Immediately screen out those who are psychologically unsuited and then selective training and day-to-day experience will be more productive.

Technical Expertise. An engineering background is extremely helpful for a person with these qualities. But then there is the problem of retaining engineers. They probably aren't going to stay in Quality if they can make more money in Engineering - so we must be ready to pay them and involve them in productive work.

Workload. It takes time to ferret out subtle omissions and exercise a healthy pessimism. If a source inspector is overloaded with work and doesn't have time for some creative thinking and digging, we might as well forget about the subtle attitude and healthy pessimism qualities.

Finally, a basic concern of this panel is the control of suppliers in regard to materials, process integrity, and lot/heat traceability. If we can work toward developing source inspectors with Rickover-type qualities, we will have taken a giant step in resolving that concern.

To prevent poor workmanship, quality must be considered as embracing all factors which contribute to reliable and safe operation. What is needed is an atmosphere, a subtle attitude, an uncompromising insistence on excellence, as well as a healthy pessimism in technical matters, a pessimism which offsets the normal human tendency to expect that everything will come out right and that no accident can be foreseen--and forestalled--before it happens. Admiral H. G. Rickover

---

- An Atmosphere
- A Subtle Attitude
- An Uncompromising Insistence On Excellence
- A Healthy Pessimism In Technical Matters

INTRODUCTION

J. A. Medina

Lefiell Manufacturing Company

Concern for Quality, as we all know, is a keynote catch phrase today. Everyone has this concern for Quality so our systems have been geared to parts inspection. From this have come slogans such as "Make it Right the first time. Don't let the Buyer Beware. Build Quality into the Product." Has anyone ever stopped to think that maybe we hinder the Quality of workmanship by establishing inspect points? Consider this. We are concerned with the knowledge that a craftsman (person) is hard to find in today's work force. A craftsman is one who considers his product a work of art, with pride expressed throughout his product. This artisan has such personal pride that he places his craftsmanship seal on his product.

Why are craftsmen nearly non-existant? Let's think about it. We want to mass produce so we manufacture. In manufacturing we stress how short a time it shall take the operator. We want to cut corners to make parts faster and cheaper. We insist on timeliness at any cost. In many cases the operator does not have time or is not allowed time to check his work. Would you put your craftsmanship seal on a product without confirmation of acceptability to blueprint requirements? Think about it. How often do we hear the following on the shop floor: "But boss, isn't it the inspector's job to check my work? Isn't that why he's there behind me? To catch my mistakes? You took my craftsmanship seal away from me and gave it to the inspector, didn't you? Doesn't that make him responsible for the quality of my work? Why then, should I have concern for quality?" Don't you hear these phrases on occassion?

Now I ask you, is the establishment of inspect points taking away the craftsmanship concept from the very person we look to for making a quality product?

Let's follow along with this thought. Are the same things being extended into the source inspector arena? Maybe the source inspector's redundancy dilutes the quality craftsmanship that companies should continually strive for. If you stop and think about it, the supplier may depend on the source inspector as his double check. The source inspector in essence affixes the supplier's craftsmanship seal. Sure we have fine print which tells the supplier he's still liable. Does this clause eliminate the need for repair, scrap, replacement, the paper mill involved? Of course not. These headaches remain.

Why not consider an alternate approach --- one which can instill pride of workmanship in suppliers; one which can make the supplier a craftsman once again. Let us consider source inspection using a ZONE DEFENSE concept. This has been an effective approach based on actual Space Shuttle Orbiter program experiences.

ZONE DEFENSE

The Apollo program concept established that Rockwell would virtually manufacture, assemble and test the Command and Service Modules. Components and assemblies were procured. During the Apollo program days, the Source Inspection quality concept was based on extensive supplier monitoring. Mandatory Inspect Points were the order of the day. Their intent was to make sure key points were not missed by the suppliers, whether it be prior to closeout operations, at final inspections of major assemblies, or elsewhere in the manufacturing cycle.

Final acceptance testing of the deliverable hardware was almost always witnessed in addition to full support of qualification testing.

Prior to shipments to Rockwell, the Source Inspection Representative reviewed the Data Package and shipping documents after verifying acceptability of the hardware's packaging and preservation. The hardware and data were acceptance stamped, then the shipment was made.

In contrast, the Space Shuttle Orbiter concept required that Rockwell manufacture the Forward and Aft Fuselage structural assemblies and subcontract everthing else. The intent was to distribute as much work as possible throughout the United States. This is why, for example, the Mid-Fuselage was manufactured in San Diego; the Wings were made in New York; Flight Control System manufacturing was accomplished in Minnesota and Florida; and the Multiplexer-Demultiplexer was manufactured in Arizona.

In addition to providing support to fifty local machine shops, the Space Shuttle Orbiter has more than two hundred and fifty suppliers providing well over five hundred procurements of significance. This includes special valves and such, but excludes off-the-shelf piece parts such as electronic components. The Orbiter concept required that Rockwell basically put it all together, test it and deliver the Orbiter to NASA.

In turn, the Space Shuttle Quality concept stressed minimizing Source Inspector/Field Representative involvement actions during performance of the subcontract effort. Why? Because we felt it was the subcontractor's contractual responsibility to deliver a quality product. In other words, the subcontractor's quality system is the control mechanism. We wanted the Field Representative to be more of a concerned observer.

To support this concept our established objective was to provide assurance that subcontractors provide a quality product. Rather than provide for resident Source Inspection personnel at subcontractors, we wanted to utilize the most cost effective method to perform subcontractor quality systems surveillance. To do this a Field Representative Surveillance program was established---ergo, a Zone Defense approach.

You may ask yourselves how we came to determine Field Representative assignments. Well, the main points used were: high contract value, article complexity, advanced state-of-the-art manufacturing techniques as in the case of the Reusable Surface Insulation Tiles, and/or the lack of test or inspection capabilities in the Rockwell Receiving Inspection function. Once Field Representative assignments were determined, they were assigned residency at major subcontractors and at suppliers where significant problems had been encountered. In some cases, assignments were also made in supplier high density areas with itinerant coverage.

The "Zones" are as you see them here. Two supervisors have cognizance over all zones: One supervisor is assigned the western half and the other is assigned the eastern half of the United States. This is what the "W" and "E" letters represent. These zones were established based on numbers of suppliers, proximity to each other in relative areas and travel coverage considerations.

Guidelines for the Source Inspection Field Representatives were established to acknowledge total adequacy of the effort. They were to assure the supplier's quality function had implemented adequate controls/inspect points. In turn, Rockwell Mandatory Inspect Points were very selectively applied to potential or known problem areas, if at all. Acceptance Test Procedures were to be witnessed but only to the extent necessary to verify procedures, processes and testing techniques. This included minimal Qualification Test support, focusing principally on test start activity as you can see. Additionally, his responsibilities included focusing attention on known and potential problem areas, monitoring and surveying the supplier's quality system and performing hardware-oriented surveys which are based on following the hardware through its spectrum of manufacture, assembly and test in order to confirm that the supplier's quality system is effectively implemented.

Of course, a key factor through it all was maintaining complete communication with management at all times so we were aware of all field effort. A most important tool was the Weekly Activity Report (WAR) which we insisted be faithfully and consistently provided. These documents were passed through all Quality Assurance management levels. They not only provided current status but eventually became a gaging tool for this program's effectiveness.

Again, where conditions warranted, the Source Inspection Field Representative initiated corrective action.

Troubleshooters in the Downey home office provided the necessary Field Support for interfunctional interface, supported Field Representatives in problem research and inquiries and were dispatched to the field to provide on-the-spot assistance in special circumstances. When occassion demanded, additional support was provided by in-house Inspection and Quality Engineering groups and/or by temporary Outside Vendor Agencies such as the Vendor Surveillance Corporation and Intertek.

From all I've told you I'm sure you realize that implementation took some considerable effort to "juggle all the balls and keep them in the air," so to speak. We maintained an average of twenty nine (29) Source Inspection Field Representatives, utilizing three (3) Outside Vendor Agency personnel and were supported by eight (8) Downey home office Field Support personnel. Forty (40) people provided support to two-hundred and fifty (250) suppliers and over five hundred (500) significant procurement packages. Yes! It is an effective approach.

We had our problems, of course. Letting go was a big one. We always, just to be sure, want to hang on, so there was no exception here. In letting go we had to determine which MIP's to establish, if any; we had to analyze problems and determine which to support; and the depth of testing support had to be evaluated. Then groundrules required formulating: to insure we knew what we wanted the Representative to do; to be able to tell the Representative during reorientation so he could effectively implement our concepts; and to establish precise groundrules for Representative support implementation.

The problem with communication and feedback was on-going. Some Representatives used the acronym "WARS" to reflect their distaste of Weekly Activity Reports at first. As a result of continually stressing their importance this drawback was overcome to the point that, as I mentioned earlier, they became a gaging tool for the program's effectiveness and allowed us to monitor the program's strengths and weaknesses. We knew whether the Representative was concentrating in the right locale and on the right problems. If this wasn't the case, the Representative was diverted to more important areas.

Although one might originally consider relocation to be a potentially high cost factor it turned out not to be. Most Field Representatives were on-site from the Apollo and B-1 programs, while the majority of the rest were hired from the area wherein they would be working. Also, by having them strategically located within respective zones, relocation was minimal. Travel costs, on the other hand, were on-going but were more related to automobile travel than to airplane. Considering how much more travel costs could have been had our manpower been in line with the Apollo program, this program was cost effective.

The last of the three major costs, outside vendor agency support, came up as an afterthought in support of a problem area requiring temporary support. Since assistance was necessary for only 6 hours a week over several days of the week, their support was effective. Utilization of outside vendor agencies was much less costly than anticipated even though more paperwork was generated in the form of Purchase Orders and related documents.

Rewards, from what you've just seen, include considerable work force reduction. We supported the Space Shuttle Orbiter Program with forty (40) people. Had we supported based on the Apollo Program concept there would be a different tale to tell. The Apollo Program had 110 plus or minus suppliers supported by eighty (80) people. Had comparable support been provided the Shuttle Orbiter Program, its two hundred and fifty (250) suppliers should have been provided support by one hundred and twenty (120) to one hundred and eighty (180) people. It was quite a reduction. This in turn considerably reduced General and Administrative (G & A) costs along with travel costs which I mentioned earlier.

Quality of the product provided us with the much sought-after result --- craftsmanship. Why ---? I feel it's because the suppliers' double check was eliminated. Overall, suppliers strived for and achieved artisanship since we had virtually no MIP's. We have learned new ways, established new concepts and successfully implemented new ideas. Let's not lose sight of them. Let us continuse to build on them, keeping our target of a quality product always in the forefront.

Oh by the way --- if you want proof that craftsmanship will provide a quality product, here it is.

ZONE DEFENSE - A SOURCE INSPECTION CONCEPT

PRESENTED BY: J.A. MEDINA
QUALITY ASSURANCE DIRECTOR
LEFTELL MANUFACTURING COMPANY

JUNE 7, 1983

OBJECTIVE

- ASSURE THAT SUBCONTRACTORS PROVIDE A QUALITY PRODUCT
- UTILIZE THE MOST COST EFFECTIVE METHOD TO ACCOMPLISH QUALITY SURVEILLANCE OF SUBCONTRACTOR QUALITY SYSTEMS, IN LIEU OF RESIDENT REPRESENTATIVE APPROACH.
- ESTABLISH A FIELD REPRESENTATIVE SURVEILLANCE PROGRAM
  - ZONE DEFENSE APPROACH

APOLLO PROGRAM QUALITY CONCEPT.

- EXTENSIVE SUPPLIER MONITORING
  - MANDATORY INSPECTION POINTS (MIPIS) FOR CLOSE MONITORING DURING MANUFACTURING CYCLE.
  - WITNESS FINAL ACCEPTANCE TEST OF DELIVERABLE HARDWARE.
  - CONDUCT ACCEPTANCE INSPECTION OF PACKAGING AND PRESERVATION, DATA PACK AND SHIPPING DOCUMENTS PRIOR TO SHIPMENT TO ROCKWELL.

SPACE SHUTTLE ORBITER CONCEPT.

- MANUFACTURE KEY STRUCTURES ONLY
  - FORWARD FUSELAGE
  - AFT FUSELAGE
- SUBCONTRACT ALL OTHER ASSEMBLIES, SUBSYSTEMS, ETC.
  - MID-FUSELAGE
  - WINGS
  - VERTICAL TAIL
  - FLIGHT CONTROL SUBSYSTEM
  - GENERAL PURPOSE COMPUTER
  - MULTIPLEXER - DEMULTIPLEXER
  - AUXILIARY POWER UNIT
  - THERMAL PROTECTION SYSTEM
  - OVER 250 SUPPLIERS
  - OVER 500 PROCUREMENTS
  - PURCHASED LABOR SUPPORT - 50 LOCAL SHOPS
- "PUT IT ALL TOGETHER"

SPACE SHUTTLE QUALITY CONCEPT.

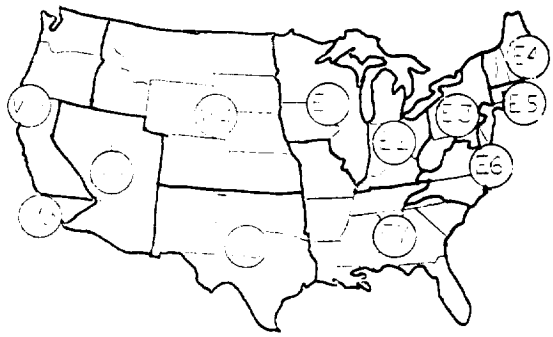
- MINIMIZE ROCKWELL QUALITY INVOLVEMENT DURING PERFORMANCE OF SUBCONTRACT EFFORT.
- IT'S THE SUBCONTRACTOR'S CONTRACTUAL RESPONSIBILITY TO DELIVER A QUALITY PRODUCT. THE SUBCONTRACTOR'S QUALITY SYSTEM IS THE CONTROL MECHANISM - NOT THE ROCKWELL REPRESENTATIVE.

FIELD REPRESENTATIVE ASSIGNMENT DETERMINATION

- HIGH CONTRACT VALUE
- ARTICLE COMPLEXITY
- ADVANCED STATE-OF-THE-ART MANUFACTURING TECHNIQUES
- LACK OF IN-HOUSE TEST/INSPECTION CAPABILITIES

FIELD REPRESENTATIVES ASSIGNMENTS.

- FIELD REPRESENTATIVES WERE ASSIGNED AS RESIDENTS AT MAJOR SUBCONTRACTORS AND AT SUPPLIERS WHERE SIGNIFICANT PROBLEMS HAD BEEN ENCOUNTERED
- ASSIGNMENTS WERE ALSO MADE IN SUPPLIER HIGH DENSITY AREAS FOR ITINERANT COVERAGE



FIELD REPRESENTATIVE ASSIGNMENT ZONES

FIELD REPRESENTATIVE DUTIES AND RESPONSIBILITIES:

- ASSURE SUPPLIER QUALITY FUNCTION HAS IMPLEMENTED ADEQUATE CONTROLS/ INSPECTION POINTS.
  - ROCKWELL MIP'S SELECTIVELY APPLIED TO POTENTIAL OR KNOWN PROBLEM AREAS.
- WITNESS SELECTED ACCEPTANCE TEST PROCEDURES (I.E 1ST ARTICLE) AND THEN TO THE EXTENT NECESSARY TO VERIFY PROCEDURES, PROCESSES AND TESTING TECHNIQUES.
  - MINIMAL QUALIFICATION TEST SUPPORT
    - REP CONCERNS PRINCIPALLY FOCUS ON TEST START ACTIVITY -
      - TEST PROCEDURES APPROVED.
      - TEST FACILITY CERTIFIED.
      - SUPPLIER QUALITY HAS OK'D TEST START AND HAS PLAN FOR TEST COVERAGE.
- FOCUS ATTENTION ON KNOWN AND POTENTIAL PROBLEM AREAS.
- MONITOR AND SURVEY SUPPLIER'S QUALITY SYSTEM
- HARDWARE ORIENTED SURVEYS WERE REQUIRED
  - HARDWARE-ORIENTED SURVEY APPROACH INVOLVES
    - ACTUAL HARDWARE BEING PRODUCED
    - METHODS AND OPERATIONS BEING PERFORMED
    - RECORDS OF HARDWARE/PROCESSES
    - FACILITIES IN USE
- MAINTAIN COMMUNICATION WITH MANAGEMENT
  - SUPPLIER STATUS
  - REPORT SIGNIFICANT PROBLEMS, FAILURES, CORRECTIVE ACTION
  - WEEKLY ACTIVITY REPORT
- INITIATE CORRECTIVE ACTION AS REQUIRED

TROUBLE SHOOTERS IN THE DOWNEY OFFICE (FIELD SUPPORT)

- PROVIDE SPECIALIZED TECHNICAL SUPPORT INTERFACE FROM OTHER FUNCTIONS.
- RESEARCH PROBLEMS/INQUIRIES FROM THE FIELD.
- PROVIDE INPUT TO FIELD REP. FROM OTHER DISCIPLINES.
- READY TO BE DISPATCHED TO FIELD TO ASSIST IN SPECIAL CIRCUMSTANCES.

ADDITIONAL SUPPORT

- IN-HOUSE QUALITY GROUPS - (QC, QE)
- OUTSIDE VENDOR AGENCIES

SPACE SHUTTLE ORBITER PROGRAM HEADCOUNT AVERAGES

- FIELD REPRESENTATIVES 29
- OUTSIDE VENDOR AGENCY SUPPORT 3
- DOWNEY OFFICE FIELD SUPPORT 8
- 40 PEOPLE SUPPORT 250 SUPPLIERS, 500 PROCUREMENTS
- IS IT EFFECTIVE? YES

THERE WERE PROBLEMS

- LETTING GO
  - ESTABLISH MIPS, IF ANY
  - DETERMINE WHICH PROBLEMS TO SUPPORT
  - TEST SUPPORT
- ESTABLISH GROUNDRULES FOR REP. SUPPORT
  - CONCEPT
  - ORIENTATION
  - IMPLEMENTATION
- COMMUNICATION AND FEEDBACK
  - WEEKLY ACTIVITY REPORTS (WAR)
  - MONITORING
- COSTS
  - RELOCATION
  - TRAVEL
  - OUTSIDE VENDOR AGENCY SUPPORT

REWARDS

- CONSIDERABLE FIELD REP. WORK FORCE REDUCTION
  - SHUTTLE ORBITER - 40 PEOPLE
  - COMPARABLE SUPPORT-APOLLO CONCEPT
    - 110 ± APOLLO SUPPLIERS SUPPORTED BY 80 PEOPLE
    - 250 SHUTTLE ORBITER SUPPLIERS SUPPORTED BY 120 TO 180 PEOPLE
  - GENERAL AND ADMINISTRATIVE COST REDUCTIONS
  - TRAVEL COSTS
- CRAFTSMANSHIP
  - SUPPLIERS' DOUBLE CHECK ELIMINATED
  - VIRTUALLY NO MIPs
  - WE HAVE A QUALITY PRODUCT

SUPPLIER AND SOURCE INSPECTION CONTROLS

AUTHOR -- A. L. Welch and C. F. Coyan

COMPANY -- Martin Marietta Aerospace

To be more precise in an imprecise world and in an even less precise occupation, we always turn to Mr. Webster for convenient definition. Once such definition of control is "to test or verify by counter or paralled evidence or experiments." By various activities, some current with the Supplier activity, and others separate and apart from that Supplier activity, we test, verify, and collect evidence on how well the supplier is fulfilling the requirements of his contract and too many times experiment with actions to assure correction based on our preceived evidence.

With this in mind, it is evident that neither the Supplier nor our people in the field can satisfactorily accomplish their tasks unless requirements are accurately and clearly defined and understood.

This definition and understanding must occur early if we are to have the requirements fulfilled as expected. Early means "up front" during pre-award activities. There are two important reasons requirements must be reviewed with the Supplier. One is to make them aware the requirements exist and the second is to assure the Supplier understands the requirements and what is expected to satisfy the requirements. This can be accomplished many ways. The pre-award bidders conference is an excellent way for everyone to review the requirements, hear what they mean and what is expected to fulfill them and to hear all the questions and answers at the same time. The important thing, however, is not "how you do it" but rather "that you do it." Don't leave the recognition and understanding of requirements up to chance. All of the technical and quality requirements that we impose on any given procurement play together to assure the reliability of the product.

Equally as important, "up front", is the assurance that our Representatives know and understand the requirements.

PAGE 18 INTENTIONALLY BLANK

Also, that the Representatives responsibilities, authority and implementation methods are clearly defined and understood. If we don't adequately prepare our Representatives before we send them to the field, they will be experimenters. The Supplier will look to our Representative for guidance, interpretation and counseling. The Representative must be consistent and positive in their understanding of the requirements, of their responsibilities, authority and their implementation methods.

Complete, accurate and understandable Procurement Documentation is a necessary ingredient to facilitate definition and understanding of requirements. If the purchase order or contract, engineering drawings, Quality Requirements Documents, special provision codes and other associated documents that make up the procurement package do not clearly and completely define requirements, you cannot expect the Supplier to be aware of, let alone satisfy, those requirements. I believe that the majority of times when errors are made or requirements are not met it is out of ignorance rather than express intent. Most Suppliers want to do a good job. We need to facilitate that want by providing clear and complete requirements definition and then assuring strict compliance to those requirements. I believe the only way that can be accomplished effectively is through a continuous in-depth surveillance program. A program that is primarily geared to evaluation and prevention rather than finding things that have already happened. I am satisfied that many times we impose totally adequate requirements, but then are lax in enforcement. Sometimes we get too engrossed in trying to inspect Quality into the hardware that we fail to enforce all the essential requirements that assure it's there.

I have always contended that if something is too long or too short or too thick or a hole is too small or too big, and etc., somewhere along the line it will be detected. However, if a particular process is not accomplished properly, the chances of it being detected during subsequent handling or assembly are rather remote. You'll find it when the failure occurs. That's too late!

In the area of Process Control, you cannot rely on the Process Specification and the knowledge or experience of the operator to assure the process is executed properly. Process Specifications are "what to" documents, but they don't tell the operator "how to." You need step by step instructions for processing particular parts, and these instructions must be validated by the Supplier and the Source Representative.

Validation is accomplished by performing the process in accordance with the instruction and determining that the desired results are obtained. Once this has been done, it is necessary to maintain an in-depth surveillance program to assure continued compliance. This surveillance must extend to Sub-tier Suppliers to be effective.

To summarize at this point, to effectively control Suppliers you need complete accurate requirements definition and understanding early in the program. You need validated "how to" instructions in the process control area. And then you need a continuous, in depth surveillance program to assure continued adherence to all the requirements.

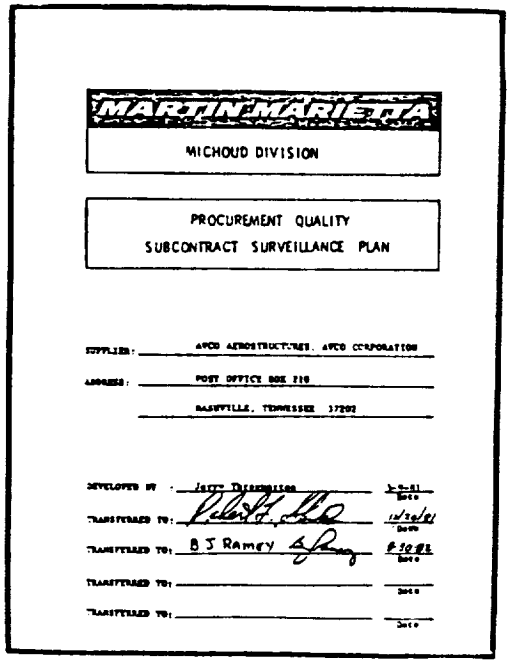
When you look at the workload the people in the field have, you quickly realize that there is no way they can inspect every characteristic of every product produced. That is why the title "Source Inspector" is a misnomer. The person in the field is a "mini" Program Manager and as such is better titled Source Representative or Procurement Quality Representative or the like. Their primary task is to see that the Supplier does the job he contracted to do--but not to do it for him.

How can we best accomplish that task? Not by inspection alone but by judicious use of many means to enforce the requirements and accomplish our end objective of "assurance." That's the definition of management--"the judicious use of means to accomplish an end."

Several times I have mentioned on-going, in-depth surveillance programs. That's our primary management tool. Effective surveillance cannot be left to recall or catch as catch can. For any plan to be effective, it must be preplanned, detailed, documented, followed and adjusted as progress dictates. A surveillance plan is no different. Once the plan is properly developed, implemented and maintained, it should provide the Source Representative, as well as his management, a detailed description of planned surveillance activities for the Quality System, Processes and the product. Further, it should provide a record of surveillance activities completed, a record of Supplier performance data which can be used as basis for adjusting surveillance (spend your time where you're finding problems--not where you never have problems) and afford you the means of affecting orderly transfer of surveillance responsibility from one Representative to another.

We at Martin Marietta have implemented a Planned Subcontract Surveillance Plan (SSP) for the External Tank Program. The SSP outlines specific surveillance requirements and provides a record of completed surveillance activities and supplier performance. All SSPs are identical in format. They will vary only as the hardware, processes, etc., vary. There is not time to review the total SSP, but I will briefly discuss some sections.

The pages I am going to show you are not "dummied" up versions, but were pulled from a currently working SSP. The cover page identifies the supplier by name and address, shows the Representative who developed the plan and the date and shows subsequent transfers.



MARTIN MARIETTA

MICHOUD DIVISION

PROCUREMENT QUALITY
SUBCONTRACT SURVEILLANCE PLAN

SUPPLIER: AVCO AEROSTRUCTURES, AVCO CORPORATION

ADDRESS: POST OFFICE BOX 210

NASHVILLE, TENNESSEE 37202

DEVELOPED BY: JERRY THOMPSON — DATE

TRANSFERRED TO: — DATE

TRANSFERRED TO: B J RAMEY — DATE

TRANSFERRED TO: — DATE

TRANSFERRED TO: — DATE

Below is the SSP outline. It outlines the specific forms to be used in the SSP for each applicable Supplier. The outline originates in the home office and is approved by the Chief of Procurement Quality. Any subsequent revision requires that same approval prior to effecting the change.

SSP OUTLINE

SUPPLIER: AVCO CORPORATION, AEROSTRUCTURES DIVISION, P.O. Box 210, Nashville, TN 37202

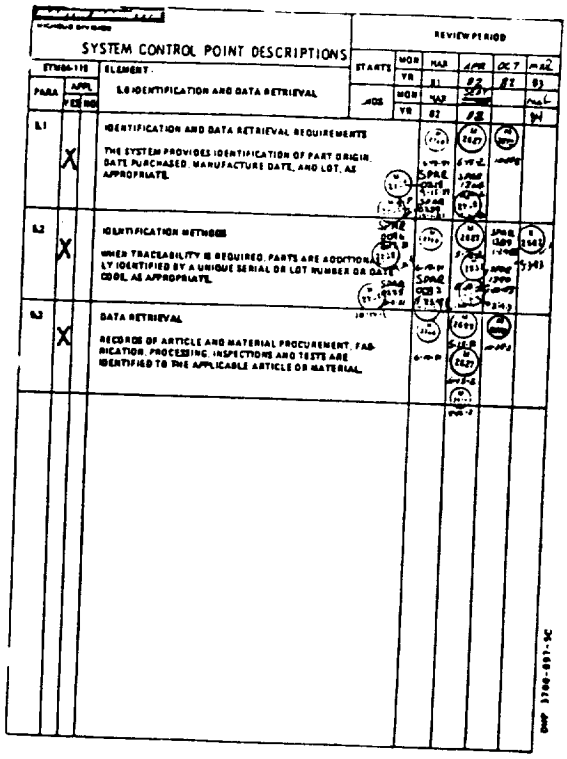
SECTION	FORM DESCRIPTION	APPLICABLE YES	APPLICABLE NO
1	GENERAL	△	
	SSP OUTLINE	●	
	COORDINATION/CONTACT INFORMATION	●	
	DOCUMENT REQUIREMENTS AND APPROVAL RECORD	●	
	QUALIFICATION/FIRST ARTICLE RECORD	●	
	TOOLING VALIDATION/TEST EQUIPMENT CERTIFICATION RECORD	●	
2	QUALITY SYSTEM SURVEILLANCE	△	
	SYSTEM CONTROL POINT DESCRIPTIONS	●	
3	PROCESS SURVEILLANCE	△	
	PROCESS CONTROL POINT DESCRIPTIONS	●	
	PROCESS VALIDATION MATRIX	●	
4	DOCUMENTATION REVIEW	△	
	DOCUMENTATION REVIEW LOG	●	
5	PRODUCT VERIFICATION	△	
	PRODUCT CONTROL POINT LOG	●	
	PRODUCT VERIFICATION PLAN	●	
	PVP APPLICABILITY PART NUMBER LISTING	●	
6	NON-CONFORMANCE DATA	△	
	WARR LOG	●	
	SPAR LOG	●	
	SUPPLIER PREVENTIVE ACTION REQUEST	●	
7	SUPPLIER PERFORMANCE	△	
	WEEKLY REPORT LOG	●	
8	SHIPPING DATA	△	
	SHIPPING RECORDS	OPTIONAL	

REVISION	DATE	APPROVAL
ORIG.	3-2-81	D. M. King, CHIEF PROCUREMENT QUALITY

DMF 3700-001 (10/81)

You can see from the outline what the SSP contains as indicated by the broad categories, general information, Quality Systems Surveillance, Process Surveillance, Documentation, Product Verification, Non-conformance Data, Supplier Performance Data and Shipping Data. The SSP is the Representative's Bible.

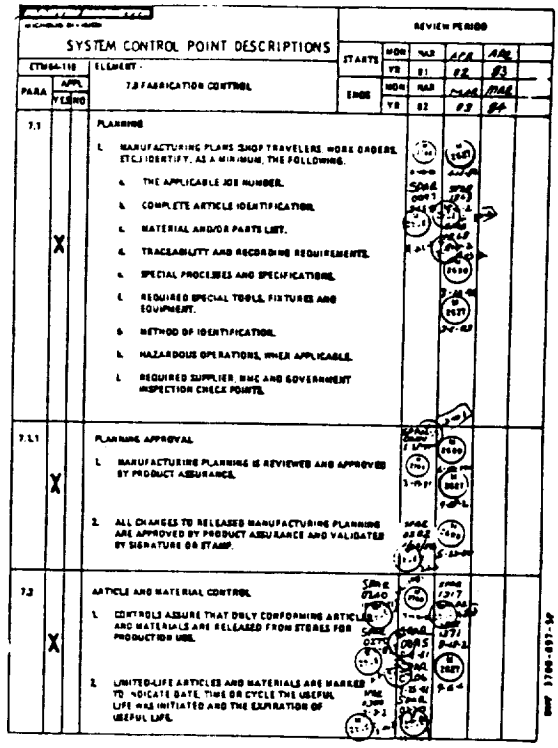
Here are two pages from the system Surveillance Section.


SYSTEM CONTROL POINT DESCRIPTIONS

REVIEW PERIOD

ETM64-110 ELEMENT: 6.0 IDENTIFICATION AND DATA RETRIEVAL

PARA	APPL YES	APPL NO	
6.1	X		IDENTIFICATION AND DATA RETRIEVAL REQUIREMENTS THE SYSTEM PROVIDES IDENTIFICATION OF PART ORIGIN, DATE PURCHASED, MANUFACTURE DATE, AND LOT, AS APPROPRIATE.
6.2	X		IDENTIFICATION METHODS WHEN TRACEABILITY IS REQUIRED, PARTS ARE ADDITIONALLY IDENTIFIED BY A UNIQUE SERIAL OR LOT NUMBER OR DATE CODE, AS APPROPRIATE.
6.3	X		DATA RETRIEVAL RECORDS OF ARTICLE AND MATERIAL PROCUREMENT, FABRICATION, PROCESSING, INSPECTIONS AND TESTS ARE IDENTIFIED TO THE APPLICABLE ARTICLE OR MATERIAL.

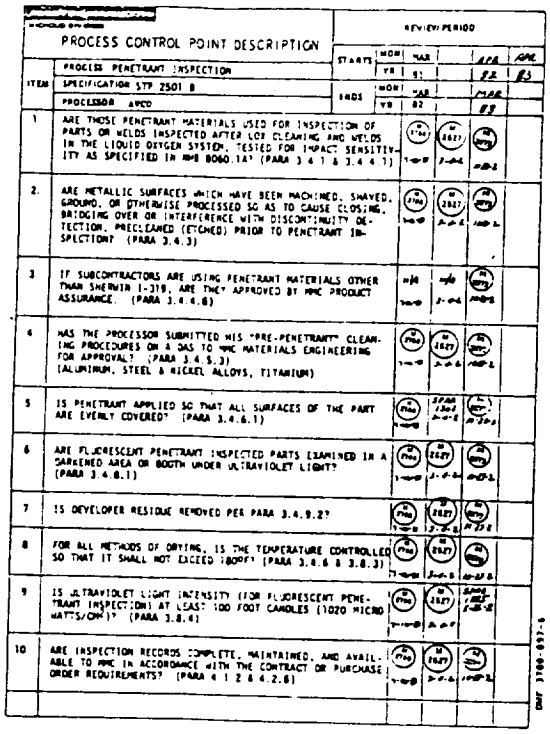

SYSTEM CONTROL POINT DESCRIPTIONS

REVIEW PERIOD

ETM64-110 ELEMENT: 7.0 FABRICATION CONTROL

PARA	APPL YES	APPL NO	
7.1	X		PLANNING 1. MANUFACTURING PLANS (SHOP TRAVELERS, WORK ORDERS, ETC.) IDENTIFY, AS A MINIMUM, THE FOLLOWING. a. THE APPLICABLE JOB NUMBER. b. COMPLETE ARTICLE IDENTIFICATION. c. MATERIAL AND/OR PARTS LIST. d. TRACEABILITY AND RECORDING REQUIREMENTS. e. SPECIAL PROCESSES AND SPECIFICATIONS. f. REQUIRED SPECIAL TOOLS, FIXTURES AND EQUIPMENT. g. METHOD OF IDENTIFICATION. h. HAZARDOUS OPERATIONS, WHEN APPLICABLE. i. REQUIRED SUPPLIER, NMC AND GOVERNMENT INSPECTION CHECK POINTS.
7.1.1	X		PLANNING APPROVAL 1. MANUFACTURING PLANNING IS REVIEWED AND APPROVED BY PRODUCT ASSURANCE. 2. ALL CHANGES TO RELEASED MANUFACTURING PLANNING ARE APPROVED BY PRODUCT ASSURANCE AND VALIDATED BY SIGNATURE OR STAMP.
7.2	X		ARTICLE AND MATERIAL CONTROL 1. CONTROLS ASSURE THAT ONLY CONFORMING ARTICLES AND MATERIALS ARE RELEASED FROM STORES FOR PRODUCTION USE. 2. LIMITED-LIFE ARTICLES AND MATERIALS ARE MARKED TO INDICATE DATE, TIME OR CYCLE THE USEFUL LIFE WAS INITIATED AND THE EXPIRATION OF USEFUL LIFE.

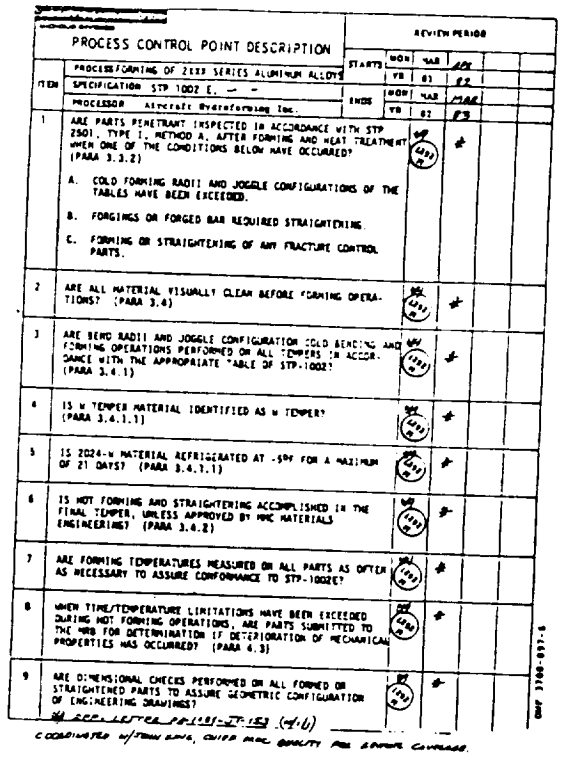
The total section is 20 pages long and addresses every element of the Quality System. These two pages are addressing "Identification and data retrieval" and "Fabrication Control." The review period is established and the requirement is that each element of the Quality System will be reviewed a minimum of once annually. These periods can be tightened but not loosened. You will see "SPAR" annotated on these sheets. When the review detects a deficiency, it is documented to the Supplier for corrective and non-recurrence action. We have a form titled Supplier Preventive Action Request (SPAR) that we use as a vehicle for documenting, tracking and resolving the deficiency.



PROCESS CONTROL POINT DESCRIPTION

PROCESS: PENETRANT INSPECTION
SPECIFICATION: STP 2501 B
PROCESSOR: AMCO

ITEM		REVIEW PERIOD
1	ARE THOSE PENETRANT MATERIALS USED FOR INSPECTION OF PARTS OR WELDS INSPECTED AFTER LOX CLEANING AND WELDS IN THE LIQUID OXYGEN SYSTEM, TESTED FOR IMPACT SENSITIVITY AS SPECIFIED IN MMS 8060.1A? (PARA 3.4.1 & 3.4.4.1)	
2.	ARE METALLIC SURFACES WHICH HAVE BEEN MACHINED, SHAVED, GROUND, OR OTHERWISE PROCESSED SO AS TO CAUSE CLOSING, BRIDGING OVER OR INTERFERENCE WITH DISCONTINUITY DETECTION, PRECLEANED (ETCHED) PRIOR TO PENETRANT INSPECTION? (PARA 3.4.3)	
3	IF SUBCONTRACTORS ARE USING PENETRANT MATERIALS OTHER THAN SHERWIN I-319, ARE THEY APPROVED BY MMC PRODUCT ASSURANCE. (PARA 3.4.4.8)	
4	HAS THE PROCESSOR SUBMITTED HIS "PRE-PENETRANT" CLEANING PROCEDURES ON A DAS TO MMC MATERIALS ENGINEERING FOR APPROVAL? (PARA 3.4.5.3) (ALUMINUM, STEEL & NICKEL ALLOYS, TITANIUM)	
5	IS PENETRANT APPLIED SO THAT ALL SURFACES OF THE PART ARE EVENLY COVERED? (PARA 3.4.6.1)	
6	ARE FLUORESCENT PENETRANT INSPECTED PARTS EXAMINED IN A DARKENED AREA OR BOOTH UNDER ULTRAVIOLET LIGHT? (PARA 3.4.8.1)	
7	IS DEVELOPER RESIDUE REMOVED PER PARA 3.4.9.2?	
8	FOR ALL METHODS OF DRYING, IS THE TEMPERATURE CONTROLLED SO THAT IT SHALL NOT EXCEED 180°F? (PARA 3.4.6 & 3.8.3)	
9	IS ULTRAVIOLET LIGHT INTENSITY (FOR FLUORESCENT PENETRANT INSPECTION) AT LEAST 100 FOOT CANDLES (1020 MICRO WATTS/CM²)? (PARA 3.8.4)	
10	ARE INSPECTION RECORDS COMPLETE, MAINTAINED, AND AVAILABLE TO MMC IN ACCORDANCE WITH THE CONTRACT OR PURCHASE ORDER REQUIREMENTS? (PARA 4.1.2 & 4.2.8)	



PROCESS CONTROL POINT DESCRIPTION

PROCESS: FORMING OF 2XXX SERIES ALUMINUM ALLOYS
SPECIFICATION: STP 1002 E.
PROCESSOR: Aircraft Hydroforming Inc.

ITEM		REVIEW PERIOD
1	ARE PARTS PENETRANT INSPECTED IN ACCORDANCE WITH STP 2501, TYPE I, METHOD A, AFTER FORMING AND HEAT TREATMENT WHEN ONE OF THE CONDITIONS BELOW HAVE OCCURRED? (PARA 3.3.2) A. COLD FORMING RADII AND JOGGLE CONFIGURATIONS OF THE TABLES HAVE BEEN EXCEEDED. B. FORGINGS OR FORGED BAR REQUIRED STRAIGHTENING. C. FORMING OR STRAIGHTENING OF ANY FRACTURE CONTROL PARTS.	
2	ARE ALL MATERIAL VISUALLY CLEAN BEFORE FORMING OPERATIONS? (PARA 3.4)	
3	ARE BEND RADII AND JOGGLE CONFIGURATION COLD BENDING AND FORMING OPERATIONS PERFORMED ON ALL TEMPERS IN ACCORDANCE WITH THE APPROPRIATE TABLE OF STP-1002? (PARA 3.4.1)	
4	IS W TEMPER MATERIAL IDENTIFIED AS W TEMPER? (PARA 3.4.1.1)	
5	IS 2024-W MATERIAL REFRIGERATED AT -5°F FOR A MAXIMUM OF 21 DAYS? (PARA 3.4.1.1)	
6	IS HOT FORMING AND STRAIGHTENING ACCOMPLISHED IN THE FINAL TEMPER, UNLESS APPROVED BY MMC MATERIALS ENGINEERING? (PARA 3.4.2)	
7	ARE FORMING TEMPERATURES MEASURED ON ALL PARTS AS OFTEN AS NECESSARY TO ASSURE CONFORMANCE TO STP-1002E?	
8	WHEN TIME/TEMPERATURE LIMITATIONS HAVE BEEN EXCEEDED DURING HOT FORMING OPERATIONS, ARE PARTS SUBMITTED TO THE MRB FOR DETERMINATION IF DETERIORATION OF MECHANICAL PROPERTIES HAS OCCURRED? (PARA 4.3)	
9	ARE DIMENSIONAL CHECKS PERFORMED ON ALL FORMED OR STRAIGHTENED PARTS TO ASSURE GEOMETRIC CONFIGURATION OF ENGINEERING DRAWINGS?	

COORDINATED W/TEAM LDR, CHIEF MMC QUALITY FOR ERROR COVERAGE.

These pages are from the process section. One shows a process being done in-house and the surveillance is done locally. The other addresses a process which is subcontracted and the Representative has to request support at the remote site for surveillance. When that has been accomplished and objective evidence has been provided to the local Representative, he up dates his record. Again, we see the sheet annotated with "SPAR" when deficiencies were noted.

We require that all Process Instructions which implement our approved process specifications be approved and validated. The validation activity is documented on a Matrix as shown here.

MARTIN MARIETTA

PROCESS VALIDATION MATRIX

SUPPLIER: AVCO Aerostructures, AVCO Corporation, P.O. Box 210, Nashville, Tenn. 37202

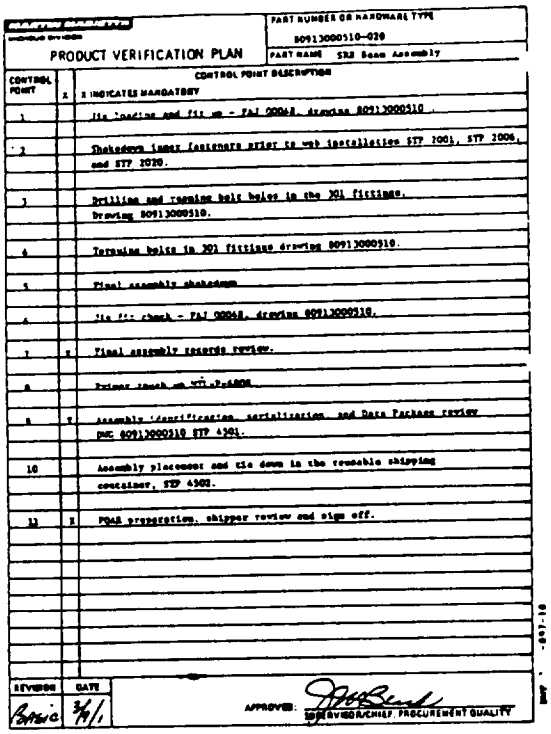
SHEET 3 OF 3

PROCESS INSTRUCTION NUMBER	SPECIFICATION NUMBER AND NAME	PROCESS PERFORMED BY	IN SE IS	DAS APPROVED	VALIDATION SIGNATURES	DATE	REMARKS
MPI 10354	STP 3003F Epoxy Coating Application	AVCO	YES		MMC / GOVT		
AVCO Process Proc. 123	AVCO Process Proc. 123 Ball Texting	AVCO	NO	AMH-014	MMC / GOVT		
MPI 1033	STP 4503+Amd.1-2 Strippable Coat-ing	AVCO	YES	AMH-030	MMC / GOVT		
MPI 10354	STP 5002A+Amd. 1 Solvent Cleaning	AVCO	YES	AMH-008	MMC / GOVT		
Q.C.D. 3017	STP 4501B+AVCO Pro. Proc. 72A to 4501B	AVCO	YES	ADI-021	MMC / GOVT		
QPI 052.12-4	M11-1-6068E Mag Particle Inspection	AVCO	YES		MMC / GOVT		
Bennett Pro. Proc. 8948	STP 8002,M11-U-6068F+Amd.1-3 H.T. of 7050 Al	Bennett Heat Tr. Newark, N.J.	YES		MMC / GOVT		
Hercules Heat Corp. Pro.Proc. 41	M11-U-6875F Heat Treat of Steels	Hercules H.T. Brooklyn, NY	YES		MMC / GOVT NO GST REQUIRED		
Pro. Proc. for Mag. Part. Insp	M11-1-6068E Mag. Part.Insp.	Liberty Ind. Fin Brentwood, NY	YES		MMC / GOVT		
LPF#25014	STP25018,M11-1-6068.Liquid Pen. Inspection	Liberty Ind.Fin. Brentwood, NY	YES	AMH-012	MMC / GOVT		
LPF#M25018	STP25018,M11-1-6068 Cleaning of Aluminum	Liberty Ind.Fin Brentwood, NY	YES	AMH-010	MMC / GOVT		
Q.C.D. 2017	STP45018 Inspecting AVCO Pro.Proc.7A	Ellanof Mac.Corp	YES	ADI-021	MMC / GOVT NO GST REQUIRED		

*REV FORMS. ALL SIGNATURES ACCOMPLISHED LOCALLY. ORIGINAL SIGNATURES ON FILE IN MMC OFFICE

Validation is accomplished at whatever level of subcontractor it occurs. This is evidenced by the listing of several sub-tier processors on the Matrix. Accomplishing this requires a considerable amount of coordination.

These pages are from the Product Verification Section. The first page outlines the control points for the particular part and indicates which of the surveillance checks are mandatory.



PRODUCT VERIFICATION PLAN

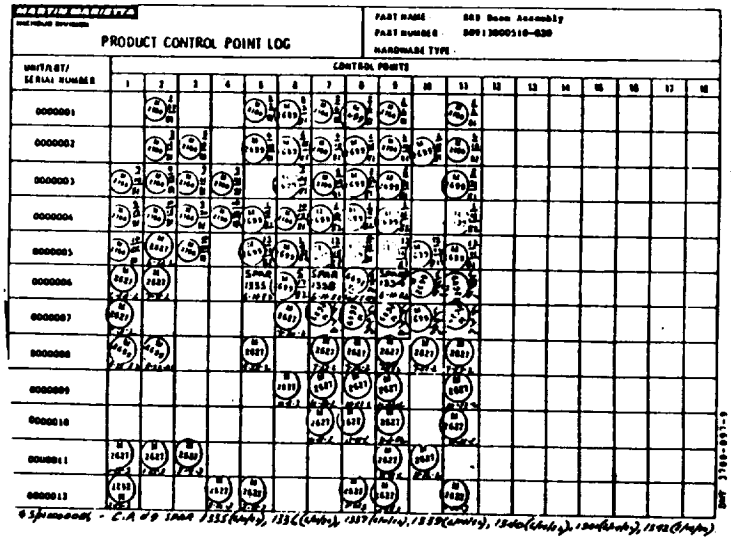
PART NUMBER OR HARDWARE TYPE: 8091J000510-020

PART NAME: STB Beam Assembly

CONTROL POINT	X	CONTROL POINT DESCRIPTION X INDICATES MANDATORY
1		Fit inspection and fit up - P&J 00068, drawing 8091J000510.
2		Takedown inner fasteners prior to web installation STP 2001, STP 2006, and STP 2020.
3		Drilling and reaming bolt holes in the JOI fittings. Drawing 8091J000510.
4		Terminal holes in JOI fittings drawing 8091J000510.
5		Final assembly shakedown
6		Fit ?? check - P&J 00068, drawing 8091J000510.
7	X	Final assembly records review.
8		Primer touch up VTL-P-6808.
9	X	Assembly identification, serialization, and Data Package review DWG 8091J000510 STP 4301.
10		Assembly placement and tie down in the reusable shipping container, STP 4302.
11	X	PDAR preparation, shipper review and sign off.

REVISION	DATE
Basic	3/1/

APPROVED: SUPERVISOR/CHIEF, PROCUREMENT QUALITY



PRODUCT CONTROL POINT LOG

PART NAME: STB Beam Assembly

PART NUMBER: 8091J000510-020

HARDWARE TYPE:

UNITS OF/ SERIAL NUMBER	CONTROL POINTS 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
0000001																		
0000002																		
0000003																		
0000004																		
0000005																		
0000006																		
0000007																		
0000008																		
0000009																		
0000010																		
0000011																		
0000013																		

The second page, titled "Product Control Point Log" documents the control points verified on each specific part or lot of parts. Again as you note on S/N 0000006, the "SPAR" is used to document deficiencies. All Product Verification Plans require the approval of the Representative's Supervisor or the Chief of Procurement Quality.

We believe the SSP is a tool which, when developed and implemented properly by a trained Source Representative, facilitates requirements enforcement and provides assurance and objective evidence that requirements have been satisfactorily satisfied. The data provided by the SSP can be used to motivate the Supplier.

Our Representative has the visibility to give proper recognition where he is doing a good job and also proper recognition where he is lax. The key here is proper recognition. Even negative factors can be motivational if presented properly. For a surveillance program to provide the desired result, it must be planned and developed so that it is a tool and not a burden.

Success in the field is dependent upon an individual's ability to manage and motivate the Suppliers through continuous in-depth surveillance and prudent application of mandatory inspection points. We must be selective in picking our Representatives and then adequately train them in the philosophy and implementation of the program.

Our Representatives need to have a good understanding of inspection methods and equipment and should be able to read blueprints and specifications with understanding. However, as important, or possibly more importantly they need to possess good communication skills. They need to like people and be able to get along with people. Think about it, in the field they will be interfacing with all types and levels of Supplier personnel and they can't be successful if they can't get along with and communicate with people. Our Representatives need to be self starters, motivated from within. Most times, there is no one there to "kick" them out in the morning or see that they are where they should be, when they should be, and doing what they should be doing. When you consider these things, it's apparent that not everyone is suited for the job. Just because a person is a good inspector doesn't mean they can be successful in the field. The job goes beyond inspection. We've already said there is no way we can inspect everything.

The persons who are truly effective and successful in the field have those qualities mentioned above and also possesses a sustained initiative and inquisitiveness. They continually wonder, ask questions and probe at all aspects of the Suppliers activities. They are "on top" of things.

From necessity, they become a "jack of all trades" and are always alert to potential problem situations. They know what they don't know and will summon "expert" assistance when needed. This is important, if someone doesn't know but doesn't know they don't know, they can make erroneous judgments . If someone doesn't know and doesn't want to admit it and won't call for assistance, the result is the same.

With one exception, I purposely have not mentioned any specific requirements we need to develop relative to materials, processes or traceability. The reason being that I believe those requirements exist, are known and understood by the Contractors. Those requirements are satisfied adequately to the extent we do a good job if imposing those requirements, assuring our subcontractors recognize them and understand them and then we do a good job of enforcing them. Just as with any requirement or law--lax enforcement breeds laxity of adherence. Don't enforce the speed laws and people will speed. Don't enforce the requirements you have imposed and requirements won't be fulfilled.

The one item I did mention was validation of "how to" process instructions. I believe there is no area where we can better spend our money than in the assurance of the adequacy of processes.

In summary, the most effective way to control Suppliers, whether it be materials, processes, traceability, etc., is to--assure that the requirements are clearly and completely defined in the purchase documentation--assure that the Supplier understands each requirement and how those requirements are to be satisfied--enforce those requirements with a planned Surveillance Program implemented by a trained Source Representative. Use the Surveillance Program to evaluate performance of the Supplier and the Source Representative. You can assure Supplier control to the extent you effectively accomplish the tasks of definition enforcement and evaluation.

PROCESS INTEGRITY, CONTROLLING THE VARIABLES

JOSEPH M. KOZMA
QUALITY CONTROL MANAGER
HEMET CASTING COMPANY

One of the axioms or truisms of manufacturing is that no two objects are ever made exactly alike. In todays high technology fields, complex and diversified processes are used to manufacture components for ultimate use in the sucess of a mission. Due to variables which may be encountered, controls must be established. One means of control, known as process integrity assists in eliminating the risk of non-conformance. No process is perfect, but sound guidelines established by the prime contractor with additional input from the subcontractor, enhances the integrity of any product. Responsibility for process integrity has taken a turn in recent years. No longer is the prime contractor procuring products sub-par to specification standards, but instead is demanding controls from the sub contractor to eliminate poor performance. With these demands, subcontractors must evaluate ways to maintain the high level of controls necessary to meet contractual requirements. Prime contractors and subcontractors alike have a responsibility to establish control guidelines. There are many ways to control the integrity of the process and much depends on the product being manufactured. The following items are not by any means all inclusive, but are areas which I have found through experience to have an impact on the end product in the Investment Casting Industry.

1. Communications - Process integrity has more often than not suffered due to a lack of adequate communications between participating groups. This failure to communicate has led to schedule delays and improper corrective action necessary to maintain process controls. Prime and subcontractors alike are responsible for the establishment of a communications system. Established lines of communication and proper personnel to contact at the onset of the contract is critical. Most prime contractors request their procurement personnel interface with the foundry and coordinate the necessary information to the proper channel. No matter who the designate may be, timely exchange of information is paramount. It has been experienced all too often that mis-communication or no communication takes place when requests are made verbally and no followup takes place in writing. The old manufacturing saying "We have been selling it that way for years" no longer applies. If the end item is not to specifications, communicate and resolve the issue. The establishment of this vital link is a step in sound process controls.

2. Standardization of Specifications - Many specifications utilized by prime contractors are old and outdated; especially the Mil-Specs. All too often prime contractors will procure investment castings to antiquated sand casting specifications. Government should coordinate with the Investment Casting Institute to bring forth specifications conducive to modern day high technology investment manufacturing and to establish sound specification guidelines by which to procure. As an example, Specifications such as MIL-I-6866 for Penetrant inspection relates no acceptance criteria, but how often do you see this callout on a contract with no additional acceptance guidelines? How does the prime contractor expect the subcontractor to maintain control of the process when technically nothing has been set up as an acceptance guideline other than unwritten practice. In the same light, interpretation of a given specification can have a major impact on the subcontractors ability to maintain integrity in a process. Other areas of concern to the subcontractor is timely transmittal of the current revisions to required specifications which will be utilized in manufacturing the product. Callout of specifications with revisions on requests for quotations (an area sadly neglected) and timely

PAGE 30 INTENTIONALLY BLANK

responses to subcontractors requests to the prime contractor for review of questionable areas within a specification. All of the above areas need to be clearly defined by prime contractors before process controls can be instituted.

3. Prime Contractors Surveys - The foundries of today who compete for the contracts of "state of the art" castings can ill afford not to comply with prime contractor requirements. Prime contractors who survey facilities for process controls should be able to recognize a subcontractors cababilities. An adequately trained survey staff who properly completes a survey of a prospective subcontract facility can determine immediately if that facility will have a problem meeting contractual requirements and maintaining the process integrity necessary to meet scheduling, dimensional or non-destructive testing controls. It is the prime contractors responsibility to insure subcontractor compliance, even before the contract is placed. An area which needs serious consideration is the number of surveys performed yearly. Many prime contractors survey a subcontractor by divisions which may mean two or three surveys yearly by one prime contractor alone. When a facility such as an investment foundry has as many as seventy surveys a year by fifty different prime contractors, time/cost/manpower become a large questionmark. We must all work (at all levels) to standardize surveys into an accepted standard, usable by all, in order to reduce the high cost, the high manhours necessary to perform the surveys and better utilization of manpower. Surveys play a key part in assuring process integrity.

4. Documented Controls - Due to the complexity of producing a product within an investment foundry, processes must be documented throughout the facility and constantly monitered by both Quality and Engineering personnel. Controls must be established in writing and constant inspection by Quality Control to insure compliance is a key element to controlling the variables of process. As an example, raw ingots received from suppliers must have proper certification of composition and immediate analysis by the foundry of the prime ingot will insure process stability through analysis. Heat treat surveys should be monitered closely for compliance with customer calibration requirements, especially when performed by an outside source. The foundry must survey and maintain a subtier control file to insure full compliance with customer specification requirements. Inspection results of dimensional findings, properly documented, will show trends, and if utilized properly, can have a drastic impact on controlling processes. All of these controls are primarily subcontractor requirements and primarily internal, but prime contractors must be assured at all times that these controls are in fact, in effect. This is best done through the prime contractors survey reports.

5. On Site Inspection - Here is probably one of the prime contractors best tools in controlling the product. The utilization of Source Inspection has several positive aspects. In a high volume subcontract facility, a resident Source Inspector can accomplish more to maintain the integrity of the product than any single or combination of elements within the prime contractors establishment. Modern cost effective means of source utilization can have a high impact on piece price reduction, higher yields in shipments and immediate interface with the prime contractors Quality or Engineering groups when problems do arise. The use of either the prime contractors own representative as a resident or the use of a contracted resident inspector, such as "FLAIR" has proven "On Site Inspection" to work and has reduced problems drastically.

6. Equipment - Todays modern foundry is constantly striving to reach new heights in the field of controlling variables inherent with the complexities of the processes. Looming on the horizon and staring the investment foundry in the face is the probability of prime

contractors requiring the foundrys to produce a product more and more in the advanced state of completion. Instead of simply producing a raw casting foundrys will be required to perform machining and finishing details. At this point, many prime contractors are realizing machine costs and finish costs are prohibitive inhouse and therefore are electing to have the foundry produce the product complete at a slightly higher cost but for less than the prime contractor can do the job. With these demands, newer equipment must be procured and additional variables will have to be dealt with. During the twentieth century, great strides have been made in foundry science and technology. Foundry equipment has undergone a major revolution, permitting a high degree of mechanization, which better controls uniformity. Within a foundry repeatability from one part to the other is best obtained by automation and removal of the human variable. Inspection equipment such as the coordinate measurement machine interfaced with a high speed computer, has been able to reduce inspection time, supply more useful data and allow interfacing computer technology to statistical control.

7. Training - Adequate training of personnel responsible for insuring process integrity is an ongoing requirement. The twentieth century forces of change have greatly affected Quality functions. Modern methods of control of various processes have little meaning if the person monitering the controls lacks knowledge of its function. Government, Prime contractors and subcontractors all have a responsibility to insure properly trained personnel are available to monitor these processes. Selection of personnel based on factors conducive to process knowledge by an entity is paramount. Personnel who are selected and trained must undergo additional skill development training periodically in order to insure keeping abreast of process changes. Areas of interest such as non-destructive testing can have mutual gains during training sessions if both prime contractors and sub-contractors partake in the training together. This type of training allows mutual understanding of interpretation of specifications and broadens the scope of expertise in the field.

8. Statistical Controls - Modern Computers have enhanced the meaning of statistics. Inspection equipment of today, automated machinery, and higher skilled personnel allow the use of statistics to maintain process integrity. Proper documentation of findings at any stage of the process allows Quality to determine trends early enough to have a positive meaning. Computer aided charts and graphs used to control time, temperatures, pressures, viscosity, dimensional results, heat treat results, chemical and physical test results, to name a few are now being implemented in the modern investment foundry to control process integrity. These statistical results enhance price controls as well as variable controls. The innovative controls of tomorrow are made by the people working to maintain process integrity today.

ABSTRACT

(PROCESS INTEGRITY)

MANAGEMENT ROLE

- MEANS OF CONTROL
- SOUND GUIDELINES
- RESPONSIBILITIES
- SUBCONTRACTOR REQUIREMENTS

COMMUNICATIONS

- ADEQUACY
- FAILURE RESULTS
- INTERFACE RESPONSIBILITY

STANDARDIZATION OF SPECIFICATIONS

- UTILIZATION
- GOVERNMENT ROLE
- MIL- SPECS

PRIME CONTRACTOR SURVEYS

- RECOGNIZING CAPABILITIES
- PROPERLY TRAINED PERSONNEL

DOCUMENTED CONTROLS

- NEED TO DOCUMENT
- ESTABLISH IN WRITING
- VARIOUS PROCESS CONTROLS
- SUBTIER CONTROL

ON SITE INSPECTION

- PRIME CONTRACTORS TOOL
- ASPECTS OF UTILIZATION
- ADVANTAGES

EQUIPMENT

- FOR USE IN CONTROLS
- REPEATABILITY
- NEED FOR MODERNIZATION

TRAINING

- RESPONSIBILITY
- SELECTING PERSONNEL
- MOTIVATION

STATISTICAL CONTROL

- ADVANTAGE
- HIGH TECH USE
- PRODUCT YIELD

Uniform Interpretation of Quality Specifications

R. E. Fear, Deputy, DCASMA Inglewood

I. FOCUS: Mil I 45208A and MIL Q 9858A Requirements - Prime Contractor control of subs/vendors

A. Disagreements between Government and Prime Contractor over sufficiency magnitude controls

1. Result from misunderstanding/misinterpretation?
2. Degree of intent?

B. Disagreements not necessarily surprising or unusual

1. Emanate from procedures review and/or product deficiency
2. Time consuming, often counterproductive, emotional polorization

C. Solution - Working together - resolve current problems and avoid future disagreements.

II. Review MIL I/Q Requirements

A. DAR XIV Guidance regarding MIL I/Q requirements
B. Essential elements MIL I/Q
C. Guidance/requirements clear enough?
D. Mentality - Contractor latitude developing controls
E. Compare NHB 5300.4 (IB) and 5300.4 (1C)

III. Reduce misunderstanding/misinterpretations

A. H50 - H51
B. QAR Certification - PQAP Course
C. Post Award Conferences

IV. Degree of Intent

A. Quality attitude (zero defects) or just get by?
B. Government Source Inspection at sub-contractor facilities

V. Recommendations

A. Current Problems

1. Dialogue - Q specialist to Q specialist
2. Impasses - Elevate contracting officers

B. Avoidance

1. Determine bredth/depth of problems
2. Ascertain cause: misunderstanding, misinterpretation, degree of intent
3. Consider joint industry/government training
4. Consider revision of specs
   a. Incorporate guidance H50, 51
   b. Incorporate NASA specifics 5300.4 (1B) or 5300.4 (1C)

ALUMINUM TASK FORCE ON PLATE SOFT SPOTS

Paul V. Mara
Aluminum Association

ABSTRACT

This paper discusses the value and the limitations of using electrical conductivity for monitoring the strength of aluminum alloy plate. The variation of conductivity and mechanical properties with quench rate is detailed and the process for distinguishing metal with a marginal quench from metal with a good quench is described.

The factors which influence conductivity other than quench rates are also discussed. These factors include lot-to-lot variations in chemistry and laboratory-to-laboratory variations in test methods and test equipment. Because of these variations, it is concluded that conductivity is an excellent tool for in-process quality control where one operator using one instrument can check the conductivity of one piece of plate for differences which may indicate differences in quench rate. It is not, however, an effective acceptance test because of the large errors which can be introduced into the measurements. The final referee for metal integrity must remain the mechanical property tests.

PAGE 36 INTENTIONALLY BLANK

INTRODUCTION

In recent years the aircraft industry has made signficant strides to improve the integrity of its aircraft by improved methods of monitoring the quality of aluminum plate. One relatively common method used for this monitoring is electrical conductivity as measured with an eddy current instrument. While the metallurgical mechanisms involved in solution heat-treated plate are well documented and understood[1-12], caution must be exercised with respect to attempts to establish direct correlations between metallurgical characteristics and conductivity values. Although electrical conductivity measurements are useful, there are limitations to their precision and accuracy. This paper describes the experiences of aluminum producers with electrical conductivity and attempts to illustrate its usefulness as well as its limitations.

Electrical Conductivity Measurements by Eddy Currents

In the eddy current method of determining conductivity, the conductivity meter generates a high frequency alternating current which is passed through a coil in the probe. When this probe is placed on a piece of metal, secondary or eddy currents are formed in the metal to a depth of about 0.030 to 0.100", depending upon the characteristics of the metal and the meter being used. For a steady state condition, the eddy currents formed are a function of the metal's conductivity. The eddy current in the metal generates its own electro-magnetic field which acts on the field of the probe producing a change in the electrical characteristics of the coil circuit. The magnitude of this voltage change can be measured by the meter and indicates the near surface conductivity of the metal. The conductivity is usually expressed as a percentage of the 100% conductivity on the International Annealed Copper Standard (% IACS).

When alloying elements are added to aluminum they can have widely differing effects on electrical conductivity depending upon whether they are in or out of solid solution. When alloying elements are in solid solution, their effect on electrical conductivity is significant. Many of the non-strengthening elements such as vanadium, chromium, manganese and titanium have a dramatic effect on conductivity with little or no contribution to strength (Figure 1).[13] When alloying elements are out of solution (i.e., in discrete second phase constituents) their effect on electrical conductivity is much smaller, sometimes reduced by as much as a factor of 10.[14] Any thermal operation may affect the amount of dissolved elements in a plate and change the conductivity. In this manner, electrical conductivity measurements are useful for following changes in the amount of solute in solid solution through the quenching and aging processes. (Conductivity is, however, affected by other factors such as measurement errors, and these will be discussed in later sections.) The following section discusses the two main uses of conductivity, as a quality control tool in monitoring quench performance and in monitoring aging for corrosion resistant tempers.

Applications of Electrical Conductivity as a Quality Assurance Tool

Monitoring Quench Performance

During solution heat treatment, the various soluble elements in an alloy dissolve into solid solution. Quenching retains these elements in a supersaturated solid solution provided the quenching is rapid enough to prevent diffusion and precipitation. The temperature range where quench rate is critical is approximately 750°F-550°F for most alloys (Figure 2).[15] Above this range the super-saturation is too small to be of importance

and below this range the diffusion rate becomes insignificant. If the quench rate is not fast enough to prevent precipitation, then whatever solution precipitates during the quench will be in the form of coarse, non-homogeneous particles and will be unavailable for strengthening. The only correction for this situation is reheat treatment.

In general, a faster quench rate will produce a lower conductivity than a slower quench rate. However, the degree of change in conductivity depends on the quench sensitivity of the alloy. Figure 3 shows three different alloys and the changes in their conductivity with varying quench rates.[16] Alloy 6061 (Al-Mg-Si) shows very little change in the conductivity over a wide range of quench rates. On the other hand, 2024 (Al-Cu-Mg) and 7075 (Al-Zn-Mg-Cu) both show a significant degree of quench sensitivity over a wide range of quench rates. Therefore, if either the 2024 or the 7075 had a large variation in quench rate within a plate it would be detectable. The quench rates normally experienced in plate are from 10-100°F/second over the critical temperature range of 750°F-550°F. The effect of quench rate on yield strength is shown in Figure 4.[17]

To use conductivity to detect a variation in quench rate, it is desirable to allow enough time to lapse after quenching for the W-temper conductivity of the plate to stabilize (Figure 5).[18] This minimizes natural aging time as a variable. However, since processing of a plate in a mill may take up to several weeks, it is desirable to measure electrical conductivity in the freshly quenched W-temper. In this manner any problems with a quench system or practice can be promptly detected and corrected. It would be desirable to develop a correlation between W-temper electrical conductivity and T-temper properties that would assure a given W-temper conductivity would meet final properties. However, because of many variables in measurement, conductivity values are imprecise numbers. As a result, the correlation developed (Figure 6)[19] is also imprecise. The spread of the date precludes the establishment of an absolute value for an acceptable conductivity. By using a conservative number for W-temper an early warning system can be established to prevent marginal metal from being processed.

The second and perhaps most valuable use (considering current quench design) is to check the uniformity of the quench rate. The many variables which affect conductivity are minimized by having one operator use one instrument to measure the conductivity in several locations on one plate. A large difference in conductivity serves as a flag that further metallurgical evaluation is necessary.

Conductivity for Monitoring of Aging

When an alloy is aged after being solution heat-treated, the alloying elements in solution are precipitated out and electrical conductivity is increased. This makes it possible to monitor the aging process by measuring electrical conductivity. Advantage is taken of this when improved corrosion resistance is desired as in the T7-tempers. It is possible to verify that the correct degree of aging is achieved by using electrical conductivity measurements in conjunction with mechanical properties. This assumes that the conductivity of a plate of a given alloy is primarily effected by the aging operation.

Limitations of the Correlation Between Conductivity and Mechanical Properties

Since electrical conductivity is helpful for in-process monitoring and for indication of corrosion resistance, it is logical to attempt to use it for verification of mechanical properties. Figure 7 shows a yield strength vs. conductivity plot for lots of 7075 with a variety of quench rates.[20] For a given conductivity measurement, the width of the 95% confidence band about the regression line is approximately 18 ksi. This generates a high degree of uncertainty when one attempts to predict mechanical properties from electrical conductivity.

Reasons for the poor correlation fall into two categories:

1. Lot-to-lot variation of chemical composition.
2. Measurement errors such as those due to meter sensitivity, calibration standards, operation variations, temperature, and geometry effects.

Effect of Chemical Composition

One way of showing the effect of alloying elements on conductivity is to calculate the conductivity for the extremes for the composition range for the alloy. Table 1 shows the effects of the alloying elements on alloy 7075 resistivity.[21] (Resistivity is used for calculations, since resistivity effects are additive.) In these calculations, an assumption is made as to the percentage of each element in solution. Using this information and the minimum, nominal, and maximum compositions for 7075, it is possible to calculate the resistivity of these compositions. When converted to conductivity, the significant effect of composition variation on an alloy's electrical conductivity can be seen. The composition limits of the registered range of the alloy 7075 could result in a conductivity variation of 4% IACS.[20]

These theoretical calculations are supported by the example shown in Figure 8. Several lots of 7075 were quenched in warm or cold water and then checked for the W-temper electrical conductivity. Since magnesium has one of the larger effects on electrical conductivity, conductivity is plotted against magnesium content. Scatter bands for the data at both quench rates are shown. It is seen that for both quench rates the composition has a significant effect on conductivity. In fact, conductivities between 30.2 and 30.7% IACS (Fig. 8) can be achieved with two widely divergent quench rates due to the composition differences.[22]

Measurement Errors

In an industry survey testing program sponsored by the Aluminum Association, 1" thick samples of 2024-T351 and 2124-T851, 7075-T651, and 7075-T7351 were supplied by each of three aluminum producers. A fourth producer supplied samples of 2024-T351 and 2124-T851. All fourteen samples were sent to 8 different locations to have conductivity measured. These locations were either fabricating plants or technical centers of various companies. The samples had a 1" circle marked on one surface and the conductivity was measured inside the circle. Each location used its own standards and procedures for their measurements. The conductivity was measured at a temperature between 68° and 80°F on all samples. A list of instruments used is given in Table 2.

Figures 9A through 9D show the distribution of conductivities measured on each sample. Several samples showed a range of measured conductivities of 1% IACS. The total distribution of conductivity readings for the four samples of each alloy is shown in Figures 10A through 10D as well as the specific data for each instrument. Some of the instruments show a range of conductivities for a sample that scarcely overlapped other instrument's measurements. This variation between instruments accounted for the largest portion of variation in the trial. Variations such as these were surprising considering the familiarity of the test locations with conductivity procedures. The fallacy of trying to measure conductivity to 0.1% and using those values for lot acceptance is pointed out by these large variations.

Figure 11 shows the differences between readings when operator, mode of operation (AC or DC) or source of calibration standards are varied. One observation for each of these conditions was chosen as an example from data supplied by the industry survey. Figure 11a shows the readings of one operator plotted as a base line with the measurements made by a second operator plotted as a deviation from the base line. All

other variables were held constant for this data plot. A similar situation is true for figures 11b and 11c. While there are some variations between readings when these variables are considered, the variations are smaller than the instrument-to-instrument variability.

Error is also introduced if a test standard is used at a different temperature from the metal being measured or if a test standard is used with a different temperature coefficient from the material being tested. Figure 12 shows the error introduced when various metal standards were used for calibration and readings taken on aluminum at various temperatures.[23]

Geometry effects are shown in Figure 13 on readings taken near the edge of a piece of metal.[24] Depending on the type of eddy current instrument and its frequency, the depth and size of the region tested vary. Consequently, some instruments are more sensitive to measurements near an edge.

Conclusion

This paper has shown the value and the limitations of using electrical conductivity to assure the quality of aluminum plate. Electrical conductivity procedures are very useful as an in-process tool for monitoring quench variations over the surface of the plate. Conductivity is also a valuable tool for verifying that 7XXX series alloys in T7 type tempers are adequately aged to provide good corrosion resistance. However, it has been found that it is difficult to assure specific mechanical properties using electrical conductivity due to the many sources of variation. It is therefore concluded that the final decision on the acceptability of heat treated plate should depend on mechanical property testing.

BIBLIOGRAPHY

1. R. A. Mueller, "Conductivity Testing At the McDonnell Company," *Materials Evaluation,* April 1969.
2. D. Hagemaier and G. J. Basl, "Evaluation of Mechanical Properties of 2014-T6 Weldments by Nondestructive (Eddy Current) Methods," *Materials Evaluation,* January 1967.

   W. D. Rummel, "Characterization and Evaluation of 2014 Aluminum Alloy by Eddy Current Conductivity Techniques," *Materials Evaluation,* June 1966.
4. W. D. Rummel, "Theory of the Use of Eddy Current Conductivity Devices To Monitor Aluminum Alloys," *Materials Evaluations,* September 1966.
5. H. V. Pellegrini, "Assessing Heat Damage in Aluminum Alloys," *Metals Progress,* March 1980.
6. L. E. Cosgrove, "Quality Control Through Nondestructive Testing With Eddy Currents," *Nondestructive Testing,* September/October 1955.
7. D. Hunter and D. D. Seltzer, "Effect of Time and Temperature Exposures on the Conductivity and Mechanical Properties of 2014 Aluminum Alloy Using Eddy Current."
8. J. T. Vietz, "The Use of Eddy Current Conductivity Measurements for Checking and Predicting the Strength of Heat-Treated Aluminum Alloys," *Journal of the Australian Institute of Metals,* 13, No. 4, November 1968.
9. D. Hagemaier, "Verification of the -T73 Heat Treatment of 7075 Aluminum Alloy by Nondestructive Methods," *Nondestructive Testing.*
10. D. Hagemaier and R. Kleint, "Evaluating Aluminum Alloys by Nondestructive Tests," *Metal Proggress,* November 1964.
11. R. E. Kleint and G. B. Mathers, "Hardness-Tensile Relationship for 7075 Aluminum Alloy," *Nondestructive Testing,* November/December 1967.
12. W. D. Rummel and W. J. Arbegast, "Characterization of 2014, 2219, 6061 and 7075 Aluminum Alloy Heat Treat Response by Eddy Current, Conductivity, Hardness, and Mechanical Properties," Paper Summaries, National Spring Conference of the ASNT, March, 1981, Philadelphia, PA.
13. See ref. 12.
14. Aluminum Handbook, Vol. I, pg. 174, ASM 1967.
15. J. T. Staley, J. W. Evancho and R. E. Hannule, "Using the C-curve to Predict Properties of Precipitation Hardenable Aluminum Alloys, presented at the Fall Meeting of the TMS, AIME, October 1971.
16. Unpublished member company report. B. W. Lifka, "Conductivity Quench Criteria For Nos. 10 and 11 Horizontal Plate Heat Treaters," Alcoa Alloy Technology Report -56-79-HQ55, November 1979.
17. Unpublished member company report. T. E. Uruggink and L. A. Willey, "RE: Evaluation of the Quench Sensitivity of Aluminum Alloys, Alcoa Report HQ-13, dated 1957-07-19.
18. Unpublished member company letter. B. W. Lifka to R. J. Kegarise captioned "Conductivity Quench Criteria for Nos. 10 and 11 Horizontal Plate Heat Treaters — Use of Alloys other than 7075-W," dated 1978-09-27.
19. Unpublished member company data. S. A. Levy and K. N. Schuler to G. E. Spangler captioned "C-curve Data for 2124-T851," dated 197-09-12.
20. Unpublished member company data. G. B. Shockley to J. W. Barr, dated 1980-12-02.
21. See ref. 14.
22. Unpublished member company letter. B. W. Lifka to R. J. Kegarise captioned "Conductivity — Composition Correlation to 7075-W." dated 1980-10-27.
23. See ref. 9.
24. Unpublished member company data. R. H. Wygonik, 1980.

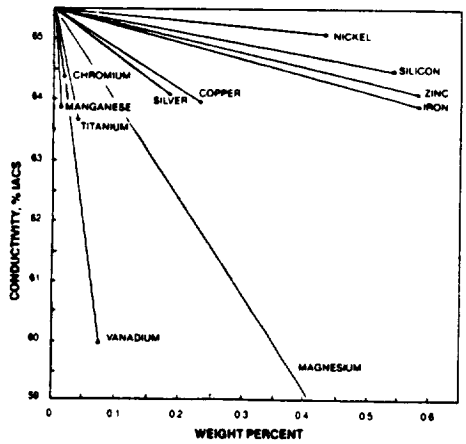


Figure 1 - Effect of alloying elements on the conductivity of aluminum.

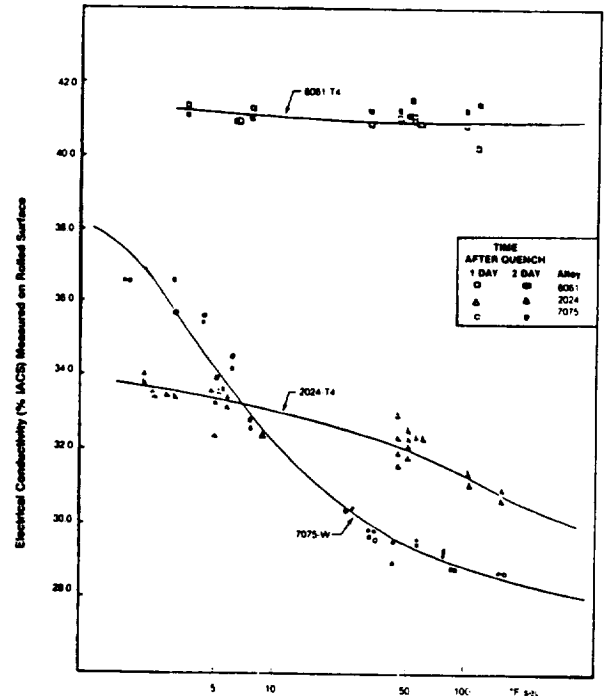


Figure 3 - Average Cooling Rate From 750 to 550° F. Effect of cooling rate during quenching on the electrical conductivities of several alloys. All readings taken by same operator with same instrument and same test conditions on adjacent sample of plate for a given alloy.

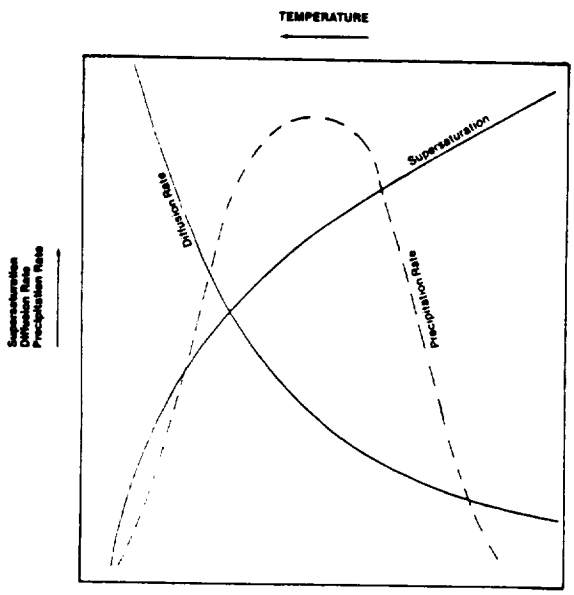


Figure 2 - This schematic illustrates how precipitation rate is dependent on both the supersaturation and the diffusion rate. The concept of a "critical temperature zone for quenching" is developed from this.

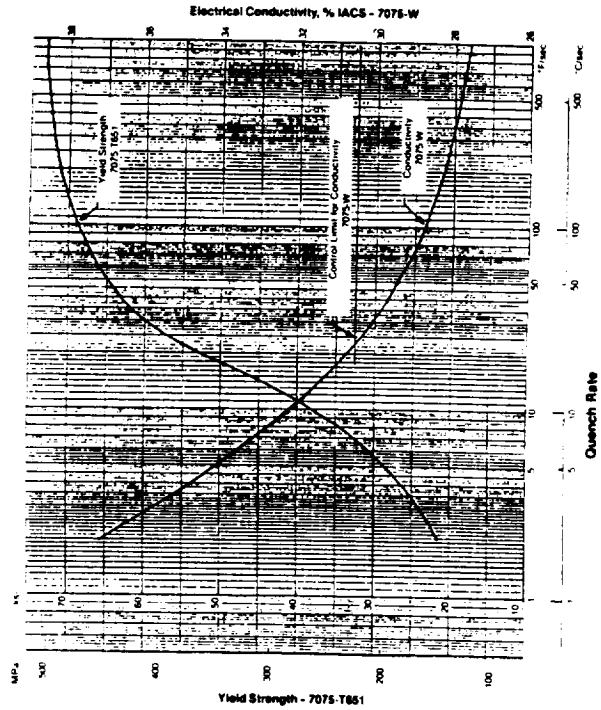


Figure 4 Average quench cooling rate between 399 to 288°C (750 to 550°F)

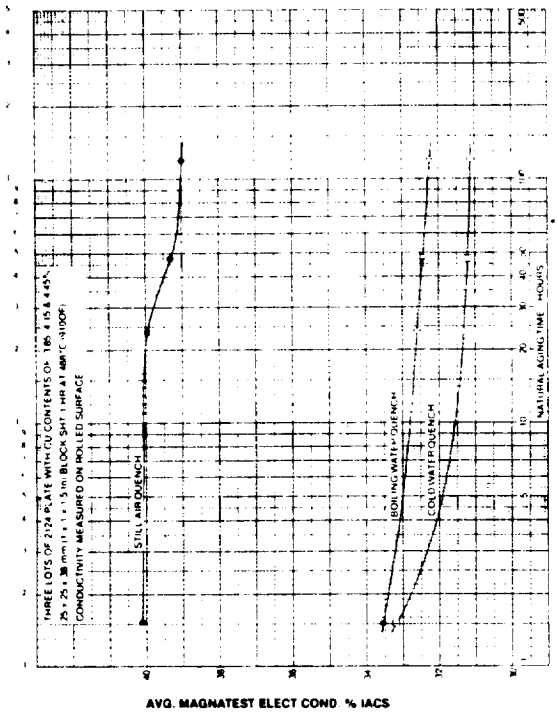


Figure 5 - Natural aging curves for 2124 quenched in various mediums

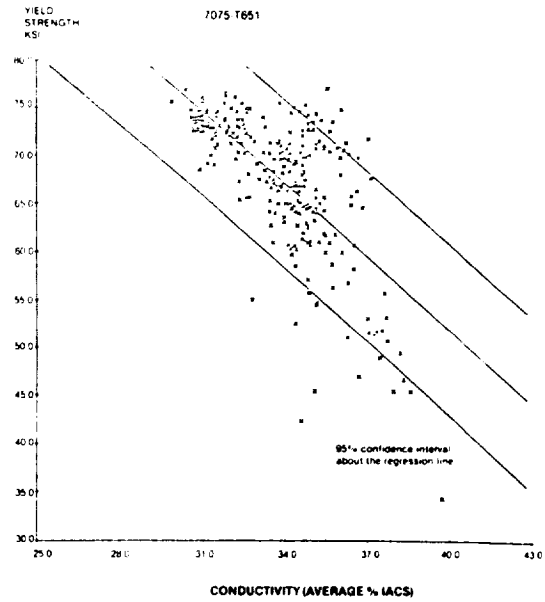


Figure 7 - The relationship between electrical conductivity and yield strength for 7075 T651 plate with a wide variation in fabrication practices. Rectangular tensile samples were machined from a location immediately below the conductivity reading

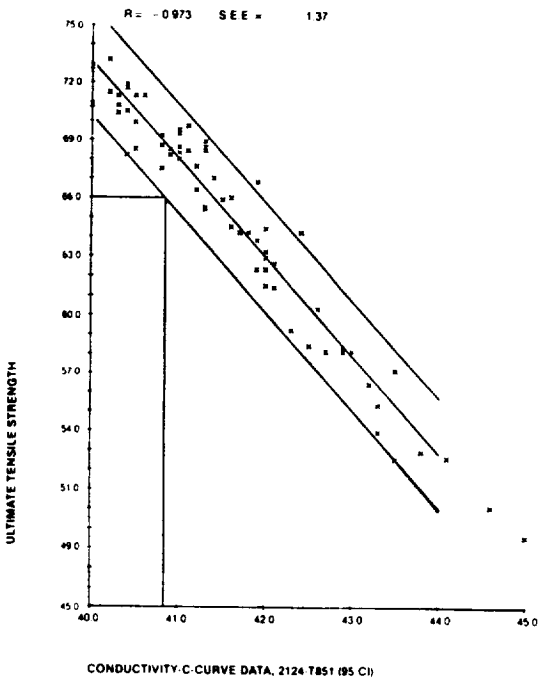


Figure 6 - Relationship between conductivity and ultimate tensile strength for 2124 T851 is shown above. Establishing a minimum conductivity of 40.8 % IACS as indicated would cause many plates with acceptable properties to be failed

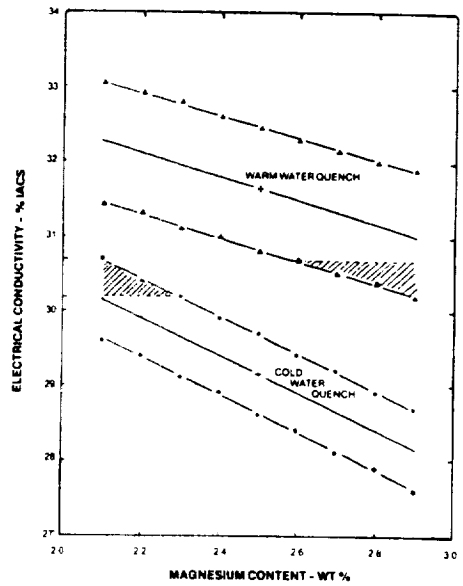


Figure 8 - Effect of magnesium content on electrical conductivity of 7075 W plate quenched in cold, 21°C (70°F), and warm, 82°C (180°F), water on a 12.7 mm (0.50) inch thick section. Plot shows the predicted mean and the 90% confidence limits. Conductivities of 30.2 to 30.7 fall into both bands. Thus, for this range, conductivity by itself cannot predict the adequacy of quenching

DISTRIBUTION OF CONDUCTIVITY VALUES
FOR ALLOY 7075-T7351

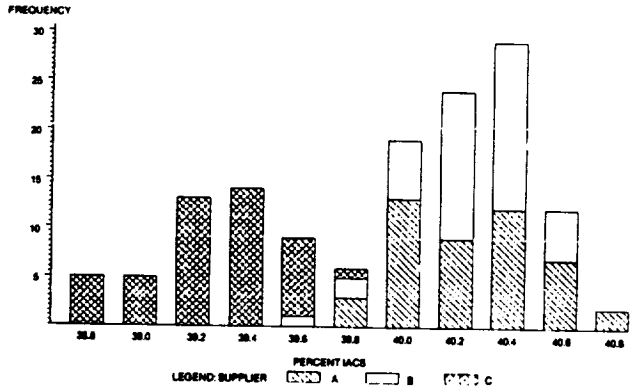


FIGURE 8A

DISTRIBUTION OF CONDUCTIVITY VALUES
FOR ALLOY 2024-T351

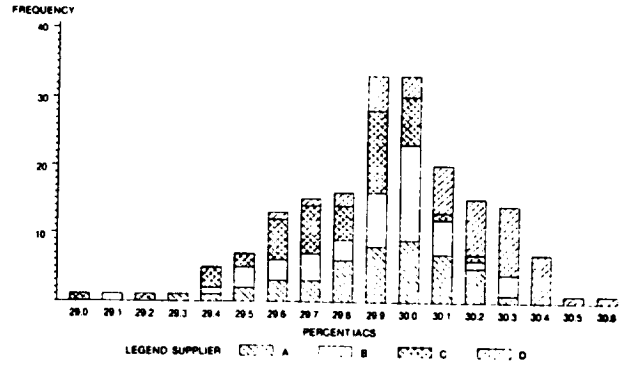


FIGURE 9D

DISTRIBUTION OF CONDUCTIVITY VALUES
FOR ALLOY 7075-T651

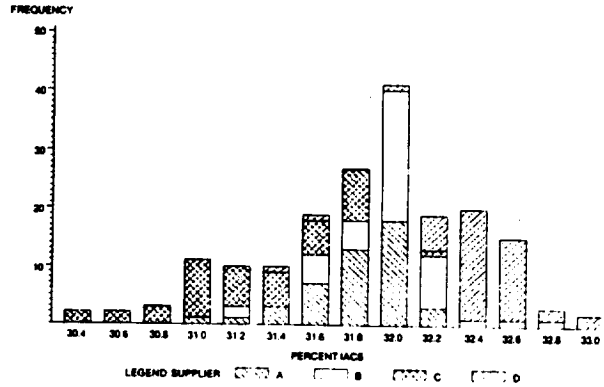


FIGURE 9B

DISTRIBUTION OF CONDUCTIVITY VALUES
FOR ALLOY 7075-T7351

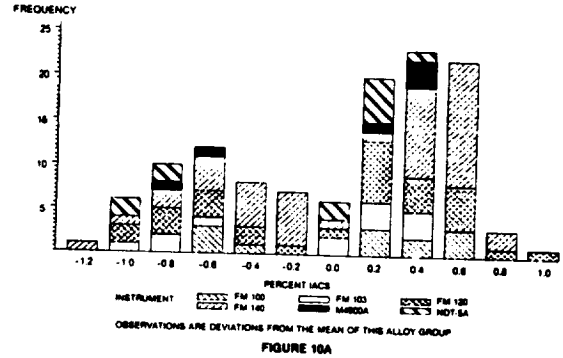


FIGURE 10A

DISTRIBUTION OF CONDUCTIVITY VALUES
FOR ALLOY 2X24-T851

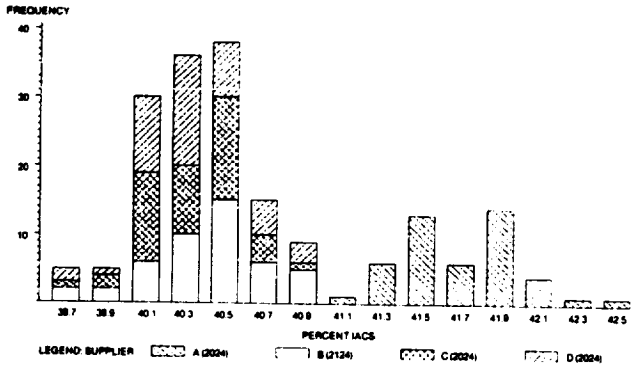


FIGURE 9C

DISTRIBUTION OF CONDUCTIVITY VALUES
FOR ALLOY 7075-T651

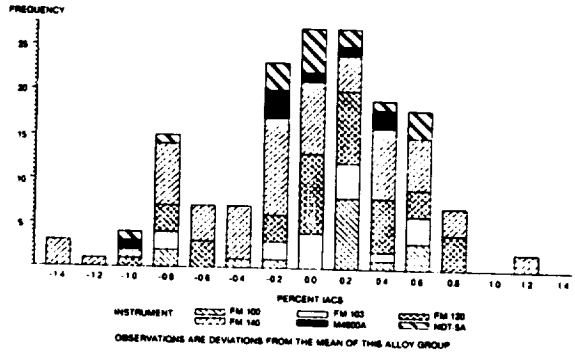


FIGURE 10B

FORGING SOFT SPOTS - THE
PROBLEM AND ITS RESOLUTION

Dennis P. Masterson
ALCOA

FORGING SOFT SPOTS

PROBLEM SOURCES:

1. Racking procedures that inhibit quench circulation through the heat treat load.

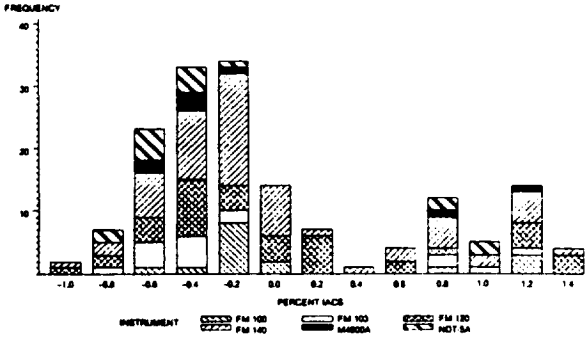
2. Sensitivity of overaged 7XXX-T7XXX tempers to the total aging process.

3. Forging configuration and its inherent influence on quench rate.

4. Homogeneous versus heterogeneous aging phenomena.

PAGE 96 INTENTIONALLY BLANK

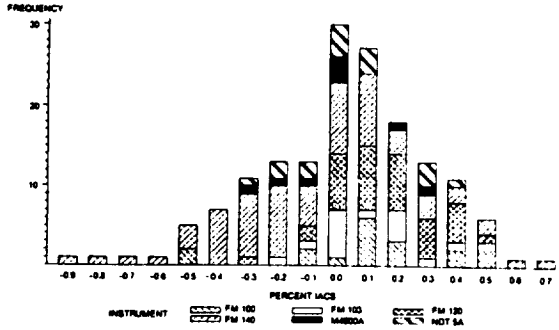
DISTRIBUTION OF CONDUCTIVITY VALUES FOR ALLOY 2X24-T851



OBSERVATIONS ARE DEVIATIONS FROM THE MEAN OF THIS ALLOY GROUP

FIGURE 10C

DISTRIBUTION OF CONDUCTIVITY VALUES FOR ALLOY 2024-T351



OBSERVATIONS ARE DEVIATIONS FROM THE MEAN OF THIS ALLOY GROUP

FIGURE 10D

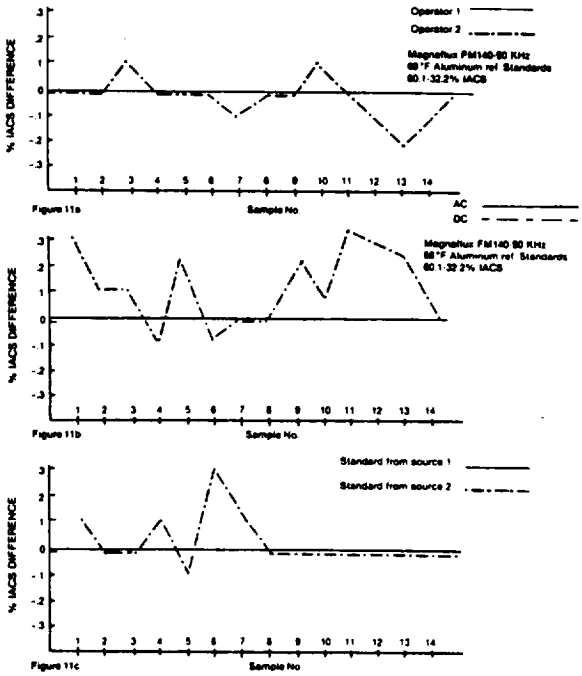
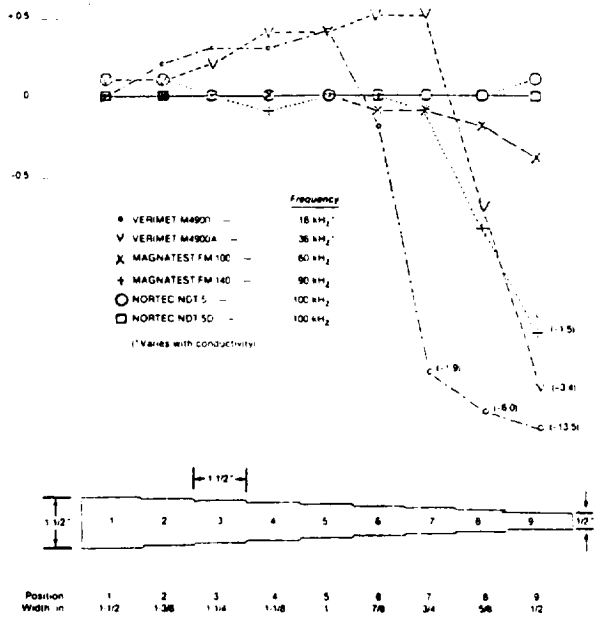


Figure 11 - Each figure shows the difference in conductivity readings when a single variable is altered. One set of readings is used as a base and the other set is plotted against it. In (a) the variable was operators, in (b) the variable was mode of operation, and in (c) it was calibrated standards



TEST SPECIMEN POSITION LOCATION

Figure 13 - Plot showing edge effects on conductivity measurements using various meters

TABLE 1

EFFECT OF MAJOR AND MINOR ALLOYING ELEMENTS ON ELECTRICAL CONDUCTIVITY OF 7075 ALLOY

Element	Resistivity Change Per Weight % of Element, Micro ohm-cm: In Solution	Out of Solution	7075 Alloy: % in Solution	Minimum Composition Wt %	Minimum Composition -cm	Nominal Composition Wt %	Nominal Composition -cm	Maximum Composition Wt %	Maximum Composition -cm
Cr	4.00	0.18	50	16	.481*	23	.481	28	.481*
Mn	2.94	0.34	100		.147*	35	.147	30	.147*
Si	1.02	0.088	50		.055*	10	.055	40	.055*
Ti	2.86	0.12	50		.045*	03	.045	20	.045*
V	3.58	0.28						05	
Zr	1.74	0.044						05	
Zn	0.094	0.023	100	5.0	.480	5.6	.526	6.1	.573
Mg	0.54	0.22	100	2.1	1.134	2.5	1.35	2.9	1.566
Cu	0.344	0.03	100	1.2	.413	1.6	.55	2.0	.688
Others				2.754		3.154		3.555	
Al				2.855		2.855		2.855	
Total				5.409		5.809		6.210	
% IACS				30.3		28.8		26.9	

TABLE 2

INSTRUMENTS USED IN ROUND ROBIN CONDUCTIVITY TRIAL

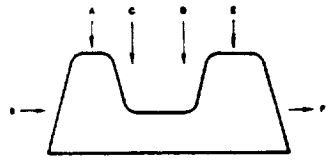
INSTRUMENT	FREQUENCY	MODE OF OPERATION	MANUFACTURER
FM 140	90 khz	AC/DC	Magnaflux Corp
FM 103	60 khz	AC	Magnaflux Corp
FM 100	60 khz	AC	Magnaflux Corp
NDT 5A	60 khz	AC/DC	Nortec
Verimet M4900A	Variable	AC/DC	K. J. Law
FM 120	60 khz	DC	Magnaflux Corp

PAGE 48 INTENTIONALLY BLANK

Inspection Reports
Thru 81-08-15
2" Grid and Multiple Location Inspect

I. 2XXX Alloys, 7XXX-T6XX Alloys (Multiple Locations)

Alloy/Temper	No. of Pcs. Inspected	No. of Pcs. Rejected
2014-T4E14	283	0
2014-T6/T61/T652	4,805	0
2024-T4/T6/T352	5,355	0
2219-T6/T352/T87	948	0
2219-T81	117	0
Total 2XXX	10,516	0
7050-T00/T74	884	0
7049-T352	76	0
7075-T6	11,857	0
7075-T652/T654	372	0
7075-T611	916	0
7075-T651	258	
Total 7XXX-T6XX	14,153	0



A		B		C		D		E		F	
R_b	BHN	R_b	BHN	R_b	BHN	R_b	BHN	R_b	BHN	R_b	BHN
77	134	82	136	75	130	74	119	77	134	76	122
83	136	83	137	74.5	119	77	134	76	136	76	122
81.5	134	82	136	75	130	76	136	80	130	77	134
80.5	132	81	133	76	136	79	136	80	130	76	122
80	130	81.5	134	80	130	79.5	139	79	136	77	124
79	126	75	126	79.5	129	80	130	76	136	77	124
75.5	123	76	126					73.5	117	77	124
76	126	76	133					74.5	119	76.5	127
80.5	133	72	114					76.5	121	77.5	128
79	136							77	134	79	136
80	130							74	119	81	133
79	136							75.5	121		
80	130							74	119		
80	130							75	126		
79.5	137							76	136		
80	130							75.5	121		
80	136							77	134		
80	130							76	122		
83	136							76.5	127		

Cond.	Cond.	Cond.	Cond.	Cond.	Cond.
40.5	40.6	40.9	41.5	40.4	40.9
40.0	40.5	40.5	41.0	40.7	40.7
40.5	41.0			40.6	40.5
40.0				40.6	
39.8				40.2	
39.6					

Inspection Reports
Thru 81-08-15
2" Grid and Multiple Location Inspect

II. 7XXX-T7XX (Multiple Location and 2" Grid)

Alloy/Temper	No. of Pcs. Inspected	No. of Pcs.* For Reheat Treat
7049-T73	823	134
7050-T736	1,503	224
7050-T73652	5,210	720
7075-T73	20,523	790
7075-T7351/53/54	4,406	250
7175-T736	5,116	30
7175-T73652	1,043	34
Total 7XXX-T7XXX	41,139	2,280

*Pieces for reheat treat defines forgings with surface EC 41.5% IACS and BHN hardness tests below specifications minima.

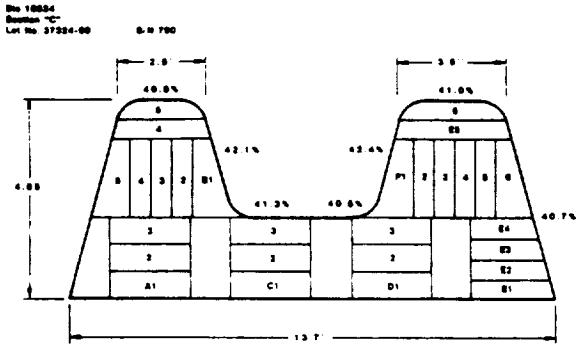
Suspect Forging Configurations
Mechanical Property Evaluation
Through 81-08-15

Alloy/Temper	No. of Suspect Configurations	No. of Ole Configurations Tested by Alcoa	No. of Ole Configurations Tested by Customer
7075-T73	40	30	10
7075-T73652	10	7	3
7075-T654	1	1	0
7049-T73	5	4	0
7050-T736	3	3	0
7050-T73652	17	14	1
7175-T736	11	2	2
7175-T73652	3	2	0
	101	60	16

Total Number of Tests Made 2000+

Suspect Forging Configurations
Mechanical Property Results
Alcoa and Customer Testing

Alloy/Temper	Configurations Tested	No. of Passing Min. Prop.	No. Failing Min. Prop.	Extend Prop. Below Min.
7075-T73	26	19	30	.3-.20 Ksi
7075-T73652	12	2	10	.1-.11 Ksi
7075-T654	1	0	1	.2-.11Ksi
7049-T73	4	1	3	.3- .8Ksi
7050-T736	3	3	0	—
7050-T73652	15	12	3	.5- .5 Ksi
7175-T736	4	1	3	.7- .6 Ksi
7175-T73652	2	2	0	—
	76	36	40	



Heat-Treat Standard Procedure

FIGURE 5

TABLE 2

History of 7075-T73/-T7352 Artificial Age Practices - Cleveland Works.

DATE (Year)	PROCESS	PRACTICE NO.
Prior to 1974	6-12 hrs. @ 225°F + 6-12 hrs. @ 325°F	---
1974 - 1976	6 hrs. @ 250°F + 4-8 hrs. @ 350°F	---
1976 - 1977	1 hr. @ 250°F + 4-8 hrs. @ 350°F	---
1977 - 1980	4 hrs. @ 250°F + 4-8 hrs. @ 350°F	1

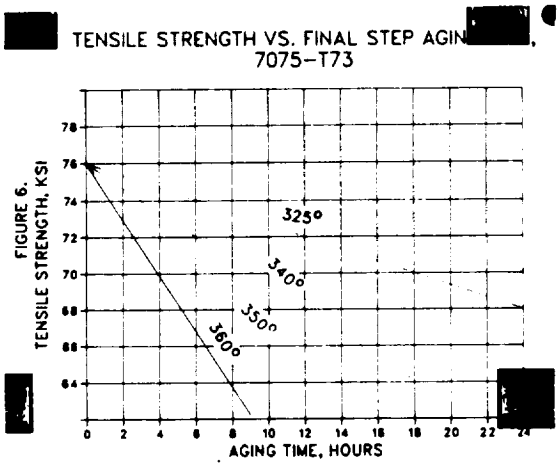
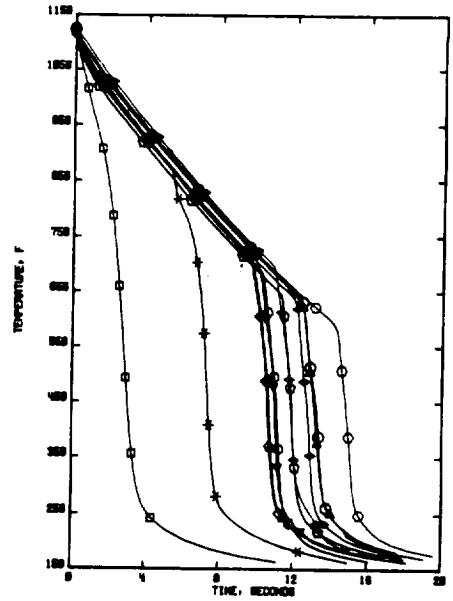
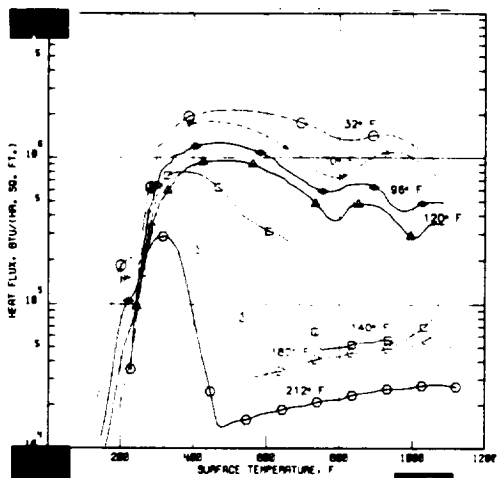


FIGURE 6.



DI WATER. IMMERSION TESTS SHOWING EXTREME SENSITIVITY OF TEST VARIABLES SUCH AS SPECIMEN SURFACE FINISH AND ABSORBED GASES, ETC. IN WATER AT 140 DEGREES F (NOMINAL)



I WATER. IMMERSION TESTS SHOWING INFLUENCE OF WATER E
FIGURE 8.

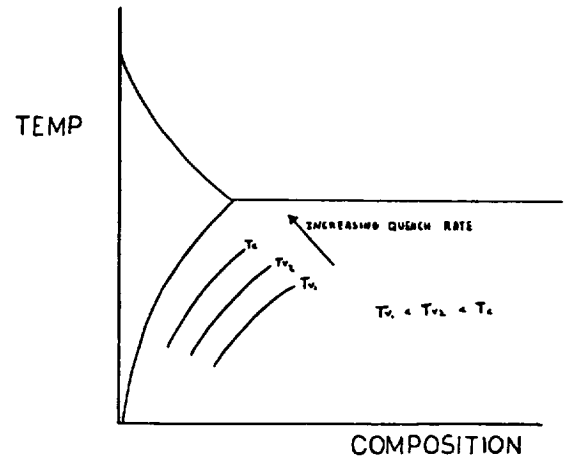


Fig. 4 Schematic Representation of G.P. Zone Solvus Temperature T_v as a Function of Composition and Quench Rate.

TABLE 3

7XXX-T7XXX Artificial Age Practice Revisions - Cleveland Works

DATE	ALLOY	PROCESS	PRACTICE NUMBER
1977 - 1980	7X75, 7049	4 hrs. @ 250°F + 4-8 hrs. @ 350°F	1
	7050	4 hrs. @ 250°F + 4 hrs. @ 315°F + 4-8 hrs. @ 350°F	1
1980-12-18	7X75, 7049	12 hrs. @ 250°F + 6-12 hrs @ 340°F	3
	7050	12 hrs. @ 250°F + 4 hrs. @ 315°F + 4-8 hrs. @ 350°F	3
1981-01-23	7X75, 7049	8 hrs. @ 225°F + 8 hrs. @ 250°F + 6-12 hrs @ 340°F	5
1981-04-15	7050	8 hrs. @ 225°F + 8 hrs. @ 250°F + 4 hrs. @ 315°F + 4-8 hrs. @ 350°F	5

TABLE 12

2" Grid Inspection Results, 7XXX-T7XXX Alloys,
Age Practice 1 (Table 3) Through 81-08-15

Alloy/Temper	Pcs. Inspected	Pcs. For * Reheat Treat	Pcs. For ** Add Age
7049-T73	180	73 (40.5%)	0
7075-T73	6,532	436 (6.7%)	55 (0.8%)
7075-T7352	1,167	348 (29.8%)	15 (1.3%)
7075-T7354	161	0	0
7175-T736	1,017	32 (3.1%)	58 (5.7%)
7175-T73652	187	0	25 (13.4%)
7050-T736	1,801	219 (16.8%)	85 (6.5%)
7050-T73652	3,965	572 (14.4%)	566 (14.3%)
	14,530	1,680 (11.6%)	804 (5.5%)

* Conductivity over 41.5%, BHN below specification minimum, thus reheat treat.

** Conductivity too low, below specification minimum.

TABLE 13

2" Grid Inspection Results, 7XXX-T7XXX Alloys,
Age Practice 3 (Table 3) Through 81-08-15

Alloy/Temper	Pcs. Inspected	Pcs. For * Reheat Treat	Pcs. For ** Add Age
7049-T73	21	10 (47.6%)	0
7075-T73	3,283	274 (8.3%)	325 (9.9%)
7075-T7352	514	274 (8.3%)	325 (19.6%)
7075-T7354	34	0	0
7175-T736	167	0	139 (83.2%)
7175-T73652	30	0	24 (80.0%)
7050-T736	0	0	0
7050-T73652	1,450	48 (3.3%)	675 (46.5%)
	5,499	332 (6.0%)	1,264 (23.0%)

* Conductivity over 41.5%, BHN below specification minimum, thus reheat treat.

** Conductivity too low, below specification minimum.

TABLE 14

2" Grid Inspection Results, 7XXX-T7XXX Alloys,
Age Practice 5 (Table 3) Through 81-08-15

Alloy/Temper	Pcs. Inspected	Pcs. For * Reheat Treat	Pcs. For ** Add Age
7049-T73	422	41 (9.7%)	34 (8.1%)
7075-T73	10,388	86 (0.8%)	166 (1.6%)
7075-T7352	2,204	8 (0.4%)	257 (11.7%)
7075-T7354	376	0	0
7175-T736	3,392	6 (0.2%)	451 (11.5%)
7175-T73652	794	32 (2.8%)	221 (27.8%)
7050-T736	202	5 (2.5%)	19 (9.4%)
7050-T73652	2,804	106 (3.8%)	315 (11.2%)
	21,122	274 (1.3%)	1,463 (6.9%)

* Conductivity over 41.5%, BHN below specification minimum, thus reheat treat.

** Conductivity too low, below specification minimum.

AIA-AA TASK FORCE ON HEAT TREATMENT OF ALUMINUM ALLOY EXTRUSIONS

R. V. Carter
Chief, Metals Technology
Boeing Commercial Airplane Company

In 1979 it was discovered that aluminum alloy plate being supplied by one of the major producers was inadequately heat treated at the mill. It was evident that this had been going on for quite some time and that suspect material was in warehouses, in fabrication by users, and in service. The deficiency was in the form of soft areas on one surface of the plate, occurring at random locations, but not generally at edges or ends of the plate. It was determined that the soft spots were easily detectable by eddy-current electrical conductivity inspection with test equipment commonly available in the producer and user facilities. The cause of the soft spots was quickly determined and corrective action was taken by the producer to prevent continued production of deficient plate. The problem was associated with steam pockets generated during spray quenching of solution-treated plate causing local interruption of the quench and allowing reheat of quenched material.

In late 1980, similar problems were discovered in aluminum alloy forgings heat treated by two of the largest producers of that product and it was again evident that this had been going on for quite some time. This time the soft spots were developed during immersion quenching of the solution-treated forgings. Specific causes were inadequate racking which allowed forgings to contact each other or heavy fixtures, too-close spacing both horizontally and vertically, racking with cavities down, quench overload, poor quenchant agitation, and racking which allowed steam generated by some forgings to interfere with quenching of others. Again, NDT methods that would have detected the problem were readily available and in common use at forge facilities.

The impact of the problem was severe and costly to producers and users alike, causing extensive testing, and re-inspection of more than 100,000 articles in stores, in fabrication, and in some cases in service.

All of these processing abuses were violations of well known basic rules of quench practice, so fundamental that they were not spelled out in heat treatment specifications. However, the real weakness in the process controls was not in lack of rules regarding racking methods or the prevention of steam formation, but in lack of surveillance requirements that would have made detection of soft spots inescapable at the heat treatment facility.

Two types of new requirements were added in early 1980 to MIL-H-6088, "Heat Treatment of Aluminum Alloys" to provide the needed coverage for spray quenching of plate. At that time, it was thought that the problem applied only to plate and to spray quenching, although minor indications of interrupted quench were seen on one or two extrusions.

In early 1981, recommendations were made to handle the problems of quenching of forgings, castings, and extrusions by both immersion and spray quenching. These recommendations took two approaches; first, general rules of good quench practice regarding spacing, racking, water temperature, fixturing, and quenchant agitation and second; facility qualification, record keeping, and product surveillance. In negotiation of the details of these changes, it was decided that quenching

practices and inspection methods for extrusions were not adequately understood and specification changes for extrusions were deleted with an informal agreement that producers and users would get together and study the problem. The changes for forgings and castings were released in mid-1981. It was later learned that some of the inspection methods levied were not appropriate for castings and these are now being addressed.

The Aircraft Industries Association and the Aluminum Association then formed a working group to address the needed coverage for extrusions. This task-group met in October 1981 and agreed on some general rules for quenching methods and product inspection. It was decided that plate could be used to qualify and monitor quality of the spray quench facility, and procedures were worked out. The details of specification language have been agreed upon by the two associations. Both groups believe the results provide necessary, reasonable, and economical requirements to protect producers and users from the threat of an industry-wide problem in heat treatment of extrusions.

This method of handling problems of this type is slow. Both associations had to circulate drafts and redrafts to their memberships and it is anticipated that the final specification change will also be circulated to producers, users and final-product users before release. Two years have passed since the decision to defer the changes to extrusions. However, we believe that this task-force approach is good, that the changes are both reasonable and effective, that a better understanding of producers and users concerns now exists.

TESTING ALUMINUM PLATE WITH AN AUTOMATIC ELECTRICAL CONDUCTIVITY TESTING SYSTEM

James E. Duarte
Corporate Quality Assurance
Reynolds Metals Co.

In manufacturing, the use of valid non-destructive tests is a desirable procedure. For aluminum plate, there are destructive tests performed in the physical testing laboratory that produce significant information about the quality of the product. Metallurgical samples are cut from the edge of the plate to be used for these destructive tests. To avoid taking random samples throughout the plate and thus, destroying it, an assumption is made that the entire plate is homogeneous. The edge samples are machined to a specific size called a coupon. Coupons are broken by a pulling procedure to obtain information related to the physical properties of the metal; e.g. ultimate yield strength. A non-destructive test that allows information to be obtained for an entire heat treated plate is also in use. The non-destructive test uses an eddy current meter to determine electrical conductivity (EC) in % IACS (International Annealed Copper Standard). A probe is moved over the surface of the plate for these measurements. In this way the entire plate can be tested to monitor quality with good confidence. This is essential when supplying heat treated plate to the aircraft and aerospace industries.

Although EC testing used as a supplement to physical testing of edge samples yields additional information about the true homogeniety of the metal, the standard EC testing procedure is time consuming. To perform the test manually on both sides of a plate requires considerable handling of the plate. With plate measuring from .250 inches up to 6 inches in thickness and several hundred inches in length, handling can become a major time consideration. Plates are moved via small cranes, one-at-a-time to racks that can be transported by lift trucks. When the racks reach the inspection area, the plates must again be handled one-at-a-time at an inspection station. After a plate is finally positioned, the probe is placed on the surface of the plate at intervals related to some grid pattern, for example, 4 inch centers. When the top surface is completely scanned, the plate is then moved by the same type of small crane to overhead saw horses. The overhead saw horses allow for inspectors to move under the plate and test the bottom surface using the same grid pattern as the top surface. During this operation only the maximum and minimum EC values from each surface are recorded. Trying to record all values would be an almost impossible task and would further delay future processing of the plate.

In an attempt to decrease the amount of time involved and increase the amount of information obtained from each plate tested, Reynolds Metals Co. contracted with Robertshaw Controls to build an automatic conductivity testing system (ACTS). Reynolds wanted a piece of equipment that would scan both surfaces of a plate simultaneously and keep the EC values for further analysis of plate quality. The venture of reading the EC from both sides of a plate at once and digesting all of those readings became quite a project. First, a roll bed to move the plate was needed to feed the plate to the EC measuring device. Next, the probes for reading the EC needed to be properly positioned with the ability to scan across the plate. Last, a mini- computer needed to be attached to the system for control, data massage and reporting of results.

At this point looking at the last step first and working backwards may be the best approach. A PDP/1103 MINC system was chosen to control the operation. The computer controls three major aspects of plate testing: 1) the speed at which the plate moves along the roll bed past the eddy current probes, 2) the speed that the probes scan the plate via a pneumatic drive, and 3) the collection of readings made by the probes. Operators enter commands to the computer through a VT100 CRT next to the testing equipment. There is also a low-speed impact printer (1200 baud) at this location for report generation.

The difficult part of physically moving the plate through the device while scanning it was manifest most dramatically when processing non-flat plate. The frame that houses the probes can be adjusted vertically to accommodate plate from about .313 inches up to 6 inches in gauge. The width is fixed at 72 inches. Depending upon the location of the probe on the plate, there are places where the plate is flatter than others. Non-flat surfaces caused the probes to change their reading angle with respect to the surface. This problem was called "lift off". It resulted in erroneously high readings. Another "lift off" type problem that haunted the system for a time was related to signal noise when the roll bed motor was engaged to move the plate past the probes. Bringing the roll bed motor to the desired travel speed all at once also gave an erroneous high reading to the computer. Rechecking the areas of the plate with hand-held instruments verified that the readings were, in fact, too high. Through software enhancements, redesigning the probe carriage and slowing the start-up, these problems have been all but eliminated.

When the plate is scanned, the computer processes the readings and prints a report on the local impact printer. The operators of the ACTS at this point have a general overview of high and low readings, the average, standard deviation and any points outside the specified tolerance limits for the alloy and temper of the plate just scanned. The data are held in the mini- computer and other programs can be run to print out "exeception" reports. These reports are printed after the operator enters a range for the EC values to be examined. This range may be as wide or as narrow as the operator wishes. Exception reports allow the operators to see if there are any patterns to the EC readings for the surface of a plate. Coordinates for any readings outside the range specified by the operator are printed. With this information the operator can recheck any areas of a plate in question with a hand-held eddy current meter. Line-printer plots also accompany this report so that the operator can see the intensity of readings outside the range. This is a rough plot and the printer is slow compared to a high speed line printer seen in computer rooms, but it is helpful to the operator inspecting the plate.

The hardware configuration of the mini-computer includes dual floppy diskette drives. With read/write capabilities on these drives, data collected after reading a plate can be stored on a floppy diskette. These data can then be transferred to other storage media for further analysis. It should also be noted that new equipment for quenching heat treated plate was installed at about the same time as the ACTS equipment. With data storage and transfer capabilities available, some sophisticated software could be interfaced to help evaluate the new quench equipment as well as monitor the quality of heat treated aluminum plate.

The first step in performing any data analysis depended upon rapid, accurate data transfer from the

ACTS to the Reynolds corporate AMDAHL computer system. A system of 1) copying data from the diskette of the ACTS to magnetic tape on a sister PDP/11 and 2) mounting the tape on the AMDAHL, became quite arduous. Several hand-carry operations were necessary because the three computers were in different physical locations. The diskettes were either mailed or carried from the Chicago plant location to Richmond, Virginia, then taken to the facility housing the sister PDP/11. After the magnetic tapes were created they were hand-carried to the Reynolds corporate computer room for mounting and transcription. Several editing routines were also necessary at that point to put the data in a usable format. All these growing pains have been overcome and the diskettes are now read and data transmission made directly to the AMDAHL via an IBM SYSTEM 38 located at the same Chicago plant facility as the ACTS.

Making the data available on the Reynolds corporate computers opened many doors to access additional software. The first program package used was the Statistical Analysis System (SAS®). A product called SAS/GRAPH™ had been introduced about that time with several procedures for graphics displays. The G3D procedure was chosen to display the EC values from readings taken on each surface of the plate (Figure 1). Different EC patterns are seen depending on the alloy, temper and gauge of the metal combined with the water pressure in the quench and the rate of travel of the plate through the quench (kickout speed). With these plots the engineers can see the results from different water spray nozzle positions and angles. Likewise, the metallurgists can see the effects of changes in kickout speeds and water pressures. The 3-dimensional conductivity plots have opened a whole new perspective on quality monitoring.

Studies were begun to evaluate not only recommended quench parameters for the new equipment, but also effects of extremes in settings for these parameters. Questions came in faster than plate could be made available to answer them. Many questions, for example, "If a water spray nozzle in the quench were to malfunction could it be detected quickly by looking at the result of a conductivity plot?", were answered within a few runs. The answer to this particular question is "yes" as seen in Figure 2. The ridge of high EC along the edge of the plot pinpoints for maintenance where, in the quench, to look for the malfunctioning nozzle. By knowing the location before going to the job, repairs can be made more quickly. Another situation was detected for a plate meeting all release specification (again, see Figure 1). This pattern had not been seen on earlier conductivity plots. An investigation revealed an abberation in the quench drive mechanism which maintenance was able to repair before unnacceptable plate was produced due to this condition. Answers to other questions resulted in subtle changes in EC patterns. The subtle changes were not as easily seen on the 3-dimensional conductivity plots, at first; consequently, an in-house program was written which draws a fine gradient continuous color plot.

The continuous color gradient plot divides the EC into ten groupings. The lowest EC's are shown in bright blue. As the EC values increase there are two lighter shades of blue then turquoise and bright green. After bright green, as the values increase, the colors go to olive drab, yellow, orange and red. These are the first nine colors in the gradient. The final color is white. White is reserved for special high values. This computer program finds the range from the lowest to the highest

EC value on each side of a plate. If the range is greater than 2.5% IACS, then all high values outside this range appear in white. These white points correspond to points that are outside of the EC range specifications.

The second graphics software package completed a system for the qualitative evaluation of the surface EC of heat treated plate. With both software systems in place, studies were designed to begin optimizing the quench parameters; i.e. kickout speed and water pressure, with respect to minimizing EC variation. Keeping the EC close to the minimum of the specification range amd minimizing the variation in the EC values, produces the most desirable heat treated plate. The assumption of homogenous properties throughout a plate was mentioned early on. This assumption of homogeniety becomes more reasonable when it can be shown that the EC values across the plate have minimal variation.

To accomplish the optimization process and answer such questions as: "What kind of range could be put on kickout speed?", a statistical analysis technique called response surface analysis was employed. Here the qualitative plots were coupled with quantitative analysis to complete the evaluation of the new quench and set the best ranges of the quench parameters for each alloy and gauge combination of heat treated plate. Parameters of the quench were supplied by the manufacturer. These values were used as a starting place to design statistical experiments that would show ranges of kickout speed and water pressure combinations that produce the best quality heat treated plate. Plots of response surfaces show the direction to move the quench parameters to find minima in the EC variation.

Additionally, to verify the consistency of readings made by the ACTS, a special plate was cut so that it would fit through the frame regardless of its orientation on the roll bed. The plate was run head first, right side first, tail first, and left side first. It was then inverted top for bottom and run four times again. Plots of these eight scans showed consistency qualitatively with ridges moving over the plots with respect to the particular plate orientation. Quantitative results in Table I showed some minor variation due to the fact that the exact same points are not read when a plate is rescanned.

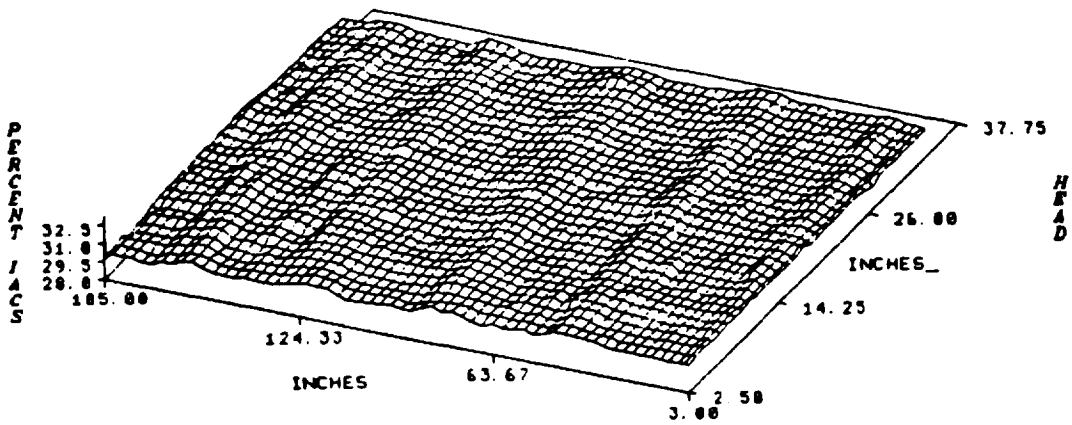
The whole process of developing computer software and using it to create a comprehensive quality monitoring system took about two years. Thoughout the project, quality inspectors have monitored production by standard means of measuring EC with spot checks as a backup to ensure customer satisfaction and compliance with all military standards. Computer graphics equipment is on line at the manufacturing site to show the quality assurance personnel the status of the quench process via regular sampling of weekly plate as specified in military standards. Combining all of the tools developed during this project brings together a comprehensive means of monitoring the quality of heat treated aluminum plate.

3-DIMENSIONAL CONDUCTIVITY PLOT

2124 PLATE 8151 WITH DIMENSIONS 185 X 38 X 6.13 INCHES

WATER PRESSURE: PRIMARY __ PSI, SECONDARY __ PSI, ANGLE: __ % KICKOUT: __ IN/MIN

HI=31.4 LO=29.7 SPREAD=1.7 STD DEV=0.26



BOTTOM SURFACE

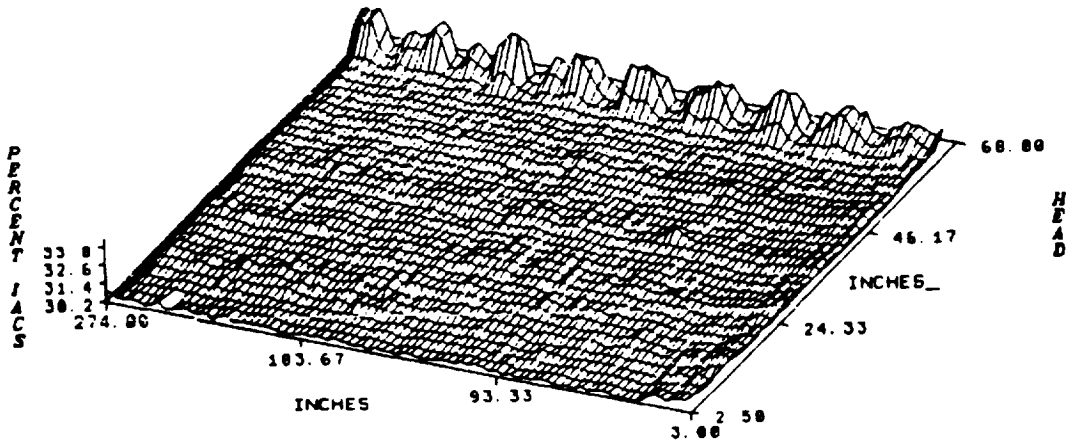
FIGURE 1

3-DIMENSIONAL CONDUCTIVITY PLOT

2124 PLATE 8151 WITH DIMENSIONS 274 X 68 X 2.50 INCHES

WATER PRESSURE: PRIMARY __ PSI, SECONDARY __ PSI, ANGLE: __ % KICKOUT: __ IN/MIN

HI=33.9 LO=30.2 SPREAD=3.7 STD DEV=0.38



BOTTOM SURFACE

FIGURE 2

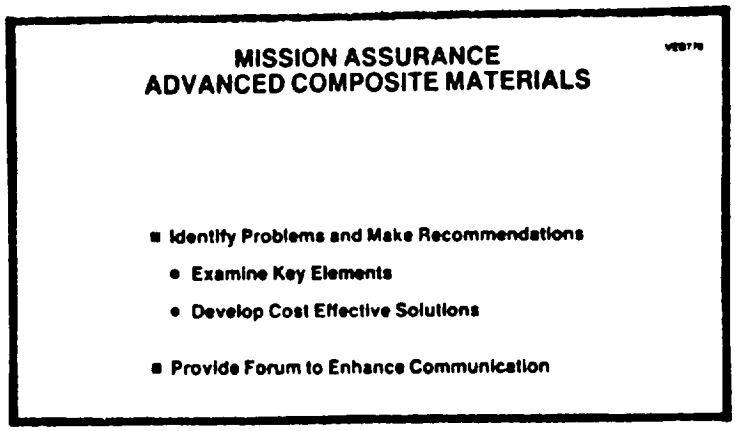
TABLE I

REPRODUCIBILITY STUDY FOR
ELECTRICAL CONDUCTIVITY READING
(Percent IACS)

	PLATE SURFACE #1				PLATE SURFACE #2			
	MAXIMUM		MINIMUM		MAXIMUM		MINIMUM	
ROTATION	UP	DOWN	UP	DOWN	UP	DOWN	UP	DOWN
0°	40.7	40.5	39.9	39.8	40.3	40.1	39.6	39.5
90°	40.6	40.7	40.0	39.8	40.3	40.2	39.5	39.6
180°	40.7	40.4	39.8	39.7	40.5	40.3	39.8	39.4
270°	40.7	40.3	39.9	39.7	40.2	40.3	39.7	39.6

ADVANCED COMPOSITE WORKSHOP SESSION
3RD MISSION ASSURANCE CONFERENCE, 1983
MODERATOR'S OPENING REMARKS

BY HANK LANGE
QA DIRECTOR, MDAC-HB

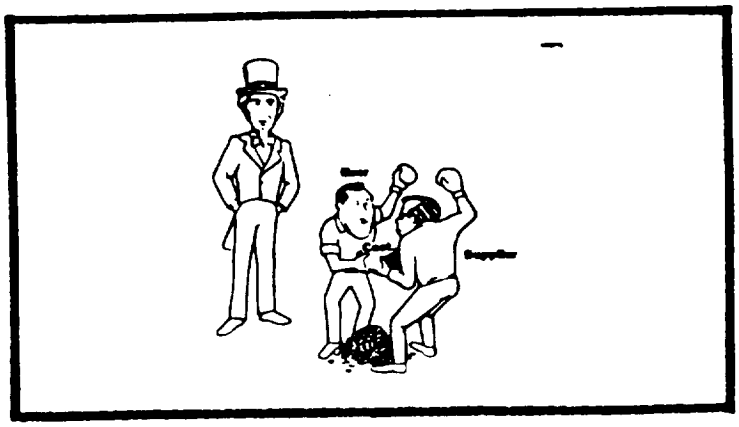


Viewgraph 1

Mission Assurance was organized with the goal of bringing together experts who can identify and discuss aerospace program related problems and make recommendations for future action. We have here today experts in the field of Advanced Composite Materials from several disciplines, both users and suppliers. In general, our task is the examination of key elements, in the procurement and use of these materials, that are essential to aerospace mission success; and, also, the development of cost effective approaches for the better utilization of this important resource. In more specific terms, we will identify those elements and factors detrimental to the efficient use of Advanced Composite Materials; and recommend solutions which will reduce cost and improve material acquisition for aerospace applications.

Another goal of Mission Assurance, and the purpose of this session, is to improve communication and dialogue by providing a forum where users and suppliers can address mutual problems.

We all well understand the mutual problems that exist between users and suppliers of Advanced Composite Materials.



Viewgraph 2

We have seen in the past ten years the emergence of Advanced Composite Materials as major structural components. However, the economics of doing business with these materials has not advanced to the same degree as with their structural metallic counterparts. Let's take a look at some of the problems that currently exist in the industry.

PROBLEMS IN THE INDUSTRY

- No Standard Specifications
- Duplication of Test Data
- Qualification Costs Vs. New Resin Systems
- Proprietary Nature of Resin Systems

Viewgraph 3

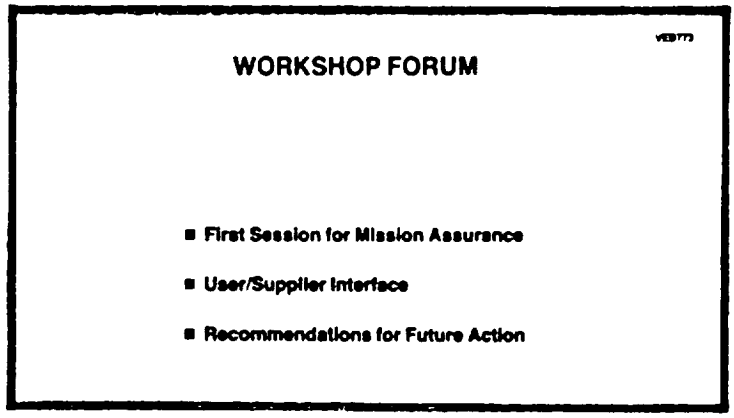
The problems existing in industry are well understood at this time. The suppliers are confronted by a variety of different user specifications because of the lack of an industry standard, e.g., an AMS or a military specification. We see suppliers conducting tests to qualify materials, and users duplicating these tests. These are very expensive tests to be repeated. For materials used in large quantities these costs may be justifiable; but what of the many materials used in small quantities?

We are also aware that we frequently continue to use the old proven resin system, even though a new, better system has been developed. This is due to the high cost required to generate a new data base to qualify the new system. Certainly, we all would like to see a way which would permit us to reduce this cost and allow us to utilize these newer, better materials for our applications.

Furthermore, inherent in our way of doing business is the proprietary nature of the resin chemical formulation. Only because of the requirements for multiple sources for military applications, have we seen selected resin systems used by two or more suppliers.

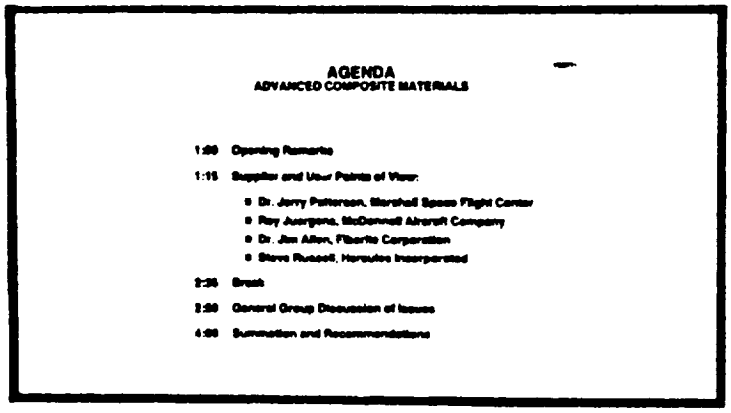
These are the problems we are here to address this afternoon. These problems influence the wider and more efficient use of Advanced Composite Materials in our aerospace programs, and will not only continue to affect the generic group of resins in use today for moderate temperature applications, but will also affect the future higher temperature resin systems. As you all know, there is considerable

effort in industry to develop these materials for use in our military and civilian aerospace programs. Such work is already in progress. Therefore, the thoughts discussed here today will not only affect current systems, but will also have an impact on the future utilization of the new Advanced Composite Materials.



Viewgraph 4

A little about the workshop forum--this is the first workshop session in Advanced Composites for Mission Assurance. It is the first forum where Advanced Composite users from industry, the military, and civilian agencies can come together with material suppliers to address common problems, and to propose recommendations for publication and for future action by Government / Industry Committee. It is our hope that from this forum a greater understanding and appreciation for the problems of users and suppliers will be established.



Viewgraph 5

We will begin by hearing presentations from four well-qualified speakers, representing user and supplier points of view. We will hear two presentations from the user's point of view. Mr. Jim Ehl of the Marshall Space Flight Center will be presenting a paper by Dr. Jerry Patterson on "Resin Systems." He will give an overview of the different problems involved with epoxy resin matrix materials and will describe how these problems relate to the effective utilization of these systems.

Mr. Ray Juergens from the McDonnell Aircraft Company will continue with "Standardized Composite Materials," and will discuss the basis for standardization, what benefits will be gained, and some of the steps necessary to establish standard material specifications.

They will be followed by two presentations representing the supplier's point of view. Dr. Jim Allen from Fiberite Corporation will speak on, "Differing User Specifications Result in Custom Formulations." He will discuss some of the unique material properties which lead to specialized formulations and will give us a little background on some of the ways differences in formulation occur as formulators strive to meet user specifications.

Our last presentation will be from Mr. Steve Russell of Hercules Incorporated, who will speak on "Variations Among User Test Specifications Require Additional Supplier Testing." Steve will show how problems arise for the supplier by the variety of user acceptance requirements, and the economic impact of these requirements on material costs.

After the presentations, at 2:35 p.m. we will take a short 15-minute break, then reconvene at 2:50 for a general group discussion, at which time you will be able to ask questions of our panel members. During this discussion, which will last until 4:00, we will list and address the key issues.

From 4:00 - 4:30 we will recap the issues, propose and discuss solutions and make recommendations.

Let us begin by introducing our first speaker, Mr. Jim Ehl. Mr. Ehl is Chief of the Tooling Branch of the Process Engineering Division of NASA's Marshall Space Flight Center, Huntsville, Alabama. He is presenting a paper by Dr. Jerry Patterson, Chief of the Polymers and Composites Branch at Marshall. Dr. Patterson is involved in the development and evaluation of cryogenic foams, ablatives, adhesives, dielectric materials and composites. Dr. Patterson's background is in organic/polymer chemistry with research interests in the area of polymer structure and property relationships. He received his PhD in Organic Chemistry from the University of Alabama and has been with NASA's Marshall Space Flight Center since 1963.

Our next speaker is Mr. Ray Juergens, Branch Chief of Technology - Material and Process Development for McDonnell Aircraft Company, St. Louis, Missouri. He is responsible for the development of nonmetallic materials and processes. His 35 years of experience in the aircraft industry includes composite technology, metallic materials, design consultation, and material research and development. Mr. Juergens has managed a variety of research and development programs including resin and metal matrix composites, nonmetallics and nondestructive testing. He received a Bachelor of Science degree in Metallurgical Engineering from the University of Missouri in 1948 and a Master of Science degree in Metallurgical Engineering from Washington University in 1960. Mr. Juergens is currently a member of the American Society of Metals (ASM) and the American Institute of Mechanical Engineers (AIME). He is listed in American Men of Science and has served as a member of the American Institute of Aeronautics and Astronautics (AIAA) Technical Committee on Materials.

Our first speaker representing the supplier's point of view is Dr. Jim Allen, Technical Director of the Advanced Composite Material Business Center for Fiberite Corporation, Winona, Minnesota. He is currently involved in research and development, and quality control of graphite reinforced organic matrix prepreg materials. He was formerly a member of the technical staff at the Tulsa Division of Rockwell International Corporation. While at Rockwell, Dr. Allen was a member of the Material and Processes team that developed the prototype for the graphite epoxy Space Shuttle Payload Bay Doors, which led to the material system used in production. He also worked as a research chemist for Phillips Petroleum Company, where he was involved in the development of specialty polymers, such as polyphenylene sulfide (PPS). Dr. Allen received a Bachelor of Arts degree in Chemistry from East Texas State University in 1958 and his PhD in Physical Organic Chemistry from Rice University in 1962. He is currently a member of Sigma Xi, the American Chemical Society and the Society for the Advancement of Material and Process Engineering (SAMPE).

Our final speaker today is Mr.Steve Russell, Manager of Carbon Fiber Development at Hercules Incorporated, Magna, Utah. Mr. Russell joined Hercules in 1969 to conduct research and development on carbon fibers, and was instrumental in the design and start-up of the first carbon fiber-line for Hercules. In his 15 years at Hercules, Mr. Russell has been involved in all aspects of carbon fiber manufacture, ranging from Production Superintendent to Manager of all Hercules carbon fiber research and development. Mr. Russell holds a Bachelor of Science and a Master of Science degree in Chemical Engineering from Ohio State University.

MATERIALS AND PROCESSES INTEGRITY WORKSHOP

ADVANCED COMPOSITE MATERIALS
RESIN SYSTEMS - - PROBLEMS/SOLUTIONS

- NASA/MSFC COMPOSITE MATERIAL APPLICATIONS -
  - SPACE TELESCOPE - - GR/E STRUCTURAL ELEMENTS
  - SOLID ROCKET MOTOR (SRM) FILAMENT WOUND CASE - - GR/E CASE STRUCTURE
  - SRM NOZZLE - - CARBON/PHENOLIC
  - SRB - - GLASS/PHENOLIC THERMAL PROTECTION
  - SOLAR ARRAY - - GLASS/EPOXY STRUCTURAL MEMBERS
  - ADVANCED X-RAY FACILITY - - GR/E STRUCTURAL ELEMENTS

MATERIALS AND PROCESSES INTEGRITY WORKSHOP

ADVANCED COMPOSITE MATERIALS
RESIN SYSTEMS - - PROBLEMS/SOLUTIONS

RESIN/COMPOSITE EVALUATION METHODOLOGY -

- DATA BASE VARIABILITY/ERROR SOURCES:
  - TEST SPECIMEN FABRICATION/REPRODUCIBILITY
  - COMPATIBLE TEST METHODOLOGY
  - VALIDITY OF MEASUREMENT TECHNIQUES
  - STATISTICAL DATA TREATMENT
- STRESS NEED FOR STANDARDIZED TEST METHODOLOGY

MATERIALS AND PROCESSES INTEGRITY WORKSHOP

ADVANCED COMPOSITE MATERIALS
RESIN SYSTEMS - - PROBLEMS/SOLUTIONS

RESIN CHEMISTRY AND CHARACTERIZATION -

- UNDERSTANDING RESIN/CATALYST REACTIVITY
  - DEFINITION OF OPTIMUM CURE RATE
  - CURE EXOTHERMS
  - B-STAGE RESIN (PREPREG) REACTIVITY VS. TIME
- CHARACTERIZATION DATA FOR MATRIX RESIN
  - UNCURED RESIN
  - CURED RESIN/CATALYST SYSTEM
- STANDARDIZATION OF TEST METHODOLOGY

MATERIALS AND PROCESSES INTEGRITY WORKSHOP

ADVANCED COMPOSITE MATERIALS
RESIN SYSTEMS - - PROBLEMS/SOLUTIONS

RESIN STANDARDIZATION/SPECIFICATION -

- GROUPING SIMILAR RESIN SYSTEMS INTO CATEGORIES
  - 250°F CURED RESINS
  - 350°F CURED RESINS
- GENERATE SPECIFICATIONS DEFINING:
  - REQUIREMENTS
  - DETAILED TEST METHODOLOGY
- PROVIDES USER WITH INSIGHT INTO RESIN PERFORMANCE AND COMPARABILITY

Materials and Processes Integrity Workshop
Advanced Composite Materials
Resin Systems -- Problems/Solutions

W. J. Patterson
Chief, Polymers and Composites Branch
Materials and Processes Laboratory
NASA-Marshall Space Flight Center

I. Introduction -

The Marshall Space Flight Center (MSFC) over the past 5 years has become increasingly involved in composite technology, primarily for utilization on the Shuttle launch vehicle and payloads/experiments. Features such as high strength-to weight ratio and dimensional stability continue to support composite materials as prime candidates for many flight hardware applications. Several examples (both implemented on flight hardware and in development) are shown below to highlight this Center's commitment to composite technology:

- Space Telescope-GR/E structural elements (CTE, stiffness, weight)
- Solid rocket motor filament wound case-GR/E filament wound structure (weight)
- Solid rocket motor nozzle-Carbon/phenolic (weight, ablative performance)
- Solid rocket booster-glass/phenolic (weight, ablative performance)
- Solar Array structural member-glass/epoxy (weight strength)
- Advanced X-ray Astronomical facility-GR/E structural elements

II. Problems/Concerns In Matrix Resin Utilization -

A number of issues bearing on the integrity and quality of composite resin systems should be addressed. The major areas of concern include resin chemistry/characterization, industry standardization and comparability of resin systems. More subtle problems relate to lack of definition of resin structure/property relationships, which may preclude proper resin selection by designers.

A. Resin Chemistry and Characterization -

One concern in this area is lack of understanding of the chemistry and definitive characterization data for the matrix resins, in both uncured and cure states. Considerable emphasis has been placed recently on analytical techniques for uncured resin characterization (HPLC, IR, GPC etc.), and a certain perferred methodology is beginning to emerge. However, this characterization is still far from definitive in terms of optimum test methods or specific procedural details that will be uniformly applicable to broad classes of matrix resins. Since several widely used matrix resins are based on variation of common chemical systems (such as TGMDA/DSS), this type of standardization should be a viable option. The resin/catalyst reactivity for commercial matrix resins is not well defined, particularly with respect to optimum cure rate and extent of cure. Little information is available on exotherms from large applications, or precise effects of B-stage resin aging on cured properties. Also, relatively little characterization data exist for the cured resins where the reinforcement is absent. These data would be valuable to assess reproducibility of resin/catalyst properties and lot-to-lot variability without the obscuring effects of the reinforcement. Dynamic mechanical analysis (DMA) is gaining recognition for cured resin characterization, and it should be exploited to a greater extent, particularly in the areas of cure advancement/optimization and assessment of environmental effects.

The progress in these areas will be paced, to a large extent, by the supplier's position on proprietary interests, and an improved supplier/user dialog is needed to establish baseline characterization data. An additional area of substantial interest to the Marshall Center is fracture toughness (or damage tolerance) of the matrix resin. A more quantitative test data base and standardized methodology are needed to assess tendency for crack formation and propagation as well as microcracking as a function of cyclic loading or environmental stress. It is realized that these properties don't transfer on a one-for-one basis to the composite system, but certainly the designer should have enough resin chracterization data to select the matrix material based on optimum performance in the anticipated use environment.

B. Resin Standardization/Specification Needs -

The question of comparability of resin systems is one that should be addressed from the user standpoint. For example, the commercial matrix resins falling within the general 350°F cure category cannot be interchanged with confidence based only on general composite properties. Some minimum level of standardization or specification should be defined which provides for adequate cure resin characterization. This would allow users to group certain matrix resins and have more confidence that the derived composite performance from one system would be comparable to another, at least for those parameters which are resin dominated. One could visualize as a minimum the generation of a military standard, or its equivalent in the aerospace community, defining material requirements and detailed test methodology for general classes of 250 and 350°F cured matrix resins. The current vendor-supplied data base of cured matrix resin properties is very limited, even for the more widely used 350°F systems. Vendors could contribute significantly to this area by generating baseline properties and defining test methodology.

C. Resin/Composite Evaluation Methodology -

One of the primary concerns in this area is questionable data bases generated from unknown or varying specimen configurations and test procedures, and lack of statistical treatment of data. The magnitude of this problem comes sharply into focus when considering the potential variability and error souces in mechanical testing of matrix resins:

- Machining and/or molding specimens reproducibly to precise dimensions (and stress relief of machined specimens)
- Producing void-free specimens with no high-energy surface flaws.
- Using appropriate test geometries, fixtures, strain rates to insure comparability with existing data base
- Verifying validity of strain-monitoring techniques (extensometer slipping, strain gage debond, etc.)
- Establish minimum number of sample for a given test condition to obtain statistically valid data.

This area of concern extends to composite test methodology as well, and more attention at this level of detail is necessary both from supplier and user viewpoints to improve the quality of the derived data base. This is a particular concern in view of the bewildering number of sample geometries, test fixture configurations, and test methods pervading this area of technology. A concerted effort is needed to standardize test methodology to improve data comparability and reduce error.

D. Long Term Resin Properties -

The evolution of long life hardware such as Space Telescope and Spacelab, and the recoverable/reusable Solid rocket booster hardware has focused attention on a number of resin-dominated properties or phenomena that appear to be subtle, second order effects during baseline testing, but which may significantly influence composite performance for long life hardware in the space environment and multi-use hardware whose cumulative service life is expected to be several years. This extended life requirement raises concerns over the long-term effects of space radiation as well as vacuum environments on the resin physical and mechanical properties. The extent to which long term radiation-induced crosslinking, moisture desorption, vacuum outgassing and simple resin aging will affect critical resin properties such as coefficient of expansion, glass transition, strain capability and microcracking is largely undefined. Similarly, the effects of cyclic or cumulative stresses on matrix resins (in the case of multiple use hardware) is an area requiring better definition. Again, these concerns point to the need for more definitive and basic resin characterization and an expanded cured material test data base.

III. Summary Of Key Issues/Concerns -

The primary concern from our perspective is definitive and reproducible characterization methodology and cured resin evaluation protocol. Additional concerns focus on the need for documentation that specifies requirements for general classes of matrix resins.

MATERIALS AND PROCESS INTEGRITY

FOR

MISSION ASSURANCE

ADVANCED COMPOSITE MATERIALS WORKSHOP

"STANDARDIZED COMPOSITE MATERIALS"

Raymond J. Juergens

Branch Chief Technology

McDonnell Aircraft

Composite materials are becoming a greater factor in the structures of aerospace systems. The structural efficiency of fiber reinforced composites, coupled with fatigue life improvement, corrosion resistance and cost saving potential, has made these modern materials prime candidates for use in new systems where performance improvement is avidly sought. In our product line, we have seen the structural fraction of composites on each succeeding aircraft rise from approximately 1% on the F-15 Eagle to 10% on the F/A-18 Hornet and 27% on the AV-8B Harrier II. Our experience has been shared by others, making high-quality carbon, organic and other fibers with appropriate resin systems a growing market. (Fig. 1)

Because of the increased production of systems using composites through the 80's, these materials will receive more attention from a supply standpoint. Thirty percent of the structure of an aircraft will provide attention equivalent to that which has been directed to materials such as titanium. Thirty percent is approximately the titanium fraction for an F-15. The titanium that we buy for the F-15 is procured on a competitive basis to a material specification that defines the material. The product is uniform, consistent, interchangeable and essentially indistinguishable from supplier to supplier. It is a standard material; composite materials will mature to a similar position as their use on production programs grows.

The existing metal specifications have all of the elements necessary and relevant to composite materials. They list a description of the material, the form, applicability, composition, processability, mechanical property capability, sampling, testing, handling, marking, safety provisions, etc. They describe well characterized standard production materials that can be reliably procured. As a result, a standard titanium alloy can be specified and used in an aircraft today much easier and with less expense, compared to composite materials.

The present industry practice for composite materials is to require qualification testing of each composite material for each new system application. We require fiber and matrix dominant tests, environmental exposure tests, perhaps some structural element or subcomponent tests,

PAGE 70 INTENTIONALLY BLANK

processability tests and tests associated with the uniformity of the product. We require tests on at least five lots of material produced by a supplier so that we can treat the data in a statistical manner. The cost of this qualification effort is repeated by other fabricators as each pursues a composite application with his selected material suppliers. The number and cost of these qualification tests can be significant, considering the numbers of fibers and resins being marketed today. The qualification data base is established and reestablished again and again.

All parties involved, the fabricator, supplier and the eventual customer could benefit from the cost avoidance offered by a standard material.

The technology necessary for a user and a supplier to define a material for multi-source procurement has been gradually evolving. The necessary information appears to be on hand and we are presently incorporating these elements into our production material specifications. There are four major technical areas that need to be defined for composite materials. These are the reinforcing fiber, the matrix resin, the processing characteristics and the cured mechanical property capabilities. While the state of technology varies for these four areas, there is the basis (as shown in Figure 2) for a standard material specification.

If standard composite materials are desirable, why don't we have them? Why have standard materials not evolved? One of the reasons is the low production material requirements of previous composite applications.

Standard materials arise from substantial production usage. Production usage creates the standard. If, by using what everyone else uses, one can avoid costs, and get better delivery and price, then the use of a standard product becomes desirable. Figure 3 shows the systems currently using or qualifying the 3501-6 and the 5208/3502 resins. These two resin types appear to be establishing themselves as standard materials through their application on several production systems. They could become standard materials solely because of their use on production programs.

Another factor inhibiting the establishment of a standard material is the proprietary aspects of resin formulation. Almost all resin formulations are considered to be proprietary because of the lack of patent protection for resin formulations. As a result, the development of multiple sources of supply for the same material (i.e. a standard product) usually follows the course shown in Figure 4.

Progress or lack of progress toward standardization results from supplier market strategies and not necessarily from a user's need. Trading, royalties, front end fees, etc. are the business prerogatives of the suppliers. However, the user's investment and risk in establishing the material thru a production commitment can be as great as a supplier's proprietary claim. The supplier's advantage in keeping a material proprietary is counterproductive to the multisource desires of the user.

The time has come for standard composite materials. Production usage will create standards eventually. One action that could accelerate standardization is mutual production agreements between suppliers. To accommodate the business aspects of such agreements it is recommended that suppliers take action as an industry group to hasten standardization so that all parties can benefit.

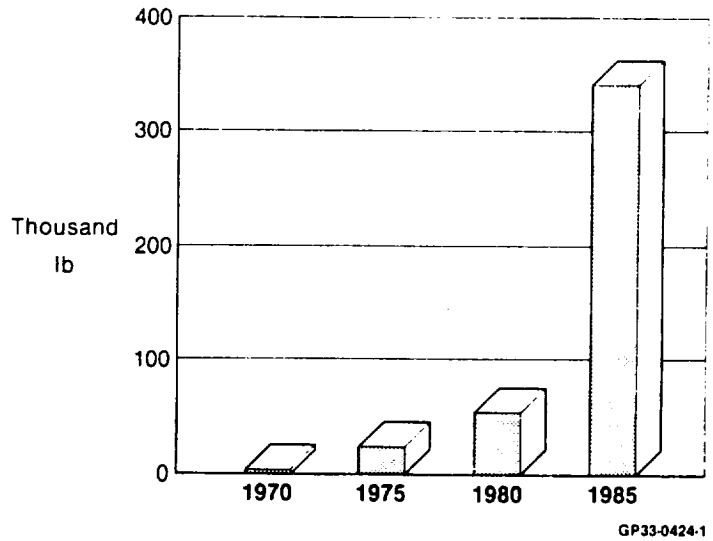


Figure 1. Composite Material Usage

Fiber		
Fibers/tow,		12,000
F_{t_u}, min.	(ksi)	500
E, Tensile,	(Msi)	32 - 35
Strain, Tensile, min.	(percent)	1.4
Diameter,	(micron)	7.5 - 8.5
Density	(lb/in^3)	0.062 - 0.066
Sizing		None
Resin	(Weight percent)	
Tetraglycidal Methylenedianaline		54 - 58.5
4, 4' Diamino Diphenyl Sulfone		23 - 29
Alicyclic Diepoxy Carboxylate		8 - 10
Epoxy Cresol Novalac		7.5 - 9.5
Boron Trifluoride Amine		0.8 - 1.4
Flow		
Viscosity; Integrated min/poise		4.5 - 8
Mechanical Property Capability	(Minimum)	
0° Tension	(ksi)	250
0° Compression	(ksi)	230
90° Tensile Strain	(μin.)	4,000
ILS	(ksi)	15

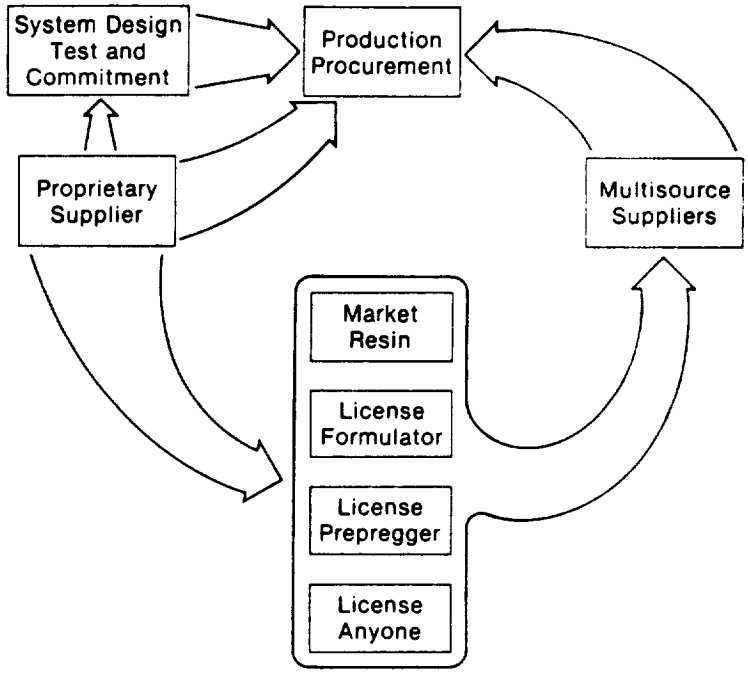
GP33-0424-2

Figure 2. Specification Requirements

	3501-6	5208/3502
Production Qualified	F-18 MCAIR/Northrop AV-8B MCAIR F-20 Northrop C-141 Lockheed ACAP Bell ES^3 (UH-60A) Sikorsky G-3 Gulf Stream MMS MDAC-W	F-15 MCAIR F-16 GD FW L-1011 Lockheed DC-10 DAC Fokker IAI Sikorsky 757, 767 Boeing, Air Italia Mirage 2000 Aerospatiale B-1B Rockwell
In Qualification	DAC Vought Boeing Bell	LAVI Vought

GP33-0424-3

Figure 3. Resin System Qualifications



GP33-0424-4

Figure 4. Material Sources

DIFFERING USER SPECIFICATIONS RESULT IN CUSTOM FORMULATIONS

ADVANCED COMPOSITE MATERIALS WORKSHOP

DR. JAMES D. ALLEN
TECHNICAL DIRECTOR - COMPOSITE MATERIALS
FIBERITE CORPORATION/A UNIT OF BEATRICE FOODS

INTRODUCTION

The subject today perhaps should be how differing user "needs" result in custom formulations in advanced composites and how these needs led to custom formulations and subsequent specifications written around these materials.

What will be discussed is how we suppliers have come to create the current proliferation of formulations, what the factors were that led to proprietary formulas and what the factors are leading to further proliferation of proprietary formulations.

MATERIAL REQUIREMENTS - HISTORICAL DEMANDS ON THE RESIN MATRIX

- ATTAINMENT OF REASONABLE TRANSLATION OF REINFORCEMENT PROPERTIES
- UPPER USE TEMPERATURE
- CURE SCHEDULE
- OUT LIFE, WORKING LIFE AND HANDLING CHARACTERISTICS

HISTORICAL MATRIX REQUIREMENTS

One of the most basic and necessary requirements placed on the resin matrix portion of an advanced composite composition was, and is, that the matrix be capable of translating and retaining a large fraction of the reinforcement strength and stiffness. Epoxy materials excel in this attribute and thus were the subject of early evaluation and selection as the preferred organic matrix material for advanced composites.

Early on, the goals for usage of advanced composite materials were at an elevated temperature, 350° F., which was compatible with the selection of epoxy matrices and dictated a particular type of epoxy base resin to attain this performance.

Cure conditions and schedules acceptable to early users of advanced composite materials were also those which epoxy materials would readily accommodate; namely, pressures of 50-100 psi and temperatures no greater than about 350° F.

The storage life, out life and handling characteristics desired in advanced composite materials to permit the facile fabrication of parts also was consistent with and directed suppliers towards a relatively unreactive latent hardener to be the material of choice to cure the epoxy.

Catalysts for the epoxy cure were also an option to the supplier/formulator and could be viewed as advantageous for some fabrication schemes. Of course, the catalysts were necessarily the latent type also.

SUPPLIER RESPONSE TO HISTORICAL MATRIX REQUIREMENTS

- SELECTION OF EPOXY BASE MATERIALS
  - SELECTION OF TGMDA EPOXY
- SELECTION OF LATENT HARDENERS AND CATALYSTS
  - SELECTION OF DADS HARDENER, $BF_3 \cdot$ MEA CATALYST
- UTILIZATION OF ADJUNCT EPOXY MATERIALS
  - CARBONYL EPOXIES
  - EPOXY PHENOLIC NOVALACS
  - EPOXY CRESOL NOVALACS

RESULT - CUSTOM FORMULATIONS

SUPPLIER RESPONSE TO HISTORICAL MATRIX REQUIREMENTS

Thus the framework of requirements for matrix materials lead to the selection by several suppliers of the epoxy base material, tetraglycidylether of methylenedianiline, and to the selection of diamino diphenyl sulfone hardener.

To this point, supplier/formulators are then moving down a rather common, narrow path. However, in response to different user preferences and also to accommodate different manufacturing preferences or capabilities of the particular supplier the compositions began to vary and become unique and proprietary; that is, some suppliers chose to catalyze their formulation and some did not; some chose one route, some chose another to adjust the formulation viscosity, tack, drape, etc.

The result then was and is custom proprietary formulation around which specifications have been written and materials qualified.

EVOLVING DEMANDS ON RESIN MATRIX

- ENHANCED ENVIRONMENTAL RESISTANCE
- TOUGHNESS
- CONTROLLED FLOW OR NO FLOW
- SELF ADHESIVE
- FLAME RETARDANT

EVOLVING DEMANDS ON RESIN MATRIX

As these materials came into usage some of their shortcomings came to be known. One of the first deficiences to surface was that in hot, wet environments the load carrying ability of these epoxy formulations decreased tremendously and some of them didn't perform as well as others. This lead then to changes in formulas and thus to additional proprietary compositions.

As different areas of application of advanced composites were considered and tried, additional performance characteristics were asked of the resin matrix; that is, the matrix formulations were asked to be of the controlled, low or no flow type, self adhesive compositions were desired to eliminate if possible, the use of adhesive in the construction of parts. Flame retardancy while always desired, became required as aircraft interior applications were considered and the resin matrix and the resulting cured composite needed and needs to be tougher.

SUPPLIER RESPONSE TO EVOLVING DEMANDS

CHANGE RESIN FORMULATIONS

- DELETE OR DECREASE ADJUNCT COMPONENTS ADVERSELY AFFECTING ENVIRONMENTAL RESISTANCE
  - DELETE CARBONYL EPOXY
- INCORPORATE COMPONENTS TO REDUCE FLOW AND OBTAIN SELF ADHESIVE CAPABILITY
  - ADD ELASTOMERS
- INCORPORATE COMPONENTS AND/OR CHANGE FORMULATIONS TO INCREASE TOUGHNESS
- MODIFY MATRIX FORMULATIONS FOR FLAME RETARDANCY
  - HALOGENATED EPOXIES
  - INORGANIC SALTS

RESULT - MORE CUSTOM FORMULATIONS

SUPPLIER RESPONSE TO EVOLVING DEMANDS

Thus the supplier response was to change the resin formulations. In the case of the state-of-the-art epoxy formulations, one way to improve the environmental resistance is to delete or reduce the adjunct components and this has been done leading to more custom proprietary formulations.

In order to reduce flow and obtain self adhesive capability in their resin formulations, suppliers have added elastomers. Again, there are a number of ways to accomplish this and thus more custom formulations arise. Also, unfortunately, these additives decrease environmental resistance and so a family of custom proprietary formulations are created for controlled flow, self adhesive usage but they are not effective in hot/wet environments.

A recent area of intense activity in advanced composites, particularly graphite composites, has been to try to toughen the matrix and thereby toughen the resulting composite. Again, there are a number of possible means of accomplishing this chemically and more proprietary formulations are resulting.

In the area of obtaining improved flame retardancy, there are also various ways to accomplish this and these options lead to more proprietary formulations.

FURTHER FRAGMENTATION

- REQUIREMENTS FOR YET HIGHER USE TEMPERATURE MATERIALS
  - 350° F. + USE MATERIALS
    BMI'S
- REQUIREMENTS FOR MORE MODERATE USE TEMPERATURE TOUGHENED MATERIALS
  - 200° F. USE MATERIALS
- ADDITIONAL EMPHASIS ON FLAME RETARDANT, LOW SMOKE COMPOSITIONS

FURTHER FRAGMENTATION

Requirements for yet higher use temperature materials are just now leading suppliers to develop custom formulations which we hope will be proprietary, such as the bismaleimide matrix materials and their modifications.

Also, while all advanced composite matrix materials are desired to be tough and resist micro-cracking, a particular sought after category of materials is moderate use temperature highly toughened composites. Effort in this area certainly can lead to more proprietary formulas because there are a

number of ways of possibly obtaining such characteristics.

Additional emphasis on flame retardant, low smoke compositions is expected and there is the possibility of applying some new chemistry in this area so more proprietary formulations may be expected to evolve from this area.

SUMMARY

In summary then, as we suppliers listen to and understand the needs of the users in terms of the performance characteristics of the matrix portion of advanced composites, we devise more and more custom and we hope proprietary formulations.

DIFFERING USER SPECIFICATIONS RESULT IN CUSTOM FORMULATIONS

ADVANCED COMPOSITE MATERIALS WORKSHOP

DR. JAMES D. ALLEN
TECHNICAL DIRECTOR - COMPOSITE MATERIALS
FIBERITE CORPORATION/A UNIT OF BEATRICE FOODS

MATERIAL REQUIREMENTS - HISTORICAL DEMANDS ON THE RESIN MATRIX

- ATTAINMENT OF REASONABLE TRANSLATION OF REINFORCEMENT PROPERTIES
- UPPER USE TEMPERATURE
- CURE SCHEDULE
- OUT LIFE, WORKING LIFE AND HANDLING CHARACTERISTICS

SUPPLIER RESPONSE TO HISTORICAL MATRIX REQUIREMENTS

- SELECTION OF EPOXY BASE MATERIALS
  - SELECTION OF TGMDA EPOXY
- SELECTION OF LATENT HARDENERS AND CATALYSTS
  - SELECTION OF DADS HARDENER, BF_3, MEA CATALYST
- UTILIZATION OF ADJUNCT EPOXY MATERIALS
  - CARBONYL EPOXIES
  - EPOXY PHENOLIC NOVOLACS
  - EPOXY CRESOL NOVOLACS

RESULT - CUTOM FORMULATIONS

EVOLVING DEMANDS ON RESIN MATRIX

- ENHANCED ENVIRONMENTAL RESISTANCE
- TOUGHNESS
- CONTROLLED FLOW OR NO FLOW
- SELF ADHESIVE
- FLAME RETARDANT

SUPPLIER RESPONSE TO EVOLVING DEMANDS

CHANGE RESIN FORMULATIONS

- DELETE OR DECREASE ADJUNCT COMPONENTS ADVERSELY AFFECTING ENVIRONMENTAL RESISTANCE
  - DELETE CARBONYL EPOXY
- INCORPORATE COMPONENTS TO REDUCE FLOW AND OBTAIN SELF ADHESIVE CAPABILITY
  - ADD ELASTOMERS
- INCORPORATE COMPONENTS AND/OR CHANGE FORMULATIONS TO INCREASE TOUGHNESS
- MODIFY MATRIX FORMULATIONS FOR FLAME RETARDANCY
  - HALOGENATED EPOXIES
    INORGANIC SALTS

RESULT - MORE CUSTOM FORMULATIONS

FURTHER FRAGMENTATION

- REQUIREMENTS FOR YET HIGHER USE TEMPERATURE MATERIALS
  - 350°F + USE MATERIALS
    BMI'S
- REQUIREMENTS FOR MORE MODERATE USE TEMPERATURE TOUGHENED MATERIALS
  - 250°F USE MATERIALS
- ADDITIONAL EMPHASIS ON FLAME RETARDANT, LOW SMOKE COMPOSITIONS

VARIATIONS AMONG USER TEST SPECIFICATIONS REQUIRE ADDITIONAL SUPPLIER TESTING

Steve Russell

Hercules Incorporated

From a material supplier standpoint, the proliferation of new users of advanced composite materials is extremely exciting, since we can see whole new markets opening up and thus additional use of our material. However, in addition to new material types that are being asked for, leading us into a "candy store" business with each customer having different requirements, we note that each new user brings his own specification to the arena, thus requiring alternate sampling and testing for two people who might be buying the same products. This minimizes the effectiveness mass production should bring. In the area of carbon fiber testing, specifications are not generally too extreme and the normal supplier acceptance data will generally suffice. In the prepreg area, however, there are as many specifications as users and quite often, two different programs in the same company will have completely independent requirements and test methods. We thoroughly understand the need for this qualification testing, since different designs require different parameters. However, in the case of routine acceptance, where our Q.A. documents tell us the materials have been built the same way as qualification materials, the difference in customer specifications is sometimes difficult to understand.

To illustrate this point, let us assume that three different customers each ordered 2,000 pounds of AS4/3501-6 tape at 145 g/mtr^2 fiber areal weight and 35% resin content. One's first reaction would be that this is a perfect opportunity to manufacture 6,000 pounds of prepreg, test it as one batch and sell it. However, Table I illustrates the different requirements for three different specifications, and one can quickly see that you are just as well off to treat them as separate 2000 pound orders. Since these are all aeropsace companies and well established users, we assume they are all extremely sophisticated and in general using the material on a high technology part. Thus we have to ask ourselves, is customer C's material better because it received significantly more testing than material "A"? In addition to leading to confusion, these variations in specifications really prohibit us from building materials, testing them and putting them on the shelf, since we would not know how to test them. Thus every prepreg order becomes a special built quantity, for which we must have an order in hand to know how to test. This both lengthens delivery time and leads to gross inefficiencies in our operation. Is it any wonder we may charge more for a small order than for a large one and that we prefer long term schedules to allow us to restore some efficiency to our operation.

Unfortunately, even if these specifications were written to perform the same tests at the same frequency, experience tells us that the burden would be only half lifted, since chances are, differences in test methods would be required. To illustrate this point, let's examine

something as apparently simple as a 0° tensile requirement in depth,

In this test, as in all advanced composite mechanical tests, there are four basic areas of concern: laminate preparation, specimen preparation, testing and data reduction. Each of these will be discussed in some detail.

Laminate Preparation

Table II illustrates some of the variables which give us problems in laminate preparation. First of all, there are basically three different types of tensile test in use today; straight sided, bonded beam or tapered edge. For purposes of simplicity, we will assume that everybody agreed the straight sided tensile test was preferred. The first question we must ask ourselves is how thick the specimen must be, which will tell us how many prepreg plies to use. In the industry today, thicknesses anywhere from 0.035 to 0.125 are used for this specimen. After selecting a thickness and a panel size to be made, we can begin fabrication.

As we lay the panel up, the questions of how many and what type of bleeder to use, what type of edge dam and bagging material to use, are the dams tight or loose, aluminum or steel caul plates, top caul plate or not must be answered. Finally, we are ready to put the part in an autoclave, now the fun really begins. One customer wants the part straight to temperature and hold, another wants a hold step at 250°F with different rates of heatup initially versus 250 to 350°F. One wants post cure, another no post cure. One prefers 60 psi pressure and full vacuum from the beginning, another applies 100 psi at 200°F, still another wants vacuum vented when pressure is applied. Can these things really matter for material acceptance? For qualification maybe, but for acceptance, just to tell what the 0° strength of the material is? However, using whatever cure cycle, the part comes out of the clave and we are ready to begin making specimens.

Specimen Preparation

Still another set of questions must be answered as shown in Table III. How will we prepare the surfaces? One customer wants his surfaces grit blasted then solvent degreased, another only wants degreasing. One wants glass end tabls with a 30° taper to the gauge length, another graphite with a square face, still another glass with a square face, and let's not forget the no tab which is just wound in emery cloth. In end tab adhesive selection, again we have a variety of choices, most of which are based on what the customer happened to have on hand when he made his first specimen, not on any scientific test. We also must worry where to place the tabs, since gauge lengths for this particular test vary from 3" to 6". Next, the method of measuring strain should be considered, some specifications call for strain gauges, some for extensometers and some for crosshead compliance.

Testing

Finally, however, we are ready to actually break the specimen. Still there are many variables to consider, as shown in Tables IV and V. Instron crosshead speeds of 0.02 inches/minute, 0.05 inches/minute or 0.5 inches per minute are all used in specifications today. Some customers want the specimen preloaded, some don't.

Data Reduction

Once we have the specimen broken, however, another nightmare exists in reducing the data. Ways of obtaining modulus from a stress/strain curve vary anywhere from a tangent slope to a three place least squares fit of the curve. Additionally, modulus is taken anywhere from initial slope to slope at ultimate load. Again, the material is not changed by the modulus method, however, the number itself is dramatically altered. In reporting the data, the question of normalization emerges. Some specifications want no normalization, others normalize to nominal or measured ply thickness and still others normalize to fiber volumes ranging from 57 to 67%. Fiber volume methods extant vary from acid digestion to resin burnout to calculation based on laminate density.

Keep in mind that this degree of variation was based on the assumption that all users asked for a 0° straight sided tensile coupon test only. The variations for all the other physical and mechanical tests listed or just as great. The saddest comment is that most frequently these variations in testing are based on "in house" expertise and material availability and could all be standardized if the specification writers were made aware of the problems and additional costs they cause in materials manufacturing.

While we have spent considerable time discussing some 0° tensile test variations, this is only an example of the types of details in a specification which cause special testing for one particular customer. All of the other common acceptance tests have similar variations as illustrated in Tables VI and VII. One of my favorite statements is "when it comes to advanced composite testing, the devil is truly in the details."

During qualification, we understand the need for extraordinary testing. We view this as a one time occurrence and are more than happy to comply with requested testing. However, for routine acceptance where all materials are manufactured to standard conditions, all raw materials accepted to specifications and internal tests performed to assure material quality, it would seem that standard tests and sampling rates could be instituted. Remember, that during material acceptance, we are only checking to see that the material has not changed, not trying to develop a design base for material utilization. Thus my challenge to the advanced composites industry is let's develop a set of acceptance criteria which we jointly feel will assure material quality, standardize on test methods for determining these parameters and begin to apply them across the board. This will relieve the material manufacturers of the burden of making special lots for each individual order and allow him to concentrate on improved production efficiency. Additionally, the user will benefit due to the larger data base available for the type of product he is buying and the cost savings associated with larger lots and utilization of tests across customers.

TABLE I - 2000 POUND ORDER TEST REQUIREMENTS
PHYSICAL PROPERTIES (NUMBER OF TESTS REQUIRED)

SPECIFICATION	RESIN CONTENT	AREAL WEIGHT	TACK	DRAPE	FLOW	VOLATILES	GEL TIME	MOISTURE	IR	GPC
A	40	40				2	2			
B	6	6	6	6	6					
C	40	40	40		2	2		40	1	1

MECHANICAL PROPERTIES (PANELS/SPECIMENS)

SPEC.	RT 0° TENSILE (PANELS/SPEC.)	RT 0° COMP.	200° 0° COMP.	RT SBS	3 PT FLEX	250° SBS	250° WET SBS	RT ILS	180° ILS
A				2/6	2/6				
B	2/6			2/6		2/6	2/6		
C	5/25	5/25	5/25					5/25	5/25

TABLE II - ACCEPTANCE LAMINATE PREPARATION VARIABLES

VARIABLE	RANGE MAXIMUM	RANGE MINIMUM
Thickness	0.125 inches	0.055 inches
Bleeders	Calculated	0
Top Caul Plate	1	0
Cure Cycle Heat Up	Hold	Straight Up
Pressure Application	Initial	At 250°F
Vacuum	Full	Vented
Pressure	100 psi	25 psi
Dwell	4 hours	2 hours

TABLE III - TENSILE SPECIMEN VARIABLES FOR ACCEPTANCE TESTS

VARIABLE	RANGE MAXIMUM	RANGE MINIMUM	NUMBER OF DIFFERENT TYPES
Specimen Length	11"	9"	2
Gauge Length	6"	3"	4
Tab Materials	Graphite	None	5
Tab Adhesive	-	-	3
Tab Angles	90°	7°	5
Strain Gauge	Yes	No	3

TABLE IV - 0° TENSILE TESTING VARIABLES

VARIABLE	RANGE MAXIMUM	RANGE MINIMUM	NUMBER OF DIFFERENT TYPES
Crosshead Speed	.5" Min.	0.02" Min.	3
Preload	5,000 lbs.	0	
Chart Drive	Strain Gauge	Time	3

TABLE V - 0° TENSILE TEST DATA REDUCTION

VARIABLE	RANGE MAXIMUM	RANGE MINIMUM	NUMBER OF DIFFERENT TYPES
Modulus Slope	Ultimate	Initial	7
Normalization	67% Fv	None	5
Fiber Volume	Burnout	Ply Thickness	6

TABLE VI - OTHER MECHANICAL TEST MAJOR VARIABLES

FLEX STRENGTH/MODULUS	COMPRESSION TEST	SHEAR
Span/Depth	Beam/Coupon	Short Beam VS ILS
Thickness	Test Fixture IITRI Rohr Celanese Boeing	3 PT VS 4 PT
Fixture Configuration	Specimen Configuration Tabs Thickness Width	Fixture Configuration
3 PT VS 4 PT		Span/Depth
Modulus Measurement	Modulus Measurement	Thickness Measurement

TABLE VII - MAJOR PHYSICAL/CHEMICAL TESTS WHERE DIFFERENCES EXIST

RESIN CONTENT	GEL TIME	FLOW	VOLATILES	TACK/DRAPE
Sample Size	Sample Size	Sample Size	Sample Size	To Numerous
# Oos Test	Method	No. of Plies	Test Temp.	To List
Solvents	Temperature	Sample Holder	Test Time	
Drying Temp./Time		Test Temp./Time		

IR SCAN	L C	THERMAL ANALYSIS
Sample Size	Sample Size	Sample Size
Solvent	Solvents	Container
Scanning Speed	Column	Time Temperature
	Gradients	Data Reduction
	Wavelengths	
	Scan Reduction	

SUMMARY OF ISSUES AND RECOMMENDATIONS

MATERIALS AND PROCESS INTEGRITY:
SESSION I, INSPECTION CONTROLS, HOW EFFECTIVE

SUMMARY OF
ISSUES AND RECOMMENDATIONS

1. Educational Background of New Employees

Issue

Source QA representatives should be better prepared to perform their functions at suppliers' facilities. New employees need formal entry level educational background.

Recommendations

NSIA, ASQC, and DLA to recommend the establishment/enhancement of QA curricula to universities, state colleges, and community colleges. This recommendation would suggest the inclusion of specific source QA topics such as a system oriented approach to the control of suppliers, interpretations of QA specifications including MIL-I-45208A and MIL-Q-9858A, and technical and engineering functions.

Suggested OPR

NSIA, ASQC, DLA

2. MIL-I-45208A and MIL-Q-9858A Requirements

Issue

Many prime contractors who have MIL-I-45208A or MIL-Q-9858A include these requirements in their Purchase Orders to their suppliers. In many cases, these requirements are excessive and unwarranted. This is especially true in the case of MIL-Q-9858A being applied to raw materials.

Recommendations

a) NSIA should encourage its member firms to review the guidance for the application of these two specifications as contained in DAR before including them in contracts/purchase orders.

b) CODSIA and DoD should clarify the applicability of MIL-Q-9858A to raw material suppliers. The aluminum industry stresses the application of MIL-I-45208A only be applied to raw material suppliers.

Suggested OPR

NSIA/CODSIA/DoD

3. Use of Cost-of-Quality Information

Issue

Industry's weakest area in QA is in the utilization of cost-of-quality information. RADM Collins surveyed 184 companies and found that only 14% of them had an effective system for the use of this information.

Recommendations

a) NSIA should encourage member firms to institute or develop an effective system for the use of cost-of-quality information.

b) DoD should institute a stronger review of this requirement in MIL-Q-9858A.

Suggested OPR

NSIA/DoD

4. Purchase Document Requirements

Issue

Suppliers continue to be confused as to the requirements contained in purchase documents.

Recommendations

NSIA should encourage member firms to use more widely informational meetings such as pre-award bidder's conference or periodic supplier's conference. Such meetings would aid in the clarification of requirements before award is concluded and to assure program continuity.

Suggested OPR

NSIA

5. Controls in Procurement Documents

Issue

Currently there are no effective "front end" controls in procurement documents which require overt QA participation in drawing and procurement specification approvals.

Recommendations

The inclusion of such controls in the prime contract would allow QA participation in the earliest stages of contract performance thus maximizing the advantage of the planning of QA tasks. It is recommended that NSIA conduct a study of such controls for inclusion into contracts of high dollar value or criticality.

Suggested OPR

NSIA

6. Duplication of Specifications

Issue

Specifications from Government and various prime contractors cover the same casting process. An example of this duplication is MIL-C-6021 (Government), PSO 23001 (McDonnell Douglas), and FPS 1001 (General Dynamics). This duplication causes confusion, delays, and excessive cost.

Recommendations

It is recommended that NSIA canvass its members for specifications covering identical or similar manufacturing processes, combine them into one specification per process, and recommend to DoD the promulgation of a common specification covering this process.

Suggested OPR

NSIA/DoD

MATERIALS AND PROCESS INTEGRITY:
SESSION II, ALUMINUM INDUSTRY RESPONSE
TO PAST PROBLEMS

SUMMARY OF
ISSUES AND RECOMMENDATIONS

1. Prevention of Quality Departures

Issue

How do we prevent additional quality departures in aluminum products?

Recommendations

Continued emphasis by suppliers on improved process controls and preventive maintenance. Establish statistical techniques to control process.

Suggested OPR

Aluminum Industry

2. Profiting from Experience of Aluminum Industry

Issue

How can other raw material producers profit from the aluminum industry's experience, i.e., prevent problems and cooperate in developing solutions.

Recommendations

Develop dialogue between users such as NASA and AIA members and the various raw material producers such as steel, titanium, rubber, semi-conductors, software, etc.

Suggested OPR

AIA and NASA

3. MIL-H-6088

Issue

How can the aluminum heat treat specification, MIL-H-6088, be more effective for both the user and producer so as to prevent unforseen problems?

Recommendations

a) Continue meetings between Aerospace Industry Association and the Aluminum Association to resolve differences and make recommendations for revision to NAVAIR.

b) Develop heat treat specification for castings.

Suggested OPR

AIA, AA, and NAVAIR

MATERIALS AND PROCESS INTEGRITY, SESSION III,
ADVANCED COMPOSITE MATERIALS

SUMMARY OF
ISSUES AND RECOMMENDATIONS

1. Acceptance Test Requirements

Issue

Confusing and proliferated acceptance test requirements result in redundant (3X to 4X) testing.

Recommendations

a) Develop uniform material specifications.

b) Develop user/producer/raw material supplier industry association or working group.

c) Develop specifications on testing first, then expand to standardization specifications on process.

d) Base acceptance on performance not process/formulation control.

Suggested OPR

Government/Industry

NOTE: Need to tie in with Workshop D regarding NDT of Composites.

2. Different Material Requirements

Issue

Excessive parochial interests stimulate different formulations in material requirements. Industry ignores society (AMS, ASTM) committees. Each company has its own experts - develops its own specifications.

Recommendations

Support aeronautical material specification writing group, Nonmetallic Committee, Society of Automotive Engineers (SAE) to develop standard material specifications.

Suggested OPR

User Industry/Material Suppliers

3. Cost of Composite Material

Issue

High costs of advanced composite material

Recommendations

a) Qualify different resin formulations to the same specification. (Allows multisupplier approach and competitive pricing)

b) Formulate supplier/user "composite industry association" similar to aluminum industry. Address and agree on business as well as technical issues.

Suggested OPR

AIA and NSIA

4. User Confidence

Issue

User has no confidence in received composites meeting designed to data requirements. Must retest and reverify.

Recommendations

a) User require approval authority and freeze of producer/supplier total manufacturing process.

b) User require producer/supplier to define his "fingerprint" (remology).

c) User establish proprietary understanding with producer/supplier.

Suggested OPR

AIA and NSIA

WORKSHOP H

THE ACQUISITION PROCESS

Chairmen:

John Lloyd, LMSC
James Mahoney, AFSD

Coordinators

Carolyn Deatherage, Hughes
Paul Scandlyn, Hughes

WORKSHOP PAPERS

Carlucci Initiatives	Col. Dana Brabson, DSMC/DRI; Chuck Culver, Aerospace Corp.
Procedures to Improve New Systems Acquisition, Draft RFP, Source Selection, Spec Tailoring, Proposal Preparation Instructions, Inspection	Joe Krueger, AFSC; Ron Verba, AFSD Jim Horn, Boeing; Arlyn Nelson, LMSC; John Morey, Hughes; Jack Godsey, Morton Thiokol
Facilities Capital Cost of Money	Jack Nelson, AFSC; Roger Edmonds, Hughes
Performance Incentives/Award Fees	Bob Powell, LMSC; Lee Ginsberg, Martin Marietta; Barney Klehman, AFSD
AFSC Project Quality	Tony De Luca, AFSD; Chic Rideout, IBM
Commercial vs. Military Practices	Jim Vint, AFSD

The Defense Acquisition Improvement Program

Colonel G. Dana Brabson, USAF
Dean, Department of Research and Information
Defense Systems Management College
Fort Belvoir, Virginia 22060

Introduction

Two years ago, many people who had been through prior reforms of the weapon system acquisition process asked, "So, what is different this time?" The difference is the continued commitment of people at all echelons of the Department of Defense. The fact that this entire workshop is devoted to the Defense Acquisition Improvement Program is evidence of this commitment.

Today, there is another question on the minds of people in the acquisition community: "Now that the original architect of the AIP has left the Department of Defense, will the original thrust be forgotten?" The primary objective of this paper is to answer this question. I shall frame the answer in terms of the progress that has been made during the first two years and the work that remains to be done. Much of the information I shall present is reflected in the *Second Year-End Report*, signed by the Deputy Secretary of Defense on 8 June 1983.

As one reviews the history of the Acquisition Improvement Program, perhaps a half dozen key dates come to mind.

- 2 March 1981—Mr. Carlucci chartered five working groups to make recommendations on improving the acquisition process.
- 31 March 1981—The working groups provided their recommendations to Mr. Carlucci.
- 30 April 1981—Mr. Carlucci published his 31 decisions.
- 27 July 1981—Mr. Carlucci added the 32nd initiative.
- 12 January 1983—Mr. Thayer took the helm.
- 5 May 1983—Mr. Thayer announced his six "consolidated Acquisition Improvement Program initiatives."

Mr. Thayer's "consolidated initiatives" are actually a composite of twelve of the original initiatives. Table 1 shows the correlation.

Table 1. CONSOLIDATED ACQUISITION IMPROVEMENT PROGRAM INITIATIVES

CONSOLIDATED INITIATIVE	CORRESPONDING AIP ACTIONS
Program Stability	4 Program Stability
Multiyear Procurement	3 Multiyear Procurement
Economic Production Rates	7 Economic Production Rates
Realistic Budgeting	6 Budgeting to Most Likely Cost
	11 Budgeting for Technological Risk
	18 Budgeting for Inflation
Improved Support and Readiness	9 System Support and Readiness
	12 Funding for Test Hardware
	16 Contractor Incentives for Support
	30 Logistics and Support Resources
	31 Improved Reliability and Support
Encouraging Competition	32 Competition

During the course of this presentation, I shall pay particular attention to the twelve AIP actions which fit within the framework of Mr. Thayer's Consolidated Initiatives. I shall also address some of the other actions so that, at the end, you will have a good idea of the progress that has been made to date and of the key themes which are running through the Acquisition Improvement Program. It is important to emphasize that the absence of one of the original 32 actions from Table 1 in no way denigrates its importance. Rather, the absence of an action from Table 1 reflects the fact that Mr. Thayer has not selected it for his personal emphasis; the actions missing from Table 1 can be characterized as "completed" or "on track."

Controlled Decentralization
(Actions 17, 24, 26, 27, 28)

The objective of this thrust is to reverse some elements of the trend toward centralization. A particularly definitive statement of Mr. Carlucci's intent is provided in his 27 March 1981 memorandum entitled "Management of the DOD Planning, Programming and Budgeting System." In the memorandum, the Deputy Secretary of Defense stated: "We will achieve better defense management by working toward a system of centralized control of executive policy direction and more decentralized policy execution." The new policies are contained in the most recent revision of DOD Directive 5000.1, dated 29 March 1982.

The first step was to reduce the number of DSARC (Defense Systems Acquisition Review Council) decisions from four to two, as suggested by Figure 1. Two other features of this figure are also worth noting. First, in a step to provide better integration of the DSARC and PPBS processes, the Justification for Major System New Start (JMSNS, formerly, the Mission Element Needs Statement (MENS)) is now submitted with the Service Program Objective Memorandum (POM) package that provides funds for its execution. Second, the new milestone entitled "Program Go Ahead" is no longer rigidly tied to the beginning of Full-Scale Development. By opting to delay this milestone, possibly as late as Complete Design Review (CDR), the DSARC can achieve a more accurate view of cost, schedule, performance, industrial base preparedness, supportability, and testing prior to a decision to commit to the completion of Full-Scale Development, Production, and Deployment.

The second step, illustrated by Figure 2, was to double the thresholds which, if breached, require that the program be reviewed by the DSARC. Equally significant is the fact that the Services have likewise doubled their review thresholds, thus decentralizing by one level of management a significant number of programs.

Figure 2. DSARC REVIEW THRESHOLD

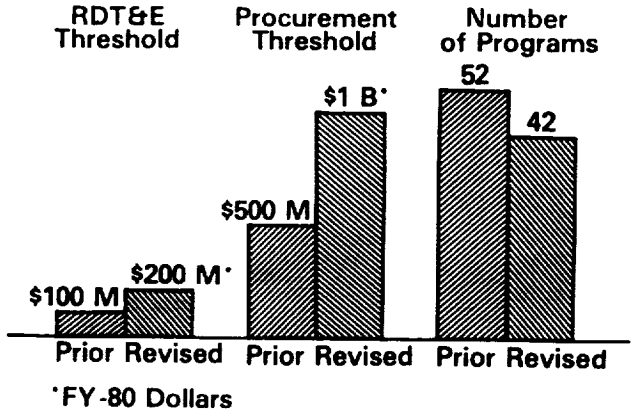


Figure 1. MAJOR SYSTEMS ACQUISITION PROCESS

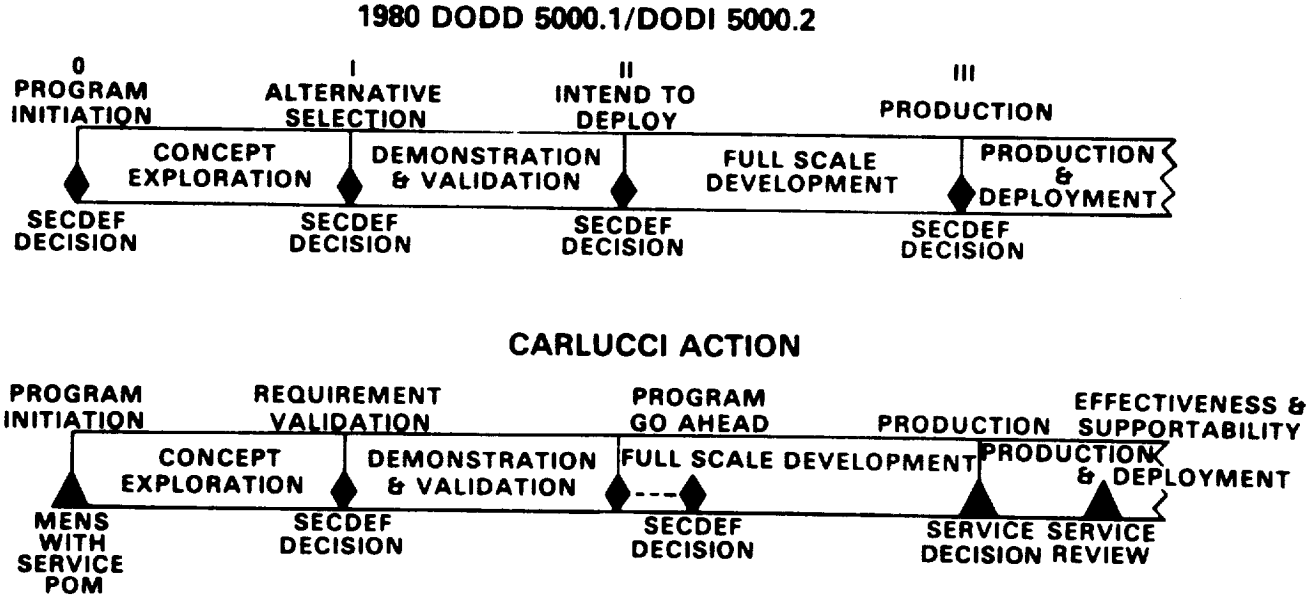
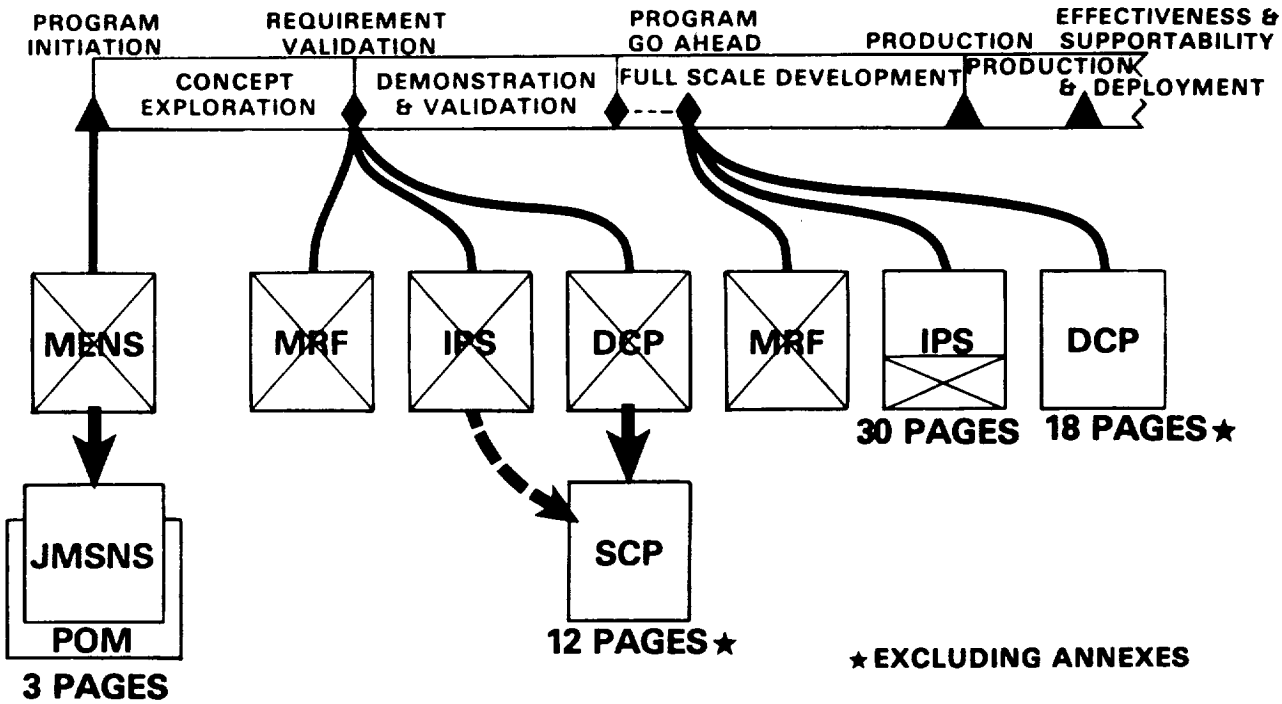


Figure 3. DSARC INFORMATION REQUIREMENTS



The third step was to reduce the number and size of documents required for a DSARC review. This step is graphically portrayed by Figure 3. These new requirements are spelled out in a newly revised DOD Instruction 5000.2, signed 8 March 1983.

Planning and Execution
(Actions 2, 3, 4, 6, 7, 8, 11, 15, 18, 19, 20, 22, 25, 29)

It is generally agreed in all sectors that cost growth is a major problem for the Department of Defense and that program turbulence is a principal contributor to cost growth. It is not surprising, then, that Mr. Thayer placed Program Stability at the top of his list of Priority Defense Management Initiatives. The dramatic impact of program turbulence is illustrated by the data displayed in Figure 4 for the F-15 program. The left-hand portion of this figure shows that, with one exception, the rate of production of F-15 aircraft has changed every year. As illustrated by the right-hand portion of this figure, one-half of the 94 percent cost growth reported in the 31 December 1981 Selected Acquisition Report is due to the fact that the OMB inflation indices underestimate the actual inflation rates. However, another one fourth of the cost growth is due to the changes in schedule portrayed by the left-hand portion of the figure. The Air Force has estimated this segment of the cost growth at $2B (in FY81 $). In an effort to reduce program turbulence, the OSD has asked the Services to nominate selected programs for a Stable Programs List. The Services have complied, and the list now stands at 27 programs.

Multiyear procurement has been adopted as the principal tool for stabilizing programs. Figure 5 compares a typical multiyear procurement with an equivalent set of annual procurements. The key provisions of the applicable portion of the FY82 Defense Authorization Bill are suggested by this figure: (1) The maximum length of a MYP contract is 5 years, (2) Congress must be notified if the cancellation liability will exceed $100M, and (3) The cancellation liability may be used to cover recurring (as well as non-recurring) expenses. In spite of the inherent flexibility provided in the FY82 bill, significant constraints have been applied to the application of the MYP concept.

Figure 4. PROGRAM STABILITY OF THE F-15 AIRCRAFT

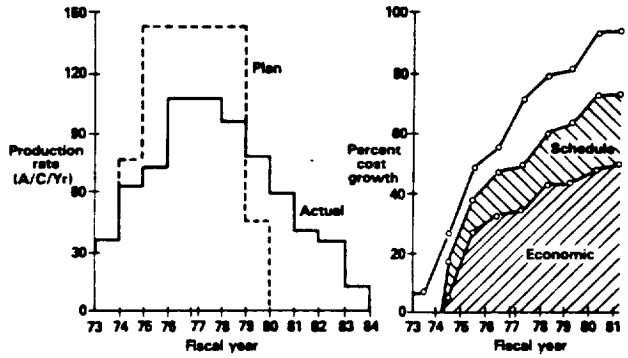
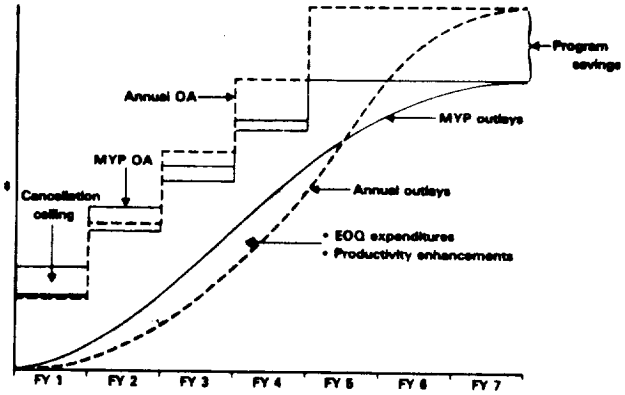


Figure 5. MULTI-YEAR PROCUREMENT



First, the DOD has stated that, as a matter of policy, the cancellation liability must be fully funded up-front. Secondly, the Congress now requires reporting on all programs with cancellation liability greater than $20M. Third, Congress requires prior notification on *all* Economic Order Quantity (EOQ) purchases. Fourth, Congress has disapproved several of the candidates proposed by the services.

Notwithstanding these limitations, MYP has been pushed vigorously. Table 2 gives the current status.

The current OSD policy on MYP programs was articulated in a memorandum signed by the Deputy Secretary of Defense on 29 December 1982. The key elements of this policy are quoted below.

> Maintain our current funding policy for major multiyear procurement programs, but retain, on an exception basis, the flexibility to allow other funding approaches when justified on a case-by-case basis. For smaller programs (those not identified by a separate budget line item), permit the Services to include recurring costs in an unfunded cancellation ceiling when justified on a case-by-case basis.
>
> Clearly articulate in the Fiscal Guidance the OSD commitment to provide up-front TOA for multiyear procurements.
>
> Establish a stable programs list comprised of the programs currently nominated by the Services for multiyear procurement.

The key fact to note here is that, in spite of stated policy limitations regarding full funding of the cancellation liability, OSD has in the past and will continue in the future to consider exceptions to the policy. Note that Table 2 is equivalent to the Stable Programs List mentioned earlier.

It is important not only to stabilize the rate of manufacture, but also to select a rate at which the fixed costs are distributed over a relatively large number of end items.

Table 2. STABLE PROGRAMS LIST: CURRENT AND PROPOSED MYP PROGRAMS

PROGRAMS • AIR FORCE EXAMPLES	STATUS	EXPECTED SAVINGS
Eight FY82 Programs • F-16 Airframe • AN/TRC-170 Radio • NAVSTAR GPS	Approved	$0.8B
Five FY83 Programs • KC-10	Approved	$1.0B
Six Additional FY83 Programs • B-1B	In FY84 Budget	$1.5B
Eight FY84 Programs	Proposed	$1.1B

This fact led to Action 7 dealing with economic production rates. The influence of production rate on unit cost (and hence on total program cost) is illustrated by Figure 6 in which the trend line is based on actual data for a series of aircraft programs. A concerted effort was made in conjunction with the FY83 budget submittal to accelerate programs to a more economic rate. In all, 18 programs were accelerated to rates higher than those proposed in the Carter FYDP; the estimated savings over the lives of these programs now stands at $2.6B. Representative Air Force examples in this list include F-15 Aircraft, F-16 Aircraft, E-3A Aircraft, AIM-9M Missiles, RF-4 IR Sensors, and Defense satellite programs. Unfortunately, some of the momentum has been lost from this initiative; for this reason, it is one of the initiatives selected by Mr. Thayer for his personal emphasis.

Figure 6. SENSITIVITY OF FIXED-WING AIRCRAFT COST TO PRODUCTION RATE

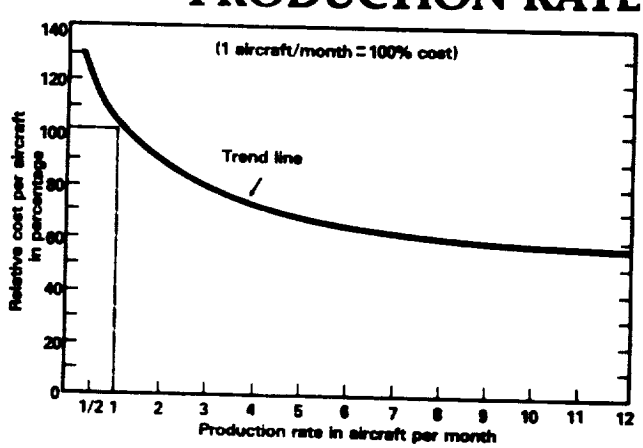


Table 3. SPECIAL INFLATION INDICES

FISCAL YEAR	ANNUAL RATE FOR MAJOR COMMODITIES*	ANNUAL RATE FOR OTHER PROCUREMENTS
1982	7.2%	5.5%
1983	6.5	5.0
1984	6.2	4.8
1985	6.0	4.6
1986	5.9	4.5
1987	5.9	4.5
1988	5.8	4.5

*Aircraft: APA, APAF, APN
Weapons: WTCV, PMC, WPN
Missiles: MPA, MPAF
Ships: SCN

The fourth area selected by Mr. Thayer is realistic budgeting; under this heading, he grouped three of the AIP actions. The first of these, Action 6, Budgeting to Most Likely Cost, deals with the problem of intentionally underestimating the cost of a system, a practice which ultimately leads to apparent cost growth, program stretchout, and criticism of our management ability. With respect to implementation of this action, emphasis has been placed on obtaining independent cost estimates as the basis for budget submissions. In 1982, OSD selected ten programs for special review and evaluation of independent cost estimates. For 1983, the list of programs has been expanded to 25.

As suggested by the data for the F-15 program (Figure 4), one of the principal sources of apparent cost growth is the difference between the OMB inflation rate we are permitted to use and the inflation rate which characterizes the real world. In an effort to achieve the ability to better plan for anticipated inflation, OSD requested and received permission to use a special set of inflation indices for selected weapon sytem procurements. These indices were published in January of 1982 and revised a year later. The current indices are shown in Table 3.

Action 11, Budgeting for Technological Risk, recognized the age-old problem that programs almost always seem to encounter unexpected difficulties and overrun. This characteristic is described schematically by Figure 7. As suggested by this figure, because of the technological uncertainties in the weapon system development business, the probability of coming in on the original cost estimate is almost always significantly less than 0.5. The good news is that a variety of techniques, usually computer based, have been devised for quantitatively estimating the delta in dollars that should be set aside as a management reserve to assure a 50:50 (or some other arbitrarily selected) chance of success. The most widely publicized technique is the Army's Total Risk Assessing Cost Estimate (TRACE) program.

One other AIP action deserves mention at this point: Action 2, Pre-Planned Product Improvement (P^3I). It is, of course, our strategy to offset the numerical advantage of our adversaries with technologically superior weaponry. All too often, however, our attempts to insert new technology as fast as possible lead to cost growth and delayed IOCs. The P^3I strategy, illustrated by Figure 8, reduces the technical risk and increases the likelihood of meeting the IOC by fielding the weapon system *without* the ultimate state-of-the-art technology but *with* provisions for incorporating the higher technology at a later date when the technology is more mature. To be effective, P^3I must be an integral part of the acquisition strategy and the following conditions must be satisfied: Planning must begin early in the acquisition cycle, funds must be set aside to develop the higher technology, and the necessary interfaces, electrical power requirements, etc., must be provided so that the improved technology can be inserted easily after IOC. In response to the FY83-88 Defense Guidance, the Services have identified 26 systems with significant P^3I efforts. Air Force systems include the F-15 Aircraft, F-16 Aircraft, B-1B Aircraft, EF-111 Aircraft, AWACS, IR Maverick, ASPJ, JTIDS, AMRAAM, and HARM.

Industrial Productivity (Actions 5, 32)

Concern for the industrial base stems both from the shrinking number of contractors who bid on Defense contracts and from the aging capital assets of these contractors. The principal OSD thrust in this area is to improve productivity by creating an environment in which improved productivity is stimulated and capital investment is

Figure 7. COST RISK

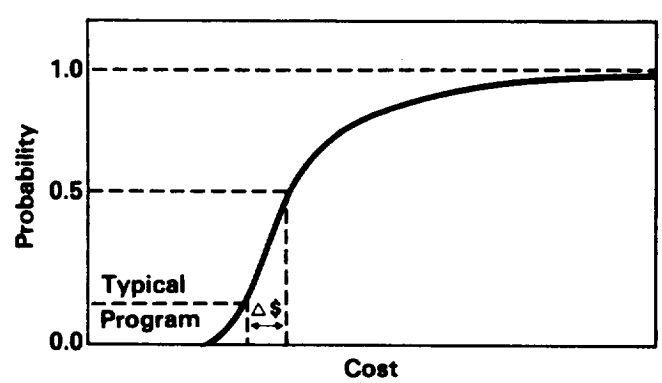
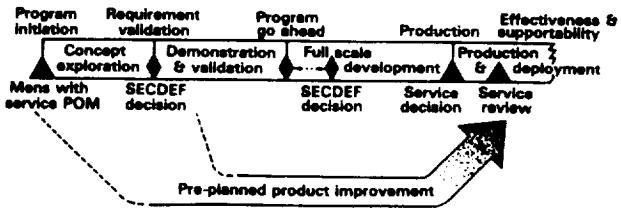


Figure 8. EVOLUTIONARY INTRODUCTION OF NEW TECHNOLOGY



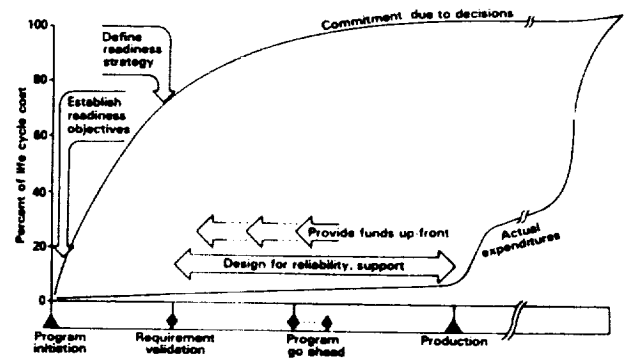
encouraged. A key ingredient in this environment is program stability, and multiyear procurement is an important tool. Given a long-term commitment by the government, the contractor is more willing to make the necessary commitment of resources.

Competition is, of course, a basic cornerstone of the free enterprise system. When applied intelligently, it stimulates development of innovative techniques for improving productivity, improves contractor performance, helps combat rising costs, increases the industrial base, and ensures fairness of opportunity for award of government contracts. Despite this, there is concern that our achievements are not adequate, and the Deputy Secretary of Defense is placing special emphasis on this initiative.

Increased emphasis on competition is far from new. In 1982, goals were established for each of the acquisition commands. Numerous techniques are being used to obtain competition. Examples of recent second sourcing decisions include the AIM-7M and AMRAAM missile programs. Second sourcing is also introduced at the subcontractor level; current Air Force examples include the F-16 canopy, ACES II ejection seat, the rocket motor for the AIM-9M missiles, and the 30-mm ammunition for the GAU-8 gun.

The objective of Action 5 is described by its title: "Encourage Capital Investment to Enhance Productivity." It contains more than a half dozen specific action items designed to stimulate capital investment and ease cash-flow problems. Some of the specific action items have already been accomplished. For example, flexible progress payment procedures have been implemented, increased progress payment rates have been authorized, and the excess profit provisions of the 1934 Vinson—Trammell Act have been repealed. In addition, the Services have been encouraged to place increased emphasis on their Manufacturing Technology (MANTECH) and Technology Modernization (TECHMOD) programs. The objective of the MANTECH program is to reduce the material acquisition costs and lead times by providing the manufacturing technology necessary to improve the productivity of the industrial base. The funding projected for the MANTECH program for the next five years (FY83-88) is double that for the previous five years. It is worth noting in passing that it is entirely appropriate to pursue a MANTECH program in parallel with an RDT&E program. Indeed, this acquisition strategy was adopted in the Air Force's AMRAAM Program.

Figure 9. RELIABILITY AND SUPPORTABILITY



The TECHMOD program was initiated to integrate existing manufacturing technologies into modern production facilities. A typical TECHMOD program is a joint Government-industry venture in which the Government invests in the manufacturing technologies and industry supports the new capital equipment and facilities. The classic example is the F-16 TECHMOD program in which a net savings of $370M will be shared by the Government and General Dynamics.

Readiness
(Actions 9, 12, 16, 21, 30, 31)

The last group of actions singled out by Mr. Thayer is the group dealing with support and readiness. The current policy is best expressed as follows.

> Improved readiness is a primary objective of the acquisition process, of comparable importance to reduced unit cost or reduced acquisition time. Resources to achieve readiness will receive the same emphais as those required to achieve schedule or performance objectives.
> —DEPSECDEF Memorandum, 30 April 1981

The principal thrusts of Actions 9 and 31 are illustrated by Figure 9. Two facts are particularly worthy of note in this figure. First, operationsl and support costs amount to about 60 percent of the total life-cycle cost for a typical weapon systems. Second, decisions made very early in the program define the majority of costs that will be incurred during the remainder of the life of the weapon system. In this context, Actions 9 and 31 require that the Readiness Objectives for the system be defined very early in the program, that the strategy for achieving these objectives be developed by and approved at the Requirements Validation Milestone, and that adequate funds be provided to assure that reliability and supportability are designed into and built into the system. In addition, these Actions require an early start of the test-fix-test process, and ask the program manager to examine the feasibility and potential payoff of concurrent development and tesing phases for "fast track" programs.

Action 16, Contractor Incentives for Reliability and Support, addresses the contract and the contractor specifically, and requires that the Program Management Office employ specific contractual incentives focused on designing for reliability and supportability.

Action 30, Program Manager Control of Logistics and Support Resources, seeks to give the program manager increased visibility of and participation in the supportability decisions affecting his program. In support of this Action, PPBS procedures were developed and implemented on a trial basis to identify more clearly the support funds budgeted in 1982 for 3 programs from each Service. The procedures will be applied to an expanded set of programs during the review of the FY85 POM submissions.

The thrust of Action 12, Front End Funding for Test Hardware, is to assure that an adequate number of test articles are provided early in the program. Two objectives are served. First, development and operational testing can be conducted concurrently, thus shortening the program without incurring appreciably greater risks. Second, the iterative test-analyze-fix-test phase and the combined environmental test phase can begin early in the development program, thus facilitating more rapid maturing of the technology, increasing reliability, and reducing supportability and maintainability costs.

As a group, the integrated logistics support actions have been vigorously pursued by the OSD. Both DODD 5000.1 and DODI 5000.2 have been revised to increase the priority of support and readiness, and DODD 5000.39 is being revised in the same vein. Implementation of these actions on individual programs is reviewed in the DSARC process. Specific programs in which changes have been made to improve readiness include .the ASW Stand Off Weapon, DIVAD Gun, AH-64 Helicopter, GLCM, LAMPS MK III, ASPJ, Patriot, and M-1 Tank.

Administrative Overhead (Actions 10, 13, 14)

The two foci in this area are first to seek revision of selected legislative requirements and second to reduce the burden imposed by DOD Directives and Instructions. With respect to the first objective, some results have been achieved and some action items are still being pursued. Thus, for example, the 1982 Defense Authorization Act raised the $10K limit for purchase order contracts to $25K, raised the mandatory threshold for contractor certification of cost and pricing data from $100K to $500K, and increased the threshold for Service Secretarial review of R&D Determination and Findings (D&Fs) from $100K to $5M. In addition, the 1982 Defense Appropriation Act doubled the reprogramming ceilings both for RDT&E (from $2M to $4M) and for procurement (from $5M to $10M). Legislative initiatives still in progress include the following representative examples.

- Amending 13 statutory thresholds to $25K
- Amending the Armed Service Procurement Act to authorize negotiation for "second sourcing."
- Amending the Walsh-Healey Public Contracts Act and the Contract Work Hours and Safety Standards Act to permit a 4-day, 40-hour week without premium overtime compensation.

With respect to reducing the burden imposed by DOD Directives and Instructions (and the Service regulations that are derived from them), a review group with members from OSD, the Services, NSIA and AIA completed its review of 132 acquisition-related DOD Directives and Instructions, and recommended that 31 be cancelled. Ten have been deleted thus far. Meanwhile, the emphasis has shifted to an examination of documentation required in DOD contracts. At the request of the USDRE, the Services and industry have conducted in-depth reviews of a small number of systems to identify excessive data and management reporting requirements. The results are being examined by the Defense Systems Management College and recommendations will be made to the USDRE in midsummer.

Summary

It should be apparent at this point that the Defense Acquisition Improvement Program is alive and well. The Deputy Secretary of Defense has concluded that 22 of the Actions are either completed or on track. He has selected 12 for his personal emphasis.

If one takes the entire group of 32 Actions as a group, a few key themes can be traced which give insight into the current management philosophy of the OSD. Possibly the most important is *program stability*. It is widely recognized that program turbulence is a primary source of cost growth. There are, of course, two dimensions to minimizing program turbulence. At the very highest levels, it is necessary to stabilize the funding levels and production rates. At lower levels, it is necessary to minimize all the actions that contribute to design and schedule changes. Multiyear procurement is viewed as a key tool in the fight to stabilize a core of the programs.

Strong emphasis is also being placed on *realistic budgeting*. The OSD views the challenge as falling jointly on the shoulders of Government and industry to increase the realism of budgets and increase the probability of coming in on the budget.

The third theme I would list is emphasis on *reliability and supportability*. As noted earlier, it is desired that integrated logistics support be given the same attention as that given to the achievement of cost, schedule and performance thresholds and goals.

Competition is percieved as a key tool in the drive to control costs, encourage innovation, and stimulate investment in the industrial base.

Another important theme is the thrust to achieve *economic production rates*. It is, of course, recognized that the achievement of an economic production rate is a key ingredient in the overall affordability of a weapon system. Moreover, the achievement of economic production rates invariably results in the production of more equipment for the field at an earlier date.

Emphasis on *incentives* appears throughout the Acquisition Improvement Program. Included are the traditional contractual incentives such as award fees. Stress is

also focused on innovative incentives such as sharing in cost savings. Incentives are suggested to encourage improved productivity and enhanced supportability in addition to the more traditional objectives of meeting cost, schedule and performance goals.

The seventh theme is *pre-planned product improvment*. The focus is, of course, both on increasing the probability of getting the weapon system in the field on time and on cost, and on providing an opportunity for the development and incorporation of a more advanced technology to meet the evolving threat.

The next theme I would cite is *concurrency*. The need for increased concurrency is evident when one takes note of the rapid development of new technologies. In a recent Army Science Board report on artificial intelligence and robotics, the authors indicate that a technological generation now spans about four years. The report goes on to note that, by comparison, a weapon system development program spans 8 to 15 years, or 2 to 4 technological generations. In this environment it is vitally important to introduce concurrency in those circumstances in which the risk can be managed; examples include concurrent development and operational testing and concurrent development and low rate production.

Tailoring of the acquisition process is, of course, closely related to concurrency. *Tailoring* was articulated in OMB Circular A-109, and has never been more appropriate than it is today. If a step in the acquisition process is unnecessary, delete it or combine it with another step. The net result will be more fighting capability in the field at an earlier date.

The last theme is *initiative*, for without *initiative*, particularly on the part of the program manager and his/her key subordinates, the good ideas that can save us money and time will never surface. It is the responsibility of each of us to search for ways to improve the process and be courageous enough to advocate our good ideas.

In conclusion, it is evident that that Defense Acquisition Improvement Program has had a significant impact on the acquisition of weapon systems. Moreover, the Deputy Secretary of Defense has accepted the challenge of adopting the toughest and most important of the initiatives for his personal emphasis. We can expect continued emphasis on this program and continued progress.

As I close, I recall a statement made by Mr. Carlucci during the very early days of the program. It seems equally applicable today.

> It's a large order and you're going to be at the center of it. A lot rides on it, not just your job, not just my job, not just this budget, but indeed the very capability of our country to sustain a defense buildup in the face of a very substantial buildup of the Soviet Union. I think we can do it. I think we can do it because we've got good people. I think we can do it because we've started out working together. I have confidence in you and I wish you well.

THE DEFENSE ACQUISITION
IMPROVEMENT PROGRAM
AN INDUSTRY PERSPECTIVE

Charles M. "Chuck" Culver
Director of Contracts
The Aerospace Corporation

The views expressed herein are the presenter's views based on his observations and contacts with industry personnel and participation with industry organizations. They are not the views of The Aerospace Corporation or of the NSIA.

Following is an explanation of the materials used:

MEMO TO MOSES
An analogy. Despite the emphasis in the initiatives on simplification, paperwork reduction, etc, industry does not see such changes but rather a greater emphasis on detail and the imposition of government micromanagement. The system is still overburdened.

CHARTS 1 - 4
The thirty-two "Carlucci Initiatives" introduced on 4/30/81.

CHART 5
The new emphasis announced by DepSecDef Thayer in the Spring of 1983.

CHART 6
E.O. 12352 "Federal Procurement Reforms" indicates current emphasis on reforms; also discussed:

- o Grace Commission Task Force (President's Private Sector Survey on Cost Controls).
- o Pending legislation involving procurement reform, HR 2545 and S-338.
- o Status of OFPP, CAS, the FAR, etc.

CHART 7
Initiatives in Trouble

Init. 2. Pre-Planned Product Improvement (P3I)
In Sept. 1981, OUSDRE report said could be most single important recommendation.

Industry still sees reaction rather than planning.

Still have problems with reliability, etc., of deployed systems.

Not on Thayer list for emphasis.

Init. 3. Multi-Year Procurement
Authorized only for DoD supply contracts.

Projected DoD savings not being realized.

Growing congressional disenchantment - notification threshold set at $20 million.

"Full Funding" is not "Full Funding".

Init. 4. Program Stability
Objective to fully fund R&D and major systems sufficient to protect acquisition schedule.

The problem - budget instability.

The culprit - congress.

Use of Continuing Resolution Authority.

Init. 5. Capital Investment Encouragement
CAS 409 not repealed.

Prompt Payment Act problems.

Difficulty with Economic Price Adjustment provisions.

Vinson-Trammel repealed - now HR 7446 to control excess profits, same factors as Renegotiation Act of 1951.

CAS 414 bacame an offset to profit/ fee.

Is the Defense Industrial Modernization Incentives Program a "ray of sunshine"?

Init. 8. Appropriate Contract Type
Still too much emphasis on fixed-price.

Not using fixed-price when could.

Fixed-price not appropriate for high risk, high tech. R&D.

Init. 13. Procurement Related Legislation
More legislative initiatives by congress than ever.

Not getting relief from costly, burdensome requirements.

CHART 8
Industry Concerns (a summary)

CHART 9
Initiative No. 1 - Management Principles.
o Emphasis on Item 8 - The adversarial relationship is growing rather than abating.

Memo to Moses from HEW

James E. Mignard, Ph.D.

The HEW staff has reviewed your application for permission to undertake a project on desert living and thinks it has some merit. However, we cannot approve it as it stands because you have not interfaced with Federal rules and regulations on socially-oriented proposals and have failed to meet several Government standards. A reading team studied your program carefully. Their criticisms and recommendations in the following paragraphs will help you understand our concerns. You may want to rewrite the proposals according to our suggestions.

First of all, it is not clear to us whether your ten words, or *ten commandments*, as some of our staff call them, are guidelines for a bona fide program for governing people in a nomadic societal setting. We assume they were more then guidelines; therefore, we must point out that we do not approve social programs that have been designed without assistance of consultants who have been certified by HEW.

The regulations require that experts with doctorates and at least ten years experience be consulted whenever a proposal writer deals in areas in which he/she has no professional competence. You speak, for example, of family relationships, neighbors and *"the sojourner who is within your gates"* (tidy up this phrase a bit), the obvious domain of sociology, but you have not listed a single sociologist as a consultant.

There is no evidence that you sought the advice of lawyers when you drafted the sections on killing and stealing. We think a scholar with a national reputation in ethics could have improved what you were trying to say about bearing false witness and committing adultery. And was there a good reason for avoiding theologians when you wrote about having no other gods or not taking the divine name in vain or not making graven images?

Your failure to utilize professional testimony, seriously weakens the proposal; indeed if we were to approve it, our own experts advise us that it would not last more than four or five years.

More specifically, we question the soundness of many of your ideas. Take the phrase, *"You shall have no other gods before me."* Not only is this unclear, but we are troubled by the absence of measurable behavioral objectives. How will you know whether this is being accomplished? Do you plan to administer pre- and post-tests, or will you use some kind of peer review? If this idea is so valuable, why is it so negative? In fact, why have you *emphasized* the negative in these propositions? We believe your proposal would be more easily implemented if it were recast in positive terms.

The *"graven image"* statement. We are puzzled by the weight you give this both in terms of the number of words you have used and its position among the ten. Did you prioritize? Does it really belong in second place? Is it necessary to include the threats? Furthermore, as you surely know, a needs assessment must accompany every application submitted for HEW approval, but you have none. Without supporting data, we have no way of knowing whether there is a genuine *"graven image"* problem among your people. Please detail this in an appendix with charts, bar grams, histograms and statistics.

"Taking the name in vain" is a fuzzy expression. A couple of good footnotes on recent research in Near Eastern onomastics would show that you know what you're talking about.

The *"sabbath day"* paragraph seems overly long and of questionable value. What exactly do you mean by *"remembering"* the sabbath day? How is *"holy"* to be understood? Have you considered a more modern word like *"unstressful"*? Our proposal readers were dismayed that you have chosen to stay with the nuclear family concept. Please review the literature.

"Honor father and mother." If this section is crucial to your project, we suggest you define *honor*. Is there some reason why father precedes mother in the rank-ordering? You state that compliance will affect the lives of the people *"in the land which the Lord your God gives you."* We studied this strange notion and could not determine the significance of real estate here. Are you implying that there is a direct relationship between parent-honoring and land-holding?

In the opinion of the HEW staff, the next five statements could be lumped together, using, however, less flamboyant language. Verbs like *kill, commit adultery, steal, lie and covet* do not sit well with most people.

In our overall examination of your proposal we were struck by the fact that you have made no provisions for the utilization of role-models. We do not think that asking the people to follow your ten points simply because it pleases God is an effective substitute for role-models. If you had attended one of our regional workshops, you would have learned that we do not approve projects that are weak on role-modeling.

Finally, we wish to point out that the rules and regulations stipulate that 14 typed copies of the proposal be sent to us. Several HEW staff members were upset because you disregarded this. Two stone tablets with chiseled letters do not comply with Federal regulations. ◘

DEFENSE ACQUISITION IMPROVEMENT PROGRAM

(INITIATIVES 1 THROUGH 8)

	INITIATIVE	INTERNAL ACTION ONLY	OMB OR CONGRESS
1.	MANAGEMENT PRINCIPLES	X	
2.	PREPLANNED PRODUCT IMPROVEMENT	X	
3.	MULTIYEAR PROCUREMENT		X
4.	PROGRAM STABILITY	X	
5.	CAPITAL INVESTMENT ENCOURAGEMENT		X
6.	BUDGET TO MOST LIKELY COSTS	X	
7.	ECONOMIC PRODUCTION RATES	X	
8.	APPROPRIATE CONTRACT TYPE	X	

DEFENSE ACQUISITION IMPROVEMENT PROGRAM

(DEP SECDEF THAYER EMPHASIS)

•	ECONOMIC PRODUCTION RATES	7
•	MULTIYEAR PROCUREMENT	3
•	INCREASED PROGRAM STABILITY	4
•	REALISTIC BUDGETING	6, 11, 18
•	ENHANCED COMPETITION	32
•	IMPROVED READINESS AND SUPPORT	9, 12, 16, 30, 31

DEFENSE ACQUISITION IMPROVEMENT PROGRAM

(INITIATIVES 9 THROUGH 16)

	INITIATIVE	INTERNAL ACTION ONLY	OMB OR CONGRESS
9.	SUPPORT AND READINESS	X	
10.	ADMINISTRATIVE COST AND TIME		X
11.	BUDGET FOR TECHNOLOGICAL RISK	X	
12.	FRONT-END FUNDING FOR TEST HARDWARE	X	
13.	PROCUREMENT RELATED LEGISLATION		X
14.	REDUCE NUMBER OF DOD DIRECTIVES	X	
15.	FUNDING FLEXIBILITY		X
16.	INCENTIVES - RELIABILITY AND SUPPORT	X	

DEFENSE ACQUISITION IMPROVEMENT PROGRAM

(INITIATIVES 25 THROUGH 32)

	INITIATIVE	INTERNAL ACTION ONLY	OMB OR CONGRESS
25.	MENS IN THE POM	X	X
26.	DSARC MEMBERSHIP	X	
27.	DEFENSE ACQUISITION EXECUTIVE	X	
28.	DSARC REVIEW CRITERIA	X	
29.	DSARC/PPBS DECISION INTEGRATION	X	
30.	PROGRAM MANAGER CONTROL OF SUPPORT	X	
31.	IMPROVE RELIABILITY AND SUPPORT	X	
32.	INSURE (INCREASE) COMPETITION	X	X

DEFENSE ACQUISITION IMPROVEMENT PROGRAM

(INITIATIVES 17 THROUGH 24)

	INITIATIVE	INTERNAL ACTION ONLY	OMB OR CONGRESS
17.	DSARC BRIEFING AND DATA R'QMTS.	X	
18.	BUDGETING FOR INFLATION		X
19.	FORECASTING BUSINESS BASE CONDITION	X	
20.	SOURCE SELECTION PROCESS	X	
21.	STANDARD OPERATIONAL AND SUPPORT SYSTEMS	X	
22.	DESIGN-TO-COST GOALS	X	
23.	ACQUISITION PROCESS DECISIONS IMPLEMENTATION		X
24.	DSARC DECISION MILESTONES	X	

EXECUTIVE ORDER 12352
FEDERAL PROCUREMENT REFORMS
(ISSUED 17 MARCH 1982)

INTERAGENCY TASK GROUPS

- ESTABLISH CLEAR LINES OF CONTRACTING AUTHORITY AND ACCOUNTABILITY
- IMPROVE GOVERNMENT PROCUREMENT SYSTEM PERFORMANCE
- REDUCE ADMINISTRATIVE COSTS AND BURDENS
- SIMPLIFY SMALL PURCHASES
- PROCUREMENT CAREER MANAGEMENT PROGRAMS

DEFENSE ACQUISITION IMPROVEMENT PROGRAM

(INITIATIVES IN TROUBLE)

2. PREPLANNED PRODUCT IMPROVEMENT

3. MULTIYEAR PROCUREMENT

4. PROGRAM STABILITY

5. CAPITAL INVESTMENT ENCOURAGEMENT

8. APPROPRIATE CONTRACT TYPE

13. PROCUREMENT RELATED LEGISLATION

DEFENSE ACQUISITION IMPROVEMENT PROGRAM

(INDUSTRY CONCERNS)

- LACK OF UNIFORMITY WITHIN THE GOVERNMENT.
- LACK OF UNIFORMITY BETWEEN THE SERVICES.
- FAILURE TO FLOW-DOWN TO FIELD.
- CONTRACT MICRO-MANAGEMENT.
- INITIATIVE 1, ITEM 8 - ADVERSARIAL RELATIONSHIP.

DEFENSE ACQUISITION IMPROVEMENT PROGRAM

INITIATIVE NO. 1 - MANAGEMENT PRINCIPLES

1. PROVIDE SERVICES WITH IMPROVED STATEMENT OF LONG-RANGE DEFENSE POLICY, STRATEGY AND RESOURCES ..... TO ENHANCE PROGRAM STABILITY.
2. PLACE RESPONSIBILITY, AUTHORITY AND ACCOUNTABILITY AT THE LOWEST LEVEL ..... WHERE A TOTAL VIEW RESTS.
3. GIVE SERVICE PROGRAM MANAGERS THE RESPONSIBILITY, AUTHORITY AND GUIDELINES ..... ADEQUATELY TO EFFICIENTLY EXECUTE THE PROGRAM. (INCLUDES FLEXIBILITY TO TAILOR .... TO ESTIMATES OF DEVELOPMENT PRIORITIES AND RISKS).
4. EXPLORE LOWER RISK EVOLUTIONARY ALTERNATIVES WHEN NEW PROGRAMS ARE PROPOSED.
5. A FUNDAMENTAL GOAL IS ACHIEVEMENT OF ECONOMIC PRODUCTION RATES.
6. SERVICES SHOULD ..... REALISTICALLY BUDGET AND FULLY FUND ..... AT LEVELS TO PROTECT SCHEDULE AND ACHIEVE ACCEPTABLE READINESS.
7. READINESS IS A PRIMARY OBJECTIVE OF THE ACQUISITION PROCESS, OF COMPARABLE IMPORTANCE TO REDUCED UNIT COST OR REDUCED SCHEDULE.
8. THE PROPER "ARMS-LENGTH" BUYER-SELLER RELATIONSHIP SHOULD NOT BE INTERPRETED BY GOVERNMENT OR INDUSTRY AS ADVERSARIAL.

SOURCE SELECTION

Joseph W. Krueger
Staff Source Selection Officer
Air Force Space Division, LAAFS

There are several acquisition elements currently being emphasized within DOD which I'd like to share with you. They have a connection with the purpose for our gathering here today. One of these is the approach which Headquarters Systems Command of the Air Force has taken with respect to the usage of the draft request for proposal or the DRFP, its acronym. I'll also touch on the emphasis which for some years the Space Division has been placing on mission assurance in its source selections. Another item is the start which the Contract Management Division has made with its Contractor Past Performance Program otherwise referred to as the CP^3 program. This is not to be confused with the old Contractor Performance Evaluation Program or CPE of the 1960's which proved to be too expensive and was cancelled in the early 1970s.

First let me address the basic subject of source selection. Every contract awarded at Space Division is not the culmination of a source selection. The bulk of our major contracts are called follow-on contracts. Way back, possibly ten or fifteen years ago we held competition, for instance, for a satellite. Once that satellite passed its qualification tests and initial satellites were delivered, we may find it advantageous or necessary to buy some more almost identical units. The economical and practical approach is to negotiate a follow-on contract for the quantities desired.

Some years we may have only four or five source selections, that is those competitions which exceed two million dollars in value. In other years we may have a dozen. The quantity is dependent on the budget and new programs coming into being. This year has been particularly active in this respect. Some of these are concept studies; others can be validation phase efforts followed by full scale engineering developments and then production. In the early phases, more than one contract is awarded so that competition is extended throughout the program. Each is handled basically the same in principle with the detail needed varying with the phase of the program. Some of our largest competitions are for operations and maintenance of such facilities as the Vandenberg Western Space and Missile Center. Then too there are integration and technical services contracts which are competed.

In all cases one person makes the decision as to who shall be awarded the contract or contracts. As you can see from the chart the dollar value and the importance of the program determines who is the Source Seclection Authority SSA. Ninety-nine percent of the time, the Commander Space Division is the SSA for major competitions. As the dollars fall off, the Vice Commander and then the program director is the SSA. The contracting officer handles the low dollar value ones. In each instance the proposals are evaluated using the evaluation factors for award contained in the RFP. Only after all evaluation is completed is any comparison of one offeror with another made. The SSA is presented the facts and makes the decision which he justifies in writing. There is no such thing as transfusion or auctioning and historically at SD the low offer only wins twenty percent of the time. The chart here reflects the SD record covering 50 competitions. The low offer may be represented by a poor proposal. Where a high priced offer wins a competition there must be very good reasons why the extra dollars represent a better buy. Mission Assurance may have been a key item of consideration.

While most competitions are decided by the Commander or his designee, the major acquisitions include advisors from higher headquarters, the using commands and the office of the Secretary of the Air Force.

These personnel review all the findings and provide significant analyses for the SSA. These people also are involved from the start, with the RFP, any competitive range determination and at the time of decision. In many cases, briefings are required up the chain of command before any award is made, to make sure all important people know the real basis for selection and not what mis-informed or losers might indicate was the basis for award.

With respect to RFP's issued, SD for quite some years has released a work statement with appended specifications and requirements in advance as a draft or partial RFP. The idea being that while the model contract and other content of the RFP is being finalized, industry has an opportunity to criticize the DRFP. Many valuable suggestions have been received so that the final document is clear. Proposal preparation time is not impacted and potential offerors have a factual basis for detail planning. In some cases, one on one conversation with the program office is encouraged. In some instances the DRFP is complete with schedule, model contract, proposal preparation instructions and the evaluation factors for award. This is the desired approach. In the past the value of the competition was not used exclusively to determine when a DRFP was appropriate.

On 4 Feb 1983, Systems Command made effective a change to its Supplement to the Defense Acquisition Regulation 3-550, Use of Draft Request for Proposal or DRFP. Heretofore the DRFP was to be used for practically all competitive acquisitions. Some contracting officers also used the DRFP for sole source and selected source acquisitions. Eventually this latter usage was made a No No by the Air Force. Now its usage is more selective.

For most competitions, the contracting officer has authority to use or not to use the DRFP. Systems Command wants its usage to be confined to the major programs or large dollar value acquisitions. Contact with industry has revealed cases where contractors spend a lot of time and money in reacting to the DRFP. They found, however, in some cases they had to do the job all over because the final RFP substantially differed from that which was originally released. Proper control and review prior to release of the DRFP has since been enforced. Desired benefits are therefore assured; offerors get a head start in planning their proposals.

For the record, major acquisitions are defined in the Department of Defense Directive 5000.1, titled, Major Systems Acquisitions. These are primarily the DSARC programs or those designated by the Secretary of Defense as requiring usage of a Defense Systems Acquisition Review Council. They are addressed in this directive as well as in its companion document DODD 5000.2, titled, Major Systems Acquisition Procedures. Systems Command, however, tempers the DRFP base line by requiring usage of the DRFP where an individual contract, with any options, totals in excess of $25 million in value. The directives talk in terms of program dollars while the Systems Command and the divisions are normally concerned with contract dollar value. Be that as it may, the use of the DRFP may be waived by the Division Deputy for Contracting, such as SD/PM. Heretofore all contracts anticipated to exceed two million were to use the DRFP. Now, also, release of the Draft RFP is to be announced in the Commerce Business Daily.

From a contractor's point of view the DRFP is advantageous, particularly for an assessment of the importance the Government places on Mission Assurance. At the Space Division for systems acquisitions, mission assurance is usually an item of consideration. This means the decision maker, the source selection authority, will definitely be briefed on the relative merits of the individual offers with respect to

proposed approach to mission assurance.

As a part of the technical proposal, quite frequently a separate mission assurance volume is requested covering such factors as Quality Assurance, Maintainability, System Safety, Reliability and Parts and Material and processes. Naturally the contract administrative office of the DCAS or AFCMD is consulted for each offeror's history of performance on these same subjects.

The role of the Contract Management Division of Systems Command in supporting the buying divisions in the selection of contractors for major acquisitions has been enhanced by their activating a Contractor Past Performance Program called CP^3. Basis for this move is the regulation AFCMDR 70-23 dated 1 March 1982. The Systems Command believes that past performance data has the potential for indicating an offeror's ability to fulfil the promises and projections contained in their proposal for a contract or program. Their objective is to provide a standardized system for collecting data, documenting the record and reporting on selected contracts. Reports are to be objective.

The major program definition is the same as that previously mentioned or that in the DOD Directives 5000.1 and 5000.2. Reporting will also be completed on subcontracts which exceed the thresholds of major programs or $100 million. Currently 37 programs have been identified for application of the CP^3 Program. Eight of these are Space Division programs. The reporting is categorized under ten specific facets. Details or precisely what will be included under each category is detailed in the regulation. All aspects of mission assurance are identified. The categories are aligned with the individual functions performed by an Air Force Plant Representative Office. Content of the data files is identified as For Official Use Only and therefore, is not releasable outside the Department of Defense.

The AFCMD has been increasingly helpful in providing inputs to Systems Command business strategy and acquisition planning. Their lessons learned impact the statements of work and contract clauses. The related past performance of offerors in competitive acquisitions, per Systems Command Supplement to AFR 70-15, must be a specific criterion and at least equal in importance to all other criteria. In summary, the formalizing and standardization initiated by the regulation AFCMDR 70-23 is of interest to our contractors and important to the Sapce Division in the analysis of contractor capability to satisfactorily perform similar contracts or programs.

The foregoing represents an approach in being to improve the acquisition of new systems. Additional aspects are being covered by other personnel in this workshop program.

MISSION ASSURANCE

THE ACQUISITION PROCESS

MR. JOSEPH W. KRUEGER
USAF SPACE DIVISION

SOURCE SELECTION CATAGORIES

CATEGORY	THRESHOLD	SSA
A	• EACH NEW DEVELOPMENT PROGRAM ESTIMATED AT $100 MILLION OR MORE RDT&E FUNDS FOR VALIDATION AND FULL SCALE DEVELOPMENT. • NEW PRODUCTION PROGRAM REQUIRING $500 MILLION OR MORE PRODUCTION FUNDS. • ANY MODIFICATION, MAINTENANCE, SERVICES OR OTHER PROGRAM/PROJECT ESTIMATED TO REQUIRE $500 MILLION OR MORE. • OTHER PROGRAMS OR PROJECTS DESIGNATED BY SEC DEF, SEC AIR FORCE OR HQ USAF.	SEC AIR FORCE WHO MAY DELEGATE AUTHORITY (AFR 70-15)
B	• ACQUISITION ESTIMATED AT $10 MILLION OR MORE BUT LESS THAN THRESHOLDS ABOVE • ACQUISITIONS VALUED MORE THAN $2 MILLION BUT LESS THAN $10 MILLION	COMMANDER/ VICE COMMANDER SPACE DIVISION "DEPUTY FOR"
C	• ACQUISITIONS VALUED AT LESS THAN $2 MILLION	PCO

SOURCE SELECTION
HOW DOES SD SELECT A CONTRACTOR?

- ISSUE A REQUEST FOR PROPOSAL (RFP) TO POTENTIALLY CAPABLE CONTRACTORS
- EVALUATE PROPOSAL RESPONSES IN TERMS OF RFP EVALUATION FACTORS FOR AWARD (SECTION M)
- CHOOSE OFFER(S) WHICH HAS HIGHEST DEGREE OF REALISM

DECISION MAKER

THE SOURCE SELECTION AUTHORITY IS RESPONSIBLE FOR THE PROPER AND EFFICIENT CONDUCT OF THE ENTIRE PROCESS, ENCOMPASSING PROPOSAL SOLICITATION, EVALUATION, SELECTION AND CONTRACT AWARD. HE SHALL HAVE (SUBJECT TO LAW AND APPLICABLE REGULATIONS) FULL RESPONSIBILITY AND AUTHORITY TO SELECT THE SOURCE (S) FOR AWARD AND APPROVE THE EXECUTION OF THE CONTRACT(S).

OBJECTIVES

- SELECT THE SOURCE WHOSE PROPOSAL HAS THE HIGHEST DEGREE OF REALISM AND CREDIBILITY AND WHOSE PERFORMANCE IS EXPECTED TO BEST MEET GOVERNMENT OBJECTIVES AT AN AFFORDABLE COST
- ASSURE IMPARTIAL, EQUITABLE, AND COMPREHENSIVE EVALUATION OF COMPETITOR'S PROPOSALS AND RELATED CAPABILITIES
- MAXIMIZE EFFICIENCY OF THE SOLICITATION, EVALUATION AND SELECTION DECISION

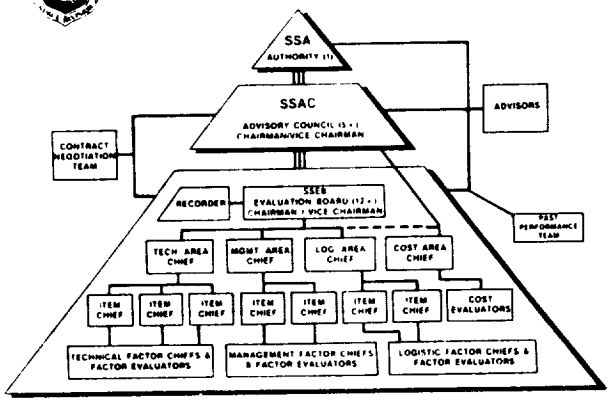
BASIS FOR AWARD

	TYPE CONTRACT				
	FFP	FPIF	CPIF/ CPAF	CPFF	TOTAL
CONTRACTS AWARDED	11	8	18	12	50
LOW OFFER WON	4	2	2	2	10
% LOW OFFER WON	36%	25%	11%	17%	20%

THE ABOVE STATISTICS COVER FY76-82 COMPETITIONS EXCEEDING $2 MILLION IN VALUE BUT DO NOT INCLUDE INSTANCES WHERE MULTIPLE AWARDS WERE MADE NOR WHERE ONLY ONE RESPONSE WAS RECEIVED OR WHERE MAXIMUM ALLOWABLE COST TO THE GOVERNMENT WAS SPECIFIED IN THE SOLICITATION.

PAGE 18 INTENTIONALLY BLANK

SOURCE SELECTION (SS) BOARD
CATEGORY "A" ACQUISITIONS



MISSION ASSURANCE
PROPOSAL EVALUATION, SPECIFIC CRITERIA
(AREA ITEMS)

TECHNICAL AREA

- SPACE VEHICLE SYSTEM (ITEM)
- TEST AND EVALUATION
- SYSTEMS ENGINEERING
- MISSION ASSURANCE
- SUPPORT OPERATIONS & SERVICES
- SPECIAL TEST EQUIPMENT

AFSC DAR 3-550 SUPPLEMENT USE OF DRAFT REQUEST FOR PROPOSALS (DRFP)

MISSION ASSURANCE
PROPOSAL PREPARATION INSTRUCTIONS

MISSION ASSURANCE VOLUME V

- QUALITY ASSURANCE
- MAINTAINABILITY
- SYSTEM SAFETY
- RELIABILITY
- PARTS, MATERIAL, PROCESSES

4 FEB 1983
REVISION

AFSC DAR 3-550 REVISED

- REQUIRED FOR MAJOR PROGRAMS DEFINED IN DODD 5000.1, MAJOR SYSTEM ACQUISITIONS
- REQUIRED FOR CONTRACTS VALUED IN EXCESS OF $25 MILLION
- WAIVER AUTHORITY—AFSC DIVISION DEPUTY FOR CONTRACTING
- MAY BE USED ON OTHER COMPETITIVE ACQUISITIONS

AF CONTRACT MANAGEMENT DIVISION

CONTRACTOR PAST PERFORMANCE PROGRAM (CP3)

AFCMDR 70-23
1 MARCH 1982

PAST PERFORMANCE DATA

DATA HAS THE POTENTIAL FOR INDICATING AN OFFEROR'S ABILITY TO FULFILL THE PROMISES AND PROJECTIONS CONTAINED IN THEIR PROPOSAL FOR A CONTRACT / PROGRAM

AFCMDR 70-23 OBJECTIVES

- PROVIDE A STANDARDIZED SYSTEM FOR:
  - COLLECTING DATA
  - DOCUMENTING RECORD
  - REPORTING ON SELECTED CONTRACTS
- REPORT OBJECTIVE INFORMATION TO PROGRAM OFFICES

PROGRAM CRITERIA

- ANY R&D CONTRACT / PROGRAM EXCEEDING $100 MILLION VALUE
- ANY FULL SCALE DEVELOPMENT AND PRODUCTION CONTRACT / PROGRAM EXCEEDING $100 MILLION VALUE
- PROGRAMS MEETING DOD/ (5000.1 & 5000.2) CRITERIA AND SO DESIGNATED
- WHEN A SUBCONTRACT'S VALUE EXCEEDS THRESHOLD'S ABOVE

CONTRACTS (37) SELECTED FOR PHASE I PAST PERFORMANCE DATA BASE

BOEING · DET 9 (9)
ALCM· ASD
ESTS · ASD
B-1B · ASD
MINUTEMAN · BMO
MX · BMO
E-3A · ESD
E-4 · ESD
IUS · SD
ASAT · SD

HUGHES · DET
F-15 RADAR · ASD
MAVERICK (IR)
HALO · SD
AMRAAM · AD
WASP · AD

WESTINGHOUSE · DET 45 (4)
ALQ · 131 · ASD
F-16 RADAR · ESD
ALQ · 153 · ASD
E-3A · RADAR · ESD

MARTIN MARIETTA · DET 10 (4)
TITAN III · SD
GSSI · SD
PIC · SD
MX · BMO

PRATT & WHITNEY · DET 4 (2)
F-15/F-100 ASD
F-16/F-100 · ASD

GENERAL ELECTRIC · DET 36 (2)
MK 12 · BMO
DSCS · SD

CONTRACTS (37) SELECTED FOR PHASE I PAST PERFORMANCE DATA BASE (CONT'D)

THIOKOL · DET 43 (2)
SRAM · ASD
MX 1ST STAGE

ROCKWELL · DET 19 (1)
MX · BMO
(GUIDANCE & CONTROL)

LOCKHEED · DET 21 (1)
C-5A WING MOD · ASD

AEROJET · DET 35 (1)
MX 2ND STAGE · BMO

FAIRCHILD · DET 44 (1)
A-10 · ASD

GENERAL DYNAMICS · DET 27 (1)
F-16 · ASD

MCDONNELL DOUGLAS · DET 47 (1)
ACES II · ASD

NORTHROP · DET 37 (1)
F-5E · ASD

ROCKWELL · DET 15 (1)
B-1B · ASD

TRW · DET 40 (1)
DSP · SD

CONTENT OF AFCMD CP3 DATA BASE

- AFPR CONTRACTOR ASSESSMENT
- AFPR CONTRACT ASSESSMENT
- MANUFACTURING OPERATIONS
- QUALITY ASSURANCE
- ENGINEERING & PROGRAM SUPPORT
- SUBCONTRACT MANAGEMENT
- CONTRACT ADMINISTRATION
- INDUSTRIAL MATERIEL MANAGEMENT
- FLIGHT OPERATIONS
- SAFETY / FIRE OPERATIONS

AFCMD DATA BASE RULES

MAY BE CONTINUED FOR PROGRAMS EXCEEDING EIGHT YEAR DURATION OR EIGHT YEARS AFTER COMPLETION

SUMMARIES UPDATED SEMI-ANNUALLY AND WHEN INCUMBENT AFPR TERMINATES POSITION

INCLUDES PROGRAM OFFICE INPUTS

AVAILABLE TO DOD CONTRACTING ACTIVITIES

DATA FOR OFFICIAL USE ONLY

AFCMD DATA BASE USAGE

- BUSINESS STRATEGY PANEL (AFSCR 70-7)
- ACQUISITION PLANNING (DAR 1-2100)
- STATEMENT OF WORK & APPENDICES
- CONTRACT CLAUSES
- SOURCE SELECTION (AFR 70-15)
- CONTRACT MANAGEMENT
- CONTRACT ADMINISTRATION

TAILORING OF TECHNICAL REQUIREMENTS IN GOVERNMENT CONTRACTS

Ronald J. Verba
USAF Space Division (AFSC)

Technical requirements are imposed in government contracts by citing Military Standards (MIL STDS) and Military Specifications (MIL SPECS) (Acquisition Management Systems) within the Statement of Work. These acquisition management systems cover technical and managerial disciplines which must be addressed in any program to ensure that the resulting item will meet the stringent user requirements of the military. These disciplines include logistics planning, maintainability, reliability, parts control, quality assurance, system safety, etc. Historically, programs that adhered to the rigid requirements imposed by these acquisition management systems have successfully met the military's need.

To fully appreciate the importance of these systems, one must understand their evolution. Prior to World War II, the technological base for military systems resided within the military services. Military laboratories designed and developed systems in direct response to the needs of that particular service. Upon completion and acceptance of the design, an arsenal, depot, or shipyard produced the system. With very few exceptions, such as aircraft, commercial industry had little or no involvement with the development or production of military systems. This method was very acceptable to the military because the military had direct control over schedule and resources as well as the methodologies employed in the life cycle of a system. Our entry into World War II demonstrated that while this procedure had the ability to serve the military needs during peace-time, it did not have the capability to expand to meet the needs of full mobilization. This fact, coupled with the rapid technological advances experienced in post-war years, caused the military to lessen its involvement in the internal development of military systems and become dependent upon the technological and industrial resources available within commercial enterprises. The military's reluctance to relinquish control of the development and production of military systems coupled with industry's lack of understanding of the customer requirements caused problems in the early acquisition programs. Military regulations and directives are not enforceable on contractors and contractors had difficulty responding to customer requirements because each service had service peculiar requirements and, in most cases, these requirements were not compatible. At industry's insistance, the Defense Department with industry involvement, developed documents (MIL STDS and MIL SPECS) that would be used by each service for imposing the requirements on contract. This limited the different methodologies that a contractor had to employ in order to meet customer requirements. Initially, these documents were invoked in total. This became expensive and imposed unnecessary requirements. In order to preclude the imposition of unnecessary requirements and to reduce costs, the government directed that the documents be tailored to the particular acquisition so that only the minimum essential requirements were invoked.

These same principles are prevalent in today's acquisition environment. Statutes, directives, regulations, policies and practices, mandate the imposition of acquisition management systems to ensure that technical disciplines are recognized and considered in the development and production of military systems. Department of Defense Directive 5000.19L requires that technical

requirements be imposed in acquisition contracts through the citing of approved acquisition management systems. The Defense Acquisition Regulation (DAR) 1-1201, requires that these acquisition management systems be tailored to include only those requirements necessary to satisfy the minimum essential needs of the acquisition. Recent changes to DAR 1-1201 permit the tailoring "in" of requirements as well as the tailoring "out" (exclusion) of requirements. This enables the acquisition manager to have the latitude when tailoring acquisition management systems to keep pace with the rapidly changing technological base being experienced in today's aerospace industry. In addition to the changes in DAR 1-1201, DAR 4-105 was changed to reinforce the government policy of specifying "what" has to be accomplished rather than "how to do it". These two changes, coupled with the current environment within the aerospace industry, demand that the industrial and military acquisition communities make a concerted effort in streamlining acquisition requirements. This can be accomplished by earlier and more productive dialogue between the potential contractors and the customer in determining the minimum essential requirements for a particular acquisition objective.

Tailoring of military technical requirements must consider the nature of the item being acquired, the environment in which the item will be used, the acquisition phase and the technological risks associated with the development of the item. Unrealistic schedules, extremely high risks, and excessive or inappropriate requirements must be identified prior to contract award. This should be accomplished early in the acquisition cycle. In the cases where Draft Requests for Proposals (DRFP's) are issued, potential offerors should submit constructive comments and alternate proposals concerning the tailoring and imposing of technical requirements. It should be noted that while alternate proposals are desired, these proposals must be addressed to existing and approved Military Standards and Specifications rather than to individual corporate policies, procedures or practices. Proposals for other than approved standards or specifications cannot be considered. In those cases where DRFP's are not issued, dialogue between the acquiring organization and the potential offerors should be established earlier in the acquisition cycle. The proper tailoring of technical requirements can reduce costs, therefore, in today's competitive and austere defense acquisition environment, judicious and prudent application of technical requirements may be the key to award or cancellation of a program. However, it should be stated, that caution should be exercised when tailoring technical requirements to reduce costs. The failure to address technical requirements in the appropriate acquisition phase, while providing an immediate contract cost savings, could prove far more costly over the life cycle of the item.

Recent recommendations resulting from the U.S. Air Force Systems Command "Project Cost" will help to enhance the acquisition process. Three recommendations are of immediate benefit to the application of technical requirements. They are:

1. Encouraging the use of complete DRFP's.
2. Certifying contractors methods and ability to satisfy technical requirements applied through MIL STDS & SPECS thus eliminating the need of detailed plans to be submitted with proposals.
3. Clarifying in each contract, the extent of compliance required by the tailoring of MIL STDS & MIL SPECS (tiering).

In order to maximize and realize the benefit to be derived from these initiatives, industry must:

1. Be responsive to the DRFP and negotiation of technical requirements.

2. Establish internal policies and practices that will be in accordance with and satisfy customer requirements.

3. Make maximum use of the MIL SPECS and MIL STDS in lieu of attempting to substitute individual corporate plans, procedures, or practices.

Mission Assurance

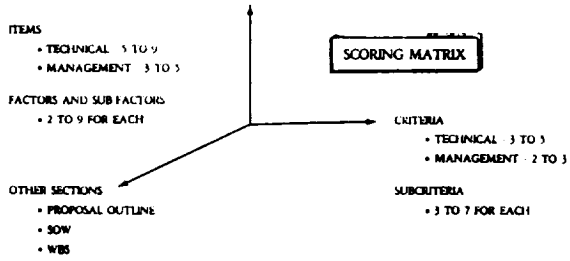
1983 Acquisition Workshop

June 7 - 9, 1983

James A. Horn
Boeing Aerospace Company

3-DIMENSIONS ARE MORE COMPLEX THAN 2

2 to 5 scored areas (Technical, management, production, past performance, cost, etc)



AGENDA

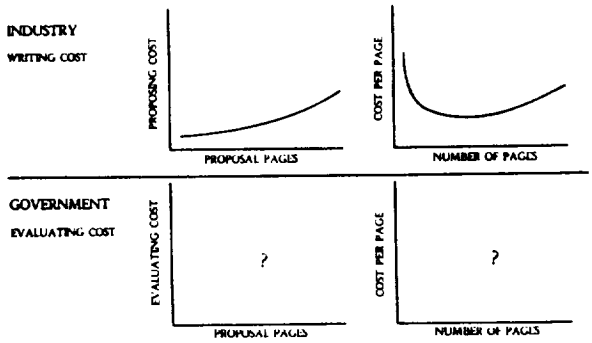
Introduction

Assumptions, Truisms and Objectives

Recommendations

Summary

SHORTER PROPOSALS COST LESS



ASSUMPTIONS, TRUISMS AND OBJECTIVES

Fairly award contracts at least cost

Better proposals result in better programs

Better RFP's result in better proposals

3-dimensions are more complex than 2

Exchange of information is helpful

Perspectives are different

Shorter proposals cost less

One page can only answer so many questions

Feedback is essential

INTEGRATE THE RFP

Integrate Sections L, M, SOW & CDRL's

Do a response matrix

Provide document outline

RECOMMENDATIONS

Improve communications

- Fog index, english vs contractual, keep it simple
- Reduce 3-dimension to 2-dimension
- Use complete draft RFP's
- Invite proposers to attend murder board
- Apply proposal principles to RFP's

FEEDBACK

At loss de-briefing

"Tell It Like It Is"

REDUCE SIZE OF PROPOSAL

Limit to deciding elements

Consider "stipulation" approach

Look at answers/page index

Reduce items/factors/criteria used in selection process

Always specify page count

SUMMARY

We have not succeeded in answering all of our questions. In fact, we have not completely answered any of them. The answers we have, however, do serve to raise a whole new set of questions about problems we had not thought of previously. In some ways, we are as confused as we used to be, but we believe that our confusion is now on a higher plane and about more important matters.

Anonymous

MISSION ASSURANCE CONFERENCE

ACQUISITION WORKSHOP

PROCEDURES TO IMPROVE THE ACQUISITION OF NEW SYSTEMS

PROPOSAL PREPARATION INSTRUCTIONS

ARLYN G. NELSON
ASSISTANT DIRECTOR, ACQUISITION MANAGEMENT

SPACE SYSTEMS DIVISION
LOCKHEED MISSILES & SPACE COMPANY, INC

Potential Impacts

MISSION ASSURANCE

- Seriously Impair Efforts to Maintain an Independent and Broad Competitive National Industrial Base
- Tendency to Dictate the Specific Kinds of Efforts Individual Contractors Should Pursue: Reducing Contractors' Flexibility
- Must Reduce B&P Cost to Maintain Current Level of Competition

Lockheed Missiles & Space Company, Inc.

Agenda

MISSION ASSURANCE

PROCEDURES TO IMPROVE THE ACQUISITION OF NEW SYSTEMS

- Proposal Preparation Instructions
  1. Introduction
  2. Use of DRFPs
  3. Proposal Outline
  4. Format Specifications
  5. Page Limitations

Lockheed Missiles & Space Company, Inc.

Use of DRFPs

MISSION ASSURANCE

PROPOSAL PREPARATION INSTRUCTIONS

ISSUE:

- **Are Seeing Them Now on Most Major Proposals**
- **Shortfall—PPIs are Frequently Incomplete**
  - **Tend to Prepare Draft Proposal to PPI That Will Change Drastically**

Lockheed Missiles & Space Company, Inc.

Congressional Concern Over IR&D And B&P Spending

MISSION ASSURANCE

- FY 83 Continuing Resolution and House Appropriations Bill Called for
  - $386M Reduction in Procurement Accounts for IR&D and B&P
  - Separate Line Item Budgeting for IR&D and B&P Beginning in FY 84
- USD R&E Memorandum, 23 Feb 1983, to Service Secretaries Directed Actions for FY 83 IR&D and B&P Funding

FY 84 Appropriations Bill Directs USD R&E to Establish Within His Office a Position Vested with the Responsibility to Monitor the Negotiation of IR&D and B&P Ceilings

Example: DRFP

MISSION ASSURANCE

USE OF DRFPs

Lockheed Missiles & Space Company, Inc.

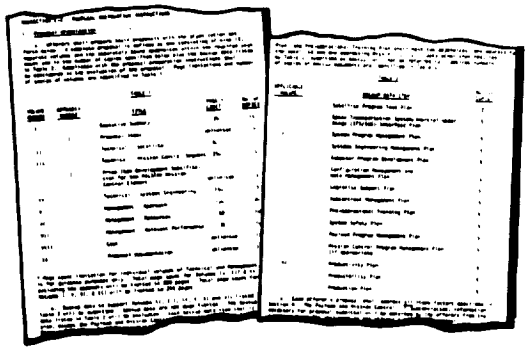
VOLUME NUMBER	APPENDIX	TITLE	(MAX) PAGE COUNT*	NO. OF COPIES
I	I	Executive Summary Proposal Prep. Instr Traceability Matrix	75 Unlimited	
II		Mgmt Prop: Resources	75	
III		Mgmt Prop: Relevant Performance	50	
IV		Mgmt Prop: Approach	200	
V		Technical: Hardware Design/ Integration/ Interfaces	150	
VI		Technical: Computer Programs	150	
VII		Technical: Test	50	
VIII		Cost Proposal	Unlimited	
IX		Contract Forms	Unlimited	
		Other Submittals: Plans	Unlimited	

*Page count limitation by individual volume number is for guidance purposes only. Total page count for Volumes I through VII, excluding Appendix I to Volume I will be limited to 700 pages.

MISSION ASSURANCE

Example: Final RFP

USE OF DRFPs



Recommendation

USE OF DRFPs

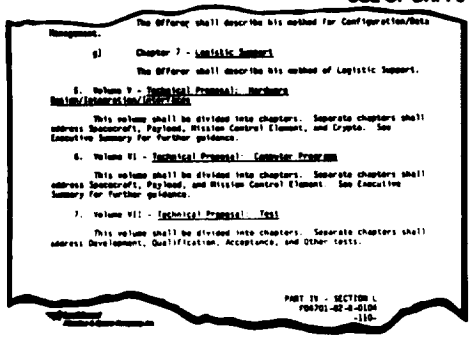
MISSION ASSURANCE

- Put as Much Time in Developing PPI as SOW and Specifications
- Feed Back Disposition of Comments as Soon as Possible

MISSION ASSURANCE

Example: DRFP

USE OF DRFPs



MISSION ASSURANCE

[redacted]CTIONS

- [redacted] Usually Provides Outline and Discussion Items
- [redacted] Very Difficult to Correlate SOW/WBS Paragraphs With PPI Outline
  - Force-Fit SOW Into Outline
  - Address SOW in Multiple Places Within Proposal
- Normally Outline
  - Study Proposal to SOW
  - Hardware Proposal to WBS

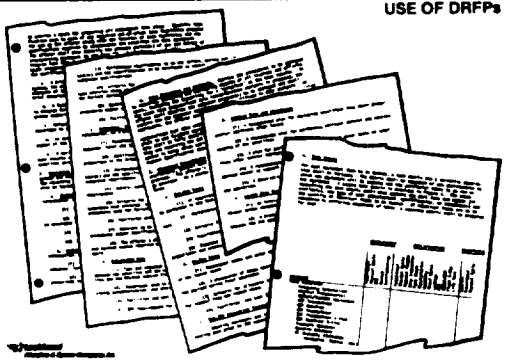
ISSUE

- Particularly Difficult to Correlate PPI to SOW and WBS on Service Type Contract—Such as System Integration, Sys. Engrg, etc.

MISSION ASSURANCE

Example: Final RFP

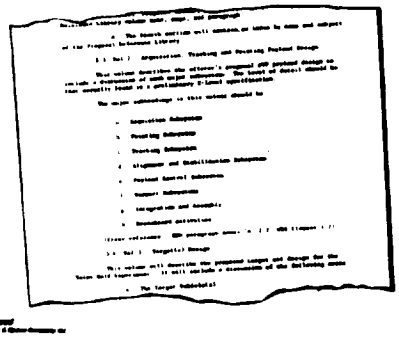
USE OF DRFPs



MISSION ASSURANCE

Example: Method 1 TALON GOLD

PROPOSAL OUTLINE



MISSION ASSURANCE

Example: Method 2

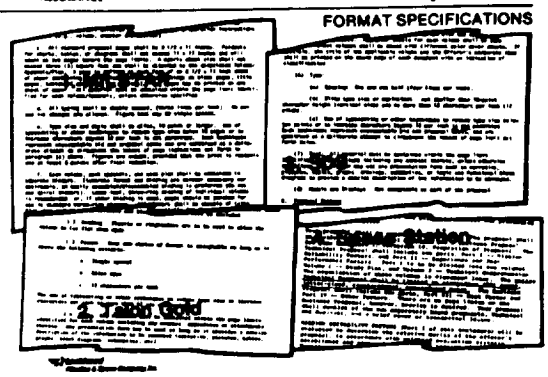
PROPOSAL OUTLINE

VOL	CHAPTER	SECTION	SOW	WBS
11	3	2	3.3.4	1.1.4

MISSION ASSURANCE

Example: RFPs

FORMAT SPECIFICATIONS



Recommendations

MISSION ASSURANCE

PROPOSAL OUTLINE

- Identify Which SOW Paragraphs Should Be Discussed in Which PPI Paragraphs
- Better Integration of PPI, SOW, etc.
- Industry Participation in RFP Review/Murder Board

BENEFITS
- Simplify Proposal Preparation
- Make Evaluation Easier
  - Data Is Where It Is Expected To Be

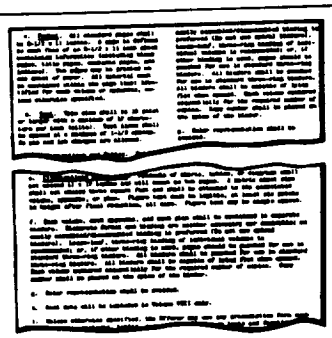
Example: Wordprocessing

MISSION ASSURANCE

FORMAT SPECIFICATIONS

2 - Col Single Spaced

1 - Col Single Spaced



Format Specifications

MISSION ASSURANCE

PROPOSAL PREPARATION INSTRUCTIONS

- Changes from One RFP to Another Cause Cost to Increase
  - Size of Type & Line Spacing
  - Image Area
- Current RFPs do not Allow Use of State-of-the-Art Labor Reducing Systems Such as Word Processing/Phototypesetting.

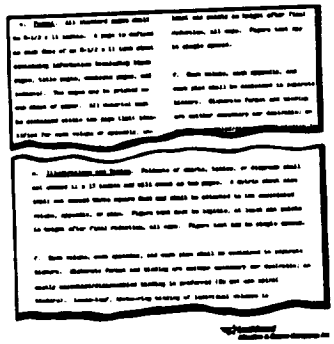
Example: Wordprocessing

MISSION ASSURANCE

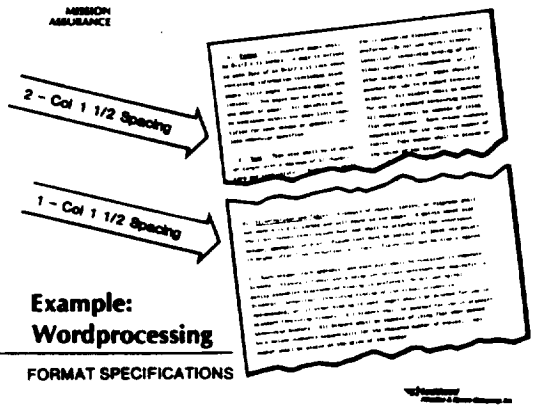
FORMAT SPECIFICATIONS

2 - Col Double Spaced

1 - Col Double Spaced



MISSION ASSURANCE



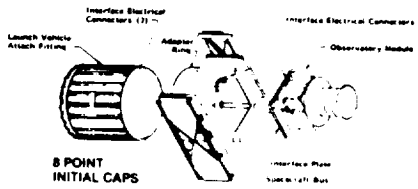
Example: Wordprocessing

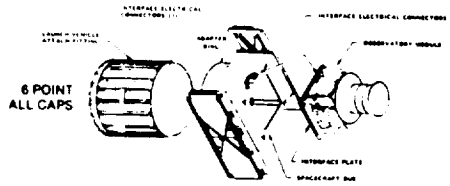
FORMAT SPECIFICATIONS

MISSION ASSURANCE

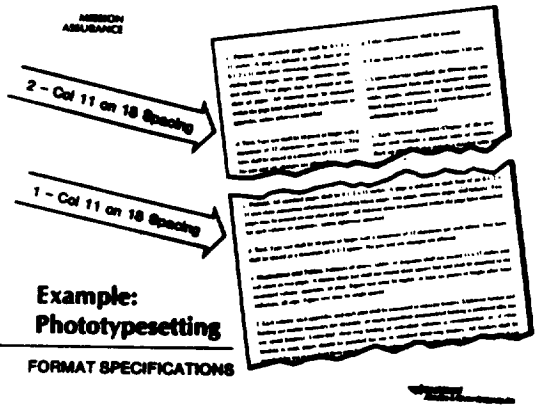
Example: Illustrations

FORMAT SPECIFICATIONS





MISSION ASSURANCE



Example: Phototypesetting

FORMAT SPECIFICATIONS

Recent Cost Driving Trends

MISSION ASSURANCE

FORMAT SPECIFICATIONS

- Different Color Covers for Each Volume
- Vugraphs of Executive Summaries
- Restriction to Printing on One Side of Page

- [redacted]ases Reading Speed by 20 Percent
  - Improves Retention
  - Improves Comprehension
- Improves Referability
  - Type Style Changes Make Important Items Easier to Locate

- Provides for Automatic
  - Hyphenation
  - Justification
  - Column and Page Breaking

Recommendations

MISSION ASSURANCE

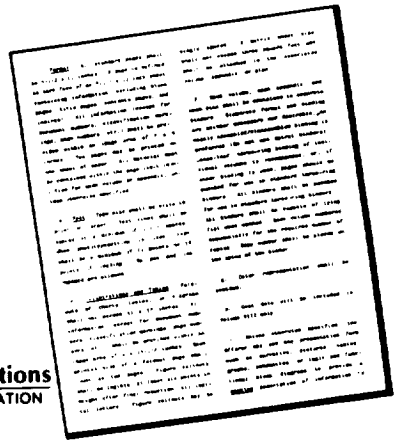
FORMAT SPECIFICATIONS

- **Standardize Type Size and Art Reduction Specifications**
- **Format**
  - Size of Page:
    - 8-1/2 × 11 Inches (Standard Page)
    - 11 × 17 Inches (Foldouts)
  - Type Style:
    - Word Processing, 10 Point "Elite" Minimum 1-1/2 Spaced
    - Phototypesetting, 11 on 18 Points
  - Illustrations:
    - 6 Points Minimum, All Caps, Single Spaced

MISSION ASSURANCE

Recommendations

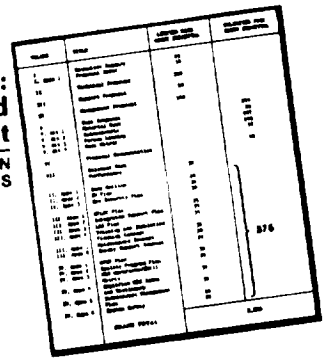
FORMAT SPECIFICATION



MISSION ASSURANCE

Example: Plans Limited Page Count

PROPOSAL PREPARATION INSTRUCTIONS



PREPARATION INSTRUCTIONS

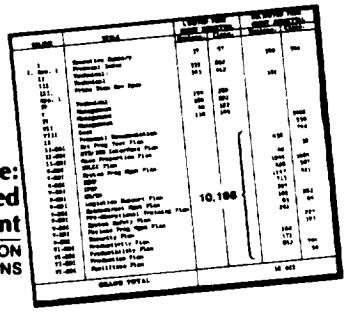
ISSUE

- Cost Directly Related to Number of Pages
- Biggest Driver is Program Plans/ Preliminary CDRLS.

MISSION ASSURANCE

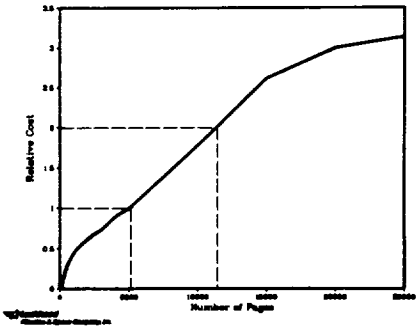
Example: Plans Unlimited Page Count

PROPOSAL PREPARATION INSTRUCTIONS



MISSION ASSURANCE **Cost Directly Related to Page Count**

PAGE LIMITATION



[redacted]endations

- Establish Page [redacted] Limits on Program Plans/CDRLs Required as Part of Proposal Submittal.

- Eliminate Redundancy of Information Required.

SUMMARY

To maintain the current level of competition for new systems with limited B&P and IR & D funding, we must develop ways to minimize procurement cost. This presentation has identified areas within the RFP Proposal Preparation Instructions (PPI) which tend to increase industry's cost in preparing proposals as well as Government's cost in evaluating proposals. The presentation also identifies areas within the PPI that are ambiguous and recommends methods to eliminate these ambiguities.

The key recommendations to reduce cost and eliminate PPI ambiguities are

1. Develop mature PPIs for Draft RFPs and notify industry as soon as possible of any changes to the PPI between release of DRFP and Final RFP.
2. Identify within proposal outline specified by PPI where each SOW and WBS item is to be discussed.
3. Standardize format and binding specifications (a recommended standard is included).
4. Establish page limitations for plans required with proposal submittal.

OVER APPLICATION AND MIS-APPLICATION OF MILITARY SPECIFICATIONS AND STANDARDS

John Morey, Hughes Quality Assurance Manager

In 1974, Bill Clemonts, then Secretary of Defense, said, "The problem with Specifications and Standards is not in the documents themselves, but rather in the way they are mis-applied". From that beginning, nearly a decade has passed during which all manner of well-intended admonitions, investigations and attempts to grapple with this task have not solved the problem.

Tailoring became a buzz word, but nobody had a solution for the big problem of 60,000 documents in the DODISS. There have been instances when a single specification was "tailored" to accomodate a specific phase of a specific procurement. There is MIL-Handbook-248 that reminds a Procuring Office that to do a reasonably through job of tailoring requirements for a routine procurement may involve reading 21,000 documents and determining which parts of those documents should be made applicable to which phase of the acquisition cycle. The admonition is completely sound; the practical application is a physical impossibility.

This paper treats with a proposal to enter the DODISS (Department of Defense Index of Specifications and Standards), identify those Specifications and Standards, and their references, that lend themselves to partial and incremental application, make prior general determinations of that application based upon the phase of acquisition, the complexity of the product and known (or anticipated) production rate and provide pre-tailored recommendations as an integral part of the document.

Some basic assumptions are made as follows:

1) The problem of over-application is essentially no different today than in 1974.

2) The task of tailoring as defined by MIL-Handbook-248 is impractical if not impossible. No procurement office has either time or expertise to read and determine such requirements.

3) The initiating agency for the Specification or Standard is best able to prepare a recommendation for applying the document to the phase of acquisition, the product complexity and the anticipated production rate.

4) The availability of such "pre-tailored" information as an integral part of the document will:

   a) Allow the generation of far more specific technical and administrative requirements in RFQ's.

   b) Reduce (or eliminate) the clamor for respondents to RFQ's to propose "innovative approaches" that are intended to do away with unnecessary requirements. The spectre of "non-responsiveness" and "consideration" generally over-shadow these well-intended clamors.

c) Allow contractors to completely put aside non-productive requirements that otherwise must be dealt with.

d) Recognize and deal effectively with the acquisition phase-sensitive degree of applicability as displayed in the "Block 2" view graph.

Why doesn't "Tailoring" as currently expoused, work? Part of the answer is in the foregoing paragraphs, but a vital part of that answer is that "Tailoring" must be done each time, for each procurement, for each phase of acquisition by each SPO. It must then be responded to in kind by each respondent to the RFQ and then of course, implemented in its program and phase-unique character at the selected contract facility. That simply isn't practical or economic; as a matter of fact, it is a terrible waste.

There is a way to fix the problem and at the same time provide a workable technique to identify those tasks that are particularily applicable to a procurement and, equally important, eliminate those requirements that are not.

First let's briefly describe what Hughes calls Partitioning. Partitioning provides an applicability matirx as an integral part of the Specification or Standard. Characteristics of the Specification or Standard, by paragraph or by sentence, are identified in the matrix as "essential", "deferred" or "continuing" in relation to the phase of acquisition, the complexity of the product and the anticipated production rate. A three-dimensional matrix graphically displays the treatment of these vital characteristics of an acquisition. It should be mentioned again that it is recommended that the agency responsible for preparation of the Specification or Standard be responsible for preparing the initial application matrix. Other agencies, at their option, may include modifications or even unique application matricies for their own procurements. There is no perceivable problem with having a Specification or Standard that includes a matrix of the preferences for Army procurement, Navy procurement, Air Force procurement, NASA procurement, etc. Much to the contrary, without such application information we remain at the mercy of the "status quo' which as I'll point out next, we must not continue to tolerate.

Let's review an example and deal with a well known, much used document. Specification MIL-E-5400 is the Genral Specification for Airborne Electronics; it is an Air Force preparatory responsibility. MIL-E-16400 is a Navy counterpart. Both these documents are best described as irreplaceable in terms of the assembly of intelligence they contain. If we didn't have them, we'd have to invent them. But they are awsome in scope, detail and particularily in the referencing chain. Applying them to a program in entirety can be an invitation to disaster. At the first level of indenture, they contain hundreds of referenced Specifications and Standards; at the second level there are thousands. For example,

they both reference MIL-STD-454 for Soldering. MIL-STD-454 has some 90+ specific requirements and as an illustration, Requirement 17 alone in turn references 86 other Specifications and Standards. By the way, there happen to be some 320+ Soldering Specifications and Standards in the DODISS. They may appear anywhere in the 2nd, 3rd, 4th or further tier of reference. So how big is this problem?

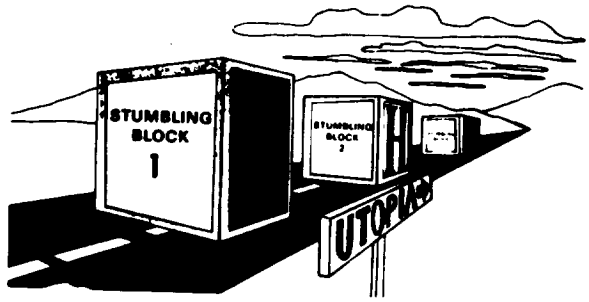
There are some 60,000 Specifications and Standards in the DODISS. The example above repeats itself over and over throughout the entire spectrum of documents. There is, for example, no easy way to enter the DODISS and extract the 320 Solder Specifications and Standards mentioned earlier; there isn't even a "rather difficult" way. There is only a plodding document-by-document method. Recognizing this, a frontal assualt in the DODISS was eliminated. In lieu, we pulled a random sample of documents and evaluated them. With reasonable confidence it can be stated that only about 6400, or roughly 10%, of the documents in the DODISS lend themselves to partial or incremental application. The balances are quite clearly "either-or-else requirements; one plate apart or does not plate it, etc. Of the 6400 documents, 5400 are "Tailorable" for only about 25% of their content, the balance of 500 are "Tailorable" for 25% to 100% of their content. So while the task to develop application matrices is formidable, it can be done.

As a result of an effort undertaken at Airlie House in the late 1970's matrices have been developed and applied to MIL-STD's 470, 480, 882, 785 and 965. By my calculations, that leaves 6395 left to do and I suggest that we get on with it.

SEARCHING FOR A SOLUTION

HUGHES

THE OVER APPLICATION AND MISAPPLICATION OF MILITARY SPECIFICATIONS & STANDARDS



SQUARE ONE

HUGHES

MEMO

SEPT. 1974

TO ALL CONCERNED:

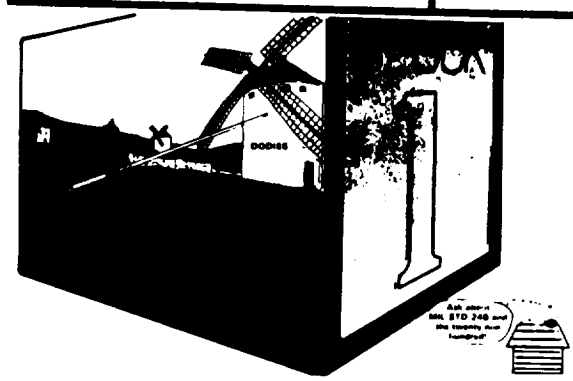
THE PROBLEM WITH SPECIFICATIONS IS NOT WHAT'S IN THEM, BUT RATHER HOW THEY ARE MISAPPLIED.

TAILOR REQUIREMENTS
OPTIMIZE EFFORTS.

DEPUTY SECRETARY OF DEFENSE CLEMENTS

A LOOK INSIDE BLOCK 1

HUGHES

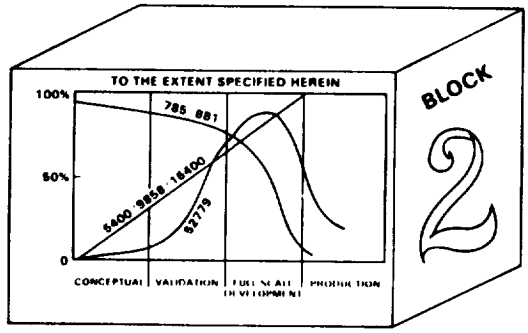


HUGHES

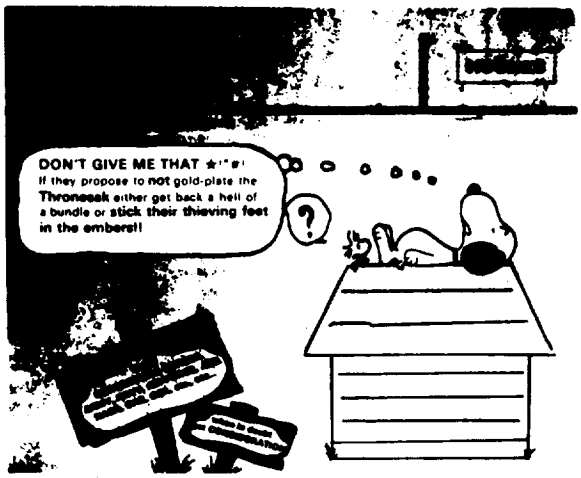
In this business it seems that getting what we thought we asked for is somehow slaved to asking for what we want to get . . . perhaps not much more complex than that, but we do manage to confuse it wonderfully.

A MIND-BLOWER IN BLOCK 2

HUGHES



PAGE 38 INTENTIONALLY BLANK



PROBLEMS WITH TAILORING

HUGHES

- MUST BE DONE
  - BY SPO
  - FOR EACH PHASE
- EXPERTISE LACKING AT SPO'S
- PRECEDENCE VS IMPORTANCE
- THE FAN OUT PROBLEM

SOLUTION

HUGHES

TAILOR SPECIFICATIONS AND STANDARDS

DOD-HDBK-248A

HUGHES

4.7.1.3 IMPACT OF REFERENCED SPECIFICATIONS AND STANDARDS. ALL REFERENCED SPECIFICATIONS AND STANDARDS RELATED TO THE SYSTEMS REQUIREMENTS SHOULD BE REVIEWED AND TAILORED TO THE APPLICATION, BECAUSE OF THE POTENTIAL HIGH-COST DRIVING IMPACT ON THE SYSTEMS ACQUISITION PROGRAM. EACH STANDARDIZATION DOCUMENT REFERENCED IN THE SYSTEM SPECIFICATION OR STATEMENT OF WORK (SOW) IN TURN REFERENCES 20 OR 30 OTHER STANDARDIZATION DOCUMENTS. THIS MEANS BY REFERENCING FIFTY OF THESE DOCUMENTS IN AN ACQUISITION, IT WOULD TYPICALLY REQUIRE COMPLIANCE WITH 1000 SECOND TIER DOCUMENTS, PLUS 20,000 THIRD TIER DOCUMENTS, ETC. PROPER TAILORING CANNOT BE ACCOMPLISHED WITHOUT READING THE REFERENCED DOCUMENTS, EVALUATING ALL THE REQUIREMENTS, INCLUDING THE SPECIFIED REQUIREMENTS OF THE DOCUMENTS REFERENCED THEREIN, AND DETERMINING WHICH REQUIREMENTS ARE APPLICABLE AND NECESSARY.

TAILORING CONCEPT

HUGHES

- REMOVE NONESSENTIAL REQUIREMENTS FROM:
  - RFP'S
  - CONTRACTS

FOR EACH SPECIFIC PROCUREMENT

AN EFFECTIVE ALTERNATIVE

HUGHES

- INSTEAD OF

  REQUIRING SPO TO TAILOR AND RETAILOR – THE SAME SPECIFICATIONS OVER AND OVER

- WHY NOT

  PARTITION THE SPECIFICATIONS AND STANDARDS TO FIT SPO NEEDS

WHAT IS PARTITIONING?

HUGHES

- ITS AN "APPLICABILITY MATRIX" TABLE (OR SET OF TABLES)
  - ADDED PERMANENTLY TO SELECTED SPECIFICATIONS OR STANDARDS
  - PROVIDING ADVANCE SELECTION OF REQUIREMENTS
  - CUSTOM FIT TO THE NEEDS OF EACH BUYING ACTIVITY USING THE DOCUMENT
  - FOR ANY COMMODITY TYPE OR ACQUISITION PHASE

ADVANTAGES OF TAILORING BY PARTITIONING

DOES NOT REQUIRE REWRITE

ALLOWS TIMELY SOLUTION

SAVES MONEY

ONE TAILORING DOES JOB

ACCOMMODATES EACH MILITARY BRANCH

EASILY COMPUTERIZED

HOW DOES PARTITIONING AFFECT DOCUMENT'S CONTENT?

HUGHES

- IT ADDS
  - ONE OR MORE MATRICES
  - PLUS ANY APPLICATION INFORMATION DETERMINED NECESSARY BY THE BUYING ORGANIZATION
- WITHOUT DISTURBING EXISTING CONTENT

SAMPLE OF PROPOSED MATRIX

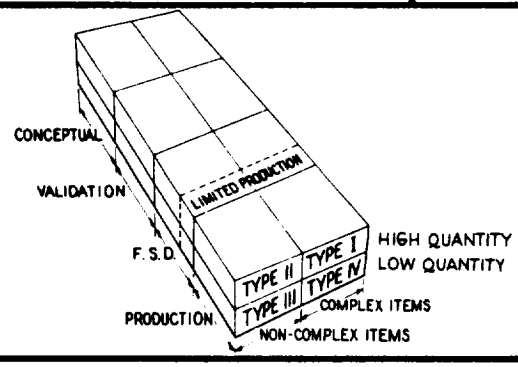
HUGHES

APPLICABILITY MATRIX

ACQUISITION PHASE	CONCEPTUAL				VALIDATION				FULL SCALE DEVELOPMENT								PRODUCTION			
HARDWARE MODEL (MIL-STD-280)	EXPLORATORY DEVELOPMENT				ADVANCED DEVELOPMENT				ENGINEERING DEVELOPMENT				PREPRODUCTION				PRODUCTION			
TYPE OF PROGRAM	I	II	III	IV	I	II	III	IV	I	II	III	IV	I	II	III	IV	I	II	III	IV
PARAGRAPH NUMBER	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
CASTINGS	-	-	-	-	-	-	-	-	-	-	-	-	D	D	-	-	E	E	-	-
1.1.1																				
NON-STD PARTS	-	-	-	-	D	-	-	-	D	-	-	-	D	-	-	-	E	-	-	-
1.1.1																				
P.C. BOARDS	-	-	-	-	-	-	-	-	D	D	-	-	D	D	-	-	E	E	-	-
1.1.11.1																				
MECHANIZED PC PROD	E	E	-	-	C	C	-	-	C	C	-	-	C	C	-	-	C	C	-	-
1.1.1.1																				
WIRE CODING	-	-	-	-	-	-	-	-	-	-	-	-	D	D	D	D	E	E	E	E
1.1.14.3																				
LABELING	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	E	E	E	E
1.1.14.2																				
HELD/SOLDER HORCTS	-	-	-	-	E	E	E	E	C	C*	C	C	C	C	C	C	C	C	C	C
1.1.14																				

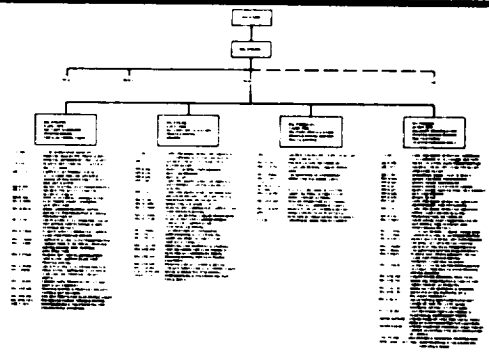
PARTITIONING: ACQUISITION, A COMPLEX PROCESS

HUGHES



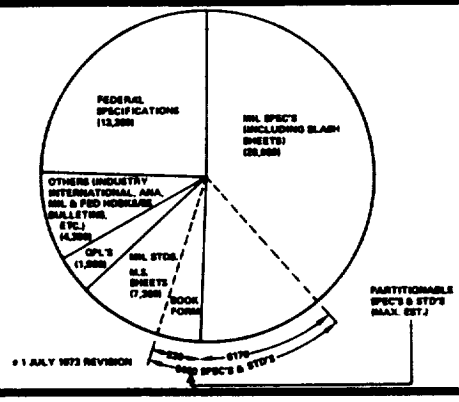
THE SPEC FANOUT PROBLEM

HUGHES



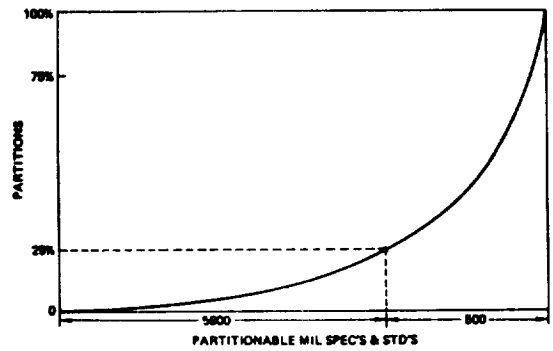
DoD ISS* BREAKDOWN
(54,900 LISTED DOCUMENTS)

HUGHES



ESTIMATED MIL SPEC AND STANDARD PARTITION DISTRIBUTION

HUGHES



PROGRESS SINCE 1974

HUGHES

SPECS WITH MATRIX ADDED

- MIL-STD-470 MAINTAINABILITY
- MIL-STD-480 CONFIGURATION
- MIL-STD-882 SAFETY
- MIL-STD-785 RELIABILITY
- MIL-STD-965 PARTS CONTROL

SPECS WITH TAILORING INFORMATION INCLUDED IN TEXT

- MIL-STD-1521 TECHNICAL REVIEW
- MIL-E-6051 ELECTRO-MAGNETIC COMPATIBILITY
- MIL-H-46855 HUMAN ENGINEERING
- MIL-STD-35 AUTOMATED ENG. DOCUMENT PREPARATION

THE TROUBLE WITH QUALITY/INSPECTION

Jack Godsey
Director, Quality
Thiokol Corporation
Huntsville Division

There is quality trouble in River City--and in other small towns across America--and in the big cities--and in the giant cities, also. One trouble with quality is inspection.

We inspect too much

We inspect the wrong things

We inspect at the wrong time

What does Acquisition have to do with quality? I'm not sure just yet.

Well, why is Acquisition interested in quality?

We have heard that our military systems are deadlined a lot of the time because of quality.

We have heard that the U.S. auto industry tottered because of quality.

We have heard that quality and productivity go up and down, hand in hand.

We have heard that we lost much of our foreign market because of productivity.

We have heard that unemployment is the result of loss of markets.

This almost sounds like the medieval tale of the loss of a kingdom for want of a horseshoe nail.

The trouble is, the quality tale is probably true.

But things aren't all bad. We are the best at aircraft, space missions, fast foods, computers, designing robots, farming and movies -- and probably other things that do not come to mind just now. I have the feeling we are better at everything, if we set our minds to it. We are setting our minds to it; that is what this conference is about as well as activities all over the country for the same purpose.

Last fall the Air Force Systems Command initiated "Project Quality" to develop contractual incentives to improve quality. Project Quality is very acquisition oriented. A survey asked tough questions about techniques from the proposal phase through the administration-of-contracts phase. Such questions as:

Should there be an award fee incentive for quality?

Should there be a "warranty of producibility" clause with lead contractor liability?

Should there be a statement of work for quality?

Should quality be given more weight in source selection?

Should DCAS delegations contain specific instructions?

This workshop includes a status report on Project Quality.

Of course, Acquisition is involved in quality.

In quest of improved quality and productivity, some serious challenges are being made to traditional management philosophy:

Participative Management

Decisions by Consensus

Quality Circles

Those subjects are being addressed at universities, management training seminars, and probably at this conference. I will not talk about that today. I want to be more specific.

We need to narrow our sights; draw a bead on one subject--Inspection. Maybe we had better close down our sights a little more before we draw a bead--say, to three inspection problems.

We inspect too much! We inspect the wrong things! We inspect at the wrong time!

We have heard so much about the Japanese's successes that we don't want to hear any more. But, they did take two of our most knowledgeable quality spokesmen to heart--Dr. Edward Deming and Dr. J. M. Juran. The accounts of how their expertise was applied is now familiar. The same thing happened to our robots. We sent them to Japan.

In Ouchi's "Theory Z" he reports the Japanese's incredulous remarks upon viewing U.S. factories.

> Why do you Americans inspect so much?
>
> Don't you trust your own people?

Why do we inspect so much?

Because we find so much wrong. We find so much wrong because we depend on inspection. We put resources there that would be better put into doing the job. We don't know how to break the cycle. It is like the chicken and the egg.

Further, we are not sure how much we have to inspect by contract. Certain "objective evidence" of conformance is required. The inspection planners don't know how much that is--so, they go by past practice. Past practice gets pushed up when someone important is embarrassed by quality deficiencies and yells, "Why was that not inspected!" Past practice is pushed down when someone important yells, "We are spending more on inspecting than on making the item!"

Frequently, that is true. It can cost as much to inspect as to build.

We must generate enough inspection data to satisfy our customers' perception of what is required. An element of cost that is maybe 20% of the total is left undefined.

Why?

It has always been that way.

OK, how can Acquisition help? I don't know the total answer, but a giant step forward would be to cause the statement of work to be explicit about "objective evidence." Exactly what hard data is required to determine acceptance? I am quite aware that the data required may depend on circumstances.

Right now the amount of inspection is determined by the subjective "convince me the item meets the spec." We need to remove as much of that subjectivity as possible.

Point No. 2. We inspect the wrong things.

Sure we do. The inspection planner does the best he can, but only the designer really knows the relative importance of features, characteristics and dimensions.

Some designs have a designation alongside each callout--saying:

> This is extremely important
>
> This is moderately important
>
> This is important

It is usually called a classification of characteristics. The classifications are:

Critical

Major

Minor

The designer can use an alternative means of communicating the relative importance of characteristics. He can do it by putting tight limits on them.

Not only does that method do a poor job of communicating, it has all kinds of bad side effects.

Classification of characteristics allows the people making the item to plan tooling, controls, and methods in consonance with real needs. It also allows that inspection planner we feel so bad about to do his job better.

Sometimes there are tough choices. We can't afford to do everything.

Some technical data packages have a "classification of defects" which should not be confused with "classification of characteristics."

The "classification of defects" if used like a "classification of characteristics" would be OK. But, its intent is to guide the decision process in determining what to do with items already built, inspected, and found to be defective. That guidance usually hinders more than it helps, because (1) the kind of people involved in MRB decisions already know the importance of the defect, and (2) the significance is logically impacted by the degree of non-conformance which the classification of defects can't anticipate.

The Navy has a classification of characteristics document; WR-43A. It also deals with other related subjects. A requirement to generate a classification of characteristics as part of the design using WR-43A definitions and guidelines would lead to better quality, I think.

My final point--We inspect at the wrong time.

We inspect too late. If an item can be corrected at all, it probably can be done more easily right where the process is. Further, the reason for any defect can be found then and there better. There are exceptions--but, they only cloud the issue; they don't change the conclusion. Back to my argument. The longer you wait, the worse it gets. After items have been delivered, and the investment is large and the schedule is pressing, differences of opinion about inspection results may lead to loud discussions about who knows how to inspect--and, to other stressful situations. The longer we wait, the worse it gets. When we wait until the user has a problem, that is as bad as it can get!

Then, why do we wait?

There are some reasons--it is more efficient to inspect in stages and in groups. For example, several steps in a sequence can be verified at once. That means the inspector can rove. Also, if a large number of units is being made, you can check them faster in a group--or, you may be able to statistically sample.

Another reason we wait is just to do the inspection by our own people in our own shop.

Another reason is that the customer (particularly the government customer) may insist on inspecting in your shop --not the vendors. If he does insist, you may as well inspect in your shop also because you are obliged to provide inspection tools, facilities and

assistance. I am not attempting to tell all sides, so I will just say that I am not trying to be critical of the government QAR. But, the QAR should have instructions that charge him with acceptance at the earliest point.

Well, should we inspect as early as possible--or wait? My argument is for early as possible. The inspector may not be able to inspect as much, but I say he will do more to insure a quality item.

Here is what I am saying we should do. Inspect less--select inspection points based on importance of the feature--inspect as soon as possible.

What can Acquisition people do?

Perhaps you can help:

(1) Include contract language that defines the amount of inspection data required for acceptance.

(2) Include a Classification of Characteristics as part of the technical documentation.

(3) Promote inspection and acceptance at the earliest possible point in the process.

My suggestions are very simple. Maybe you will have some better ideas.

WHAT BENEFIT FACILITIES?
A CONTRACTING OFFICER'S DECISION

Jack Nelson, AFSD

With the advent of "Profit 76" payment was provided to the contractor as considerations for capital investment risk associated with the facilities employed in performance of a contract. Defense procurement circular (DPC) 76-3 dated 1 September 1976 directed this consideration to be 6 to 10 percent of the net book value of the facilities employed. This was revised to 16 to 20 percent by Defense Acquisition Circular (DAC) 76-23 dated 26 February 1980. This 16 to 20 percent factor is based on consideration of whether the facilities are general or special purpose, the age of the equipment or facilities, the undepreciated value of the facilities and the relationship of that undepreciated value to the contract effort. In making this value determination general purpose equipment is given less weight than special purpose equipment which contributed to the efficient production of the instant contract items. Also considered are projected new equipment and facilities investments. These new investments must be part of an approved investment plan, have achievable benefit to the government, primarily defense business, and reduce the life cycle cost of the defense products produced.

DPC 76-3 also added DAR 3-1300 "Facilities Capital Employed" which is the DOD implementation of Cost Accounting Standard (CAS) 414 "Cost of Money as an Element of Cost of Facilities Capital". This cost accounting standard establishes criteria for and the approved method of measurement and allocation of facilities cost of money to overhead pools. DAR 3-1300.1 policy states that it is the policy of the Department of Defense to recognize facilities capital employed as an element in establishing price of certain negotiated defense contracts. It further states and I quote,"The inclusion of this recognition is intended to reward contractor investments, motivate increased productivity and reduce cost through the use of modern manufacturing technology, and to generate other efficiencies in the performance of defense contracts."

These changes in profit policy also revised the relative weight mix of the major profit determinants. The contractor's input to total performance which was 65 percent of the total was revised to 50 percent of the total by applying a 30 percent adjustment factor to the profit/fee calculated for the above the line contractor effort. Cost risk was adjusted to 40 percent of the total from the previous weight of 30 percent and investment to 10 percent from the previous total for all other factors of 5 percent.

Three elements are required to develop cost of money factors (CMF). These factors are used to allocate the cost of money to a specific contract:

1) Business unit facilities capital data

2) Overhead allocation base data

3) And the interest rate promulgated by the Secretary of the Treasury pursuant to Public Law 92-41.

Variations in these three elements therefore effect an impact on the development of the cost of money factors applicable to each cost accounting period. Two of these, business unit facilities capital cost and the overhead allocation base are controllable or dependent on the contractor's management decisions - the third - the Secretary of Treasury interest rate is the result of the government economic outlook. The cost of money factors are developed using form CASB-CMF which is prepared by the contractor and analyzed by the auditor and contracting officer for compliance with Cost Accounting Standard 414. Estimates of the capital cost of money and capital employed are made using the cost of money factors and DD Form 1861 "Contract Facilities and Cost of Money". This form provides for the listing of overhead pools in the same structure as the contractor's proposal and Form CASB-CMF by year. Each allocation base is multiplied by its cost of money factor to calculate the facilities capital cost of money estimated to be incurred annually for each allocation base. Capital employed can be determined by dividing the contract cost of money by the Treasury rate. These end results are used in the determination of the profit/fee objective using DD Form 1547 "Weighted Guidelines Profit/Fee Objective". Defense Acquisition Circular 76-23 directed the use of three separate weighted guidelines which employed different weighting factors for different types of contract effort (manufacturing, research and development and services contracts). In determining whether a contract should be classified as manufacturing, research and development or services primary reliance should be placed on the nature of the work to be performed. Many contracts for research and development and for services will require a significant amount of facilities for efficient contract performance. DAR 3-808 states that when in the judgement of the contracting officer a significant amount of facilities is required for efficient contract performance manufacturing weighted guidelines shall be used. In making this determination the contracting officer is required to assess the facilities needed including contractor owned and leased and Government owned facilities. When there is a relatively small amount of facilities capital cost of money allocated to a contract because some of the facilities are provided through operating leases and by the government this does not always mean an insignificant amount of facilities are required.

When a method other than the manufacturing weighted guidelines is used for profit/fee determination DAR directs the profit objectives shall be reduced by the amount of the facilities capital cost of money allowed. DOD policy is that it is neither desirable or equitable to apply an investment oriented profit policy to contracting situations which require relatively few facilities and there is no significant productivity gain to be realized by increasing facilities investment. Therefore, the facilities cost of money is considered only when the manufacturing weighted guidelines is used.

We should now examine the impact the contracting officer's determination of WGL application has on the profit/fee objective of a contract by comparing a manufacturing determination to an R&D.

The above the line weighting (Part I "Contractor Effort") on the DD Form 1547 for manufacturing weighted guidelines and for research and development weighted guidelines are identical except for the 30 percent adjustment factor which does not apply to research and development weighted guidelines. As stated before application of the 30 percent factor in the manufacturing weighted guidelines shifts the relative

weighting of the major profit determinats to 50 percent for contractor's efforts, 40 percent for cost risk and 10 percent for investment. In the case of research and development weighted guidelines the directed non-use of the adjustment factor coupled with reduced cost risk ranges shifts the relative weighting of the profit/fee mix to one based predominately on the contractor's effort.

If we make a comparison of the resultant profit objectives for a manufacturing contract versus a research and development contract where the subtotal profit/fee for the contractor's effort is 9.5 percent after the 30 percent manufacturing adjustment is applied, the profit factor is 6.65 percent for the manufacturing method while it remains at 9.5 percent for the research and development. However, let us assume in this instance the ratio of facilities cost of money to total cost excluding facilities cost of money is 3 percent; according to DAR requirements, the research and development profit/fee factor is reduced by 3 percent to calculate the total objective. For this hypothetical contract the cost risk is determined to be 3 percent based on the contract-type and other determination factors. The research and development contract receives no credit for capital employed. Assuming the current Treasury rate is 12 percent and an 18 percent facilities factor is applied, the capital employed profit/fee factor for the manufacturing method is 4.5 percent. The basic profit/fee objectives are 14.15 percent for the manufacturing mehtod and 12.5 percent for the research and development method. DAR requires an offset for allowed facilities capital cost of money applicable to research and development and services weighted guidelines and the profit/fee factor for the research and development method is reduced by 3 percent to a total profit/fee objective of 9.5 percent vice the 14.15 percent profit/fee objective calculated using the manufacturing method. This amounts to a difference of 4.65 percent or $46,500 of profit/fee opportunity for each million dollars of estimated cost.

SEE ATTACHED COMPARISON TABLE

An Air Force Systems Command (DCS/PMM) Policy Letter dated 24 March 1982 on the subject of Facility Capital Cost of Money which provides guidance with regard to the method to be used to select the appropriate weighted guidelines factors states:" - The principal determinant in deciding which column is most appropriate is not the product being acquired, but rather the amount of inhouse capital facilities needed to perform the contract. If extensive capital facilities are required, the manufacturing column should be used. Incidentally, Government-owned and leased facilities as well as contractor-owned facilities should be taken into account." It further states "it is recognized that adherence to this guidance will cause the manufacturing guidelines to be used a majority of the time. This usage pattern was anticipated when the R&D and services guidelines were created and is in keeping with the DOD policy of encouraging contractor investment in productivity enhancing capital equipment."

To this point I have stressed the regulations, policies and guidance which apply to the contracting officers determination andthe important impact his determination has on the contractor's profit opportunity. To make this determination the contracting officer is required to assess the facilities needed and whether they contribute significantly to contract performance. In most instances the contracting

officer is faced with a considerable lack of information upon which to base judgement. Proposal preparation instructions or the DD 633 instructions do not require the submittal of support information for the cost of money factors nor is there a requirement for the contractor to submit details on facilities usage. The amount of proposal facilities capital cost of money cannot be used for measurement purposes because a considerable amount of the facilities to be used may be Government owned or leased and not appear in or have only a minor impact on the cost amount.

Additionally, facilities capital investment includes buildings as well as production equipment and numerous other capital investment items. Some questions that must be answered are: Do new modernized buildings and furniture contribute to efficient contract performance? Do computer intensive management information systems, Do employee health and welfare facilities and equipment, Do compatible well maintained and furnished offices contribute? Is the contribution of these capital items significant?

Cost of money distribution is principally allocated to direct labor bases and research and development and service contracts are labor intensive contracts, the disallowance of the capital employed profit on these type efforts as directed by DAC 76-23 decreases the amount of recognized capital employed. AFSC "Profit Study 82" points out that the allocated versus the recognized capital employed in profit objectives on AFSC contracts in 1981 was 60 percent. Under DAC 76-23 the contractor's recovery of investment through capital employed profit is dependent on the contracting officer's decision on the WGL method. How much significance can be given to the capital employed factor in a corporate capital investment decision under current policies? If this factor has not received significant weight in corporate capital investment decisions it has failed to motivate capital investment and the DAC should be revised or rescinded.

I recommend, and these are my personal recommendations and opinions and not those of the Air Force or Space Division, immediate action to apply capital employed profit to all DOD contracts using the current applicable allocation methods and capital employed profit factors. This is to serve as an interim method during which revisions are made in the capital employed profit guidelines. I recommend that labor remain as the primary allocation base but that contractor capital be categorized into several major classifications such as production equipment, robotics, general purpose tools and equipment, computer and data processing, buildings and leases and that each of these classification to be eligible for a range of profit ranging from high for productivity enhancing production equipment to a low for such capital items as furniture and general equipment.

In closing - the DOD profit policy objective is to stimulate investment in productivity enhancing facilities - reward productivity improvement and foster a reasonable level of interest in defense contracts by providing a reasonable return on investment. The facilities cost of money profit is one of the methods toward achievement of these objectives. Profit alone will not achieve these goals. other programs with a more direct effect on capital investment are being undertaken and thru them hopefully these objectives will be achieved.

COMPARISON

MANUFACTURING VS RESEARCH AND DEVELOPMENT

		MFG (%)	R&D (%)
SUBTOTAL PROFIT/FEE		9.5	9.5
LESS: ADJUSTMENT FACTOR (30%)		2.85	0
SUBTOTAL PROFIT/FEE		6.65	9.5
COST RISK		3.0	3.0
CAPITAL EMPLOYED	25 TIMES 18	4.5	0
BASIC PROFIT/FEE OBJECTIVES		14.15	12.5
LESS: FACILITIES COST OF MONEY		0	3.0
TOTAL PROFIT/FEE OBJECTIVE		14.15	9.5

This hypothetical calculation illustrates the wide disparity in profit/fee opportunities possible as the result of the weighted guideline determination. I assure you this is not an isolated incident but is rather medium in impact when compared to some where the facilities capital cost of money has a ratio of 5 to 7 percent of estimated cost.

WEIGHTED GUIDELINES PROFIT/FEE OBJECTIVE

1. CONTRACTOR IDENTIFICATION	a. COMPANY NAME			b. DIVISION NAME (If any)	
	c. STREET ADDRESS		d. CITY	e. STATE	f. ZIP CODE

2. WEIGHTED GUIDELINES CATEGORY (Check one)
a. ☐ MANUFACTURING b. ☐ RESEARCH AND DEVELOPMENT
c. ☐ SERVICES

3. TYPE OF CONTRACT (Ref DAR, Sec III, Part 4)

4. BASIC PROCUREMENT INSTRUMENT IDENTIFICATION NO

5. SPIIN

a. PURCHASING OFFICE	b. FY	c. TY-PROC-INST-CODE	d. PRISN

6. WEIGHTED GUIDELINES PROFIT FACTORS (DAR 3-808.4)

PROFIT/FEE FACTOR OR SUBFACTOR (a)	MEASUREMENT BASE (b)	PROFIT WEIGHT RANGES MFG (%) (c)	R&D(%) (d)	SVCS(%) (e)	ASSIGNED WEIGHT (%) (f)	PROFIT/FEE DOLLARS (g)
PART I – CONTRACTOR EFFORT						
7. MATERIAL ACQUISITION						
a. SUBCONTRACTED ITEMS		1 TO 5	1 TO 5	1 TO 5		
b. PURCHASED PARTS		1 TO 4	1 TO 4	1 TO 4		
c. OTHER MATERIAL		1 TO 4	1 TO 4	1 TO 4		
8. ENGINEERING						
a. DIRECT LABOR		9 TO 15	9 TO 15			
b. OVERHEAD		6 TO 9	6 TO 9			
9. MANUFACTURING						
a. DIRECT LABOR	50%	5 TO 9	5 TO 9		70%	
b. OVERHEAD		4 TO 7	4 TO 7			
10. SERVICES						
a. DIRECT LABOR				8 TO 15		
b. OVERHEAD				4 TO 6		
11. OTHER COSTS						
12. GENERAL MGMT – G & A		5 TO 8	5 TO 8	5 TO 8		
13. SUBTOTAL PROFIT/FEE						
14. LESS ADJUSTMENT FACTOR		35				
15. TOTAL EFFORT						
PART II – CONTRACTOR RISK						
16. COST RISK	40% (Total from Col b)	6 TO 8	6 TO 7	30% 6 TO 4		
PART III – FACILITIES INVESTMENT						
17. CAPITAL EMPLOYED	10% (Line 8, DD Form 1861)	16 TO 30				
18. BASIC PROFIT/FEE OBJECTIVE (Items 14 + 16 + 17, Col g)						
PART IV – SPECIAL FACTORS						
19. SPECIAL PROFIT/FEE OBJ						
a. PRODUCTIVITY	(See DAR 3-808.5(a))					
b. INDEPENDENT DEVELOPMENT	(See DAR 3-808.5(b))	1 TO 4	1 TO 4			
c. OTHER	(Total from Item 16)	-5 TO +5	-5 TO +5	-5 TO +5		
d. TOTAL SPECIAL PROFIT/FEE OBJECTIVE						
20. SUBTOTAL PROFIT/FEE OBJECTIVE (Items 18 + 19d, col g)						
PART V – COST OF MONEY OFFSET (Applicable to Research and Development and Services Weighted Guidelines only.)						
21. LESS FACILITIES CAPITAL COST OF MONEY (DAR 3-808.5(a)(1)(ii))						
22. TOTAL PROFIT/FEE OBJECTIVE (Items 20 - 21, col g)						

DD FORM 1 JAN 80 1547 PREVIOUS EDITIONS ARE OBSOLETE.

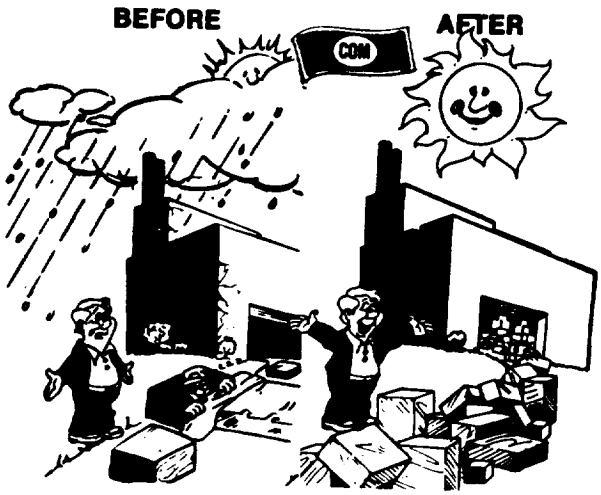
NSIA CONFERENCE
MISSION ASSURANCE
HUGHES AIRCRAFT COMPANY
JUNE 8, 1983
R. L. EDMONDS
(213)616-2132

HUGHES

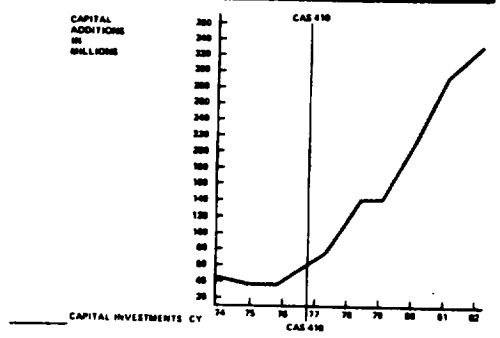
HUGHES CAPITAL INVESTMENT

- CONSISTENT WITH FACILITIES COM
- $1.3 BILLION; 5 YEAR PLAN

- DIGITAL COMPUTERS AND MICROELECTRONICS
- CO-LOCATE ENGINEERING AND MANUFACTURING



HUGHES RESPONSES TO COST OF MONEY RECOGNITION



HOW WELL IS COM POLICY WORKING

HUGHES

- REGULATIONS
- OVERALL HUGHES RESPONSE
- EDSG EXPERIENCE
- CONCERNS
- BENEFITS — GOVERNMENT AND INDUSTRY
- PROBLEMS
- RECOMMENDATIONS

FLEX FAB SYSTEM VS CONVENTIONAL MACHINING CENTERS

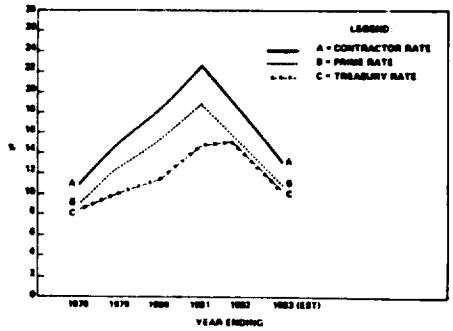
	FLEX FAB SYSTEM	MACHINING CENTERS
NUMBER OF MACHINES	9	25
COST	$7.5M	$8.3M
NUMBER OF OPERATORS PER SHIFT	3	25
LABOR COST OF OPERATORS PER YEAR-2 SHIFT	$130K	$1100K

COMPARISON IS BASED ON ABILITY TO ACCOMPLISH SAME AMOUNT OF MACHINING PER SHIFT

IS COM POLICY WORKING?

- YES – BASED UPON EXPERIENCE TO DATE
- MISSION ASSURANCE ENHANCED
  - IMPROVED PRODUCTIVITY
  - BETTER QUALITY
  - LOWER COSTS
- SOME CONCERNS

HISTORICAL COMPARISON PRIME RATES, CONTRACTOR BORROWING RATES, TREASURY RATES



SUMMARY OF SENATOR ROTH (DEL) HEARING
(SENATE GOVERNMENT AFFAIRS COMMITTEE — 19 MAY 83)

- GAO REPORT – COM
- SEVEN RANDOM CONTRACTORS – $500M SPENT OVER 5 YEAR PERIOD
- OSD – DATA BASE USED WAS INACCURATE
- CONCLUSION: PRESENT COM POLICY IS WORKING – GOVERNMENT PLANS NO BASIC POLICY CHANGES

FACILITIES CAPITAL/COM POLICY BOTH SIDES BENEFIT

CONTRACTORS

- REWARDED/INCENTIVIZED TO MODERNIZE FACILITIES
  - MODERN, EFFICIENT FACILITIES
  - INCREASED PRODUCTIVITY
  - BETTER QUALITY OUTPUT
  - MODERNIZATION WITHIN OUR CONTROL
  - KEEP PACE WITH TECHNOLOGY
- LOWER COST, COMPETITIVE

GOVERNMENT

- LOWERS GOV'T FACILITY EXPENSE
  - PAY AS YOU GO
  - REDUCED ADMIN EXPENSE
- ALL CONTRACTORS MUST MODERNIZE TO COMPETE
  - FACILITY COMPETITION
- LOWER COST END ITEMS
- MORE DIVERSIFIED BASE

BOTTOM LINE – MORE MODERN FLEXIBLE INDUSTRIAL BASE AVAILABLE TO ASSURE THE DOD/NASA MISSIONS ARE ACCOMPLISHED

SOME PROBLEMS

COM CONCERNS

- HIGH COM FACTORS
- COMPLEX, NOT UNDERSTOOD
- COMPETITION – COST IMPACT
- ALLOWABILITY OF COM COST

COM IMPLEMENTATION PROBLEMS

- R&D CONTRACT COM OFFSETS
- CPAF CONTRACT COM OFFSETS
- PROFIT + COM CEILING
- ZERO PROFIT/ZERO COM

COM OFFSET TO PROFIT

- WHEN THE R&D GUIDELINES ARE USED, THE PROFIT OBJECTIVE IS REDUCED BY THE AMOUNT OF FACILITIES CAPITAL COST OF MONEY
- THE PROFIT FOR MANUFACTURING CONTRACTS SHALL BE COMPUTED USING THE MANUFACTURING WEIGHTED GUIDELINES METHOD — NO OFFSET
- FOR R&D CONTRACTS, THE R&D WEIGHTED GUIDELINES ARE USED UNLESS, IN THE JUDGEMENT OF THE PCO, A SIGNIFICANT AMOUNT OF FACILITIES IS REQUIRED FOR EFFICIENT CONTRACT PERFORMANCE, IN WHICH CASE THE MANUFACTURING WEIGHTED GUIDELINES SHALL BE USED — NO OFFSET ON R&D TYPE

PROFIT IMPACT

	R&D METHOD	MANUFACTURING METHOD
BASIC PROFIT OBJECTIVE	$2,745.0K (9.0%)	$2,745.0K (9.0%)
ADJUSTMENT FACTOR (30%)	0	(823.5K)
PROFIT ON CAPITAL EMPLOYED	0	2,300.0K
LESS: PROPOSED COM - OFFSET	$1,220.0K (4.0%)	0
PROFIT OBJECTIVE	$1,525.0K (5.0%)	$4,221.5K (14.0%)

R&D CONTRACT — NO. 1 PROBLEM

- R&D DEFINITION
- LOWER PROFIT
- TIME CONSUMING NEGOTIATION
- NO REWARD FOR INVESTMENT

CUSTOMER COM POSITION

- DELETE COM
- STAND FIRM ON OFFSET
- OVERSTATE PROFIT – USE OFFSET

R&D CONTRACT EXCEPTION

- THE PROFIT OBJECTIVE FOR R&D CONTRACTS . . . . . .

  USE THE R&D WEIGHTED GUIDELINES . . . . UNLESS A

  SIGNIFICANT AMOUNT OF FACILITIES IS REQUIRED . . .

  IN WHICH CASE THE MANUFACTURING WEIGHTED

  GUIDELINES SHALL BE USED. – NO OFFSET

SECOND PROBLEM — CPAF/COM OFFSET

- DAR REQUIRES THAT COM BE SUBTRACTED FROM BASE FEE
- POSSIBLE NEGATIVE BASE FEE?

DAR CONFUSION

- DAR 3-405.5 – DEFINES CPAF CONTRACT

  "THE WEIGHTED GUIDELINES METHOD SHALL NOT BE APPLIED TO CPAF CONTRACTS WITH RESPECT TO EITHER THE BASE (FIXED) FEE OR THE AWARD FEE."

WGL PROVISION CONFLICTS

- DAR 3-808.2

  "ON COST-PLUS-AWARD-FEE CONTRACTS, THE BASE FEE SHALL BE REDUCED BY THE AMOUNT OF FACILITIES CAPITAL COST OF MONEY OR THE CONTRACT SHALL CONTAIN A PROVISION TO DIS-ALLOW THE COST."

CPAF COM OFFSET NEGOTIATION EXPERIENCE

- SUBTRACT COM FROM AWARD FEE POOL, NOT BASE FEE
- MUTUALLY AGREE DAR IS INCONSISTENT – DISREGARD OFFSET
- ADJUST BASE FEE AND AWARD FEE POOL
- CONVERT TO CPFF OR CPIF CONTRACT

THIRD PROBLEM — ARTIFICIAL PROFIT + COM CEILING

- SOME BELIEVE THAT COMBINING FEE AND COM CANNOT EXCEED THE DAR FEE LIMIT – 15% ON R&D CONTRACTS – 10% ON SUPPLY
- COM IS ALLOWABLE ELEMENT OF COST, NOT SUBJECT TO FEE LIMITATION

CUSTOMER 10.0% MAXIMUM POLICY

- PROPOSED: (SUPPLY CONTRACT CPIF)

TARGET COST	$10,000.0K	
COST OF MONEY	400.0K	(4%)
TARGET FEE	1,200.0K	(12%)
TOTAL CPIF	$11,600.0K	(16% COMBINED)

- ACTUAL SETTLEMENT:

TARGET COST	$10,000.0K	
COST OF MONEY	400.0K	(4%)
TARGET FEE	600.0K	(6%)
TOTAL CPIF	$11,000.0K	(10% COMBINED)

COM RECOMMENDATIONS

- REMOVE R&D OFFSET FROM DAR
- REMOVE CPAF OFFSET – DO NOT SUBTRACT COM FROM BASE FEE OR AWARD FEE POOL
- COM IS A COST, NOT A PROFIT – DO NOT APPLY ARTIFICIAL MAXIMUM TO COMBINATION OF FEE & COM
- RESCIND OLD INSTRUCTIONS AND POLICIES THAT CREATE CONFUSION – SUCH AS OFPP LETTER STATING ZERO PROFIT RESULTS IN COM DISALLOWANCE
- CLARIFY INCONSISTANCIES AND AMBIGUITIES IN DAR PROVISIONS

ACQUISITION PROCESS RECOMMENDATIONS

- ISSUE POLICY STATEMENTS FROM EACH COMMAND HEAD-QUARTERS TO PCO'S . . . EMPHASIZE THAT COM IS A COST AND SHOULD NOT BE CONSIDERED IN NEGOTIATION OF APPROPRIATE FEES AND PROFITS

EFFECT OF
PERFORMANCE INCENTIVES/AWARD FEE
ON
MISSION ASSURANCE

R. M. POWELL

Lockheed Missiles & Space Company, Inc.

CPIF/AF EXAMPLE

	1ST CONTRACT	2ND CONTRACT
● TARGET FEE	10%	12%
● AWARD FEE	5%	3%
● PERFORMANCE	8%	6%
● COST	6% { 1% D. B. 90/10 1ST 10% O. R. 80/20 2ND 20% O. R.	4% { 1% D. B. 90/10 1ST 10% 80/20 2ND 10%
● SCHEDULE	3%	5%

FEE BILLING:

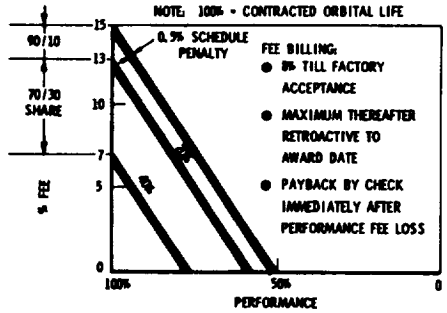
- INITIAL RATE OF 8% (1ST CONTRACT) OR 9% (2ND CONTRACT)
- AWARD FEE 2 MONTHS AFTER CUSTOMER EVALUATION
- PERFORMANCE FEE LOSSES PAID UPON DEMAND OF PCO

KEY FEATURES: (1) SIX-MONTH AWARD PERIODS; (2) CRITERIA SET AT START OF EACH PERIOD; (3) CUSTOMER ABILITY TO IMPACT CONTRACTOR; (4) EARNBACK ON PERFORMANCE PAST CONTRACT LIFE POSSIBLE FOR COST, SCHEDULE AND PERFORMANCE LOSSES.

NOTE: (1) AWARD FEATURE SAME ON BOTH CONTRACTS EXCEPT 8 NOT AWARDED ON 2ND CONTRACT GO INTO POT FOR LATER CONSIDERATION

(2) "CHANGE IN SCHEDULE EMPHASIS BETWEEN 1ST & 2ND CONTRACT"

CPIF EXAMPLE



CPFF/AF EXAMPLE

	1ST CONTRACT	2ND CONTRACT
FIXED FEE	4.0%	6.0%
AWARD FEE	4.5%	3.25%
AWARD FEE FACTORS PERIOD	6 MO.	6 MO.
COST	2.5%	2.0*
PERFORMANCE	2.0%	1.25%

EVALUATION CRITERIA - SUBMITTED 30 DAYS PRIOR TO START OF NEXT PERIOD TO CUSTOMER FOR REVIEW, REVISION AND APPROVAL

SELF EVALUATION/CUSTOMER EVALUATION

*FINAL DETERMINATION POINT - 50% OF AWARD OR LOSS THEREOF ON THE COST FACTOR CAN BE RESCINDED OR AWARDED RESPECTIVELY AT TWO TIMES DURING LAST HALF OF CONTRACT

FEE BILLING: ● FIXED FEE BILLED MONTHLY
● AWARD FEE TWO TO THREE MONTHS AFTER EVALUATION

CPIF EXAMPLE

WITH EARNBACK PROVISION

FEE: 15% MAXIMUM

INCENTIVES:

- COST: -7.5%
  SHARE RATIO 80/20 1ST 15% O.R., 70/30 2ND 15% O.R.
- SCHEDULE: -0.7% CUMULATIVE AT 0.0388% PER DAY FOR PRELIMINARY DD-250 AND/OR 0.0033 PER DAY FOR LAUNCH
- PERFORMANCE: -15%
  DIVIDED LINERALLY OVER CONTRACT ORBITAL LIFE

BILLING:

8% T.C. UNTIL PRELIMINARY DD-250, MAXIMUM FEE THEREAFTER

CPAF EXAMPLE

FEE: 8% AWARD

FEATURES:

- 75% PRELAUNCH - 25% 30 DAYS POST LAUNCH
- SIX-MONTH PERIOD
- CRITERIA SET FOR EACH PERIOD FOR THREE FACTORS --
  - ACHIEVEMENT
  - COST CONTROL
  - BUSINESS MANAGEMENT
- SELF EVALUATION/CUSTOMER EVALUATION
  - ONLY ONE RATING OUT OF NINE DISPUTED TO DATE

FEE BILLING:

- 4% IN ADVANCE ON MONTHLY BASIS
- AWARD AND BILLING OF BALANCE APPROXIMATELY TWO MONTHS AFTER SUBMITTAL OF CONTRACTOR EVALUATION

OBSERVATIONS AND CONCLUSIONS

- AN INCENTIVE WITH MAJOR EMPHASIS ON PERFORMANCE CREATES A MANAGEMENT ENVIRONMENT THAT IS FAVORABLE TO PREVENTATIVE AND CORRECTIVE ACTION WHICH LEADS TO MISSION ASSURANCE.
- PERIODIC AWARD FEE CAN HELP ASSURE UPDATED AND CONGRUENT PRIMARY OBJECTIVES OF CUSTOMER PROGRAM OFFICE AND CONTRACTOR PROGRAM TEAM.
- THE SIMPLER THE INCENTIVE ⇒ THE EASIER TO UNDERSTAND ⇒ THE MORE EFFECTIVE IN MOTIVATING PERSONNEL.
- A WELL THOUGHT OUT INCENTIVE DISCIPLINES THE CUSTOMER AS WELL AS THE CONTRACTOR.

ACQUISITION WORKSHOP
EFFECT OF PERFORMANCE INCENTIVES/
AWARD FEE ON MISSION ASSURANCE

Lee J. Ginsberg
Contracts Manager
Space Launch Systems Division
Martin Marietta Aerospace

INCENTIVE CONTRACTING

Incentive contracting is a way of life both within the Department of Defense (DOD) and the National Aeronautics and Space Administration (NASA). It is currently accepted by industry as a means to achieve its fiscal and growth objectives. Incentive contracting is being applied by the government to cost reimbursable and fixed price type contracts. The rewards and penalties provided by such contracts gets industry's top management attention.

NOT A NEW INNOVATION

Performance Incentives are not a new innovation. They have been used during the earliest days of government contracting. The first iron navy ship, the "Monitor", during the Civil War was contracted on a total performance contract. It had to be floated, attain a specified minimum speed and win its first battle before the contractor was paid. The Monitor met the first two conditions, fought the "Merrimac" to a draw on March 9, 1862 but the contract was paid.
The Wright Brothers contracted with the United States Department of War in 1908 for the first military aeroplane. The contract was for a fixed price of $25,000 with a $5,000 bonus for exceeding target speed by 2 miles per hour. This incentive was earned.

A RECENT INNOVATION

Although, according to the NASA CPAF Guide, subjective fee determinations have existed in government contracts since the 1950s for aircraft maintenance and overhaul, the first full-fledged CPAF contract ever negotiated was by NASA in 1962. Bendix Radio received a 2 year $10 million CPAF contract from NASA for operations, maintenance and engineering services for Project Mercury tracking and communications stations. Since then NASA has used CPAF contracts extensively. NASA formalized its award fee methods in a cost-plus-award-fee Guide in 1967 to promote greater uniformity of practice. The Navy let its first pure CPAF contract in 1964 for maintenance of instrumentation and range faciliities in Los Angeles and has continued to widely use this contracting method.

QUESTIONS NEED ANSWERING

Considering the broad application of incentive contracts the government has funded many research projects to find the answers to these questions:

- Do incentive contracts accomplish the government's objectives?
- Do the benefits outweigh the cost?
- Can improvements be made?

Initially, studies to provide answers to these questions were theoretical. As more use of incentive contracting was made and more data became available empirical testing was made possible. However, many conclusions are still based on limited samples and results are not always conclusive.

JUSTIFICATION FOR USE OF INCENTIVES

A study conducted by the Logistics Management Institute on "An Examination of the Foundation of Incentives Contracting (1968)" found the following four primary justifications for the government's use of incentive contracts:

- Incentives motivate efficient contract management and achievement of a high performance product.
- Incentives enable the government to reward contractors on the basis of demonstrated management ability and product performance.
- Incentives assign to the contractor a larger portion of contract risk than he would bear with a cost-plus-fixed-fee contract.
- Incentives provide explicit communication of the government's contracting objectives.

These four items are just as valid for justifying today's use of incentive contracting as first stated in 1968.

TYPES OF INCENTIVES

The following four types of incentives are directly related to contactor performance. When properly applied, they assure achievement of the government's objectives and goals of mission assurance within acceptable cost and schedule constraints:

- o Cost
  - Shared monetary and rewards or penalties over range of incentive effectiveness
- o Schedule
  - Usually penalty in terms of fee/profit
- o Performance
  - Fee/profit, reward/penalty based on discrete measurable technical characteristics.
- o Award
  - Fee based on subjective unilateral customer determination on preestablished evaluation criteria.

MOTIVATORS

The DOD/NASA Incentive Contracting Guide and the DAR considers the profit motive essential to successful implementation of an incentive contract. The somewhat dated guide states that "The profit motive is the essence of incentive contracting." Financial gain under risk conditions is gained through increased profit by:

- o Attaining cost levels more beneficial to the government
- o Meeting or exceeding technical performance goals
- o Outstanding contractor overall performance

However, the DAR and the Guide recognize that there are extracontractual incentives that motivate the contractor:

- o Firm perpetuation
- o Sales maximization
- o Socio-economic considerations

The government's use of extracontractual incentives should be considered as motivators. Although they are limited because of being beyond the control of the government and their use is not in the DAR to any significant manner, understanding and innovative use of such motivators can increase the effectiveness of incentive contracts.

CONTRACTOR MOTIVATION STUDY

A research study was conducted by Dr. Robert E. Williams of the Army Procurement Research Office in 1981 on contractor motivation. The objective was to attempt to identify motivational practices to improve contractor performance and constraints on the use of these practices. Results were gathered from questionnaires provided to Army procurement personnel and industry members of the National Security Industrial Association (NSIA). Government and industry perceptions were received on twelve defense contractor incentive objectives and on the effectiveness of twenty one government incentives.

WHAT DOES THE CONTRACTOR REALLY WANT?

Some general conclusions of the Williams study revealed that:

- o Motives of contractors are complex but amenable to incentive contracting.
- o Industry is interested in long-term profit incentives followed by short-term profit objectives.
  - Providing a good product was by far the most important objective followed by
  - Maintaining a long term continuing business relationship, then
  - Improved cash flow
  - Profit and
  - Development of new capabilities.
- o Industry does not "want" things not readily translated into profit impact
- o Government buyers perceive industry is interested in short-term profit objectives
- o Government and industry indicate that long-term profit incentives are most important.

Based on this it appears the govern-

ment doesn't understand what industry really wants.

USE OF TECHNICAL PERFORMANCE INCENTIVES

Technical performance incentives are used by the government to motivate the contractor to achieve outstanding technical performance. Performance incentives force the identification of technical objectives in advance of contract award. Close support by technical personnel is required to determine adequate characteristics, targets, measurement criteria and relative importance. Inadequate determination of such items can make the administration difficult.

ARE PERFORMANCE INCENTIVES NECESSARY?

Dr. Raymond Hunt concluded in his study on "Extracontractual Influences in Government Contracting" that most organizations are oriented more toward performance than cost reduction. And that performance incentives are probably redundant with the "natural" organizational tendencies and therefore contribute little to program management and performance control. Studies by Belden and Parker came to the same conclusion that performance incentives were typically earned regardless of cost overrun/underrun outcomes. This tends to support Dr. Williams conclusions that contractors view providing a good product by far the most important objective. However, lack of performance incentives may send a negative message to contractors that the government has no real interest in performance.

DO PERFORMANCE INCENTIVES CONTRIBUTE TO MISSION SUCCESS?

Performance incentives have contributed to the overall success record of Titan launch vehicles. One hundred and nineteen of the last one hundred and twenty two launches were successful. This is an outstanding 97.5% success record. The government dollar commitment for launch vehicle performance was small in comparison to value of payloads exposed to loss. These dollars were well spent and the government received value for their expenditure. The Titan launch vehicle success record was made possible in part by a comprehensive mission success program.

MISSION SUCCESS ORGANIZATION

A dedicated Mission Success organization supports the assurance of mission success for Titan launch vehicles. Acting as a dedicated central agency, the organization is directly responsible to senior management and the customer. Dedicated to mission assurance, the organization monitors nonconformances and failures for impact on the program, assesses launch readiness, assures corrective action is taken and lessons learned are incorporated. Mission Success conducts program audits, oversees motivational and training activities down to the lowest levels and institutes assurance programs to continue the launch of high quality and reliable vehicles that earn contract incentives.

DOD/NASA AWARD FEE CONTRACTS

DOD procurement actions of award fee contracts have averaged 3.2% of total net value of contracts issued over the past five years. Air Force procurement actions during this same period have averaged 2.2%. The higher DOD percentage over the Air Force may be attributed to the greater use of award fee contracts by the Navy. The significance of award fee contracts is out of proportion to the total procurement value for DOD. However, NASA procurement actions of total net value of contracts over the same period have averaged 66.5%. NASA relies heavily on award fee contracts to achieve its research and development objectives.

CONCEPT

Because of the government's uncertainty of contractor's ability to perform in the changing and uncertain weapons systems acquisition, a retroactive subjective evaluation of contractor's performance was adopted.

The government's strategy concerning uncertainties is to adapt to it and reduce it by participating with the contractor in the detail planning and management of the acquisition. Through this strategy, the government provides the contractor the ability to earn profits rather than negotiate them. The award fee concept may be considered a method of management, "a tool", not a contract type. Therefore, a learning curve should apply. Of thirteen CPAF contracts studied, Mel Byers of the USAF found that the "percentage of award fee the contractor earned seemed to increase" as the contract progressed.

IMPLEMENT THE STRATEGY

Ways which would support successful implementation of the award fee strategy and reward performance judged outstanding by the Air Force include:

- Encourage government - contractor cooperation
- Assure active role for government managers
- Stimulate both formal and informal communication
- Recognize variablity of human organization motivation
- Let contractors motivate own personnel
- Provide flexibility and room for human judgment
- View the acquisition process as dynamic
- Simplify contractual provisions

SOME FEARS

Sherer, in his book "The Weapons Acquisition Process" mentioned some objections to after-the-fact evaluations raised by the NSIA. These include:

- Socialism
- Unilateralism
- Unfair allocation of responsibility for program outcomes
- Bureaucratic proliferation and delay of awards
- Vagueness of contract objectives and evaluation criteria
- Subjectivity and bias in performance evaluation

We continue to hear similar objections made to award fee contracts. Sherer generally discounted many of these objections and viewed these items as manageable and necessary elements in making the award fee process successful.

Other objections raised by industry are:

- Contractor over responsiveness
- Demanding to administer
- Regulatory restrictions

APPLICATION

After-the-fact evaluation was limited predominately to support and services efforts by the Government in the past. However, today this concept is being applied to all types of efforts such as:

- R&D
- Build
- Operations

Expanded use of the award fee concept now requires reevaluation of exisiting government regulations governing award fee contracts.

REGULATORY RESTRICTIONS

The DAR and agency supplements provide the regulations used by the government to formulate and implement award fee contracts. These regulations control:

- Base fee, DAR 3-405.5(d) and AFSC DAR Sup 3-405.5(d)(3)
- Unearned award fee, AFSC DAR Sup 3-405.5(d)(2)
- Cost of money, DAR 3-808.2(a)(1)(g)
- Partial payment of award fee, DAR 3-405.5(e)(i)
- Evaluation/Payment, AFSC DAR Sup 3-405.5(e)(7)

SUGGESTED IMPROVEMENTS

Because of the broader application of award fee contracts and to assure that the intended incentive features are maintained to provide maximum contractor motivation, the following improvements are recommended:

- Keep award fee periods short or
  - Provide monthly advance fee payments
  - Provide interim feedback to the contractor on performance

- Institute an overage award fee tracking system
- Allow CAS 414 Facilities Cost of Money because of the broader application of award fee to facility intensive contracts.
- Provide roll forward of unawarded fee to provide maximum contractor motivation
- Provide a small base fee not subject to evaluation

EFFECT OF PERFORMANCE INCENTIVES/

AWARD FEE ON MISSION ASSURANCE

INCENTIVE CONTRACTING

- o A WAY OF LIFE WITHIN DOD AND NASA
- o ACCEPTED BY INDUSTRY
- o USED TO MOTIVATE IN MONETARY TERMS
- o APPLIED TO FIXED PRICE AND COST REIMBURSABLE TYPE CONTRACTS
- o GETS TOP MANAGEMENT ATTENTION

NOT A NEW INNOVATION

- o PERFORMANCE INCENTIVES USED IN EARLIEST DAYS OF GOVERNMENT CONTRACTING
  - THE "MONITOR" CONTRACTED ON A TOTAL PERFORMANCE CONTRACT (1861)
    - o MUST BE FLOATED
    - o ATTAIN A SPECIFIED MINIMUM SPEED
    - o WIN FIRST BATTLE
  - THE WRIGHT BROTHERS "HEAVIER THAN AIR MACHINE" (1908)
    - o FIXED PRICE - $25,000
    - o BONUS FOR EXCEEDING - $5,000
      TARGET SPEED BY 2M/HR

QUESTIONS NEED ANSWERS

- o DO INCENTIVE CONTRACTS ACCOMPLISH THE GOVERNMENT'S OBJECTIVES?
- o DO THE BENEFITS OUTWEIGH THE COSTS?
- o CAN IMPROVEMENTS BE MADE?

JUSTIFICATION FOR INCENTIVES

- o LOGISTICS MANAGEMENT INSTITUTE FOUND INCENTIVES:
  - MOTIVATE EFFICIENT CONTRACT MANAGEMENT AND ACHIEVEMENT OF A HIGH PERFORMANCE PRODUCT
  - ENABLES GOVERNMENT TO REWARD CONTRACTORS FOR DEMONSTRATED MANAGEMENT ABILITY AND PRODUCT PERFORMANCE
  - ASSIGN TO CONTRACTOR LARGER PORTION OF RISK THAN WITH A CPFF CONTRACT
  - PROVIDE EXPLICIT COMMUNICATION OF GOVERNMENT'S OBJECTIVES

TYPE OF INCENTIVES

- o COST
  - SHARED MONETARY REWARDS OR PENALTIES OVER RANGE OF INCENTIVE EFFECTIVENESS
- o SCHEDULE
  - USUALLY PENALTY IN TERMS OF FEE/PROFIT
- o PERFORMANCE
  - FEE/PROFIT REWARD/PENALTY BASED ON DISCRETE MEASURABLE TECHNICAL CHARACTERISTICS
- o AWARD
  - FEE BASED ON SUBJECTIVE UNILATERAL CUSTOMER DETERMINATION ON PREESTABLISHED EVALUATION CRITERIA

MOTIVATORS

- o DOD/NASA INCENTIVE GUIDE STATES, "PROFIT MOTIVE IS THE ESSENCE OF INCENTIVE CONTRACTING", REALIZED BY:
  - ATTAINING MORE BENEFICIAL COST LEVELS
  - MEETING OR EXCEEDING TECHNICAL PERFORMANCE GOALS
  - OUTSTANDING CONTRACTOR OVERALL PERFORMANCE
- o EXTRA-CONTRACTUAL INCENTIVES
  - FIRM PERPETUATION
  - SALES MAXIMIZATION

CONTRACTOR MOTIVATION STUDY

- o DR. WILLIAMS RESEARCH STUDY ON CONTRACTOR MOTIVATION - (1981)
- o OBJECTIVE
  - IDENTIFY MOTIVATIONAL PRACTICES TO IMPROVE CONTRACTOR PERFORMANCE
  - IDENTIFY CONSTRAINTS ON USE OF THESE PRACTICES
- o QUESTIONAIRES SENT TO ARMY PROCUREMENT PERSONNEL AND MEMBERS OF NSIA
- o PERCEPTIONS RECEIVED ON
  - DEFENSE CONTRACTOR INCENTIVE OBJECTIVES
  - EFFECTIVENESS OF GOVERNMENT INCENTIVES

PAGE 64 INTENTIONALLY BLANK

CONTRACTOR MOTIVATION STUDY CONCLUSIONS

- MOTIVES OF CONTRACTORS ARE COMPLEX BUT AMENABLE TO INCENTIVE CONTRACTING
- INDUSTRY IS INTERESTED IN LONG-TERM PROFIT INCENTIVES
- INDUSTRY DOES NOT "WANT" THINGS NOT READILY TRANSLATABLE INTO PROFIT IMPACT
- GOVERNMENT BUYERS PERCEIVE INDUSTRY INTERESTED IN SHORT TERM PROFIT INCENTIVES
- GOVERNMENT BUYERS AND INDUSTRY INDICATE LONG-TERM PROFIT INCENTIVES ARE MOST IMPORTANT
- GOVERNMENT BUYERS NEED TO UNDERSTAND WHAT THE CONTRACTOR REALLY WANTS

USE OF TECHNICAL PERFORMANCE INCENTIVES

- MOTIVATES CONTRACTOR TO STRIVE FOR OUTSTANDING TECHNICAL ACHIEVEMENT
- CREATES A BALANCE IN EMPHASIS ON COST AND SCHEDULE PERFORMANCE WHEN COMBINED WITH SUCH INCENTIVES
- FORCES IDENTIFICATION OF TECHNICAL OBJECTIVES IN ADVANCE OF CONTRACT AWARD
- REQUIRES TECHNICAL PERSONNEL TO DETERMINE:
  - CHARACTERISTICS TO BE INCENTIVIZED
  - WHAT CONSTITUTES "SUPERIOR" TECHNICAL ACHIEVEMENT
  - WHAT CONSTITUTES A REASONABLE TARGET
  - HOW THE CHARACTERISTICS WILL BE MEASURED
  - RELATIVE IMPORTANCE OF DIFFERENT CHARACTERISTICS

ARE PERFORMANCE INCENTIVES NECESSARY?

- DR. HURT CONCLUDED
  - MOST ORGANIZATIONS ORIENTED MORE TOWARD PERFORMANCE THAN COST REDUCTION
  - PERFORMANCE INCENTIVES PROBABLY REDUNDANT WITH "NATURAL" ORGANIZATION TENDENCIES
  - ORGANIZATIONS HAVE A PERFORMANCE BIAS
- BELDON AND PARKER ALSO CONCLUDED
  - TYPICALLY EARNED REGARDLESS OF COST OVERRUN/UNDERRUN OUTCOMES
- LACK OF CONTRACT PERFORMANCE INCENTIVES MAY SEND A NEGATIVE MESSAGE
  - "GOVERNMENT HAS NO REAL INTEREST IN PERFORMANCE"

DO PERFORMANCE INCENTIVES CONTRIBUTE TO MISSION SUCCESS?

- PERFORMANCE INCENTIVES HAVE CONTRIBUTED TO SUCCESS OF TITAN LAUNCH VEHICLES
  - 119 OF LAST 122 WERE SUCCESSFUL LAUNCHES (97.5%)
- DOLLAR COMMITMENT FOR LAUNCH VEHICLE PERFORMANCE INCENTIVE SMALL IN COMPARISON TO VALUE OF PAYLOADS EXPOSED TO LOSS
- THESE DOLLARS ARE WELL SPENT; GOVERNMENT RECEIVES VALUE
- SUPPORTED BY A MISSION SUCCESS PROGRAM

MISSION SUCCESS ORGANIZATION

- CENTRAL AGENCY
- DIRECTLY RESPONSIBLE TO TOP MANAGEMENT AND CUSTOMER
  - STRONG MANAGEMENT FOCUS DOWN TO THE MOST JUNIOR LEVEL
- MONITOR NONCONFORMANCES AND FAILURES FOR IMPACT ON PROGRAM
  - ASSESS READINESS STATUS
  - CORRECTIVE ACTION
  - PEDIGREE
  - LESSONS LEARNED
  - AUDITS
- OVERSEE PROGRAM MOTIVATIONAL AND TRAINING ACTIVITIES
  - ATTITUDE
  - RECOGNITION
  - AWARDS

A RECENT INNOVATION

- AFTER-THE-FACT EVALUATION
  - EVOLVED FROM USE OF SUBJECTIVE TECHNIQUES SINCE THE 1950'S WITH COST-PLUS-INCENTIVE FEE CONTRACTS
- FIRST FULL-FLEDGED COST-PLUS-AWARD FEE CONTRACT EVER NEGOTIATED WAS BY NASA IN 1962
  - BENDIX RADIO, 2 YEARS, $10 MILLION
  - 3% BASE FEE, 7% AWARD FEE POOL
  - OPERATIONS, MAINTENANCE AND ENGINEERING SERVICES FOR PROJECT MERCURY TRACKING AND COMMUNICATIONS STATIONS
- NASA FORMALIZED AWARD FEE METHODS IN A 1967 GUIDE TO PROMOTE GREATER UNIFORMITY OF PRACTICE

DOD/NASA AWARD FEE CONTRACTS

FISCAL YEAR	% OF NET VALUE			% OF ALL CONTRACTS		
	DOD	AF	NASA	DOD	AF	NASA
'78	2.9	2.6	---	0.5	0.4	---
'79	2.7	1.9	61.9	0.5	0.5	---
'80	3.0	2.0	65.6	0.5	0.5	41.6
'81	3.3	1.7	69.6	0.5	0.5	46.4
'82	3.9	2.6	68.9	0.6	---	42.5

CONCEPT

- UNCERTAINTY OF CONTRACTOR'S ABILITY TO PERFORM
- GOVERNMENT'S STRATEGY
  - ADAPT TO IT
  - REDUCE IT
  - PARTICIPATE IN MANAGEMENT OF THE ACQUISITION
  - PROVIDE EFFECTIVE MANAGEMENT AT THE LOCAL LEVEL
  - PROFITS ARE EARNED NOT NEGOTIATED

IMPLEMENT THE STRATEGY

- ENCOURAGE GOVERNMENT - CONTRACTOR COOPERATION
- ASSURE ACTIVE ROLE FOR GOVERNMENT MANAGERS
- STIMULATE BOTH FORMAL AND INFORMAL COMMUNICATION
- RECOGNIZE VARIABILITY OF HUMAN ORGANIZATIONAL MOTIVATION
- LET CONTRACTORS MOTIVATE OWN PERSONNEL
- VIEW ACQUISITION PROCESS AS DYNAMIC
- PROVIDE FLEXIBILITY AND ROOM FOR HUMAN JUDGMENT
- SIMPLIFY CONTRACTUAL PROVISIONS

SOME FEARS

- SOCIALISM
- UNILATERIALISM
- UNFAIR ALLOCATION OF RESPONSIBILITY FOR PROGRAM OUTCOMES
- BUREAUCRATIC PROLIFERATION AND DELAY OF REWARDS
- VAGUENESS OF CONTRACT OBJECTIVES AND EVALUATION CRITERIA
- SUBJECTIVITY AND BIAS IN PERFORMANCE EVALUATION
- OVER RESPONSIVENESS
- DEMANDING TO ADMINISTER
- REGULATORY RESTRICTIONS

APPLICATION

- PREVIOUSLY APPLIED TO SERVICES AND SUPPORT EFFORTS
- NOW BEING APPLIED TO ALL TYPES OF EFFORTS
  - R&D
  - BUILD
  - OPERATIONS
- REQUIRES REEVALUATION OF EXISTING GOVERNMENT REGULATIONS

REGULATORY RESTRICTIONS

- BASE FEE

  DAR 3-405.5(d) - ". . . . BASE FEE SHALL NOT EXCEED THREE PERCENT . . . ."

  AFSC DAR SUP 3-405.5(d)(3) "THE "BASE" FEE WILL BE ZERO . . . ."

- UNEARNED AWARD FEE

  DAR - SILENT ON DISPOSITION OF UNEARNED AWARD FEE.

  AFSC DAR SUP 3-405.5(d)(2) "UNEARNED AWARD FEE WILL NOT BE CARRIED FORWARD FOR POSSIBLE AWARD IN SUBSEQUENT EVALUATION PERIODS FOR INCENTIVE FEE PURPOSES."

- COST OF MONEY (CAS 414)

  DAR 3-808.2(a)(1)g ". . . . THE BASE FEE SHALL BE REDUCED BY THE AMOUNT OF FACILITIES CAPITAL COST OF MONEY OR THE CONTRACT SHALL CONTAIN A PROVISION TO DISALLOW THE COST."

REGULATORY RESTRICTIONS - continued

- PARTIAL PAYMENT

  DAR 3-405.5(e)(1) ". . . . PARTIAL PAYMENT OF FEE WILL GENERALLY CORRESPOND TO THE EVALUATION PERIODS . . . ."

- EVALUATION/PAYMENT

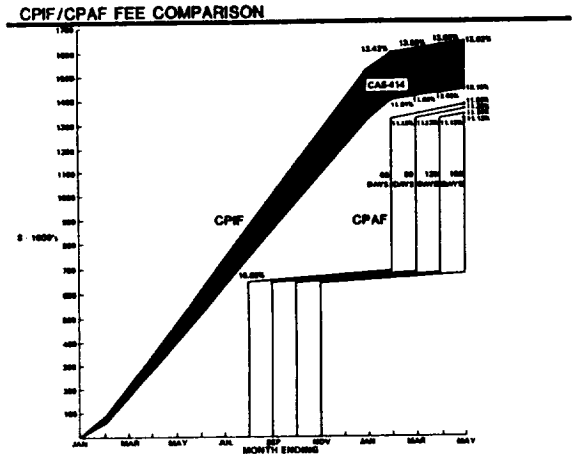
  AFSC DAR SUP 3-405.5(e)(7) - "WITHIN 30 DAYS AFTER EACH EVALUATION PERIOD THE FDO WILL ADVISE THE CONTRACTOR AND CONTRACTING OFFICER IN WRITING OF HIS DECISION . . . .

  . . . . THE CONTRACTING OFFICER WILL UNILATERALLY AMEND THE CONTRACT WITHIN 30 DAYS AFTER WRITTEN NOTIFICATION FROM THE FDO . . . ."

CPIF/CPAF FEE COMPARISON BASIS

	CPIF	CPAF
PERIOD OF PERFORMANCE (MOS)	12	12
AWARD FEE PERIODS		2 - 6 MONTHS
TARGET/ESTIMATED COST	$12,000,000	$12,000,000
CAS 414	$ 180,000 (1.5%)	0
FEE	$ 1,300,000 (11%)	$ 1,440,000 (12%)
AWARD (90%)		$ 1,296,000 (10.8%)
BILLINGS		
COST	BI-MONTHLY	BI-MONTHLY
CAS 414	MONTHLY	0
FEE	MONTHLY	"X" MONTHS AFTER PERIOD
ANNUAL INTEREST	12%	12%

CPIF/CPAF FEE COMPARISON



SUGGESTED IMPROVEMENTS

- KEEP AWARD PERIODS SHORT OR
  - PROVIDE MONTHLY ADVANCE FEE PAYMENTS
  - PROVIDE INTERIM FEEDBACK ON PERFORMANCE
- INSTITUTE AN OVERAGE AWARD FEE TRACKING SYSTEM
- ALLOW CAS 414 FACILITIES COST OF MONEY
  - BROADER APPLICATION OF AWARD FEE TO FACILITY INTENSIVE CONTRACTS
- PROVIDE ROLL FORWARD OF UNAWARDED FEE
  - ADDS ADDITIONAL INCENTIVE DURING LATTER PART OF CONTRACT
- PROVIDE A SMALL BASE FEE NOT SUBJECT TO EVALUATION

EFFECT OF PERFORMANCE INCENTIVES/ AWARD FEES ON MISSION ASSURANCE

Barney Klehman
Senior Contract Reviewer, AFSD

WHAT ARE WE ASSURING?

- o PERFORMANCE DURING THE CONTRACT
  - o Flexibility
  - o Responsiveness
  - o Understanding what the customer feels is important
  - o Objective vs Subjective Measurement
- o PERFORMANCE "AFTER" THE CONTRACT
  - o Program Requirements
  - o Objective Measurement Emphasized
  - o Subjective Measurement
- o CONTRACTOR INTERESTS VS GOVT INTERESTS
  - o Are they really different
  - o Are they incompatible
  - o Effect of performance incentives
  - o Effect of award fee

THE CONTRACT

Is it a means to an end, or an end in itself?

- o WHAT ARE THE TRUE OBJECTIVES?
- o CONTRACT TYPE
  - o AFSC Business Strategy Panel
  - o Draft RFP
  - o Political Considerations
  - o Theoretical Considerations
  - o Motherhood
- o THE INCENTIVE PROCESS
  - o Multiple Incentives-Do They Work?
  - o Positive Incentives-Do They Work?
  - o Negative Incentives-Do They Work?
  - o Baseline vs "Locking In"

CONTRACTOR PROBLEMS WITH AWARD FEE

1. No "Control" over outcome- subject to whims
2. Personality change may be required
3. Administrative burden of self-assessment preparation
4. Administrative burden of briefing management
5. Conflict of interest-managing the program vs satisfying the customer? Who assumes final responsibility?
6. Cash flow and financial forecasting
7. Tradeoffs in a multiple incentive arrangement
8. Late payment
9. Inability to recapture lost award fee in later periods
10. Base fee vs target fee vs satisfactory performance

SPACE DIVISION EXPERIENCE

- o MISSION RESPONSIBILITY

PAGE 68 INTENTIONALLY BLANK

- o ROLE OF THE CONTRACT
- o PERFORMANCE INCENTIVES
  - o Track Record
  - o The Program Manager View
  - o The Contracting Officer View
  - o Problems
  - o Conclusion
- o AWARD FEE
  - o Track Record
  - o The Program Manager View
  - o The Contracting Officer View
  - o The Contractor View
  - o Problems
  - o Conclusions

THE BOTTOM LINE

- o IS CONTRACT TYPE IRRELEVANT?
  - o Effect on Contractor's Management
  - o "Business as Usual"
- o MONEY TALKS
  - o Near Term Profitability may be crucial
  - o Cash Flow
  - o Return on Investment
  - o Non-Financial Measurement
- o TRENDS
  - o What's Happened Since Last Time
  - o The Resistance to Change
  - o Line vs Staff
  - o Local vs Command
  - o Inertia
  - o The importance of individual personality
  - o Current Developments at SD and AFSC
  - o Comparison to USAF, NASA, other Services

CONCLUSION

- o What are the obstacles to mission assurance?
- o Are there better approaches available and attainable?

CONTRACTOR PROBLEMS WITH PERFORMANCE INCENTIVES

1. Difficulty of defining performance criteria
2. Lack of performance control-failure attribution
3. Cash Flow
   - a. Incremental Payments
   - b. Minimum Time Lapse
   - c. Mean Mission Duration
   - d. Recoupment (Negative Penalties)
   - e. Profit Erosion (Time Value of Money)
4. Difficulty of Radically Changing Organization/Plant

PROJECT QUALITY

'INCENTIVIZING QUALITY IN CONTRACTS'

MR. A. J. DELUCA
AFSC/PM

METHODOLOGY

- BRAINSTORMING
- CATEGORIZATION
- SURVEY DEVELOPMENT
- GOVERNMENT / INDUSTRY MEETINGS
- DATA ANALYSIS
- RECOMMENDATIONS PRODUCED
- DISSEMINATE REPORT / TEST HYPOTHESES

OVERVIEW

- PURPOSE
- METHODOLOGY
- CAUTIONS
- FINDINGS
- PROPOSED TOOLS
- BOTTOM LINE

CAUTIONS

- ACTIONS SPEAK LOUDER THAN WORDS (LOW BIDDER SYNDROME)
- DON'T PAINT US ALL WITH THE SAME BRUSH
- COST OF COLLECTING DATA MAY EXCEED BENEFITS
- DON'T SQUEEZE THE BALLOON
- CLEARLY DEFINE SR^2 INGREDIENTS
- DON'T ASK FOR COSMETIC QUALITY - PAGE LIMITATIONS
- LOOK OUT FOR GAMING ON INCENTIVES
- DIFFICULT TO DO CONTRACT BY CONTRACT
- HAND ME BACKS ARE TRUE HIDDEN COSTS

PURPOSE

DEVELOP CONTRACTUAL TOOLS AIMED AT REDUCING THE COSTS OF SCRAP, REWORK, AND REPAIR (SR^2)

FINDINGS

- TOP MANAGEMENT EMPHASIS KEY
- INSTANT CONTRACT TOOLS AVAILABLE – EFFECT LIMITED
- APPLICATION OF TOOLS DEPENDS ON ACQUISITION PHASE
- SR^2 ONLY PIECE OF THE PUZZLE
- MAXIMUM BENEFIT THROUGH PLANT-WIDE PROGRAM

PROPOSED TOOLS

SHOULD COST

- OBTAIN SR^2 DATA IN PROPOSAL
- IDENTIFY SR^2 AS DISCRETE ELEMENT FOR REVIEW
- APPLY PRODUCE TO YIELD PHILOSOPHY

APPLICATION

CONCEPT DEFINITION	TECHNOLOGY DEMONSTRATION	VALIDATION	FULL SCALE DEVELOPMENT	INITIAL PRODUCTION	FOLLOW-ON PRODUCTION

• CONTRACTING →

- FILL-IN RFP
- FENCE QUALITY DOLLARS
- SOURCE SELECTION
- DEEP LOOK RFP
- DESIGN SUBMISSION/ REVIEW GESTATION
- AWARD FEE
- SOURCE SELECTION
- DEEP LOOK RFP
- PROPOSAL INSTRUCTIONS
- QUALITY CDR
- PROGRAM QUALITY PLAN
- PRODUCE TO YIELD
- YIELD INCENTIVES
- SHOULD COST
- MMPCR INTEREST
- PROGRESS PAYMENTS
- PARETO INCENTIVES

DESIGN TO YIELD

- REQUEST PROPOSED YIELD RATES FOR PRODUCTION
- INCLUDE RATES IN CONTRACT AS GOAL
- PENALTIES/REWARDS IN FOLLOW-ON CONTRACTS FOR PAST EXPERIENCE
- IN-PRODUCTION INCENTIVES BELOW AGREED TO GOAL

DEEP LOOK DRFP

- REQUIRE FLOW DOWN TO SUBS
- INCLUDE PURCHASE ORDER IN DRFP
  - ASSEMBLE INDUSTRY REPS
  - "ALL STAR" REVIEW

PRODUCE TO YIELD

- REQUEST FIRST TIME YIELD DATA IN SOLICITATION
- AUDIT RESPONSE
- PARETO ANALYSIS TO SELECT DRIVERS
- NEGOTIATE FIXED RATE FOR FIRST TIME YIELD
- SELECT YIELD GOAL
- SHARE REDUCTION TO GOAL
  - MUST HAVE AUDIT RIGHT

AWARD FEE

- OBTAIN PROGRAM QUALITY PLAN WITH MILESTONES
  - CONTRACT
  - PROGRAM
  - PLANT
- AGREE ON EXPECTED RESULTS
- PREPARE AWARD FEE PLAN
- MONITOR PROGRESS
- AWARD ACCORDINGLY

FENCE QUALITY DOLLARS

- CLEARLY STATE QUALITY REQUIREMENTS IN RFP (TOLERANCES, REPORTING, ETC.)
- HAVE CONTRACTORS PROPOSE COSTS TO ACCOMPLISH
- ISOLATE $ IN CONTRACT FOR QUALITY TASKS

SOURCE SELECTION

- SPECIFIC REQUIREMENT – QUALITY PROGRAM DATA
  - NOT OVERALL QUALITY PLAN
    - PASS/FAIL
    - TIED TO COLOR CODE
- USE PAST PERFORMANCE AS CRITERIA

BOTTOM LINE

FILL IN RFP

- LEAVE SCHEDULE DELIVERY DATES BLANK IN RFP
- SELECT CONTRACTOR WITH MOST REALISTIC SCHEDULE
- CAN WORK IN CONJUNCTION WITH DEEP LOOK DRFP

BOTTOM LINE

- RELEASE REPORT
- CONTINUE HIGH LEVEL MANAGEMENT INTEREST
- WORK TOGETHER TO SCOPE TRUE COSTS OF NON-QUALITY
- RECOGNIZE SR^2 HEAVILY INFLUENCED BY EARLY DECISIONS
- TAILOR THE TOOLS TO FIT THE SITUATION
- LOOK BEYOND INSTANT CONTRACT

INDUSTRY'S RESPONSE - AFSC PROJECT QUALITY

Mission Assurance Conference
Acquisition Workshop
June 9, 1983

W. C. Rideout
AIA/NSIA

PROJECT QUALITY REPORT

Forward

Provide a summary of findings and recommendations of Project Quality with guidelines for application in AFSC contracts

INTRODUCTION

- Quality
  - Not only in terms of product performance
  - But also in overall cost

Key element

- focus on cost of scrap, rework and repairs[2];
- Commonly referred - as the hidden factory

Hidden factory or non quality costs are the focus of the attack.

- Government must drive down SR[2] costs
- Through contractual arrangements with industry

- A systems approach towards quality must consider manufacturing, purchasing, engineering, etc.

- Observation - ability to incentivize contractors by individual contract provision is greatly limited

CHARTER

PROJECT QUALITY GROUP

- DoD committed to modernize our military forces.

We seek affordable, capable & reliable systems of high quality.

- Unfortunately many manufacturers have attempted to "insert in" quality after production rather than designing it in

- Consequence - (DoD) invested heavily in source inspection systems, scrap, rework and reinspection facilities

This has been called America's "hidden factory".

Project quality will address the hidden factory in American defense industries and devise contractual approaches aimed at eliminating it.

Signed

Robert T. Marsh, General
Commander AFSC

APPLICATION

Introduction

- Selected most workable and effective techniques
- 6 categories
  - I - Preproposal Actions
  - II - Specification and S.O. W. Preparation
  - III - Negotiations
  - IV - Performance, Administration of Contracts
  - V - Policy
  - VI - Other Considerations
- Recommendations basically geared to individual contracts
- Recommendations - application of spin-offs of existing techniques

CATEGORY I

I-1. Draft RFP

- Instructions - identify provisions adversely affecting SR[2] rates.

Ind. - Generally being done - subjective
- Inputs treated as proprietary
- Erode competitive edge

*I-2. "Down Load" RFP

- Existing procedures don't receive best efforts
- Purchase Order to prospective offerors
  - reduce air fare/per diem costs
  - one location - company rate.
  - not for attribution
  - data delivered - Government with limited rights

Ind. - Potential collusion - exposure
- Offerors - not "invited"
- Contracts - value/coverage
- Loss of data - key to winning
- Reaction - biased

*New recommendations since last review by industry associations.

*I-3. "Fill in the blanks" - reasonable delivery requirements

- Contractors forced into unrealistic delivery requirements
- Tied with "Down Load" RFP

Ind. - Unrealistic - each contractor giving "optimum" schedule.
- Tell Government - what they want to hear
- Invites game playing

I-9. Require a detailed quality plan

- CDRL - deliverable
- Reviewed and approved - incorporated in contract
- Subject to same material requirements and remedies for non-compliance in progress payments.

Ind. - Program Plan generally Contract Plan
- Experience Quality Plan is contract requirement
- and deliverable

I-10 Require detailed procedures to secure good subcontractor performance

- Incentives prime will use
- Pre-award assessment procedures
- Information on tradeoffs between price performance and quality

Ind. - MIL-Q - 9858A requirement
- Key subcontract approval clause
- MIL STD 1535 (Subcontract Quality Plan)
- Not a trade-off item

I-12. Increase use of award fees

- Often extremely effective
- Can be used to "target" management activity.

Ind. - Award fees successful in other areas of contract performance
- Quality decisions could be improved
- Worthy of further discussion
- NASA experience
- Limitations - extraordinary requirement, clearly set forth up front

*I-16. RFP's - "Design to Unit Cost" basis

- Remove "award to low bidder" syndrome
- Must be used with caution
- Used with stringent use final test inspections

Ind. - Don't understand"
- Quality & cost - different aspects
- Look at D.T.U.C. - make sure getting quality

*I-17. RFP - Quality requirements specified as contract deliverables

- Ex. - submission of acceptable quality plan
- Required contents for each deliverable and price

Ind. - Usually done on major procurements
- CDRL - Data Item Description - explicit
- Example - corrosion control plan

CATEGORY II

II-4. Use of "definition period" early in contract

- Require use of a tolerance study prior to beginning of production
- Would identify critical areas deliverable and priced

Ind. - Accomplishment very unlikely
- FSD contract generally includes productivity study
- Engineering changes developed during all phases of test, manufacturing and field performance
- Design freeze is really a goal
- Changes to optimize vs. cost

*II-7. Establish vigorous reliability demostration test

- Awards for 1st, 2nd, 3rd "no problems" reviews
- Model after Navy RAP - Reliability Acceptance Program

Ind. - Generally done - supportive
- Part of planning effort
- Reliability growth tests - want failure

*II-8 Promote use of service warranties
Span it has time after delivery
Contractor rebuild/servicing
Delivery - term production runs

Ind Possible conflicts
Quality service contract

CATEGORY III

III-1&2 Involvement of Quality personnel (Government and Industry) in negotiations and quality considerations used in source selection.

Certainly on No. 1 and caution on No. 2

Past performance on quality is difficult to quantify with variables as.
- anticipated design risk
- equivalent product lines
- time since last requirement improvements made in interim

CATEGORY IV

IV-2 S.D.W. require senior contractor quality personnel in all design reviews

- Also presence of some on Government side
- No designs accepted until both sides "buy off"

Ind. - Review is appropriate
Definition of "senior" technical (rather than management)

IV-6 Ensure MRB's address why nonconforming part was produced

- Ultimate goal: improve manufacturing process used

Ind - Contrary to Mil Std. 1520 - corrective action document where MRB addresses proper disposition
- Disposition of hardware
- Corrective Action boards or functions address why

*IV-8 Customers put together team of users - meet with contractor

- Demonstrate importance of program
- Public relations - employee morale

Ind. - Good idea
- Not done often enough

*V-9 Establish ombudsman at AFSC - "court of last resort"

- Subcontractors and small business - no place to go for assistance

Ind. - Support procedural guidance
- Fox in the chicken coop
- Disputes clause - "court of last resort"

CATEGORY V

*V-5. Policy - Quality given emphasis early in program

- Monies (non-transferable) set aside for prototype testing, etc.
- Caution - time pressures, funds deleted during negotiations

Ind. - Agree
- Who backs off first?

*V-6. Policy requiring feedback to contractors from users

- Address at PMR/CAR briefing

Ind. - AF 66-1 doesn't do job
- Poor and untimely data
- Need early data on prototypes

*V-7. Establish AFSC fund pool - through "Program Tax"

- For "Quality Technology" or plant-wide award fee contract
- Designed to improve quality, reduce scrap, rework and repair
- Plant-wide approach

Ind. - Sounds like other programs
- Man Tech - Tech Mod - IMIP
- Get Price (ESD)

*V-8. Require ECP evaluations to consider Quality improvements, life cycle costs

- Require Government quality assurance review of ECP's

Ind. - Positive use of VECP - some problems
- Experience - counter-productive
  - Rejections
  - Delays

CATEGORY VI

VI-1. Advance cost agreement (DAR 15-107)

- Direct charge to contracts for quality improvement training

Ind. - Good idea
- No DAR change necessary
- Navy doing it

*VI-4. Develop scrap/rework/repair standards on commodity-wide basis

- Requires long term analysis/commitment

Ind. - Not sure what is intended?
- Defects, value/man hr. or repairs?
- Rates/common wide commodity basis?

WHAT MAKES COMMERCIAL SPACECRAFT CONTRACTS DIFFERENT?

James E. Vint
Staff Contract Analyst, AFSD

The commercial spacecraft world has significant differences from the military spacecraft. Although the spacecraft are similar in function, i.e., both are Communications satellites, there are differences related to the required use of military satellites during wartime necessitating more stringent design and production criteria.

However, commercial satellites face the same natural environment and perform virtually the same mission as military communications satellites. In this context differences in means of procuring these spacecraft become even more significant. This chart lists some notable points about commercial contracts.

COMPARISON OF COMMERCIAL VS MILITARY SPACECRAFT PROGRAMS

Comparison between a typical commercial and military communication satellite program. Perhaps the major difference reflected on this chart is that the commercial customer needs the satellite and had based revenue projections on service from the satellite. The customer seemingly does not perturb the builder and this results in a stable fabrication cycle and a higher probability of on-time delivery.

BUILDING-IN PROBLEMS

This chart portrays the problems of performing high risk research and development on an essentially production contract i.e., a build of 2 - 4 spacecraft. Inevitably problems develop and a schedule stretch results leading to overruns, late deliveries and unfavorable publicity.

APPLICABILITY OF COMMERCIAL APPROACH

Carrying the commercial approach too far can result in problems also. If an achievable baseline does not exist many of the concepts applied to a commercial contract cannot be applied. For example, a well defined requirement spec may not be possible. A firm fixed price contract may not be possible or desirable.

The approach makes the most sense when the technology is in hand and we know what we are building.

PAGE 78 INTENTIONALLY BLANK

WHAT MAKES COMMERCIAL SPACECRAFT CONTRACTS DIFFERENT?

COMMERCIAL CUSTOMERS HAVE PURCHASED SPACECRAFT SINCE THE MID-60'S WITHOUT DIRECTING:

- SPECIFICATIONS - OTHER THAN PERFORMANCE REQUIRED
- STANDARDS
- PARTS AND PROCESSES

COMMERCIAL CUSTOMER DOESN'T PERFORM IN-PLANT QA DURING FABRICATION

COMPARISON OF COMMERCIAL VS MILITARY SPACECRAFT PROGRAMS

COMMERCIAL		MILITARY
LOW	COMPLEXITY	HIGH
LOW – FFP CONTRACT	RISK	MODERATE - HIGH – CP OR FPI CONTRACT
CRITICAL – DELIV ON TIME – PENALTY FOR LATE DELIV – MFED DATES DRIVE SERVICE PROMISE & REVENUE PROJ	SCHEDULE	CRITICAL – DELIV OFTEN LATE – CHANGES OFTEN DELAY DELIVERY
STABLE	STABILITY	SUBJECT TO CHANGE – CONGRESS - REQMT & FUNDING – OSD/OSD/AF/AFSC – INDUSTRY MARKETING – GAO INFLUENCE
GOOD BUSINESS PRACTICE	CONSTRAINTS	BAR PPBS HIGH LEVEL PERSONNEL TURNOVER – PRIORITIES LOW LEVEL PERSONNEL TURNOVER – DOCUMENTATION INCREMENTAL FUNDING

BUILDING-IN PROBLEMS

CAPABILITY NEEDED BUT TECHNOLOGY NOT AVAILABLE

SPACECRAFT CONTRACT AWARDED
- SOME DEVELOPMENT
- SOME RISK
- SOME TBD + IN SPEC

TECHNICAL PROBLEMS
- SCHEDULE STRETCH
- HIGH LEVEL CONCERN / REVIEW

NATIONAL PRIORITIES CHANGE
- BUDGET REDUCED / SPREAD

MORE STRETCH OUT

LONGER DURATION INCREASES COST
(MANPOWER) + (MORE TIME) = HIGHER CO
INFLATION OVER TIME INCREASES COST

UNFAVORABLE PUBLICITY

APPLICABILITY OF COMMERCIAL APPROACH

NOT APPLICABLE FOR HIGH RISK TECHNOLOGY DEVELOPMENT

APPLICABLE WHERE BASELINE DESIGN:

- EXISTS
- IS ACHIEVABLE
- MEETS USER REQUIREMENTS

APPLICABLE FOR MODERATE RISK TECHNOLOGY DEVELOPMENT WHEN

- LOW RISK BASELINE IS PURSUED IN PARALLEL
- CONTRACTOR SEES AND SHARES IN BENEFITS

PAGE 80 INTENTIONALLY BLANK

SUMMARY OF ISSUES AND RECOMMENDATIONS

THE ACQUISITION PROCESS

SUMMARY OF TOPICS,
OBSERVATIONS, AND ACTIONS

1. Government/Industry Communication - Acquisition Processes

Actions

a) Document observations and recommendations in Mission Assurance proceedings.

b) Promote continuing dialogue between AFSD and Industry.

2. Carlucci Initiatives
(Defense Acquisition Improvement Program)

Observations/Actions

a) Implementation incomplete (6 in trouble)

b) Motivate Program Manager

3. Draft RFP

Observations/Actions

a) Endorsed but spotty results

b) Improve

- Integration - WBS, PPI, SOW, Section M, etc.
- Reduce proposal size
  - Page limits
  - Delete unscored plans
- Provide response matrix
- Standardize format and binding specifications
- Eliminate redundancy in RFP and requests for redundant proposal data
- Encourage bidders to suggest improvements in RFP and tailoring through early coordination and feedback.

4. Specification Tailoring

Observations/Actions

Endorsed two approaches

a) Specific reference to para. or reference spec para.

b) "Partitioning"

5. Cost of Money (COM)

Observations/Actions

a) Clarify COM regs, instructions and policies

b) Treat COM as cost not profit (don't subtract COM from profit)

6. Incentives/Award Fees

Observations/Actions

a) Not common understadning of value to Mission Assurance

b) Keep award fee period short

c) Incorporate fee awarded more rapidly

d) Roll unused fees forward

e) Simple, objective incentive fee structure

7. Project Quality Findings

Observations/Actions

a) Not common agreement on 64 recommendations

b) Government controls environment

c) Industry in reactionary mode to Government and public opinion

8. Commercial versus Military Practices

Observations/Actions

a) Fundamentally different

b) Fourteen factors cited for cost/schedule growth of Government programs

c) Commercial lessons for Government (reduce micromanagement)

WORKSHOP I

SOFTWARE

Chairmen:

Bill Porter
Manager, Data Systems Resource Management
General Electric

Major Ed Koss
Director, Computer Resources
USAF Space Division

Coordinator:

Robert Borland
General Electric

SESSION 1A — SOFTWARE MANAGEMENT

The Role of Planning in a Project Environment	Michael Evans, Integrated Computer Engineering

SESSION 1B — TRANSITION TO ADA

USAF Strategy for Introducing Ada into Mission Critical Systems	Maj. David Hammond, USAF
Impact of Ada on the Software Life Cycle	Susan Mickel, General Electric
"An Ada Application Experience"	John Bailey, General Electric

SESSION 2A — SOFTWARE DEVELOPMENT MANAGEMENT

Software Development Management — Grappling with the Intangibles	Dr. Sol Volansky, Hughes
Decade of Software Engineering Problems and Solutions — MPD Case History 1973-1982	E. Reese, General Electric
Overview of Software Cost Estimating Methodology	Thomas F. Masters, NSA
The Constructive Cost Model	Barry Boehm and Thomas H. Bauer, TRW
A Method of Measure Effective Productivity in Building Software Systems	Lawrence H. Putman, Douglas T. Putnam, Lauren P. Thayer
PRICE S	Peter B. Korda, RCA/Price Systems
Adjusting Software Estimating Models to Obtain Comparative Results	Bernard H. Rudwiek, Planning Research Corp.
Software Estimating and Scheduling Using Parametric Models	William C. Cheadle, Martin Marietta

THE ROLE OF PLANNING IN A PROJECT ENVIRONMENT

Michael W. Evans

INTEGRATED COMPUTER ENGINEERING

2672 Bayshore Frontage Road
Mountain View, California 94043

Telephone: (415) 965-8921

ABSTRACT

This paper describes the need for effective project planning and why, in the context of the project this planning is essential. The paper discusses the Software Development Plan and its significance in the project environment also. Discussed is the relationship between this plan and other areas of project planning and the role of software manager in the planning process. An actual project situation is presented where planning was poorly done and the problems which resulted are described. Much of the information is abstracted from a book being written by the author "Productive Software Test Management" to be published by John Wiley and Sons in early 1984.

Author's Background

Michael W. Evans is Vice President of Integrated Computer Engineering (I.C.E.), a company specializing in software management, quality assurance and productivity improvement.

Mr. Evans is a recognized expert in the field of software management, project recovery, and methodology selection and application. He has successfully managed several projects ranging from new applications through recovery of projects experiencing significant development difficulties. The project applications ranged in difficulty from commercial point of sale systems through large real-time communications software. He has published over twenty-five articles and papers on related subjects of software project management and is currently under contract to John Wiley and Sons to produce two books, "The Productive Management of Software Development" to be published in October 1983, and "The Productive Management of Software Testing" to be published in early 1984.

Introduction

Software development is a complex process requiring the careful integration of diverse disciplines, technical activities and administrative project controls. Too often software projects are implemented without first defining and implementing firm plans to project how the software will be developed. The true measure of project effectiveness is how well the management orchestrates the development activities, and the effectiveness of the tools, techniques and methodologies applied to the project and the smoothness of the data flow and transition of responsibility as the implementation proceeds. Management controls must be defined and implemented to assure schedule and budget compliance, software quality, and correspondence to user requirements. In short, the software manager must plan and manage the disciplines required of software development not monitor their progress.

For the software manager, the pitfalls in software acquisition are numerous. To the initiated, the labyrith of software development is overwhelming. The terminology is unclear, the technology is complex, and the acquisition, administration, and management of technical personnel and development activities seemingly impossible.

The manager too often throws up his hands in the midst of the development crisis failing to recognize that although the form of the software management problem may be unique, the specific management techniques, practices, and procedures required to develop, integrate, and demonstrate software are based on good engineering management principles.

The characteristics of the software product which contribute significantly to the management dilemma are the lack of physical products associated with software development and the complexity of the management program necessary for control of the software development.

Through rigorous application of discipline and judicious application of tools, techniques and methodologies, software development can become an engineering discipline rather than a technological art form.

The software development process is the integration of several distinct, and often overlapping technical, management, and control methodologies. Each of these methodologies supports different project areas as follows:

1. Project Planning – The techniques and procedures, which take basic project, technical, contractual administrative, and development requirements, translating them into a project structure which support the productive development of software.

2. Requirements Definition – The process by which user technical and operation requirements are defined in a traceable, testable

form which serves as the technical basic for functional decomposition and software design.

3. Functional Decomposition – The techniques procedures by which user requirements are functionally decomposed into a set of functional and derived requirements which may be allocated to individual software components for detailed software design.

4. Software Design – The specific procedures, techniques, technical requirements, and data required to decompose the software functional requirements and derive the software design.

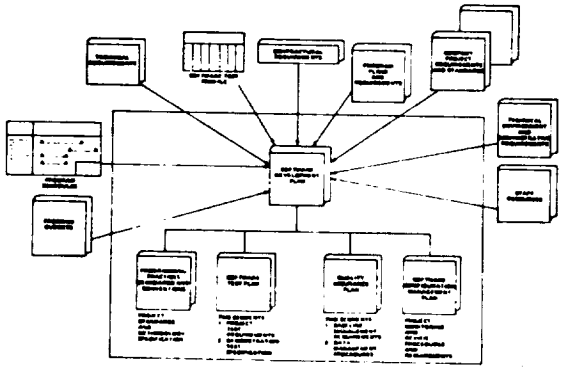
5. Software Code And Test – The procedures, techniques, data, and supporting environments which are applied to support and control the development of software code and conduct preliminary software testing.

6. Test And Integration – The specific technical procedures and techniques applied to integrate the software into an operational configuration and the management policies and practices implemented to control the process.

7. Configuration Management And Control Policies – Practices and techniques implemented to coordinate and control data baselined by the project or approved through a project review.

8. Automated Support – The tools and automated support facilities developed or installed in support of the project methodologies.

The degree to which these methodologies play together, the data compatibility between each of the individual methodologies, the smoothness and control of the data flow between each of the individual methodologies and the various components of the project is the measure of how efficient the project implementation is proceeding.



The Software Development Plan – The Starting Point

As illustrated in Figure 1, the planning of the software project should be hierarchical based on the contractual and technical requirements of the project, the resources available to support the development, the constraints imposed by the development environment, and the standards, development requirements, and expectations of the company, customer and user. This software development plan is the essential first step in planning the project environment, technical and management controls, resources and support requirements and project review, defining requirements for development monitoring, and essential program, customer, and company interface requirements. This plan forces the manager to define and communicate what will be done by the project at each point in the development cycle, what tools, techniques and methodologies will be applied, and how the process is to be managed and controlled.

In the software development plan, the software manager describes the criteria to be followed in the technical areas of the project as well as general requirements for planning and implementing these criteria in the context of the project structure. These general criteria must be translated will define the technical and support requirements of the project and establish the specific methodologies to be followed in development of the software.

Secondary Planning Documentation – How to do it

From this plan a series of 'how to' plans may be developed for the project. The four most important to the project are:

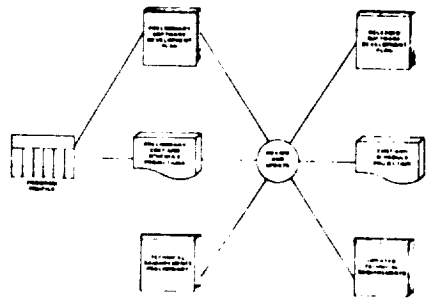
1. The Programming Practices Standards And Conventions (PP S&C) which details the project standards for designing, implementing, reviewing, and describing the software product. The requirements and detailed requirements for the development, code, test, integration and support requirements are defined and the interfaces between each of the methodologies is defined.

2. The Software Quality Assurance Plan defines how each of the formal and informal reviews are to be conducted and how these reviews are to be integrated into the project environment. This plan should also describe how the quality, performance, and integrity of the software application is to be assured.

3. The Software Test Plan should identify how the test and integration requirements described in the software development plan are to be implenented in the project environment. This plan should be developed in two parts. The first describes in detail how the project will support each of the various test levels and the second release details the software demonstration requirements which, when completed will qualify the software resulting in customer acceptance.

4. Software Configuration Management Plan – This plan is the document which how details the approved customer baselines are to be managed and controlled as well as detailing how, in the project environment data is to be managed and controlled. The baseline management functions control the in ternally approved software products during the period of the development.

As illustrated in Figure 2 the production of the Software Development Plan is only one of several planning functions essential for project success. The technical requirements must be defined and the cost and schedules should be developed.



The Management Dilemma

The early period of the project is the most critical from a managers standpoint since it is during this period that the structure of the project is defined, the technical requirements and characteristics of the project are developed, the cost, schedule, and resource constraints which limit the perogatives of the software manager are identified, and what is done by the project during the course of development and the specific management and project control procedures and techniques are defined and implemented. In short, it is during this period that the software manager decides what he is going to do to ensure project success.

Despite the importance of this early planning to the ultimate success of the project it is often difficult for a manager to find time or resources early in the project to perform these functions. The typical project is typically front loaded in with a myriad of essential short term tasks which require the direct participation of the software manager and his staff to complete. Early customer reviews, program planning meetings, project staffing, budget negotiations, and project organization development all compete for the software managers time and sap the limited resources available to the software project early in the implementation. Longer term problems such as software project planning, meaningful cost and schedule projections, critical early test planning activities, and planning and development of a productive project environment are often deferred in lieu of these short term requirements in the hopes that, "After these crises are over their time will be available to address the pressing long term issues of the project." Unfortunately, the reality is that there is never time available in a project situation.

Once the manager falls into the trap of trading short term project requirements for long term planning requirements, he or she has sown the seeds of poor productivity. The commitments on the manager and staff increase, requiring more time rather than less to complete. As the development proceeds through the design and development stages the time that the manager hoped would be available for planning evaporates. The situation is compounded since the early project planning which would bring order to these latter project phases – requirements definition, design, coding, test and integration has not been accomplished resulting in, at best, an inefficient project environment and, at worst, development chaos.

The software manager must recognize the importance of this early planning to the long term health of the project. This planning, and the specific requirements for it should be integrated into the project environment sharing equally in importance with the short term and early technical activities and milestones. These activities, conducted prior to the Systems Requirements Review should be under the direct supervision of the software manager. In the case of the Software Development Plan the manager should have primary responsibility for defining the plans content and in determining the requirements for the tools, techniques, methodologies, and project management procedures and controls.

A Planning Example

The planning steps described above are critical components of a successful software project. Unfortunately, the planning requirement is often ignored due to conflicting priorities or, in many cases a lack of experience in the management of software. The following example is a description of an actual project situation in which the planning was poorly done, the symptoms of the resultant problem, the cataclysm which resulted and how the problem was resolved.

Project Background – Early Seeds Of Disaster

In the mid 1970's a contract was awarded to develop a state-of-the-art communications network. The project began with the usual flourish, customer and contractor personnel congratulated each other on the quality of the specifications and agreed that, in this case the development would be done right and prove to be an example to be followed by the industry. For the software manager, the first several months were a whirlwind of of meetings, briefings, interviews and short term project requirements. There was no time for planning but, in the mind of the manager, as soon as these short term commitments were completed, there would be more than enough time to plan. This did not prove to be the case.

Early Schedule Drift – Project Control Problems

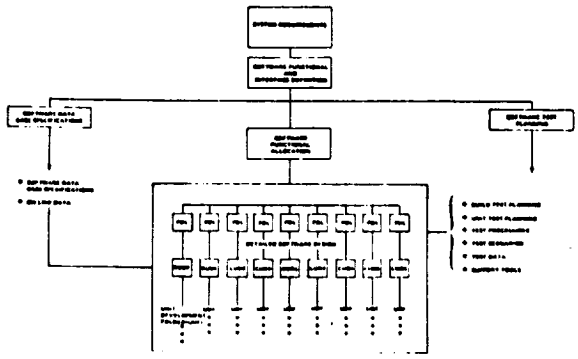
The first major project milestone was the System Requirements Review (SRR). This review was the responsibility of the program office, however, because of the customers requirements and the nature of the system a major segment of the presentation was presented by software. Everyone on the project worked hard getting ready and, although the preparation for the review was disorganized, the presentation, was reasonably well received and established a feeling on all sides that the project was proceeding well. This was not the case. Right after this review the project began to experience difficulty. People working on the project did not have an understanding of how tho develop or document the design. Schedules and budgets were ignored. The application of resources was ineffective and the generation of the functional specification became an insurmountable roadblock not achievable by the project.

The software project was hostage to the quality and effectiveness of the previous project activities. Seemingly small problems encountered during the early stages of the project now had major impact on the project success, technical, cost and schedule performance, and the ultimate reliability and quality of the software system.

The software manager was told late in pre-PDR segment of the project, when budgets, schedules and resources had been depleted, that major design problems had been found requiring significant time and resources to correct.

The software manager found himself on the horns of an unresolvable dilemma. On the one hand, he had to maintain schedules and budgets while, on the other, he had no control over the magnitude or complexity of the technical problems being found or the resources required to correct them. He had lost control over the technical commitments required to satisfy project goals and objectives.

As illustrated in figure 3, a requirement for early project focus was essential because of the nature of data production in the software project environment.



Early in the development, the project was required to control a small, manageable quantity of data expressed as requirements and interface specifications. These requirements were reasonably static, not undergoing frequent and substantial modification. The software manager and his staff found that, even with poor, ineffective configuration control procedures the integrity of the data could be maintained.

From these basic requirements, the software project had to define a set of software functional, interface, and operational requirements upon which the software development was to be based. The amount and complexity of the data to be controlled by the project increased substantially, and manual and ineffective control practices were becoming cumbersome. This data, once defined was reasonably static not requiring frequent or significant change.

There was, however, an explosion of information as the project moved through the latter stages of functional and software design, code and test phases of development. In a short period of time the amount of data which had to be produced and controlled by the project increased by an order of magnitude. By not being planned for, this data explosion caught the project without sufficient controls, discipline, and support facilities to manage the process of development. The data explosion continued throughout the remainder of the development and when not dealt with effectively, resulted in project chaos.

A second problem was not dealt with early. The long lead times associated with planning, implementation, and application of tools, data, and project controls essential to success. The development of the essential components of the project environment should have been accomplished in parallel with the technical aspects of requirements definition, design and coding. They should have been available when needed by the project if project productivity was to be maintained. A late start of development of these items virtually assured that the components would not be available in time to avoid development impacts.

What Happened – Disaster

The Project problems began to snowball shortly after the functional specification delivery schedules were missed. The software manager found that, since there was no organization to the project and no plan for doing the essential development tasks even the simplest project task could not be successfully accomplished. The project slowed and slowed, eventually stopping prior to PDR.

Corrective Procedures – Difficult But Effective

When the project blew up the program manager was faced with a difficut decision. He felt that he had only six possible options, none of which were attractive:

1. Reorganizing the project, shifting personnel assignments in an attempt to gain management expertise and perspective.

2. Keeping the same management while rescheduling and rebudgeting the software project to provide sufficient time and resources to complete the effort.

3. Bring in outside "experts" to evaluate the software development problem to restructure the existent project, and to get the project focused toward a goal.

4. Descoping development to reduce technical requirements, problems, and schedule requirements.

5. Cancelling the project or, at least reassigning development responsibility to a different organization.

6. Reassigning software management responsibility initially, keeping the same schedules, budgets, and staff while giving the new manager freedom to reestablish development parameters as required.

The manager initially selecting option 3. The consultant reviewed the project and identified 35 project areas which should be looked at. The areas ranged from methodology application through cost and schedule controls. When faced with the magnitude of the recommendations the manager was concerned. He grouped the problems and found that the five most significant were:

1. A general lack of motivation on the part of personnel assigned to the project.

2. An inability on the part of software personnel to meet schedule or cost milestones and to identify specific causes for the slips or overruns.

3. Lack of understanding, prior to development of how tools, techniques and methodologies were to be integrated into the project structure.

4. An inability to clearly state project goals, objectives, and development methodologies.

5. An overall lack of clear project organization and communication sources.

Based on this analysis a new software manager was assigned and told to "fix it".

Planning – How It Turned The Project Around

The software manager took the following steps:

1. Planning – An overall software structure, project organization, and control and development environment were defined and documented in a Software Development Plan, Configuration Management, Standards and Conventions, Test, and Quality Assurance Plans.

2. Staffing – The staffing segment of the task was extremely difficult due to the critical requirement to get the job started. An initial analysis of the taskindicated that staff was required in five primary areas:

   a. Development – The development team was organized under the technical leader and staffed with personnel previously assigned to the project.

   b. Tool Development – An initial evaluation of project tools indicated that they were inadequate to support development and test of the software. A group was indentified to develop tools for supporting project methodologies.

   c. Test Planning – An independent test planning and development function was identified to verify software quality and adherence to baselines.

   d. Documentation – To maximize productivity it was decided to offload as much documentation responsibility as possible from the development team. Personnel were identified to document the software.

   e. Configuration Management – To control the myriad of

software products and baselines a strong configuration management structure was required. The functional and staff requirements were identified.

An organizational structure was then identified and an aggressive staffing plan was pursued to fill the required positions. Using a combination of known outside consultants, outside body shops, and personnel within the company, the project was fully staffed in a short time. Each individual assigned was brought into a documented position and assigned a documented task.

3. Scheduling – Scheduling of the job was accomplished in two steps:

   a. A somewhat arbitrary set of software builds was defined to provide a technical basis for incremental development and testing of the software.

   b. Scheduling personnel were brought in to work with technical personnel in deriving and overall software PERT by build. This PERT was developed within a short time and identified the critical interrelationships among the tasks involved.

4. Establishment of Software Configuration Review Board (SCRB) – SCRB was a forum where any project member could submit a problem report on any aspect of the project he felt needed attention and that each problem would be reviewed amd resolved. The SCRB was described in the Software Development Plan and the first meeting was held within 6 weeks of project restart, thus firmly establishing its existence. It was also established at this time that the SCRB meetings would be held weekly.

The management activities described above enable the project to start at a full run. Each staff member came on board fully cognizant of his responsibilities as well as overall project goals and the way development was to proceed.

Management functions during implementation consisted of forcing adherence to the plans and structures defined at project start, monitoring schedules and making adjustments as necessary when slippages occured, and instigating immediate action as problems arose. Project momentum was maintained mainly by the continued active management participation.

Several key decisions during the planning phase proved especially significant during the implementation period:

1. The early development project planning documentation proved to be critical to the ultimate success of the project. The new manager was forced to think through what he was going to do on the project and, from these requirements, define how the project requirements were to be satisfied.

2. The decision to initiate development of tools early in the planning phase facilitated productivity and was the single most important factor allowing a key training milestone to start on time.

3. The independent test effort proved an essential part of the development process and, along with various configuration management and quality assurance functions, provided positive means for maintaining, monitoring, and controlling all software and support products throughout the development and test period.

4. The PERT schedule provided specific information concerning the development status. On many occasions, early recognition of problem in development schedules allowed additional resources to be applied to the effort to recover lost schedules.

Project Experience – Implementing The Plan

This early project planning was the means by which the software manager gave early form and structure to the project. It was through the planning that the myriad of development tools, techniques, and methodologies are selected and applied to the project. The rigorous development and application of the plan provided a documented structure for the project and communicated upward through the customer and program organizations the plan for development and downward through the Software organization the roles, resources, responsibilities and development requirements of each organizational components.

Generation of the Software Development Plan was the first step in laying out the requirements for the software project. This level of planning was based on the project and development requirements and defined requirements for each of the categories of design, development, and test.

The plan was the definition of what was to be done during each step in the development, resources to be applied to the project, and a plan for applying them and controls for monitoring the project in the context of the development effectiveness.

Practices and procedures for managing and controlling the development were also defined. Development of the Software Development Plan was the direct responsibility of the software manager although segments of the plan were delegated to other areas of the project for development. The manager was committed to the resultant plan and to implementation of the project requirements specified within.

Once the plans, schedules, budgets and project requirements were clearly defined the manager took an active and effective role in ensuring their implementation. Technical personnel threatened to quit, managers at all levels threatened funding restrictions and career impacts, customer personnel threatened program impacts if their ideas were not incorporated in the project environment. A variety of observers outside the program pointed out "critical shortcomings" and "grave concerns".

The manager kept his focus, however, only changing the project environment when it made sense in the context of the approved, implemented, and documented project environment. As a result the project returned to productivity and, ultimately succeeded.

The nature of the software development process demanded that this early planning be rigorous, complete, and tailored to the development requirements of the project. The complexity of the disciplines essential for development, the non-tangible nature of the early data products and the essential relationship between requirements, design, code and testing data required carefully planned and orchestrated technical and administrative tasks be accomplished.

Project Experience – Success In Spite Of Itself

Because of the disciplines decribed above, the project was successful. The rapid progress made early in the project was rewarded by the acceptance and certification of the system for use. The delivered system contained some 350,000 line of code, of which approximately 75% were rewritten. This system was fully documented, tested, and under configuration control when delivered. This kind of productivity was

possible through the availability of technically competent personnel, their hard work and dedication, and active, aware management.

REFERENCES

1. Boehm, Barry. "Trends in Software", IEEE Transaction on Computers, Vol C – 25 (12), December 1976.

2. Evans, Michael, and Dolkas, James, "Improving Productivity Through Acceptance of Software Methodology with the Project Organization", IKD Congress, Berlin, October 1980.

3. Aron, Joel D. "The Program Development Process", Addison-Wesley, 1974.

4. Metzger, Harold. "Managing the Software Development Project", Hempstead Press, 1975.

5. Evans, Michael, and Dolkas, James. "Defining Tools and Methodology For The Development Environment," Computer and Government Twenty-First Annual Symposium, ACM, June, 1982.

6. Evans, Michael, Piazza, Pamela and Sonnenblick, Paul. "Large Scale Software Development Management Techniques," NAECON, 1982 Dayton, Ohio.

7. Evans, Michael, Piazza, Pamela and Sonnenblick, Paul. "How to Salvage A Faltering Software Project", NSIA Conference, Washington, D.C., October 1981.

8. Evans, Michael, Piazza, Pamela and Dolkas, James. "Principles Of Productive Software Management", John Wiley & Sons, Inc., New York, N.Y. 1983.

9. Evans, Michael, Productive Software Test Management, John Wiley and Sons Inc., New York, N.Y. To be published in 1984.

USAF STRATEGY FOR INTRODUCING ADA INTO MISSION CRITICAL SYSTEMS

MAJ DAVID A. HAMMOND
Deputy Directory for Achitecture and
Language Standardization
Directorate of Computer Resources
DCS/Acquisition Logistics
Andrews AFB DC 20334

ABSTRACT

Air Force Systems Command has developed a four-phased plan for introducing Ada into Air Force Systems, and has established definite criteria for advancing to each successive phase. The objective is to start using Ada as soon as possible, but to do it prudently. The four phases are: laboratory development and exploration; product division parallel operational system development; use of Ada on selected programs; and mandatory use of Ada on nearly all programs. At each phase there are readiness criteria related to compiler maturity, completeness of tool sets, availability of documentation, maintenance, training, and adequacy of the experience base. The criteria can prevent errors of haste that have plagued past Air Force language introductions; they also provide near-term goals for projects in the early phases. The next step in the planning process is to design projects for each phase that will accomplish the goals.

Introduction

The Air Force plans to introduce Ada into its systems deliberately, so as to avoid the errors of haste that have plagued previous language introductions. We found, for example, that our haste in making JOVIAL J73 mandatory caused severe problems for both the people trying to develop it and the systems trying to use it. Programs using J73 compilers that were still under development tended to drive the compiler development, to the detriment of the general user community. The problem was especially severe when two or more programs committed to the same developmental compiler. Then we had changing and conflicting compiler requirements, competition for scarce compiler writing talent, poor configuration control, and general confusion. The situation was equally bad for the using programs. They often had to adjust to late compiler deliveries. In some cases, young compilers produced unacceptably inefficient object code, and compiler writers were too busy to improve them. In other cases, programs had failed to arrange for adequately responsive compiler maintenance support. (Sometimes the support was simply unavailable.) The programs had to put up with schedule slips and increased costs while they waited for critical compiler bugs to be fixed or needed features to be implemented. In the meantime, the operational forces had to go without the capabilities they needed to perform their missions.

Clearly, we cannot afford to use immature languages and compilers on critical programs. On the other hand, if we wait for total maturity before using a language or compiler, it will never achieve maturity. Accordingly, we have developed a four-phased introduction strategy, whereby we apply Ada in situations where we have successively more riding on its success.

The four phases are: 1) laboratory developments and explorations; 2) product division parallel operational system developments; 3) use of Ada on selected operational system development programs; and 4) mandatory use of Ada on nearly all programs.

We have included in the strategy a set of criteria for deciding when we are ready to move on to the next phase. Used conscientiously, the criteria can keep us from biting off more than we can chew in any given effort. They can also serve as a set of near-term objectives for early efforts.

It needs to be emphasized that since many of the criteria are related to specific application areas and the tools to be used, there is no hard and fast cutoff for saying the Air Force as a whole is in one phase or another. We can expect to have some programs in each of the first three phases at any given time.

Phase 1: Laboratory Developments and Explorations

Phase 1 consists of two types of programs at our laboratories: development of Ada compilers and related tools; and explorations of how to use Ada in the areas in which each laboratory specializes.

We have Phase 1 programs of both types at each of our major laboratories: The Air Force Armament Test Laboratory (AFATL) at Eglin AFB Florida; the Air Force Wright Aeronautical Laboratories (AFWAL) at Wright-Patterson AFB Ohio; and the Rome Air Development Center (RADC) at Griffiss AFB New York.

This manuscript has been cleared for publication IAW AFR 190-1.

AFATL has two compiler development programs. In house, they developed a partial implementation of an Ada compiler. This compiler runs on the CDC Cyber, is written in PASCAL, and produces code for the Zilog Z8000 microprocessor. It is a partial implementation in that it accepts many, but not all, of the constructs of the Ada language. Work on this compiler is considered complete. The compiler has successfully been used to develop Ada code for other AFATL programs. AFATL's other Ada compiler is being developed under contract by Florida State University. It also runs on the CDC Cyber, is written in PASCAL, and generates code for the Z8000. Its difference from the in-house compiler is that it is intended to implement the entire Ada language. AFATL considers this compiler to be coming along well. They are attempting to find funding to develop a version of this compiler to generate code for the MIL-STD-1750A microprocessor.

AFATL has proposed an exploratory effort to rewrite the code for the Advanced Medium Range Air to Air Missile (AMRAAM) to determine the best design approach. It is currently being written in J73 to run on a proprietary processor. The rewrite is to be in Ada to run on a MIL-STD-1750A microprocessor.

AFWAL is negotiating a Memorandum of Agreement (MOA) with the Army to develop a MIL-STD-1750A target capability for the Army's Ada Language System (ALS). This effort will give the Air Force a VAX-hosted 1750A-targeted Ada capability.

AFWAL has several exploratory efforts underway. They have agreed with the Army to be a beta test site for the ALS and are participating in the initial installation and checkout of the ALS. As a beta test site, they plan to develop a 1750A simulator in Ada, to support Air Force Institute of Technology thesis work in Ada, and to conduct work on artificial intelligence in Ada. They are participating in the Tri-service KAPSE Interface Team. They are investigating the use of Ada in a distributed multiprocessor avionics system. They recently completed a study on translation of J73 code to Ada, which reached the conclusion that translation is neither desirable nor feasible.

RADC is developing the Ada Integrated Environment, which is a complete Minimal Ada Programming Support Environment, as defined in the DOD Stoneman document. The initial version is IBM 370-hosted and targeted. RADC also intends to develop a 1750A-targeted version. This effort is complementary to, not competitive with, the Army's ALS. Between the two of them, we will have a 1750A target capability on both IBM and VAX hosts. Previous surveys have shown that these are the two most important hosts for the development of embedded software.

RADC has several exploratory efforts going. They recently completed a study of the feasibility of using a federated system of microprocessor Ada development systems tied to a central data base to develop embedded Ada software. The study concluded that this is a reasonable way to develop code for moderate sized programs. RADC is monitoring industry's attempts to use Ada as a program design language. They are watching Air Force funded efforts, such as the Satellite Control Facility's work on the Data System Modernization program, and independent industry funded efforts. RADC is planning to rewrite, in Ada, an existing FORTRAN command and control system, both to evaluate feasibility and to determine the best design approaches.

Phase 2: Product Division Parallel Operational System Developments

One of the most important lessons that we learned from our J73 introduction is that transitioning a new language from the laboratories to the product divisions is much more difficult than we envisioned. Laboratory efforts are often feasibility demonstrations. The labs can afford to work with and develop systems that are finicky and fragile. Lab people are technically oriented and motivated. Some laboratory people can even modify compilers to correct errors. Many are capable of diagnosing compiler errors and devising work arounds. Some of them even actually enjoy that kind of work. Product divisions, on the other hand, are expected to deliver usable supportable products to the operational forces. The products must be robust. Product division people are management oriented and motivated. They expect delivered products to work. They view compiler errors as cost and schedule impacts, not as technical challenges. The same contrasts can be drawn between the contractors the laboratories use and those that the product divisions typically use.

Accordingly, we have concluded that we have to consciously transition Ada from the laboratories to the product divisions; we cannot afford to just let it happen. We need to provide a means for the product divisions and their contractors to develop experience using Ada without putting operational system developments at risk. We have concluded that the best way to do this is to conduct a series of product division parallel operational system development efforts, at least one at each of the product divisions. The idea is to identify system programs that are developing moderate amounts of code in some mature language and fund a parallel development of the same software in Ada.

The exact details for Phase 2 programs have yet to be worked out, but we expect that the guidelines will be roughly the following: 1) the product division computer resource focal points will identify candidate programs and obtain preliminary agreement from the program manager to take on the additional management workload. Headquarters Air Force Systems Command will make the final selection and provide formal tasking; 2) funding for the Ada part of the effort will come from separate program elements that have been created specifically for laboratory to

product division transitions. This will avoid cost impacts to the mainline effort, prevent transfer of funds between efforts, and preserve program manager accountability; 3) the two efforts will start from a common system specification and then go their separate ways. The Ada effort will continue through hardware/software integration and possibly some system test, but will avoid the costs of formal qualification test; 4) the product division will be free to use either the same or different contractors for the two efforts. The development teams, however, will be separate; and 5) the Ada effort will not be schedule-constrained. This will avoid panic mode and will provide time to fix compiler errors in a rational manner.

The criteria for deciding that we are ready to start a Phase 2 effort are a compromise. To obtain maximum experience from Phase 2 efforts we want them to be as similar to real system developments as possible. This implies that the language and tools should be as mature as they need to be for a real system development. But we don't want to wait that long to get started on our first Phase 2 efforts. Accordingly, we have taken a careful look at the requirements for supporting a real system development and have backed them off as much as we can without excessively increasing the risk that we won't get useful experience from our Phase 2 efforts. The criteria we need to retain fall into the general areas of compiler readiness, completeness of tool set, documentation, maintenance support, and estimation statistics.

In the area of compiler readiness, we cannot afford to use a compiler that is hot off the presses. The compiler we use must have previously been used to develop code, most likely in a laboratory. This requirement gives some indication that the compiler's performance and human interfaces are at least usable. We also need the compiler to have compiled a large body of existing Ada code so that we can have some faith in its capacity. The existing code can be the compiler itself, if it is written in Ada, or it can be the Ada validation suite. We do not require totally error-free compilation, but we do require that all discovered errors be documented. Finally, since we are admitting the possibility of known compiler errors, we require Ada Joint Program Office approval for use of the compiler.

In the area of completeness of the tool set, we require a minimum set of tools that are compatible with the compiler. Experience has shown that it is not feasible to develop software without at least a cross-assembler, linking loader, downloader, and either a simulator or target debugging capability, in addition to the compiler.

In the area of documentation we require full documentation of all the tools including the compiler. Full documentation includes disclosure of the tool design, so that developers can devise work arounds when they discover errors in the tools during the system development process. In the interest of time, we are willing to accept contractor format in lieu of full MIL-SPEC format.

Recognizing that there will always be residual errors in the tools, we require that there be a maintenance capability in place. The maintenance capability must cover all the tools, not just the compiler. However, it does not have to be, and most likely will not be, a government capability. Since we do not have a system initial operational capability riding on the completion of the Ada development, we can tolerate a less than quick-reaction capability.

Finally, in the area of estimation statistics, we need some way to determine the scope of the proposed effort. The needed productivity statistics should come from previous laboratory efforts.

The minimum capabilities listed above are certainly less than any prudent program manager would require before committing to a system development. However, we believe that they are sufficient for a parallel development where success of the system does not depend on success of the parallel development. By encouraging parallel developments as soon as we have these minimum capabilities, we can start using Ada several years earlier than would otherwise be the case.

Phase 3: Use of Ada on Selected Programs

Following completion of one or more Phase 2 efforts, we can start to use Ada as the primary development language for new operational systems. The development of these systems is subject to the normal cost, schedule, and technical performance requirements that are imposed on all operational system development programs. Success or failure of the developments directly impacts combat capability and national security. Accordingly, the development language must be carefully chosen to maximize the chances of success. We expect that Ada will be the language of choice for some systems long before it is the language of choice for all systems. We have two ways of identifying these systems: 1) we can allow systems to volunteer and obtain Headquarters AFSC approval to use Ada, and 2) the Headquarters can direct non-volunteer programs to use Ada. Of the two approaches, we much prefer the use of volunteers. We expect that the first programs to volunteer will be small, low-risk programs. This is exactly the kind of program that should be the first to use a new language. After small, low-risk programs successfully use Ada, we can expect larger, higher-risk programs to select it.

Since the success or failure of each Phase 3 effort directly impacts national security, we must satisfy a more rigorous set of readiness criteria before selecting a language. The criteria that we have identified are equivalent to the criteria that any conscientious program manager evaluates when choosing any development language, with suitable adjustments for the fact that Ada is a new language. The criteria we have identified fall into the general areas of compiler maturity, completeness of the tool set, documentation, maintenance support, and training availability.

In the area of compiler maturity, we require that the compiler to be used must have previously been used on a Phase 2 effort in the same application area. (By compiler we mean the precise host/target combination. This is because quality of code generation is extremely important for operational systems. By the same application area, we mean avionics, electronic warfare, command-control-communications-intelligence, training devices, and automatic test equipment. This is because different application areas stress different language features, and because we expect that run time systems will be application area dependent.) The compiler must be fully validated, as defined by the Ada Joint Program Office. The performance characteristics of the compiler and run time support package must be known, including the code and run time expansion ratios. These criteria are equivalent to a program manager's looking for a compiler that has been used successfully before, with due recognition of the fact that some program has to be first.

In the area of completeness of the tool set, we require the same minimum tool set required for Phase 2, plus a symbolic debugger and a measurement capability. Ideally, of course, we should also have additional tools, such as configuration management aids, a library manager, and a prettyprinter.

In the area of documentation, we have to realize that the tools that are used to develop the system will also be used to support it over its entire life cycle. Accordingly, all of the tools must be documented to full MIL-SPEC requirements. Additionally, Air Force Logistics Command must approve the tool set and the associated deliverables.

In the area of maintenance support, there must be a quick reaction maintenance capability in place for the compiler and all of the other tools. This is because every day loss caused by tool problems translates directly into a delay in system operational capability. (This assumes that software is on the system critical path, as it usually is.)

Finally, in the area of training availability, the contractor must have some method for training his people in designing, coding, and testing in Ada. Training is required for the experienced programmers who have to adjust to Ada's new way of doing business, and for the new programmers hired off the street.

Satisfaction of the above criteria will not eliminate all risks, nor even reduce them below the risks associated with using a mature language. It will, however, ensure that no unnecessary risks are being taken.

Phase 4: Mandatory Use of Ada

The Air Force (and Department of Defense) goal is to use Ada for virtually all system developments. Accordingly, we plan to make use of Ada mandatory at some point in its life. We will do that by changing the appropriate regulations. Of course, not even Ada will be suitable for all programs. Recognizing this, we will also have to develop waiver criteria and set up a process for obtaining waivers.

Making any language mandatory is a big step. It is effectively taking a technical decision out of the hands of the engineers and putting it in the hands of the regulation writers. Effectively, it is deciding a priori that nearly all programs satisfy the criteria for Phase 3. Accordingly, we must be sure we are ready before we do it. Our criteria for deciding we are ready to fall into the areas of compiler maturity and program management.

In the area of compiler maturity, there must exist sufficient compiler host/target combinations to cover the majority of potential applications. These compilers must have previously been used on operational system development programs, and must be supported by quick reaction maintenance.

In the area of program management, we must have acquired sufficient productivity information on previous programs to be able to estimate cost and schedule for future programs. We must also have the management experience to be comfortable with the adjusted milestones that use of Ada is expected to cause.

Where Are We Now?

The foregoing strategy was jointly developed over a several month period by members of the Ada Joint Program Office, Air Staff, Headquarters AFSC, the Embedded Computer Standardization Program Office, and the Computer Resource Focal Points. It has been briefed to the JOVIAL-Ada Users Group, MIL-STD-1750 Users Group, industry meetings, the Second Air Force Systems Command Standardization Conference, and various Air Force Working Groups. To date, no significant disagreement has surfaced.

The next step is to flesh the strategy out into a full-blown plan. Electronic Systems Division is now writing the plan under the 64740F computer resources management program. When they complete it, it will get a thorough Headquarters AFSC review and will then get approval from the other product divisions, the various major commands (especially Air Force Logistics Command), and the Air Staff. After it becomes a formal Air Force plan, it will go to the Ada Joint Program Office for incorporation into the Tri-service Ada Program Management Plan.

As mentioned above, the plan will contain the detailed guidelines for selecting and managing Phase 2 efforts. As soon as the guidelines are firm, we can start actively looking for candidate Phase 2 efforts and identifying sources of funding. Then we will be ready to start some Phase 2 efforts as soon as the criteria are met.

Impact of Ada on the Software Life Cycle

By
Susan B. Mickel

ABSTRACT

The introduction of the Ada programming language will affect every aspect of software development and every phase of the software life cycle. It is the intent of this paper to examine some of the issues involved in the introduction of this language and explore some ways of addressing them in a beneficial manner.

1.0 INTRODUCTION

The Ada programming language was designed to support modern software engineering principles in order to produce software that is

- o modifiable
- o efficient
- o reliable
- o understandable.

In designing this language, it was recognized that a programming language is a very powerful tool in the software development process. However, it is just one of many tools. Further, the effectiveness of any tool is limited by the way it is applied. As the well-known maxim says, "The programming language does not exist in which it is the least bit difficult to write bad code". It is the intent of this paper to examine some of the software engineering, management and training issues associated with the introduction of the Ada programming language and to recommend ways of addressing them.

2.0 THE LIFE CYCLE MODEL

A typical software life cycle model is represented in Figure 1. As a model it is an abstraction of the actual life cycle process, with distracting details eliminated. It is an ideal that facilitates discussion and analysis of the software development process. It represents little of the complexities of a real software development project. In the real world, some or all of these activities occur simultaneously and cyclically. New requirements can be introduced during any phase of the life cycle; testing can expose problems that cause re-design, etc. The model, after all, represents the software life cycle, not the project life cycle. In spite of this, the life cycle model forms the basis for

- o cost estimation and scheduling algorithms
- o staffing profiles
- o management practices
- o military standard review procedures and documentation.

Lines of code are not the only products required of a

software development project. Intermediate products and review points are necessary to assure the quality and maintainability of the software produced and the satisfaction of all system requirements. Especially on large software projects, a great deal of formalism is required. This formalism is captured in military standards which specify the products (deliverables) and reviews associated with each phase of the life cycle. Figure 2 illustrates the products and review points currently required on a DoD embedded system development project. These deliverables in turn drive the project schedule, staffing profiles, and the automated tools to support their production. Thus, we have a tightly integrated system all based upon the life cycle model and all intended to produce an acceptable end product. The Ada programming language will have a profound effect upon the life cycle model and, by implication, upon all the items discussed thus far.

3.0 ISSUES

In this section, the issues involved in transitioning to Ada are discussed within the framework provided by the life cycle model. It is recognized that these issues are closely interrelated and interdependent. For organizational clarity, however, they will be divided into several broad areas of discussion.

3.1 Life Cycle Phase Transitions

The transition from one phase of the software life cycle to another has traditionally been characterized by context switching. During the requirements analysis phase, a requirements language like SREM or PSL/PSA is used. During design, a program design language (PDL) is used. During implementation, a programming language is used. Traceability from phase to phase is difficult and costly. In addition, the transition from one language to another is likely to introduce errors. Different automated support tools must also be built and maintained for each phase.

With the introduction of Ada, that scenario could become a thing of the past. One of the advantages of Ada is that as a language it contains features that make it feasible for use in requirements analysis and design as well as programming. The advantages are obvious. Traceability is improved. Errors introduced through context switching are eliminated. Separate support tools are no longer needed. However, there are difficulties.

Together with the lack of context switching comes a blurring of the phases of the life cycle. At what point does one decide that requirements analysis is complete and design beginning? Where does detailed design leave off and coding begin? The use of a common language, while having immense advantages, also is difficult to control. How does one prevent the writing of detailed code during the requirements analysis phase? How does one communicate the design at design reviews? What are the appropriate review points?

These questions can only be answered by a careful study of how the life cycle phases are

changed by Ada. The deliverables must be natural by-products of the development process. Thus, changes to the life cycle model necessitate changes to the deliverables which necessitates modifications to the military standard specifications. The recommendation is, therefore, for a general and extensive overhaul of the military standard specifications, types and forms to conform to a new software life cycle model.

3.2 Design Methodology Issues

Over the last 20 or more years of coding military applications in FORTRAN, Jovial and assembly languages, the structured programming methodology of software development has been widely applied to the development of embedded systems. Functional decomposition, design and code walkthroughs, and other structured techniques have become standard operating procedure for defense contractors. If that methodology is applied to the development of Ada code, little of the expected benefits of Ada would be realized. For example, structured programming techniques were never intended to design packages to maximize information hiding. There is no support for the definition and management of data abstractions. The Ada language itself cannot force the application of good software engineering techniques like information hiding. It merely permits their application.

Thus, new methodologies are needed that support the design principles embodied in the Ada language. Methodologies are being proposed, such as the object-oriented methodology [1], which appear promising. However, most of these methodologies are still merely concepts. Much work is needed to validate and evaluate them. Even more work is required to make them manageable, measurable procedures that can be applied in a disciplined manner to large software development projects involving hundreds of software professionals.

The difficulty in selecting a design methodology for Ada is that, ultimately, a large amount of experience with the language itself is required to intelligently evaluate it. Many systems were written in FORTRAN as structured programming evolved. Also, the problems in transitioning to Ada will influence the selection of methodologies. For example, one technique may appear superior simply because designers learning Ada for the first time find it easy to understand and apply. Experienced Ada designers might find it inefficient and error prone.

In the meantime, something must be done. Recognizing that the best methodology to use with Ada is some years away, advances can still be made. One approach is to evaluate promising new methodologies on realistic software development projects. A "Methodman" document [2] has been published that lists criteria for the evaluation of such methodologies. Projects that are jointly funded by government and industry are preferred, staffed by experienced systems developers from industry. The benefits of this arrangement are

- Technology transfer by means of the training of experienced software professionals in Ada,

- Evaluation by those who have actually built large systems and understand the management problems as well as the technical problems,

- The exposure of problems in transitioning to Ada experienced by software development staffs.

One of the first controversies surrounding the introduction of the Ada language concerned the use of Ada as a PDL. While this seems a valid point of discussion, it is really begging the question. A PDL is simply a tool for program design. For systems developed in first generation languages like FORTRAN, PDL was a tool to assist the developers in transitioning from a textual description of the requirements to the low level, tightly specified code. One sentence in the requirements document might result in thousands of lines of code.

Ada is a much higher level language than FORTRAN and other languages currently used to develop embedded systems. It can be used in the requirements analysis and design stages to avoid context switching. The issue is the process by which to apply Ada to those activities and not whether compilable Ada or a PDL should be used. In any case, the latter issue mainly affects the tools that will be built to support the design process.

3.3 Software Reusability Issues

The packaging and multi-tasking facilities of the Ada language provide support for the definition of what is colloquially called "software chips". The reusability of code would save the government millions of dollars a year in new system development, increase the reliability of new systems through the incorporation of large portions of validated software, and reduce the shortfall of software professionals. As such, it is a high priority goal of the DoD and an expected result of the introduction of Ada.

There are difficulties, however, in implementing software reusability on a large scale. The reuse of code developed by one contractor by another contractor may raise licensing or infringement issues.

There are technical difficulties, also. First, in order to be able to define packages for general use, it is necessary to develop a standard abstract view of the capability to be supplied. Take as an example a matrix manipulation package. It is easy to define such a package since the abstract object matrix is well defined as are the operations to be performed upon it. It is a much more complex task to abstract an orbit determination system or a satellite command generation system. In many cases, there is no standard nomenclature from customer to customer or from contractor to contractor. The problem is similar to those faced by the developers of the so-called expert systems. One must be able to abstract the knowledge about a given task, define a standard nomenclature, and to abstract objects, types and operations.

Second, assuming that libraries

of reusable software are available, some mechanism is required to locate and retrieve the proper packages. It must be easy and quick for potential users to identify the needed packages and sufficient information made available to easily and quickly determine its applicability. Many methods of retrieval will be required, as for example, by types, objects and/or operations. Again, a standard nomenclature would be helpful in efficient retrieval algorithms.

Finally, programmers by their nature are likely to develop their own routine if there is one minor difference between what they want to do and what a pre-existing, tested routine does. They also tend to believe that they have a better, faster, or more elegant algorithm and so resist reusing code - the "Not Invented Here" philosophy. Management practices are therefore necessary to encourage and enforce the use of package libraries.

One method of overcoming these difficulties is to organize several projects to define generic type systems. Typical embedded systems or subsystems of moderate size should be selected for the broadest applicability, such as an orbit determination system for a satellite. In the first step, a standard nomenclature for the application is defined. This necessitates the participation of experts in the particular application area. Then abstract types, objects and operations are defined. Finally, these items are organized into the specifications for reusable packages. Several of these projects operating in parallel would increase the reliability of the results. The developers of one system can review the packages from another system, etc. This is another area where industry must participate for the reasons cited in Section 3.2.

3.4 Training Issues

It is already realized that Ada is not just another programming language that can be learned by examining the language reference manual. Training in the design principles that Ada was developed to support is mandatory. In addition, training will be necessary for every professional involved in the development of systems in Ada. This includes system engineers, designers, coders, testers, quality assurance personnel, configuration managers, project managers, etc. As Ada will affect every phase of the life cycle, so, too, it will affect everyone involved in every phase. This generality of training is required if the desired benefits of software reusability and maintainability are to be achieved.

Ada cannot be learned by studying individual features. The way in which the language features interact is one of the most difficult and important aspects of the language. One of the most effective methods of teaching this type of concept is by example. The development of a system of graduated examples is therefore recommended. Simple examples would involve only one feature of the language, such as a generic subprogram. More advanced examples would involve the interaction of several features. The reusability paradigms of Section 3.3 would

contribute greatly. Further, all examples would be written to demonstrate good Ada programming style.

4.0 SUMMARY

Many recommendations have been made in the preceding sections. Briefly, they are:

- Revision of the military standard specifications, types, and forms
- Evaluation of methodologies for designing Ada software
- Modification of software acquisition practices to promote reusability
- Development of standard nomenclature for embedded systems and development of programming paradigms of reusability
- Development of training courses using graduated examples

All of these recommendations are made for the near term, i.e., the next 1-3 years. During that period, our knowledge concerning Ada will increase exponentially. Therefore, it is infeasible now to propose what should be done in the long term. An iterative approach is recommended, in which a reassessment is made periodically, new issues recognized, and adjustments made accordingly.

5.0 CONCLUSION

Many of the issues raised in this paper are not peculiar to Ada but are general software development issues that have existed for some time. Ada alone cannot solve all the problems and complexities associated with the software development process. Disciplined software engineering methods applied consistently throughout the life cycle are required. However, to the extent that Ada does provide powerful support for these methods, it has great potential for improving the current practice. Regardless of other benefits, the expected introduction of this language is prompting an examination of software development issues and experimentation in innovative solutions. As D. A. Fisher has stated in reference to Ada, "the major payoffs will come from better programming methods and techniques, more software commonality and more useful and easily accessible software tools and aids" [3].

REFERENCES

1. G. Booch, "Software Engineering with Ada", Benjamin/Cummings Publishing Company, 1983.

2. R. Abbott, "Ada Programming Style Guide", 1982.

3. D. A. Fisher, "A Common Programming Language for the Department of Defense - Background and Technical Requirements", Institute of Defense Analysis, Report P-1191 (June 1976): 6.

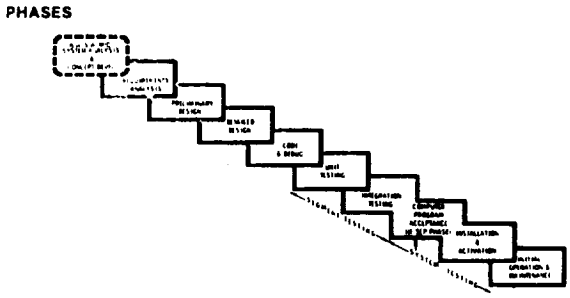


Figure 1. Life Cycle Phases

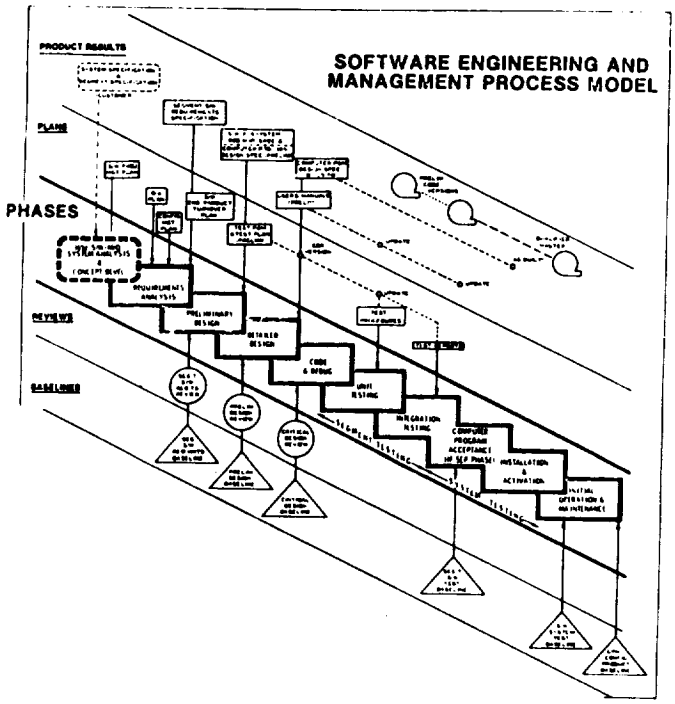


Figure 2. Products and Reviews

TRANSITIONING TO ADA*: EXPLOITING A PILOT DEVELOPMENT

John Bailey
General Electric Company
Arlington, Virginia

ABSTRACT

The effective transition to the use of Ada implies much more than learning the syntax of a new programming language. Ada supports the application of many advanced principles of software development. However, these principles are not automatically associated with the use of Ada. What is required is an approach to learning the language which is integrated with a disciplined approach to the application of a well-conceived methodology. Through our experiences we have seen that an effective way to prepare a software development organization for the proper use of Ada is to conduct a pilot development. Careful monitoring and tracking of such a development will provide an organization with valuable insight into the actions and steps required to facilitate subsequent uses of the language at that location.

Introduction

Ada should be viewed as a vehicle for bringing both known and suspected beneficial techniques to the industry of software development. Throughout this paper, the proper use of Ada is meant to imply the successful incorporation into the software life cycle of those techniques which Ada was intended to facilitate. The author recognizes that Ada is not required for the implementation of a new methodology; indeed, the techniques predate the language. However, from a typical industry point of view, Ada represents an impetus for both a reassessment of the current status and an evolution to new and hopefully better software development practices.

General Electric's Space Systems Division desired to gain experience with the Ada life cycle before Ada is required for the development of large and critical systems. A pilot development was conducted using Ada to provide this experience. This paper has been motivated by the results of a study to monitor this pilot development and to ultimately recommend metrics which are germane to the Ada life cycle.

This paper is divided into four sections. The first section is a brief description of the Ada development project. This is followed by a summary of the most interesting questions raised by this effort. The third section is the longest and most important. It elaborates upon the set of insights and discoveries which resulted from our study of the development effort. The last section covers some issues which will require additional future study.

The Pilot Project

In 1982, General Electric's Space Systems Division conducted a redevelopment in Ada of a subset of an existing software system. The system was originally written in FORTRAN. The part which was selected for redevelopment in Ada was able to stand alone and included real time processing, batch processing and data base manipulations. Although the size of the original FORTRAN which performed these tasks is difficult to estimate due to the interdependence of the system, a rough estimate would be in the range of six thousand lines. The team which performed the redevelopment in Ada was not associated with the original development and had not been previously exposed to any parts of the system or its documentation. The team was comprised of a lead programmer who had substantial experience in the application area, a second programmer who had exposure to a variety of application areas and languages, a third programmer who had just graduated from college but who had no industry experience, and a librarian. They worked entirely from the Part 1 software requirements document with input from the original developers as required to resolve ambiguities.

New requirements were derived, primarily by the lead programmer, which were taken from the original requirements document but which allowed the subset to stand alone. The development environment was a VAX 11-780 running VMS. The only automated software development tools used by the team were a screen editor and the most recent version of the NYU Ada/Ed compiler. The decision to avoid keyword-prompting editors (which were available) and other aids to the design and code process was made to minimize the chance of bias imparted to the error data which we were collecting on this initial use of the language.

The generation of the new requirements was interrupted for about one month to provide training in Ada as well as an introduction to several software development techniques. The bulk of the training was based on short

*Ada is a trademark of the United States Department of Defense.

PAGE 15 INTENTIONALLY BLANK

programming examples which illustrated each of the features of the language. The techniques covered as an introduction to development methodology were design and code walkthroughs, the use of PDL, formal test plans, data abstraction, top down design, stepwise refinement, and the use of a librarian. Approximately one day was spent describing these topics.

After about one third of the overall time allotted to the development had elapsed, a set of requirements was formally reviewed and baselined. In many ways this document paralleled the original document but it was more complete and was written to be more comprehensible to a programmer who had no previous experience in the application area. The requirements received two more updates during the remainder of the development, each incorporating all approved changes since the last release. Most of these changes were to make the document internally consistent, more complete, or clearer.

After the formal requirements review, the development effort shifted to the generation of design documents. The work assignments were determined by the lead programmer and the designs were expressed in an Ada-like syntax (basically Ada but with optional embedded English). Although our initial goal was to have the team use Ada as a PDL, we soon learned that such a directive without the proper framework amounts to a license to generate code directly from the requirements. To avoid this, large parts of the system were designed at a sufficiently high level to be presented at a one or two-hour walkthrough.

After walkthroughs had been completed on the high-level design for each major subsystem, a more detailed design was developed. This used the same syntax, with the optional embedded English, but was sufficiently low-level to be used as the input for a coding assignment. These documents were also presented at walkthroughs. A formal design review was conducted in October of 1982, after about two-thirds of the development time. This review was attended by the cheif designer of the original implementation. At this review the results were judged reasonable and viable. However, we were surprised to learn that the current design quite closely paralleled the design of the original FORTRAN implementation. This occurred in spite of the fact that the only information common to the two developments was that contained in the software requirements document. Although we were initially perplexed about this, we have come to understand much more about how Ada should be used as a result of this revelation. This will be discussed in the following section.

Coding was performed for the last three months of the development and an attempt was made to compile the code with the NYU compiler. Due to the large size of the system, however, the programming team was unsuccessful in managing to keep more than a small part of the system in the library at any one time. For this reason, hope of execution of the system before the end of the development was abandoned. The compiler was used only to syntax-check the individual units of code. An attempt was made to continue as though the system were to be delivered at the end of development but the lack of machine assistance and feedback was clearly disappointing for the team.

Questions Raised

The foregoing is a brief description of the redevelopment in Ada of a modest piece of software which was representative of the type of work done by the Space Systems Division in the General Electric Company. Although our initial goals were to investigate the impact of Ada on the software development life cycle, we discovered, perhaps not surprisingly in retrospect, that an Ada development can take on whatever characteristics one imposes upon it. During the study of the development, many questions were raised about the nature of the cause and effect relationships which existed during the development effort. For example, we questioned the conventional definitions of the phases and the expected effort per phase of a software development. Also, we questioned what factors were present in our development which resulted in a conventional FORTRAN design rather than a design which might have used Ada more appropriately. Furthermore, we wanted to know how to measure the difference between such styles. Finally, we wanted to better understand the meaning of the use of Ada as a PDL on an Ada application project. These questions will be described in more detail below followed by a summary of the resulting insights in the third section.

Phase Definitions

Although it is not necessarily easy to admit, it is entirely possible that the amount of time spent developing the requirements document was driven by the overall allotment of time for the project and not necessarily for the purpose of reaching any identifiable point in the analysis. It is probably reasonable to assume that the time spent developing requirements (or producing design) was driven by what programmers are accustomed to rather than by the use of Ada. Since this conventional approach may not be optimal, however, we cannot state that there is not a more reasonable or better division of the overall effort required when developing an application program in Ada.

A corresponding argument, however, is to assert that there is no apparent reason why there should be any difference from the conventional breakdown of effort when developing with Ada. Although this may be true, we believe that experience with the Ada life cycle will suggest not only spending more time in earlier phases of a project but possibly also a revision of the goals (deliverables and outputs) for those phases.

Factors Influencing Design

Around the time of our design review, it became clear that one of the major factors driving the modularization of the design was the modularization of the requirements document. The requirements document was organized around three major functional capabilities, as was the original document for the FORTRAN implementation. In general, these capabilities were the setup and execution of a major batch-oriented process, the graphical display of the results of this process, and the monitoring of the information carried by a real-time data stream. In the design chosen by the programmers on this project, three principal Ada packages were used, each providing a collection of related subprograms and each performing one of the major functions. In this way, packages were used to define little more than the software boundaries within the system which were taken from the organization of the requirements. Other packages were designed to handle specific processing requirements such as a library of math routines and a library of plotting routines.

It was emphasized by the programming team that this organization for the design simplified the process of reviewing the design for correctness and completeness since it closely paralleled the requirements document. This simplified requirements traceability was welcomed as a way of streamlining the design walkthroughs.

We also suspect that the background of the programmers and the training and exposure to Ada provided as part of this project substantially influenced the design of the system. Although the programming team came from diverse application areas and had varying amounts of experience, the language most familiar to the team as a whole was FORTRAN. As part of the training provided on this project, an education was provided in the features and semantics of the Ada language. The elegance of the use of Ada to solve many classical programming problems, such as readers-writers, the dining philosophers, eight queens, and others, was illustrated. A sizeable (500-line) class assignment to construct an interpreter for arithmetic expressions was completed and executed with the NYU compiler. The topics of data abstraction and virtual machine programming were also presented. However, the most complicated illustration of the use of a package to implement an abstract data type was a pushdown stack.

Ada As Its Own PDL

As was explained, our goal was to use Ada as a PDL for the design phase of this project. That sounded like a good idea. We soon realized, however, that we provided insufficient direction to accomplish this goal. We were unprepared for the level of understanding about the design process that is required to implement this outwardly simple idea. Undirected, we discovered that the use of the programming language Ada as a PDL was tantamount to skipping the design phase altogether. Our minimal understanding of the purpose of a PDL, however, told us that at least it should be useful to express successive levels of algorithmic abstraction.

The use of Ada as a PDL was controlled by calling for the expression of large parts of the system in a mix of Ada and English syntax. Embedding English within the Ada used to document a design was found necessary to simplify the expression of those processes or conditions which could not be easily identified at a given point. This allowed the system to be expressed at a high level without unwanted detail. (Technically, embedded English was allowed to replace any non-terminal in the Ada grammar. This controlled use of English simplified the automatic processing of these documents since the grammar is identical to Ada's except for the additional production from every non-terminal which allows a terminal English string. Set brackets were used as delimeters of those strings to further simplify their recognition.)

English used in this way was for different reasons than that used in comments. Comments explained, illustrated, or clarified while the Ada together with the embedded English formed the actual design language. This approach has been the source of considerable discussion and debate ever since it was first proposed as a solution to the "straight-to-code" problem. It provides some interesting insights into what we expected of a PDL and what design strategy was being followed. This will be discussed more fully in the following section.

Two levels of design were used with the second using the same basic syntax but expanding upon the first, and leaving only rudimentary processing unspecified. Finally, Ada code was produced using this detailed design. The choice to use two levels of design was rather arbitrary. Our goals for the design phase were never quantifiable. We were only aware that design is necessary and that Ada has been heralded by some as the ideal design medium. The programmers found the development of the second level of design tedious to produce and even more tedious to review. Perhaps not surprisingly, two programmers later admitted to sketching out implementations in Ada of the key aspects of the system and only submitting design documentation when they were satisified that the chosen approach was viable. The use of the implementation language as the design language made the documentation produced as a result of the design phase seem unnecessary. It was argued that maintainers would surely look at the code, especially if it were well styled and commented, and avoid external design documentation which is notoriously incorrect. We were already taking great pains to keep the requirements document consistent with the implemented system. Why, therefore, was it necessary to keep three versions of the system (requirements, design, and code) up to date?

During the development, the difficulties we encountered using a PDL and the apparent conflict between the use of Ada as a PDL and our approach to designing was puzzling. We were willing to believe that Ada simply could not be used as a PDL without extending it with English as we did. Even with this extension, we seemed to be asking programmers to work in an unnatural way. We attended workshops and conferences to speak of this and other problems and received substantial empathy from fellow Ada pioneers.

Resulting Insights

Toward the end of the development, answers to several of our questions started to surface. The answers to why we had trouble using Ada as a PDL and why the resulting system was structurally very similar to the existing FORTRAN system were discovered to be interrelated. The questions about phase definitions, training, and measurement are not yet solved but we have a better perspective on these issues and are currently persuing the answers through further research. Some speculations based on our current feelings about these latter topics are noted in the final section.

A Conventional Approach to PDL

As primarily FORTRAN programmers, experienced in designing with algorithmic abstraction, our programming team produced successive functional decompositions of the requirements culminating in the Ada code. A PDL which supports these levels of system description needs to be fundamentally algorithmic, with the ability to show various degrees of processing detail. The features of Ada which support this are procedures, functions, tasks, exceptions, and the control statements. Packaging is useful for collecting similar processing into one major module and for locating definitions which will be common to large parts of the system.

To use Ada in a way which supports such a functional design implies the ability to insert stubs for any processing which is too detailed for inclusion in a given piece of design. Stubs may also be required to describe certain conditions which might exist during the processing and which can affect the flow of control. Since it was not always convenient to select identifiers which represented processing or conditions, which would have been necessary to compile these design documents, we allowed descriptive English at these points, as previously explained. This allowed the ability either to process the design or to convert it into correct Ada without restricting the expressiveness of the design language.

In one sense, then, our views of the desig. process as one of algorithmic abstraction drove the selection of the PDL syntax. The PDL, in turn, reinforced a functional decomposition familiar to FORTRAN programmers. This has been cited as a main reason that the resulting system was more similar than dissimilar to the original FORTRAN implementation.

An Alternative

What we failed to realize until much later was that there is a fundamentally different way of decomposing a system which does not rely on algorithmic abstraction. Basically, this method identifies hardware objects, data structures, or resources in the system and isolates the characteristics of each inside a module. The system becomes a collection of these modules which is commanded by comparatively simple main processing steps. This style of designing has been associated with greater modifiability since a change to one of the objects or data structures will be localized to one module while a change to the top-level behavior of the system is localized to the commanding unit. Also, since other systems will likely share some of the hardware objects or even the data structures, a large part of the software could be reused in other, related systems.

The PDL which supports this style of design, however, is fundamentally different than one which supports algorithmic abstraction. This design strategy can be directly supported with the Ada package. The system becomes a collection of Ada packages which is commanded by a main procedural unit. The decisions about what to package and at what level of detail or abstraction (which are opposite concepts) become the most important ones made during design. The Ada package specification is the vehicle for expressing these decisions. A package specification is separately compilable and a collection of them can be checked for consistency. Prototypes of the processing units can be generated to verify the correctness of the approach. This whole process produces compilable Ada which is exactly the same as a portion of the future source code. Nevertheless, it is design. Here was the paradox which confused us. With this style of designing, there is superficially no difference between design documentation and portions of the Ada code. In fact, the benefits of the use of an Ada PDL for an Ada development may be a moot issue: PDL of the nature described here will automatically be developed as a system is decomposed.

Because of their ability to partition a system into independent units, Ada packages are also expected to provide a mechanism to establish libraries of reusable software components. This implies a dual goal of both designing systems with the expectation that parts of them will be general enough to be reused as well as establishing design techniques which consider the currently available packages as components in any new system. Clearly, we had no concept of designing with components. Most of the productivity gains theorized to be associated with Ada are based on this idea that a package which isolates a hardware object, data structure, or resource can be written once and used many

times. A strategy behind realizing this goal involves anticipating how to design a system such that the components make sense individually and independently. Ultimately, it should be possible for a software system designer to adopt a strategy to make use of pre-existing packages much the same way that a hardware designer makes use of available products and components. This concept of designing from components is not new. It has recently been realized through the Smalltalk operating system and language of Xerox and is commonly known as object-oriented programming [1]. The mechanism with which Ada allows programming from components is somewhat different, but the underlying concepts are similar.

The main conclusion to be drawn from this description is that the nature of the design strategy will drive the selection of the style of any PDL used. We were predisposed to a certain style of designing and we discovered a syntax for a PDL to support that style. We also had certain beliefs about why a PDL should be used and of what help the resulting documentation should be. For example, among the benefits of a PDL, we cited it as a programming aid and as maintenance documentation. When it became clear that a maintenance document was not detailed enough to serve as a coding assignment, we asked for a second, more detailed level of design. Perhaps if we had undertaken the project with a better understanding of some of these issues we would not have experienced many of these problems. As a result of this project, however, it is hoped that such understanding will be present in our future Ada developments.

A Conventional Approach to Requirements

In addition to the foregoing explanation as one reason that our Ada implementation looked so much like the original FORTRAN one, we also suggest that the software requirements document influenced the design chosen. The requirements document actually contained a substantial amount of design detail which expressed internal aspects of the software and not only its outward affect. Not only was this considered typical and acceptable, this information was used as a point of departure for further design. Also, the desire to be able to trace individual requirements between the design and the requirements document led to a fairly strict adherence to the organization of the requirements document during design and implementation.

Something which was not clear at the onset of the development was that the software requirements document (called a Part I specification, in this case) was intended to convey high-level design information and essential processing algorithms. Specifically, the batch processing was described in one major section of the document along with the algorithms to be used to accomplish the computations. The display of the results of this processing was the topic of another major section. The FORTRAN implementation was organized around this subdivision, and we understand that this was the intent of making that division in the requirements. A more appropriate use of Ada to implement these functions was more recently described to us by two consultants from General Electric's Corporate Research and Development. Their view was that these two major functions actually operated on the same underlying hardware objects and so should be packaged together while the specific tasks required of these objects for this application should be separate. This enables the reuse of the hardware-oriented package by keeping it sufficiently general.

Alternatives

There seem to be several possible solutions to the conflict between the conventional style of expressing requirements and the proper use of Ada to find a software solution. If it will be the case in future Ada developments that the requirements will continue to be organized in a fashion which suits a functional description of the system, it will be necessary either to accept the design decisions made in these specifications or to learn how to read the documents in a way which separates essential features of the system from the organization chosen to describe the system. If the former path is chosen, reusability, maintainability, and reliability will probably not improve over the current situation and, beyond portability, there seem to be few remaining advantages of using Ada. An exception to this might be where the writer of the requirements understands the impact of the decisions conveyed by this phase and ensures that they are compatible with a proper Ada design. This means, however, that the writer of this documentation must have as much ability to properly design for Ada as any other designer on the project.

Alternatively, the requirements for Ada developments could be written in a way which clearly distinguishes between the external description of the desired system, including hardware constraints, and the internal features of the processing, such as algorithms. It may be the case that what is typically considered to be the complete requirements will diminish in size while some additional documentation will appear which becomes an appendix to the true requirements. This approach is currently being studied by us as a result of our experience with this redevelopment.

Toward the Future

In addition to the insights gained into the influence of the style of the requirements document and the approach to system design, general issues about training, development milestones, and measurement became apparent through our case study.

Training

Perhaps the most important aspect which contributes to the nature of a development is the

experience and training of the individuals involved [2]. As noted, we provided a thorough training in the syntax and semantics of the Ada language. However, few useful examples were provided in the techniques to derive the alleged benefits of the Ada language, such as reusability, maintainability, and reliability. Ada is a sufficiently general language to be used in almost any fashion with which a programmer feels comfortable. This may be a detriment in that business as usual is obviously not acceptable to our software industry. Ada and the associated STARS effort have been sponsored by the DoD to improve upon the status quo. Therefore, our training will need to include the correct use of Ada and not just facility with it, a distinction which is only subtle on the surface.

In retrospect, we observe that, although it would have taken considerable preparation and class time to illustrate more complex and realistic examples, without such examples we could not expect a successful transfer of the principles of programming with packages. The extrapolation of these principles from academic examples into actual complex examples simply cannot be performed in a vacuum.

Clearly, this type of training cannot be designed in a short time. There is currently insufficient experience in the use of the language to design the optimal course. Nevertheless, several key principles to software development are now known (such as organizing around objects, data structures, and resources) which are language-independent and can serve as a beginning. It is interesting to note that one of the foremost examples of the use of these preferred principles of software development, the A-7E Operational Flight Program which is being redeveloped by Kathryn Heninger Britton, David Parnas, and others at the Naval Research Laboratory, is being implemented in assembly language [3].

A Shift in the Conventional Milestones

Earlier it was asserted that a change in the manner of estimating the life cycle phases for Ada will have to evolve. For example, we have noticed that a proper design strategy can result in measureable progress toward what is commonly considered code development. Further, the use of reusable components will drastically reduce the code development to only the driver modules and any new modules required. When reusable modules can be automatically verified against a set of assertions upon which the rest of the system relies, testing effort will also be reduced. Finally, at the other end of the life cycle, we believe that the establishment of precise requirements before proceeding to design will be particularly important as the later phases become more automated. Notice this does not require that the requirements cannot change; designing for anticipated change is one of the single most important concepts in design [3]. Nevertheless, the ability to evolve the requirements for a system as the feasibility of various implementation strategies is investigated will probably be absent from future developments.

What this means is that the general shift of effort from implementation phases into the earlier phases of requirements and design, which has been observed over the history of software development methodology, will continue. To compensate for this in terms of milestone planning and deliverables, new milestones will probably have to be inserted into the requirements and design phases. Without such visibility into a development, a customer is not likely to authorize, say, half of the allotted funds for the development of the requirements when conventional wisdom indicates this should take only ten to twenty percent of a project's resources [4]. If, however, case studies and pilot projects can demonstrate the value of such an approach, we may soon see half or more of the time for a development being absorbed by what we now know as the single phase of requirements definition, with perhaps a third of the time being allotted to design. The actual implementation and test may be quite fast, since the proper division of work assignments may make it possible to employ many additional individuals for these phases.

Comparative Measurement

In was mentioned in the second section that we wanted to know how to measure the difference between the products of a conventionally-styled development and one which uses Ada more appropriately. Unfortunately, this becomes a chicken-and-egg paradox where we do not know how to measure the difference until we can define it better but it is difficult to define the difference without tangible measures. What is required are measures which directly reflect the distance from each proposed goal of the methodology. For example, if one desires to measure the independence of modules of a system, variations of the ideas of cohesion and coupling [5] or segment-global usage pairs [6] might be revealing. However, direct measures of the reusability or modifiability of a module might be quite hard to develop. We are currently in the process of identifying that which we believe makes sense to measure and compare among Ada developments. The identification of actual metrics is yet a subsequent step.

Summary

It is felt that the experience gained through the study of a pilot Ada development is quite worth the expense. Our goal is to attempt to develop a consistent and integrated approach to software development with Ada. We feel that the insights to accomplish this only come through trial experiences such as the one described in this paper. In lieu of organizing a complete laboratory-style development such as the one described here, it would be reasonable to expect that the same benefits could be derived from the careful examination of "live" Ada developments. A software development organization could take

advantage of internal software developments or developments of a less critical nature to develop insights into the impact of Ada on the software life cycle within that environment. Only given those insights can an organization develop a strategy for an effective transition to the proper use of Ada.

ACKNOWLEDGEMENTS

This paper is only a snapshot of one transition to Ada. A useful understanding of such a process is only possible through the combined efforts and observations of many individuals. The thoughts covered in this paper were the result of countless discussions with Thomas M. Chapman and Elizabeth Kruesi who were involved in the pilot Ada development and the study of it, respectively.

This research program is monitored by the Office of Naval Research (ONR) under contract #N00014-82-K-0225 to the University of Maryland with funding from ONR and the Ada Joint Program Office, however, the views expressed are not necessarily those of the Office of Naval Research, Ada Joint Program Office or the Department of Defense.

REFERENCES

[1] D. Ingalls. "Design Principles Behind Smalltalk," BYTE, August 1981, pp. 186-198.

[2] B. Boehm. Software Engineering Economics, Prentice-Hall, 1981.

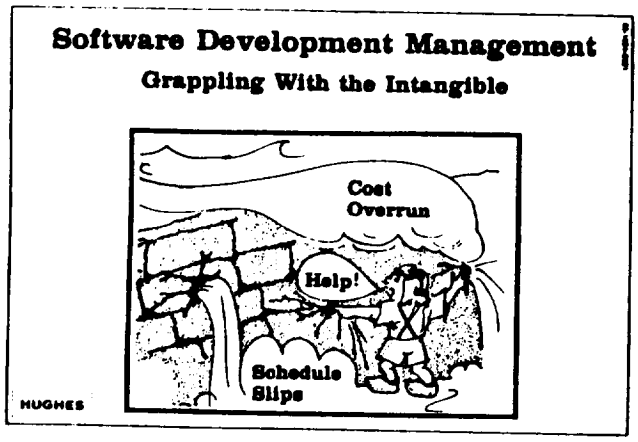
[3] K. Heninger, J. Kallander, D. Parnas, and J. Shore. Software Requirements for the A-7E Aircraft, NRL Memorandum Report 3876, 1978.

[4] R. Pressman. Software Engineering: A Practitioner's Approach, McGraw-Hill, 1982, p. 85.

[5] E. Yourdon and L. Constantine. Structured Design: Fundamentals of a Discipline of Computer Program and System Design, Prentice-Hall, 1979.

[6] V. Basili and A. Turner. "Iterative Enhancement: A Practical Technique for Software Development," IEEE Transactions on Software Engineering, December 1975.

SOFTWARE DEVELOPMENT MANAGEMENT
GRAPPLING WITH THE INTANGIBLE



S. A. VOLANSKY
PROGRAM MANAGER
DEFENSE SYSTEMS DIVISION, SCG
HUGHES AIRCRAFT CO.

Data Processing is big business within DOD and software is the major component of that

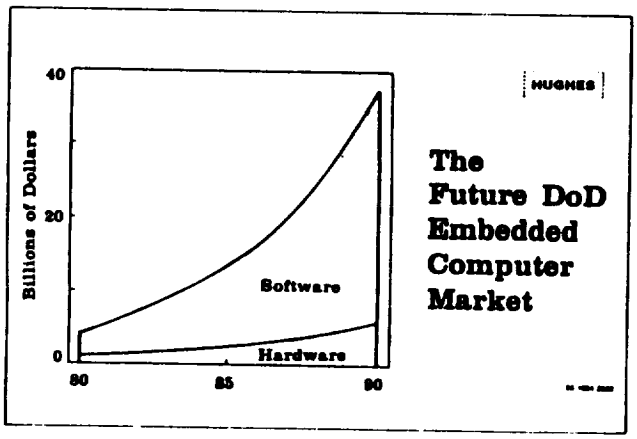


Figure 1.

business. The Electronics Industries Association has forecast that the DOD Embedded Computer market will reach $38 Billion by the end of the decade - almost an order of magnitude increase over the 1980 level. As depicted in Figure 1, the overwhelming contributor to this increase is software. It is almost impossible to imagine how DOD and Industry will manage a $32 Billion software investment. Historical software management techniques are woefully inadequate to this monumental task. This is aptly evidenced by GAO's 1979 summary of the management

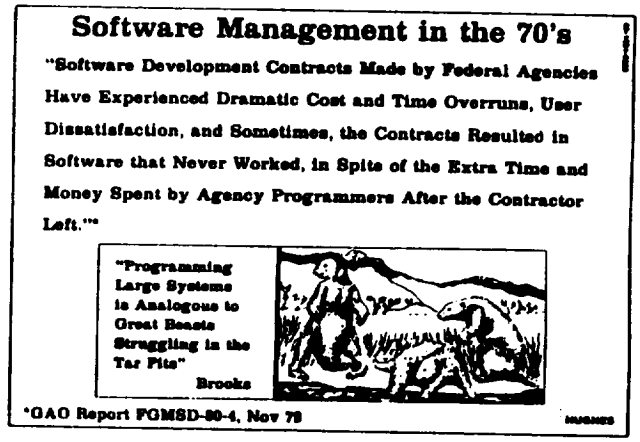


Figure 2.

situation shown in Figure 2. The projection of exponential software investment growth coupled with a demonstrated inability to control the development process imposes the critical management challenge for the 80's.

In 1980, the C(3)I community appraised the software acquisition environment and arrived at the set of conclusions presented in Figure 3. Several of the key problem areas focus directly on software development management. In particular, lack of good management practices, software uniqueness, cost estimating, productivity tools, and procurement policies

PAGE 26 INTENTIONALLY BLANK

are all entwined in the management dilemma. Numerous initiatives, within both DOD and

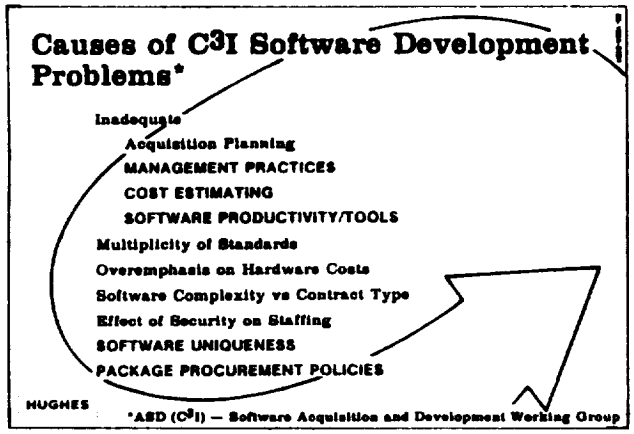


Figure 3.

Industry, have been undertaken in response to the challenge of extricating ourselves from the software management tar pits. One such initiative was undertaken in the late 70's in the Defense Systems Division of Hughes Aircraft's Space and Communications Group. The balance of this discussion concentrates on initiative specifics, their application to a major development activity, and the lessons learned from several years of real-world struggle.

The development milieu (Figure 4) was a major, state-of-the-art upgrade of an operational C(3)I System. Over a several year period, processing capacity was increased 10-fold. In excess of 2 million (source) lines of operational software were delivered, which more than doubled the installed base. Technically, this upgrade covered the software spectrum - operating systems, multicomputer networking, graphic and alphanumeric displays, interactive and real-time applications, as well as support and maintenance tools. It involved numerous contractual actions, multiple associate contractors, and both integrator and developer roles. In short, it had all the typical precursors for disaster.

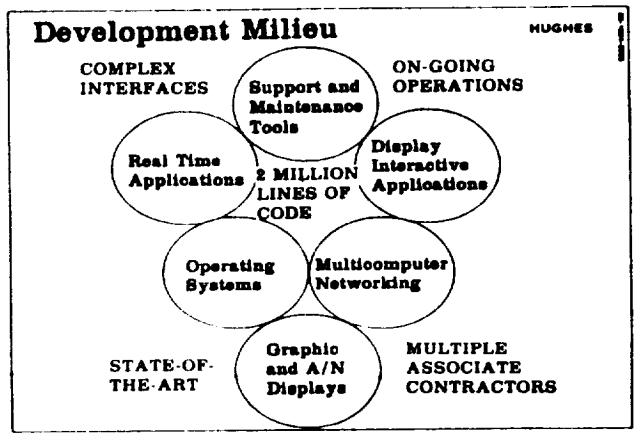


Figure 4.

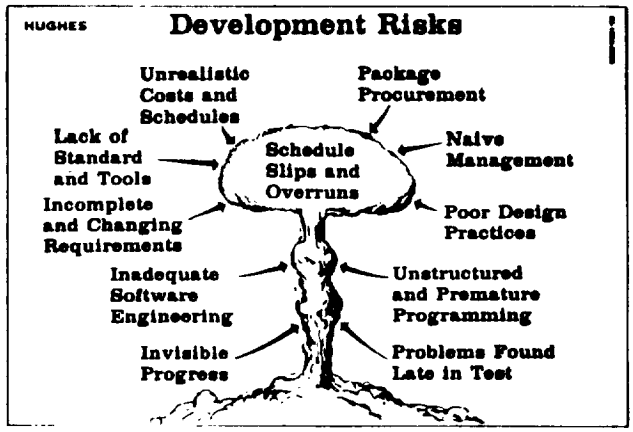


Figure 5.

From the outset, many of the risk factors inherent in the program were recognized. A concerted up-front effort was initiated by both customer and

contractor management to address and abate the risks illustrated in Figure 5. The essence of the development methodology, which emerged after considerable engineering and intensive customer/contractor negotiation, was embodied in the Data Processing Management Manual (DPMM) which was formally issued by the Air Force. This Baseline Management approach, summarized

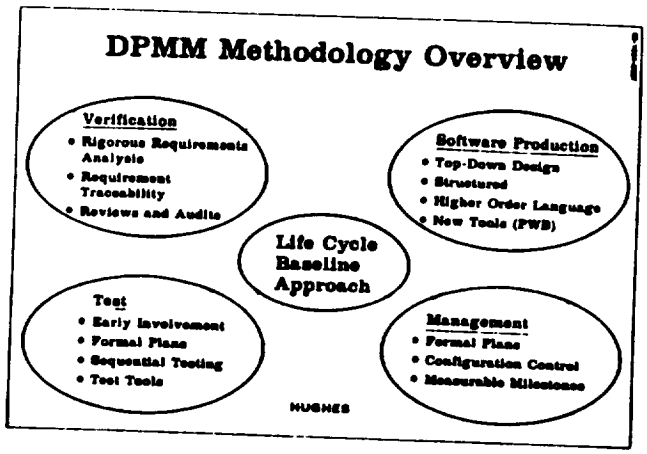


Figure 6.

in Figure 6, embraced the entire software life cycle. It identified the various customer and contractor roles and associated responsibilities. Formal management plans and development standards and practices were mandated. Rigorous requirements analysis and traceability, independent testing responsibility, and formal baseline configuration control were integrated to foster end-to-end discipline.

The heart of the DPMM is a four-phase data processing life cycle (Figure 7): Requirements Definition, Development, Transition, and Operations. Each phase contains one or more major activities. Data Processing Configuration Items (CI's) - aggregates of hardware and/or software - are identified, specified, developed, tested, and eventually accepted into the operational inventory in an orderly and controlled manner. Embedded in this life cycle are a series of configuration management baselines, associated with the transitions between the major activities. Reviews and/or audits of each baseline (documentation and/or products), conducted at the end of each activity, verify that all precursors for the succeeding activity have been satisfactorily completed.

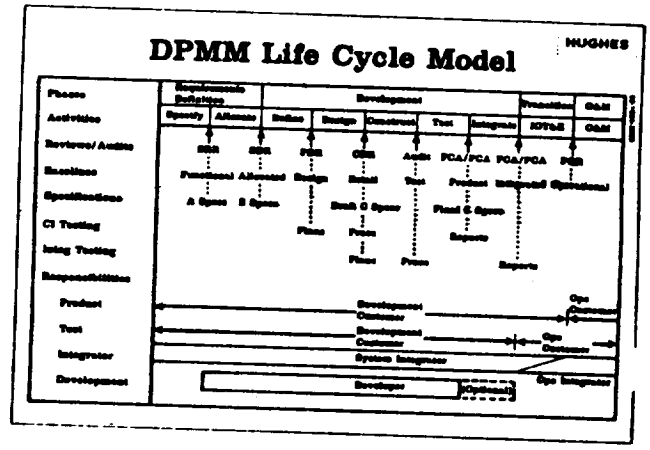


Figure 7.

Management roles and responsibilities are fully integrated with the technical aspects of the life cycle. On the customer side, product responsibility was vested in the development customer until a product was formally accepted into operations (FQR). Testing responsibility transferred from the development customer to the operational customer at the start of the Transition phase, which marks the boundary between specification-oriented testing

and operations-oriented testing. Each customer component was supported by a separate integrating contractor. The system integrator, in support of the development side of the customer, had responsibility for interface engineering among the various development contractors and for the overall technical integrity of the entire upgrade - at the segment or A specification level - until products were actually integerated into operations. In several instances, the system integrator actually performed the Integrate activity (with developer support). The site or operational integrator, in support of the operational side of the customer, was responsible for conducting the Transition phase - bringing products to a state of full operational readiness. Various associate development contractors had specific product responsibilities for elements of the functional baseline (established at SRR) until formal selloff (via DD250) at either the product or integrated baseline (as determined by technical complexity and risk).

Recognizing that, over the course of a multi-year development, neither the development baseline nor the operational baseline could remain frozen, a formal configuration management process (Figure 8) was instituted as part of the DPMM methodology. Two configuration control boards (CCB's), one for development related changes and one for changes to the ongoing operational environment, were constituted. Each CCB, chaired by an appropriate customer representative, had responsibility for its specific sphere. The System and Site integrators, as members of both boards, effected the coordination to ensure that impacts were fully identified across both operations and developments. Change requests that could not be dispositioned through inter-CCB coordination were referred to a separate Ground Segment Configuration Policy Board, comprised of representatives of both sides of the customer and both integrators, for ultimate disposition.

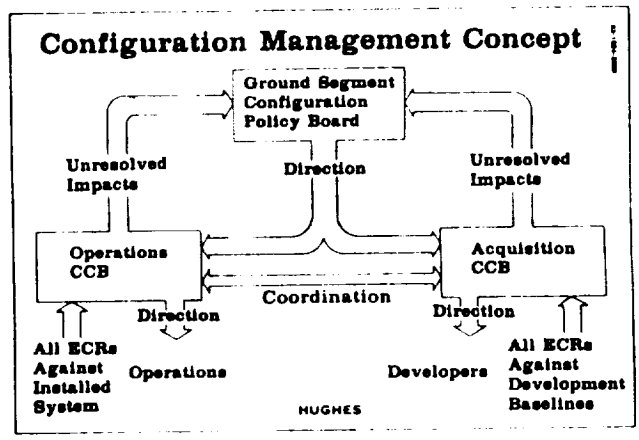


Figure 8.

A major key to the success of the overall DPMM methodology was the establishment of a phased approach to procurement as illustrated in Figure 9. Segment-level requirements engineering for the entire upgrade led to the definition of a complete functinal baseline. This baseline, validated at the System Requirements Review (SRR), was further refined by a time-phased series of System Design Reviews (SDR's) which authenticated specific elements of the allocated baseline. The system architecture was refined from the inside outward. The kernel resource environment,

consisting of computers, operating systems, networking and display support capabilities, was the first element of the allocated baseline solidified. The Mission support layer, planning and vehicle support software capabilities, was solidifed next, based on the defined resource environment. The outer layer of payload applications was built upon a well-defined support base. End item developments were initiated in time phases to ensure that key development dependencies were satisfied. The resource environment, for example, was basically completed to support initiation of the Construct activity for the next product layer. As end items were completed, they were transitioned and placed into actual operations. This not only minimized impacts to ongoing operations, but also enhanced product maturity and stability.

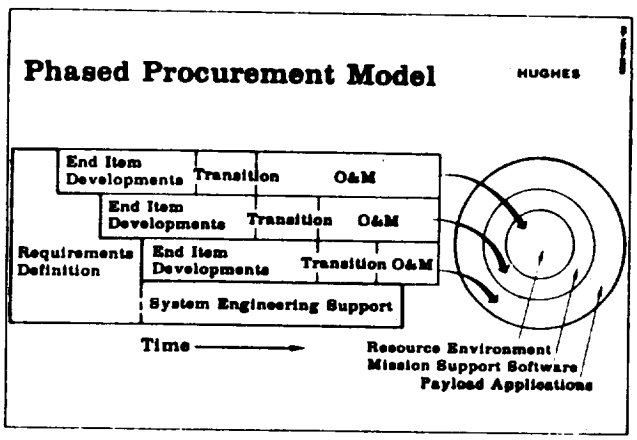


Figure 9.

Not only were end item procurements phased over time, most end item developments were procured in a multi-step fashion. Requirements definition evolved from generic, A-level specifications, through end item (B0/B1) specifications to specific CI (B2/B5) specifications in a hierarchical structure. Requirements definition culminated with a rigorous design-to-cost, or more truthfully a specify-to-cost, activity. Requirement sets were evaluated in light of budgetary constraints and specifications were tailored to be compatible with customer provided funding profiles. Contractual steps were paced by requirement evolution so that a firm validated requirement base was established prior to initiating actual development. In this way, the actual scope of the development efforts could be more accurately assessed before committment to contractual strictures.

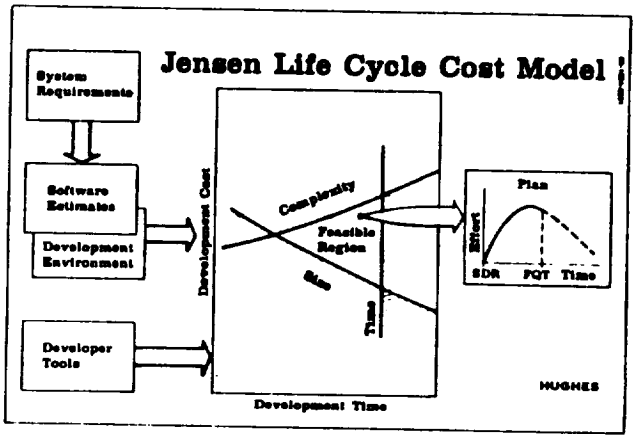


Figure 10.

Since the requirement base was fairly well-defined at the start of the real development activity, it was possible to employ software cost/schedule estimating modeling with a high degree of success. The Jensen model, depicted in Figure 10, was developed at Hughes as a

refinement of Putnam's Rayleigh curve model of the software development activity. It is a relatively accurate model of the period from System Design Review through Formal Qualification Test. The development time and cost are modeled as a function of software size and complexity and the development environment and tool availability. The intuitive basis of the model is that development cost and time grow with increasing software complexity and size and are strongly influenced by the environment - defined in terms of development methodology and tools - in which the development occurs. Use of the model requires estimation of software size and complexity (usually on a CI by CI basis) and characterization of the development environment based on both historical performance and anticipated changes in methodology and tools. A feasible development region, in the cost-time domain, is defined in terms of a complexity constraint (based on type of software), a productivity constraint (based on both size and development environment), and a time constraint (usually arbitrarily specified). Any point in this region can be selected as the basis for the development plan - which defines effort over time. Figure 11 provides definitions of the various model parameters with the input parameters highlighted. Development size (delivered source instructions) is an obvious parameter. The difficulty or complexity parameter spans the spectrum from high logical complexity microcode with critical interfaces and control logic to simple standalone straightline software with minimal interfaces and simple input/output. The technology constant measures the developer's production technology - the ability to implement the project. It includes factors such as availability of computing resources, organization strategies, development tools, personnel capability, experience and familiarity with the development environment. The model input parameters can be treated as independent random variables, rather than point estimates, and used to support quantitative risk analyses. Feedback of actual performance into the model facilitates parameter refinement and, more importantly, provides a rigorous data base for downstream management decision making.

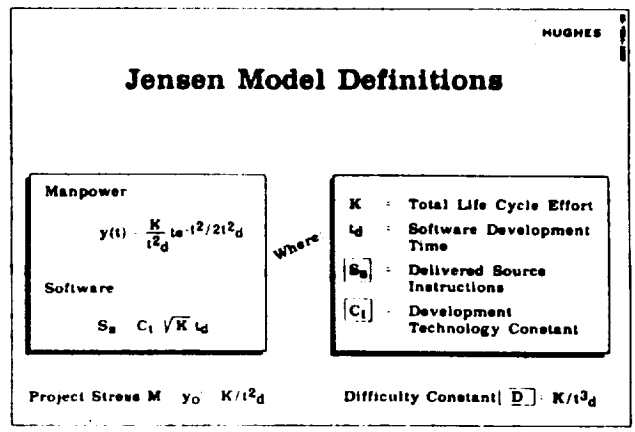


Figure 11.

A structured approach to the allocation and verification of requirements, as depicted in Figure 12, underlay the development process. Allocation proceeded via stepwise refinement from generalities to specifics. Requirements were documented in a hierarchal set of specifications with complete interspecifiction traceability.

Identification of verification methods and test assignment was coincident with the allocation process. Actual verification proceeded via incremental buildup from specific CI-level testing to general segment-level testing. This incremental approach to integration, shown in Figure 13, allowed verification of more complex capabilities to be conducted on a base of previously verified components.

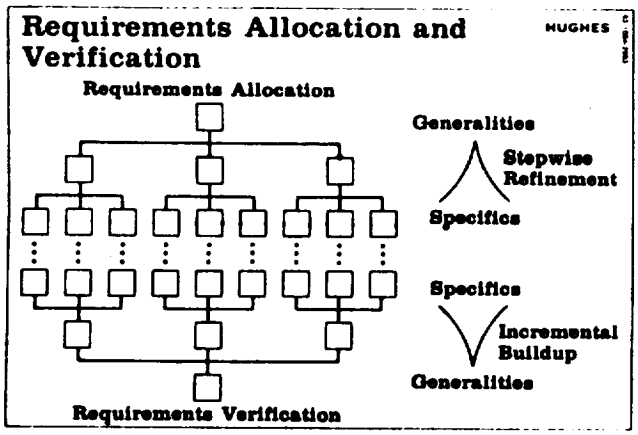


Figure 12.

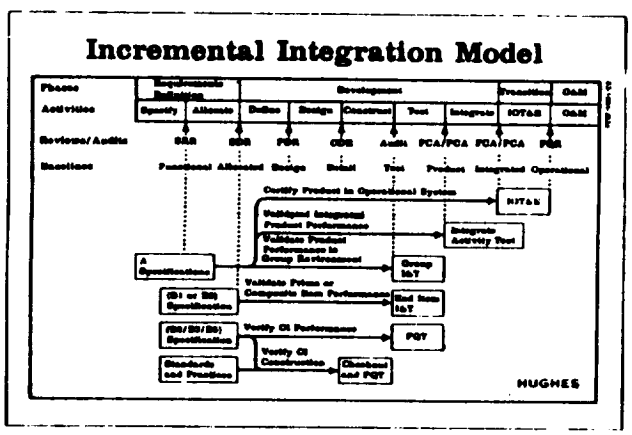


Figure 13.

Hughes had responsibility for overall system integration as well as for a major portion of the actual development end items. The organizational structure in which these activi-

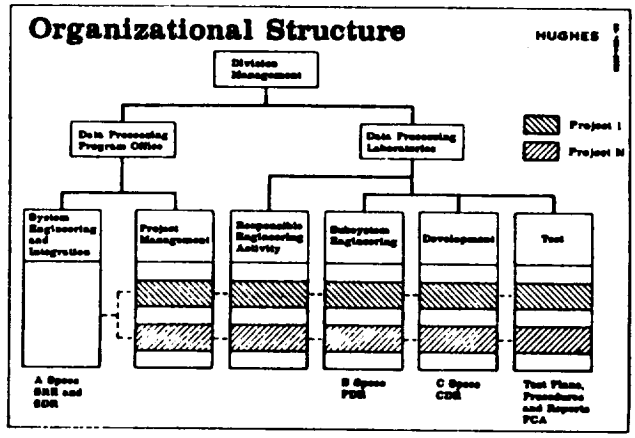


Figure 14.

ties were accomplished is shown in Figure 14. A separate Data Processing Program Office was chartered with overall management responsibility. The system integrator and development roles were assigned to different managers within the Program Office to maintain integrator integrity. The integrator role had responsibility for all A-level specifications and associated reviews along with overall system engineering. A separate Development Project Manager was identified with each major set of end items. The Project Managers interfaced with the Data Processing Laboratories, which actually performed the developments, through Responsible Engineers, consistent with an overall matrix management approach. The DP Labs were organized into Engineering, Development, and Test departments. The separation of test at an equal level with

engineering and development ensured an independent approach and objectivity in the test program. Responsibilities for the various baseline documentation products and reviews were delegated to the various functional departments. Project teams, spanning the various organizational entities, carried each project from beginning to end.

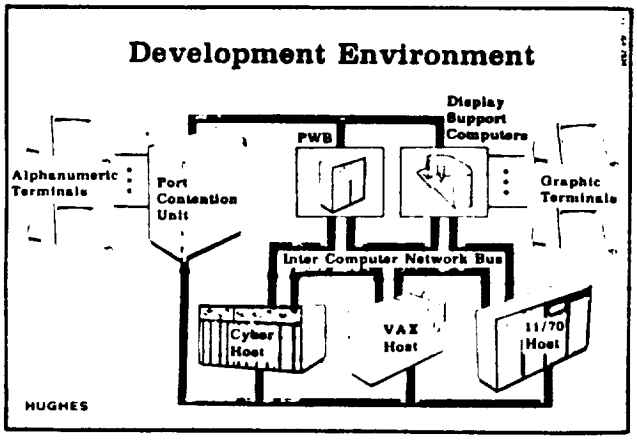


Figure 15.

The physical development environment (Figure 15) played a major role in development productivity. The heart of the development environment was Programmer's Workbench (PWB), a UNIX-based system of development tools. Source code and documentation were entered and maintained via alphanumeric terminals distributed throughout the development area. These terminals, connected via a Port contention unit, could access not only PWB, but any of the host and display support computers in the development computer center - allowing a programmer to enter code, compile, execute and review output from a single location. PWB, the various host computers, and the display support computers - which provided a distributed graphics capability - were all interconnected via a high speed multidrop data bus which allowed great data transfer flexibility. This configuration not only fostered high development productivity, but provided a robust test bed for multi-computer test and

Development Tools HUGHES

- **Interactive Text Editor**
- **Source Code Control System**
- **Structured FORTRAN Precompiler**
- **Variable Description Program**
- **Processing Logic Tree Plotting System**

Figure 16.

integration activities. Key development tools are identified in Figure 16. They include a full screen text editor, a source library maintenance and configuration control system, a structured FORTRAN precompiler, a package for maintaining internal CI interfaces (calling sequences and common blocks), and plotting system for Processing Logic Trees - a form of structured flowchart.

A disciplined approach to library maintenance and control was greatly facilitated using the tools and capabilities in the development environment. Figure 17 summarizes the major components of the library maintenance system. Programmers, using their local

terminals, could access working files, make changes, temporarily merge changes with controlled library elements, and then transfer the resulting program files to the appropriate host for compilation and execution. Upon checkout completion, programmers transfer code changes (deltas) to proposed change files for access by the librarian. The librarian transfers proposed changes to review files for approval prior to incorporation in the controlled library. Library maintenance via delta files allowed complete flexibility and easy recreation of prior versions. Source libraries were maintained on PWB while object libraries were maintained on the hosts using PWB. Coupling the power of PWB with a powerful distributed resource pool resulted in a responsive and flexible develoment environment significantly enhancing overall productivity.

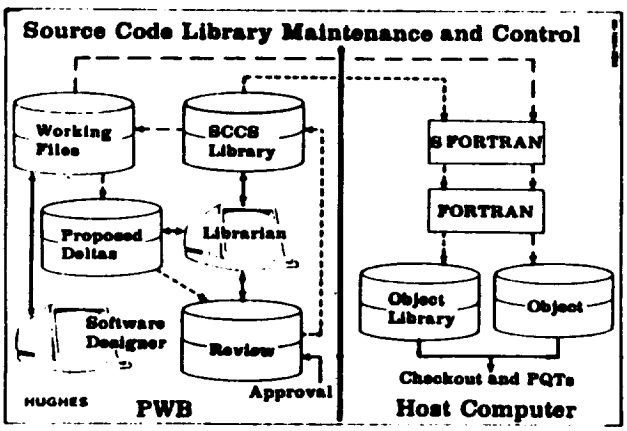


Figure 17.

The inability to quantify programming progress has often been cited as a reason for failure of software development management. Granting that software intangibility renders direct inspection of progress impractical, a well-disciplined methodology nevertheless provides numerous progress measures. Startng at CDR, the classical programming dark period, there are several parameters suitable for a "rate chart" form of progress monitoring. The prime requisite for an acceptable progress measure is an objective rather than subjective criterion for completion. For example, number of routines coded is open to rather wide latitude in programmer interpretation. Whereas the number of routines on the controlled library, coupled with a disciplined library procedure, provides a reliable and effective measure of progress. In our environment, routine size was

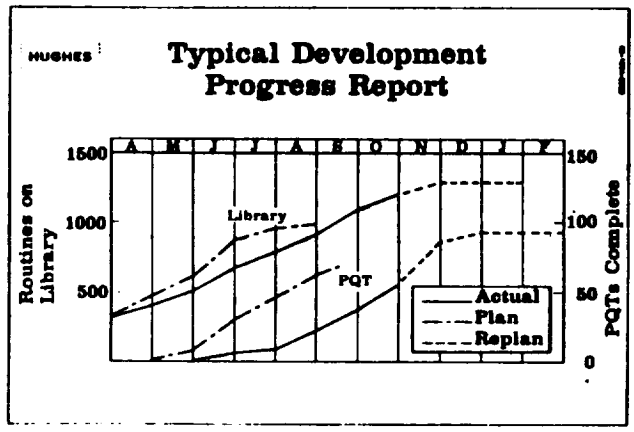


Figure 18.

typically between 50 and 100 lines of source code. Functionally related routines were grouped into units. For each unit written checkout plans and acceptance criteria were prepared. Progress through checkout was monitored via counting the number of

Preliminary Qualification Tests (PQT's) completed - all acceptance criteria satisfied. Rate charts, as typified by Figure 18, reviewed on a weekly basis, gave reliable and straightforward progress measurements. Statistics maintained on PWB made generation of these charts essentially automatic. Similar techniques were used during formal testing. In that activity, the number of specification requirements validated and the number of problem reports written and dispositioned were the operative parameters.

Quantification of progress in measurable "work units" is a natural precursor to effective management in an earned value domain. As shown in Figure 19, work packages, defined in terms of measurable parameters and costs over time, are aggregated to portray overall project status in terms of dollars. This allows complete assessment of project status from beginning to end in consistent management-oriented terms.

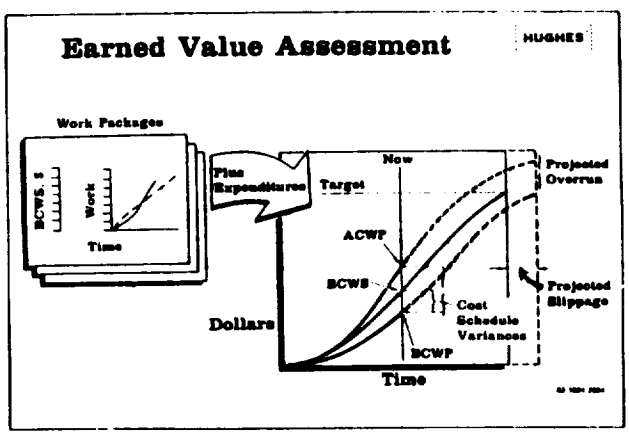


Figure 19.

Before recapping the software development management approaches and critiquing their overall effectiveness, some thoughts about the hardware side of software development are appropriate. As summarized in Figure 20, there are several key hardware points that can make or break a software development activity. A stable, predictable hardware environment is an absolute necessity for successful software development. Without it, problem diagnosis is exponentially more complex and the inevitable finger-pointing slows progress to a crawl. Not only should concurrent hardware and software development be avoided but the natural tendency to upgrade in the middle of development to the latest greatest wizzbang widget the vendor is pushing is fraught with pitfalls. Especially critical is the need for hardware-specific support software. A box full of silicon chips is a distinct liability to the development process without the associated diagnostics, executives and interface software. Considering the relative costs of hardware and software, it is unbelievable that so many projects flounder into massive cost and schedule overruns by trying to cram ten pounds of software into five pounds of hardware. Software seems to inevitably grow in size over the development cycle. The added complexity and pain of overlaying and tailoring code for minimal size is rarely worth the effort for ground-based systems where deliberate oversizing of hardware is an inexpensive and very effective technique.

Some Thoughts About Hardware

Avoid	• Concurrent Hardware/Software Development
	• Tendency to get Latest Vendor Offering
Recognize	• Support Software is as Important as Hardware Itself
Anticipate	• Hardware to Meet Inevitable Job Growth

HUGHES

Figure 20.

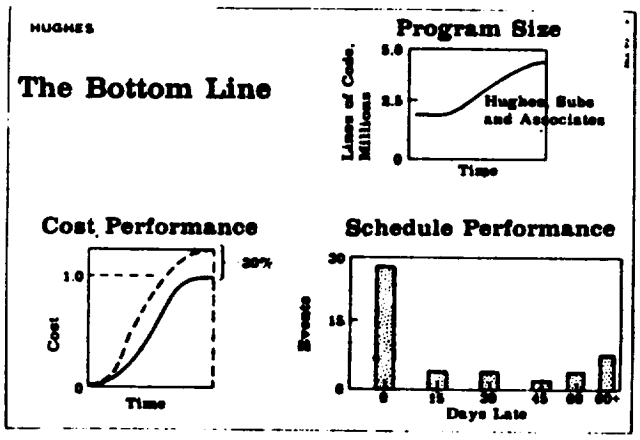


Figure 21.

In summary, five years ago Hughes undertook deliberate action to define and apply a software development methodology that addressed the major failings of the then current software development process. Now, at nearly the completion of a major upgrade to an ongoing C(3)I operation, one can begin to assess the effectivity of the methodology employed. The development spectrum to which this methodology was applied was wide - many classes of software were developed. There were many successes and a few painfully outstanding failures. The overall bottom line is shown in Figure 21. The successful integration of approximately 2 million source lines is nearly complete. Cost overrun was held to under 30% and 85% of major schedule milestones were achieved within 2 months of the agreed to schedule dates. While by no means an outstanding success, the methodology, coupled with a dedicated and talented staff, lots of hard work, and more than a little luck, resulted in very respectable performance.

From today's vantage point, one can reflect over the various constituents of the methodology and summarize the major lessons learned both - positive and negative. Some things worked well and some were disappointing. Figure 22 highlights the major positive lessons - things which will be carried forward to future development activities. Baseline management really works. One can channel and control the development process through well-defined documentation and a series of thorough reviews and audits. A phased approach to procurement is the best risk reduction technique. A comprehensive bilateral understanding between contractor and customer of the requirements before tieing the process in contractual red tape eliminates many nasty surprises for both sides. Especially when coupled with an effective method of cost/schedule estimation. A friendly development environment makes happy and productive programmers. A robust programmer's tool kit and adequate

development resources are a major contributor to success. Development progress can be measured - rate charts are extremely effective - and monitored in management terms via an earned value approach.

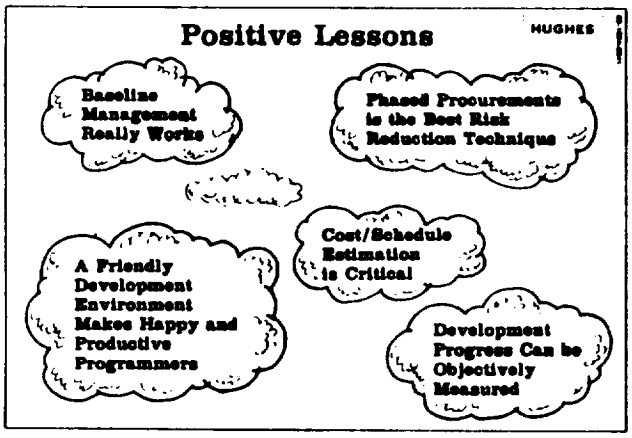


Figure 22.

On the other hand, there were characteristics of the methodology that proved to be less successful than anticipated. Figure 23 highlights these negative lessons - things which will be modified for future developments. Excessive formal control wastes effort. Formal customer concurrence and change control on a huge volume of documentation often added significant delays to the development process. Careful assessment and tailoring of which documents are critical enough to warrent specific customer involvement would go a long way toward reducing the paperwork log jam. The overly structured transitions between development activities discouraged code development prior to CDR and thus tended to limit prototyping - resulting in more than a few "oopses" long after CDR. Provisions for prototyping must be included in the formal baseline management approach to encourage use of this valuable technique. While independent test was a decided technical asset, the organizational separation of test turned out to be noticeably expensive. Training of test engineers was notably underestimated and many disconnects occurred between test and engineering as to the intent of requirements. Both these drawbacks could be eliminated by combining requirements and test engineering while still providing organizational separation from development. CDR timing is most critical - too early and its a waste - too late and its expensive. Occassionally, the shear bulk of documentation overwhelmed the review process and camouflaged design deficiencies. A series of working design walkthroughs with knowledgeable reviewers, as opposed to a formal dog-and-pony show, would seem to ameliorate this situation. Finally, software documentation must be useful. Software documentation is already huge and producing unnecessary paper is simply ridiculous. For example, the detailed build-to documentation for CDR (draft C5 Specification) is necessary for a comprehensive review of the detailed design. Delivery of this plethora of detail, which must be kept current, with the actual product at selloff has turned out to be wasteful overkill. Given structured well-commented program listings and overall architecture and structure charts, the maintenance programmers ignore the vast bulk of the as-built documentation. Again, the proper balance must be struck to ensure necessity and sufficiency

of delivered documentation.

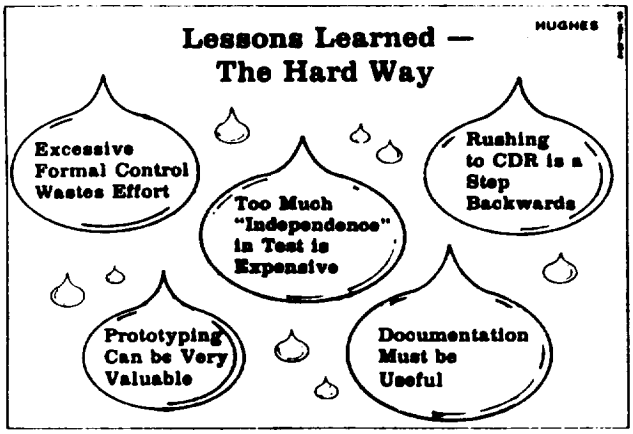


Figure 23.

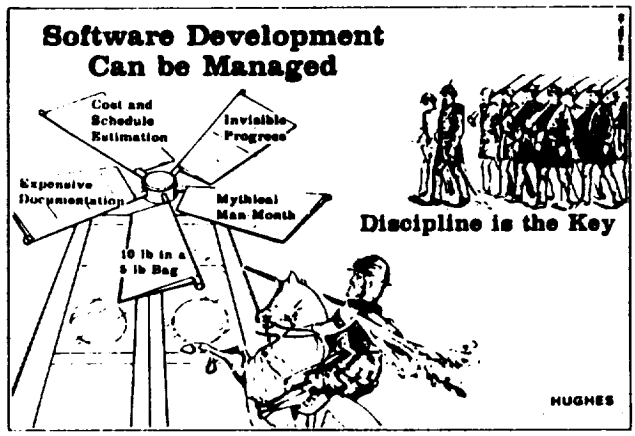


Figure 24.

To recapitulate, the Hughes experience has shown that a great deal can be done to provide effective management of the software development process. Proven methodologies and techniques are available. While a great deal of progress is required before software development management reaches the maturity commonly associated with hardware development, a great deal has been accomplished. The poor software program manager (Figure 24) is no longer a lonely figure tilting at the development windmill. There is help. A well thought-out and disciplined approach is the key.

A Decade of Software Engineering Problems and Solutions
A Case History: MPD 1972-1982

Edward H. Reese, Jr.
General Electric

To borrow from Charles Dickens in the opening of "A Tale of Two Cities": "It was the best of times, it was the worst of times...!".

It was the best of times because it represented the largest opportunity the General Electric Space Division ever had to put into place a software engineering effort.

A small nucleus of engineers, with a background of command and control experience, spacecraft avionics, simulation, and analytical programming, were brought together in support of a number of related studies, which would lead to a large command and control system.

Recent experience on the ill-fated Manned Orbital Laboratory Program, as well as some attempts at software engineering in support of General Electric commercial ventures, convinced us of the need for engineering discipline in the development of software—but not necessarily of the need for new development tools and techniques.

Customer climate at this time was also appropriate for establishing a more disciplined approach to the development of computer programs. This, perhaps more than any other single factor, led to whatever success was gained. Because it was as true then, as it is now and always will be, that the sale of an idea requires an enlightened customer!

Two major aspects were brought together in our then-current related study efforts: first, a very large software development activity was looming on the horizon—as the command and control concepts were merged with significant scientific computational elements and large simulation activities; and second, as this new system concept was being developed, it was decided that a unique focus on software was needed. The focus was correct, but the uniqueness later turned out to be both inappropriate and distracting to the overall system engineering of the resulting projects.

It was the worst of times, because the techniques available to perform an effective software engineering activity did not match the need, or the zeal, for software engineering.

Furthermore, the size of the activity was such that many of the concepts of discipline for configuration management and documentation were to be stretched. Merely scaling up the concepts available would not handle the problem.

It was also the worst of times because, while many were enlightened, not all were enthusiastic recipients of the cult of software—with its own vernacular and obscure products.

This lack of acceptance was frequently exacerbated by little things like the corruption of perfectly good English language—words to mean apparently anything the software engineering community wanted them to mean!

For example: words like "activity", "event", and "process" were taken out of circulation as generic descriptions, and given very specific meaning, much to the consternation and confusion of many. New terminology, with imprecise definition such as real-time and off-line, merely fueled the fire of discontent in many, and in particular, the program management cadre. Many of these self-inflicted wounds, while appearing trivial, made it difficult for those committed to convincing the immediate world that software was

a mature scientific product, worthy of the significant dollar investment being made, especially when things were not going well in the development process.

The early years of this case history were filled with both progress and frustration—in just about equal doses—shades of the age of wisdom, and the age of foolishness!

The wisest thing that was done was the application of a baseline management approach to the software development process.

This was not the first attempt to employ these concepts, but it was probably the most rigorous application of the methodology that we had experienced up to that time.

Many of the team members had participated with the Air Force, and with Aerospace Corporation, in the application of these techniques—on precursor command and control developments, including the MOL Program, as I have mentioned before; however, most of the time had been spent generating the foundation of the concepts, and much less in actually employing the prescribed disciplines.

This time the emphasis had shifted. Shifted to performance against the standards rather than inordinate bureaucratic arguments by working groups, discussing the roles and responsibilities of the various participants, i.e., contractor, government, advisor, integrator, etc., in accomplishing a baseline activity. The emphasis could now be focused on performance, because, under customer direction, a set of "software implementation directives" were put in place and compliance was made mandatory from the outset—as opposed to trying to develop these concurrent with the development. The foundation of these directives was the earlier initiatives; and quite clearly, if it were not for the significant body of work done for MOL and other projects run out of the AFSCF, the task would not have been carried out. I would be lying if I said that all of the bickering over the form of documentation, and the criteria for successful completion of baselines like a PDR, were all behind us in 1972. What was behind us, was the establishment of a clear authority for determining the form and criteria issues. What was present, were people who were willing to make decisions, and to make them stick. In short, we had discipline and background. What we did not have was a shortage of problems to test the process. Figure 1 shows a brief summary of the class of problems that we had to deal with, which also made this time the worst of times.

As I proceed, I will attempt to share with you a few examples of these problems, how we dealt with them at that time, what, if anything, we learned from the experience, and how we deal with it today.

But, before we go into the problems, let's take a look at the baseline development methodology that was put into place.

I'm sure that, by now, you are all familiar with this methodology. Either your company or your agency employs these techniques in developing software, or you have read about it in the literature.

The most remarkable thing about this process is that it is not very different from the techniques, employed for years, in the development of hardware systems. The biggest hurdle back then was convincing management that software was not a black art, and also, convincing the artisans that what they were producing needed to be managed. As I said at the outset, some of us were believers—not all of us!

In Figure 2 I show the baseline management approach employed by the GE Space Division today; it is only a slightly modified version of the process developed in the late 60's and employed on the projects with which I was personally associated. These techniques were also employed on the DSCS III ground software de-

velopment in the late 1970's, and it is no accident that we adopted this as the foundation for our division standard for software development.

The remainder of this discussion will be loosely coupled to the chronology of the development process model.

Systems Analysis
Concept Development
Requirements Analysis

The baseline development process acknowledges, in concept, the timely and orderly development of requirements: the literature is full of data relating to the price one must pay for not finding errors early in the development cycle. Poor requirements equate to errors. Figure 3 is intended to remind you of the shape of the curve which characterizes the relative expense of finding problems at various stages. The figure is also annotated with the general causes of errors introduced at various states of development. A problem that existed at the start of this case history, and is unfortunately still with us, is the failure to recognize the impact of what I call late developing requirements. These are the open requirement areas that are allowed to persist, because of the belief that there will be plenty of time to incorporate these new modes of operation later: "After all, how hard can it be when you are already doing something like that...it's not like you're bending metal, you know!"

These are not fuzzy requirements; these are requirements that are deferred under the false premise that it does not matter. All of this comes from a lack of sensitivity to the software developer's needs. As I have shown in Figure 4, the programmer usually has different sentiments about open requirement than does the requirements setter, because it is exactly as though you were bending metal! The flexibility to deal with changes that come about through the application of good structured design practices, such as modularity, cannot overcome the difficulties which ensue from not specifying all of the major modes of operation required to achieve total system operation at the same time. In the case that I'm talking about, the requirements associated with initializing the system, and those associated with dealing with contingencies, were the project open areas. They remained undefined until well into the detailed design of the nominal performance processes. These requirements were of such significance that they disrupted the design progress of approximately 50% of the previously accomplished design effort! It was not only that it was pervasive, but also that it was not clearly appreciated that the newly defined requirements were going to be difficult to incorporate. In fact, it was so poorly understood, that meeting after meeting was held to explain not just how difficult it was, but why it was so difficult. Clearly, if we had been building a means of transportation to go between fixed points in a straight line, and had designed a cable car as a solution, only to find out later that the open requirements were such that contingency handling demanded a self-contained power plant, and furthermore required maneuverability characteristics, no one would have doubted the difficulty of adapting the on-going design to the new requirements.

The solution to this problem is partly one of educating requirement setters into the complexity of software interfaces, and into the labor intensive nature of programming; but it also requires that the developers have a strong capability within their own organization. In our organization, system engineers are charged with the responsibility of assuring that requirements are both comprehensive and timely. They are specifically charged with assuring that there are no major open items when we arrive at PDR. This requires that they develop candidate requirement statements

and impact explanations, to bring to the attention of the requirement setters the full implications of the open requirements. This aggressive pursuit of requirements is the best safeguard you have to avoid what Melier Page-Jones, in his book "A Preliminary Guide to Structured Design", characterized as: "...Trying to arrive at an unknown destination, in the dark, without a map!"

To accomplish requirements development, that is, to assure requirements are consistent and complete, also requires a disciplined methodology. In the early 1970's, the methodology we employed was a manual approach to requirements allocation and traceability. As depicted in Figure 5, this process consisted of physically extending the driving document; by that I mean that the System Segment in question, Segment specification, and all of its interface specifications with other Segments in the System being developed, were physically appended with data to describe the interpretation of the requirement. The data provided a definition of how this requirement was to be handled by the various end items into which the segment design had been partitioned. On the surface, this appeared to be a reasonable and rigorous approach to assuring that all requirements were being addressed, and that they were being allocated to deliverable end items of software, hardware and personnel. The problems, however, were manifold, and all related to the scale of the effort both in terms of the source requirements and the recipient end items. All of this was further compounded by the dynamics of the requirements development process. It would have been nice if the requirements all matured at the same time, but that can never happen. Even if the open area situation that I discussed earlier had never happened, the requirements development activity would still be one that matured as a function of time.

You can well imagine the physical logistic problems of maintaining the paper monstrosity which evolves from the methodology described. And this does not even begin to address the requirements for downward traceability to the verification documentation, nor the backward traceability from the end item functional and verification documents to the driving requirements to assure that the entire process promises closure. Those too were accomplished in a similar manual process. The process, of necessity, could only be accomplished as a "snapshot in time" of the baseline, and was out of date the day it was produced. It did however serve the purpose.

You have to keep in mind, that in the time frame that I am referring to, we had no automated tools available, not even word processing support. The inefficiency of the process has motivated us, over the decade, to investigate all of the requirement development tools which have come along, for example, PSL/PSA and SREM; but, to date, we have found that the most effective mechanism for documentation and control of the requirements traceability problem, which allows us to cope with both the dynamics and the extensive upward and downward traceability needs, is the "Data Dictionary" approach (see Figure 6). The dictionary is created with linkages from the driving requirement, through the segment design specification, through the end item functional specification, through the end item verification plans, procedures and reports. The appropriate records are updated as required. The data base management system, through its connectivity checking and report generation features, provides the mechanisms for audits and reports which aid in assuring that the process closes. Furthermore, it allows for a process which can remain current without the labor intensive effort of its predecessor.

PRELIMINARY DESIGN

The Preliminary Design problems which were experienced were largely related to two things that still plague a lot of developments. The first of these deals with the visibility provided into the overall design, and the second has to do with the appropriate level at which we treat the verification program.

In early 1972, in the course of developing the design of a large command and control application, a serious breach of the concepts of structured development occurred in our development of the PDR baseline. The segment level design document, which was characterized in my previous discussion of requirements development, was not appropriately structured to clearly depict the overall design approach. What happened was that the baseline documentation directives had defined a specification that was to contain the requirements allocated to software, without addressing those requirements allocated to the personnel subsystem and hardware elements. Furthermore, the document was defined in such a manner that there was considerable overlap between it, and the functional specification for the software end items. As a consequence of the overlap, it was decided that the specification which had the capacity for becoming the segment design specification, and which could have, and should have been, the purveyor of the design structure, was abandoned.

On the other hand, the need for a document to contain the design level operational concepts and the allocation of requirements to the personnel subsystem was not fully recognized. In no small way was this related to the problem that I alluded to earlier—that emanated from the "special" attention given to the software.

To our credit, the informal specification which was prepared did go a long way toward providing a repository of a great deal of the segment design structure. The problems that arose then are the natural result of faulty specification structures in general. Some of the design team was marching to the implied and specific requirements of the document, others treated it as a guideline, and still others ignored it altogether.

Since it was not in the recognized specification tree, the operations requirements specification, as it was known, could not, in the beginning, be referenced in the software end item specifications. This, of course, led to a great deal of extra labor on the part of the system design team, charged with assuring that the total design was unambiguous and held together. That chore is difficult enough under normal circumstances; imagine how hard it is when you can't even derive the conventional benefits of document configuration management to aid in the control of the baseline.

Through the normal maturation process, a great deal of persistence, and no small quotient of missionary work, the appropriate level of document was recognized as a required element of the system. *Note: I said system; not just software! document tree.*

In Figure 7 I have depicted the specification tree as it exists today, and show its relationship to both the software and operations products which support the personnel subsystem. I can't stress enough how important it is that the structured design of an embedded computer system take all of the design elements into consideration, i.e., hardware, software, and personnel in a single overall design effort. Not as sequential nor isolated aspects of a related design, but as a concurrent orchestration of the one and only design.

The type of system that I am talking about was not one in which we were engaged in converting an existing manual procedure into an automated process, but was one in which there was no existing end user. I don't believe that the documentation or design process would be

altered if it had been, but this kind of situation focuses on the need for the design team to act as surrogates of the eventual end user in every step of the design process to avoid end user dissatisfaction with the delivered product. The changes that result from such dissatisfaction and discontent are euphemistically characterized as system maintenance, but are frequently due to the neglect that I'm suggesting can occur in the early design phases. I have also indicated in the figure that the DOD specifications Mil Std's 490 and 483 still do not make official and adequate provisions for the operations-related products. Hopefully, the current "Software Initiatives" being worked by DOD will rectify this hole in the document tree.

As I said earlier, there were two problems which plagued us in the PDR phase. The second of these, which is related to the verification program, is a problem of emphasis. In fact, it is two problems of emphasis (see Figure 8). The first of these is over-expectation in terms of design maturity on the part of reviewing authority; the second is the second class treatment of the verification process in the structure of the Preliminary Design Review event. This form of behavior is clinically referred to as schizophrenia. On the one hand there is the desire to understand, in minute detail, how a design which is still embryonic will be verified, but on the other hand, the review event typically relegates the verification element a very short, and usually low priority, slot on the review agenda.

The first part of this problem stems from the fact that while the performance requirements have matured, and while the verification requirements must follow the functional and performance requirements to the letter, these requirements and the preliminary design shall form necessary, but insufficient conditions, for the development of test cases and acceptance criteria. Attempts to demand a high quality test plan of this type at PDR time comes from a failure to recognize how the design matures. It is impossible to have the design team, let alone an independent test group, develop a high quality end item test plan, reflecting appropriately designed test cases, with relevant acceptance criteria, in the same time frame during which they have barely struggled through the specification and preliminary design activities. The net result is frustration on the part of the preparer, and dissatisfaction on the part of the review team. If you then couple this with a review event that is pressed for time on the agenda, you have all the necessary ingredients for a major open item growing out of the PDR. The natural resolution of this open item is to put in place some form of task force to drive this to closure. Sure enough somewhere around CDR time, when the design has matured sufficiently to support the process of test case design, the test plans are produced; proving that if it were not for the tiger team, the test program would really be in trouble.

In the meantime, because of the attempt to define the details, the program has not focused on the essence of the test program. How is the test program per se designed? How are tests designed to flow? How is data to be generated? Are common scenarios to be generated to minimize data generation and to maximize related performance evaluations? What test support tools are to be employed? How will configuration integrity be assured during test and problem repair cycles? In short, the key and timely questions do not get addressed in a timely manner. Of course, these questions will come up, but that will be the problem for another anachronistic tiger team to solve.

It is possible to avoid this syndrome! All that is required is to address the verification process at the right level, and at the right time. At PDR, or maybe even at a baseline review after PDR, but before CDR, a review of

the verification process should be conducted, with adequate time and appropriate criteria to assure that the verification process is maturing consistent with the design. This concept should be carried out in conjunction with the CDR baseline, as well, to assure appropriate focus, at the appropriate time, throughout the development process.

CRITICAL DESIGN REVIEW

Let's move to the CDR baseline to observe what, by now, must sound like an endless series of disasters. I can assure you that there were times during this period that it seemed like that to a number of people. Any series of problems, regardless of how rational they are, if they lead to cost and schedule difficulties, are going to be met with a great deal less understanding and forgiveness than they deserve.

Considering some of the things that I told you earlier, like not being able to code TBD's—the diversion of the designer to unachievable tasks related to verification in the wrong time frame—it does not take too much imagination to see chaos brewing. But unfortunately, not everyone reacted the way they might have. In fact, the gentleman I was working for at the time did not react as I expected, even though the signs were clear that we were in big trouble—and that we needed big help!

The team had inadequate time and tools, and needed revitalization. When my boss was confronted with the prospects of going to his management and declaring a form of limited bankruptcy, he declined. I will never forget the little analogy that he gave me, which summed up his philosophy of the period. It was the story of "The Sultan's Request": "It seems that, at one time in history, there was this very cruel Sultan, who ruled his empire with an iron fist and whose prisons were full of persons guilty of misdemeanors, or sometimes, guilty of nothing more than irritating the Sultan. His penalties were uniformly long imprisonment, followed by torture, and then by death. Well, it came to pass, as they like to say in these kinds of tales, that the Sultan was given an elephant that, with the right kind of training, could be made to fly. He sent word out to the prisons—if any one could teach his elephant to fly, that prisoner would be given his freedom. However, if at the end of a month, the elephant could not—fly the perpetrator would be filleted in the public square.

In one of the prisons, a brave soul raised his hand and said that, as luck would have it, that was, in fact, his trade prior to incarceration in the Sultan's jail. As the courier went off to arrange for his release, one of the prisoner's friends rushed over to him. The friend said to the volunteer, "Have you lost your mind; you have never trained an elephant to fly; surely they will extract your bones one by one at the end of the month." The volunteer assured his friend that he knew what he was doing. He told his friend that perhaps the Sultan, who had many enemies, would be overtaken by a foe that was more benevolent, or perhaps the Sultan, who was not in the best of health, would die in that month and he could just ride the elephant away. Finally, he said to his friend, "Who knows, maybe I *can* teach this elephant to fly!!"

I don't know how the prisoner made out—but I do know that that philosophy did not prevail over the problems that we had. One of us was naive, and I am not sure that I can say that it was him. He may have known that his fate was inevitable, or it may have been that he did not recognize the danger sign associated with a program in trouble. It does not matter whether these problems are self induced or externally stimulated, the results are the same.

In the associated figure (Figure 9), I have listed the danger signals that tell you that you

are involved with a problem project. I obtained this specific list from a paper entitled "How to Salvage a Faltering Software Project," by M.W. Evans, P. Piazza, P. Sonnenblick, of Ford Aerospace and Communications Corporation. This paper appeared in the Proceedings of The Society for General System Research, January 1982. I have never met the authors, but I know we all share similar experiences.

I realize that this is not uniquely tied to the CDR baseline, but I think it is appropriate to treat it here, because that is where, in the development chronology, that it starts to become clear that a problem project may be with you.

A CDR related problem, from which we have evolved a methodology that appears to be a major contribution to our productivity, is in the area of detailed documentation (see Figure 10).

In the beginning we had a great deal of difficulty generating "build-to" specifications in a media that could survive and mature with the design, and not always be out of date the minute that coding began. In the beginning we used conventional flow charts to convey the detailed design at CDR. While they graphically conveyed the design intent they could not be kept up to date by any other means that by redlining of the CDR material, and after you have redlines on top of redlines, you begin to become at least slightly confused.

Today we are using a Program Design Language to form the shell at PDR and provide the details at CDR. In fact the PDL is carried along with the source code and maintained in essence as part of the code.

As we move on from CDR toward our verification baselines, we find ourselves in the middle of the process of both code and debug of the prime item, that is, the deliverable code, and in the process of developing our test data. We are in the part of the development process where we are most sensitive to the computer environment in which we must conduct our development.

Now we have more people affected by how well we have planned to support the development with the right tools.

Here too we had some hard lessons to learn. As I have indicated in Figure 11, this too has a war story to go with it.

In this case, it was associated with software that could function only after the sun went down!

It was, unfortunately, not a characteristic of only one program, it was a malaise that was affecting every program. The typical scenario had the designer attempting to do some form of program verification, only to find out that the results were not repeatable. He would work diligently into the night trying to get, and finally obtaining, repeatable results—only to find out the next day that his code or data base had regressed to some different state. Sabotage, as well as the occult, was considered; but the simple truth was that the controls on shared libraries, and on shared data bases, were not adequate for the development of a large software system. It was not until a global approach to the controlled library structure for the code, and a flexible, yet controlled, data base structure was developed, that mysterious behavior of the software was brought under control.

Figure 12 depicts the library control structure that we have successfully employed for the development and configuration of the delivered product, including the control of multiple versions of delivered products.

Access to the library structure, and connectivity to system data bases, are controlled through the log on mechanism which grants write access to libraries in a hierarchical manner.

It should be noted that, regardless of loca-

tion within the structure, you can always read from above, thereby assuring you access to the best version available.

Finally, with the proper environment in place and with our late requirements closed out, we marched into the verification process. During this process we learned two of the most valuable lessons one can ever learn about the software development process.

The first is that this is a very people sensitive process. Through the dedication of a large number of people, and through their hard work, no matter how frustrating it was, they made it work. These things that I am telling you about are the lessons that they have learned, and the innovative solutions that they generated are the product of their collective genius. I was fortunate enough to be there, and even more fortunate to have survived it in the career sense (not everyone was so fortunate). Hopefully, I have learned enough to be able to communicate it to you, and to others, and hopefully, they can avoid the problems that we had to experience.

The second lesson that we have learned is that we haven't solved all of the problems. Like the chart implies, just as we feel like we have gotten control of the problem, the problem also begins to grow. The systems we are working on now are larger, and we are finding that some of our solutions don't scale up to the new problem—just as our previous solutions were inadequate for the problems we discussed. We find ourselves in a daily confrontation with the issues shown here.

Since I started this out with the first line from "A Tale of Two Cities", I'd like to use another quote from the same Dickens' novel: "It is a far, far better thing that I do, than I have ever done"; and that is to end this case history.

Edward H. Reese, Jr.

GENERAL ELECTRIC

SOFTWARE DEVELOPMENT - PROBLEMS/SOLUTIONS

E.H. REESE

SOFTWARE DEVELOPMENT - PROBLEMS/SOLUTIONS

A CASE HISTORY

MPD FROM 1972 TO 1982:

- "IT WAS THE BEST OF TIMES . . ."
  - LARGEST SOFTWARE DEVELOPMENT OPPORTUNITY THE DIVISION EVER HAD
  - INNOVATIVE PROCESSING CONCEPTS
  - SMALL NUCLEUS OF PEOPLE THAT BELIEVED THAT SOFTWARE COULD BE ENGINEERED
- "IT WAS THE WORST OF TIMES . . ."
  - REQUIREMENTS DEVELOPERS INSENSITIVE TO SOFTWARE DEVELOPER'S NEEDS
  - TOOLS?
  - NAIVETE (DEFICIENT IN WORLDLY WISDOM OR INFORMED JUDGEMENT)

Figure 1. A Case History

SOFTWARE DEVELOPMENT - PROBLEMS/SOLUTIONS

SOFTWARE ENGINEERING AND MANAGEMENT PROCESS MODEL

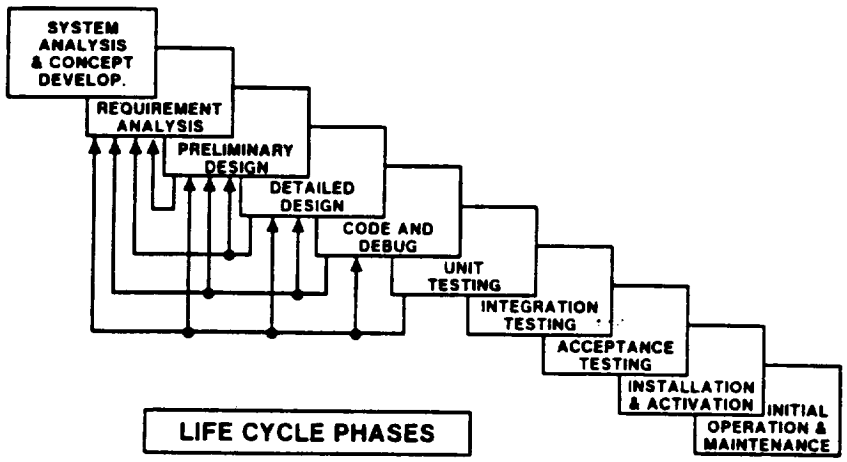


Figure 2. Process Model

SOFTWARE DEVELOPMENT - PROBLEMS/SOLUTIONS

THE PRICE YOU PAY

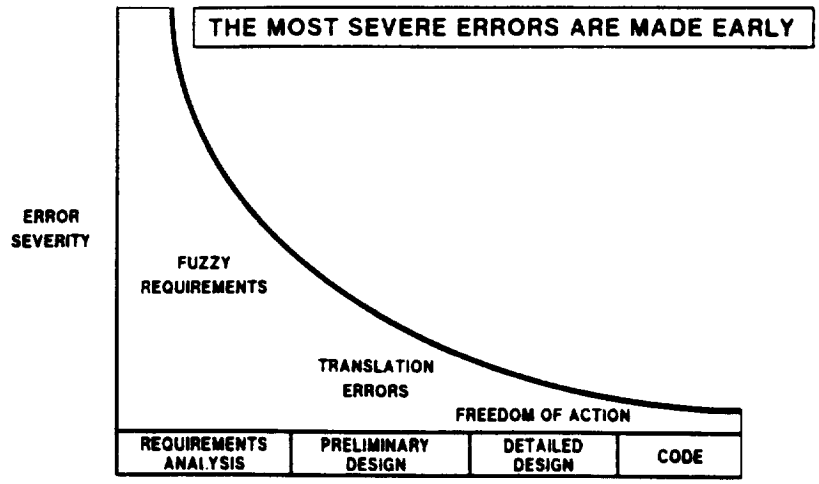


Figure 3. Error Severity vs. Phase

SOFTWARE DEVELOPMENT - PROBLEMS/SOLUTIONS

INSENSITIVITY TO SOFTWARE DEVELOPER'S NEEDS

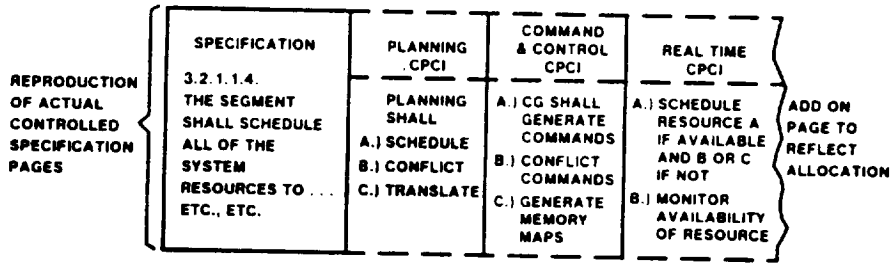
- "YOU CAN'T CODE A #$!#?'ING TBD!"
  - LATE DEVELOPING REQUIREMENTS
  - LACK OF INSIGHT INTO THE COMPLEXITY OF SOFTWARE INTERFACES
- "YES, BUT YOUR ALREADY DOING SOMETHING LIKE THAT!"
  - PROGRAMMERS ARE GOOD, NOT OMNISCIENT
  - FAILURE TO APPRECIATE THE LABOR INTENSIVE NATURE OF PROGRAMMING

Figure 4. Insensitivity to Developer's Needs

SOFTWARE DEVELOPMENT - PROBLEMS/SOLUTIONS

REQUIREMENTS DEVELOPMENT TOOLS

- A SIMPLE METHODOLOGY



- A "SCALING" PROBLEM
  - ~10,000 DRIVING REQUIREMENTS PARAGRAPHS
  - DYNAMIC REQUIREMENTS DEVELOPMENT
  - MULTIPLE END ITEMS (SEVEN SOFTWARE, HARDWARE)

Figure 5. Requirements Development Tools

SOFTWARE DEVELOPMENT - PROBLEMS/SOLUTIONS

REQUIREMENTS DEVELOPMENT TOOLS

THE DATA DICTIONARY APPROACH:

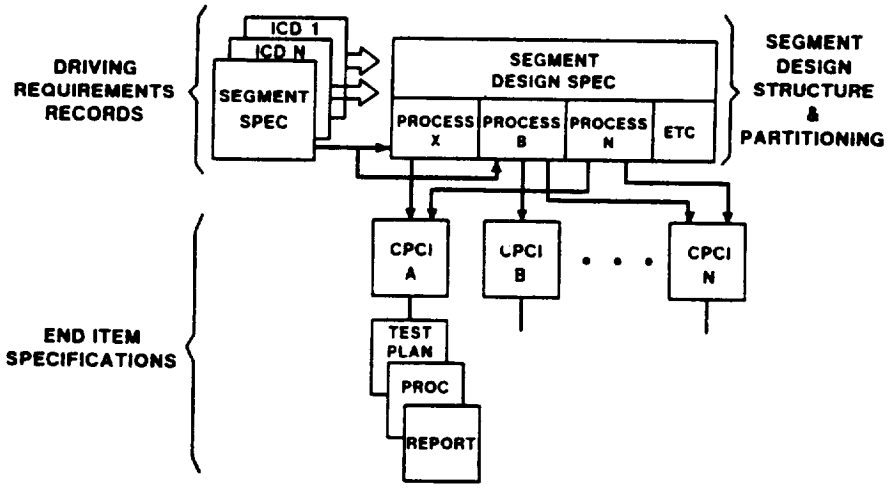


Figure 6. Data Dictionary

SOFTWARE DEVELOPMENT - PROBLEMS/SOLUTIONS

REQUIRED SPECIFICATION TREE

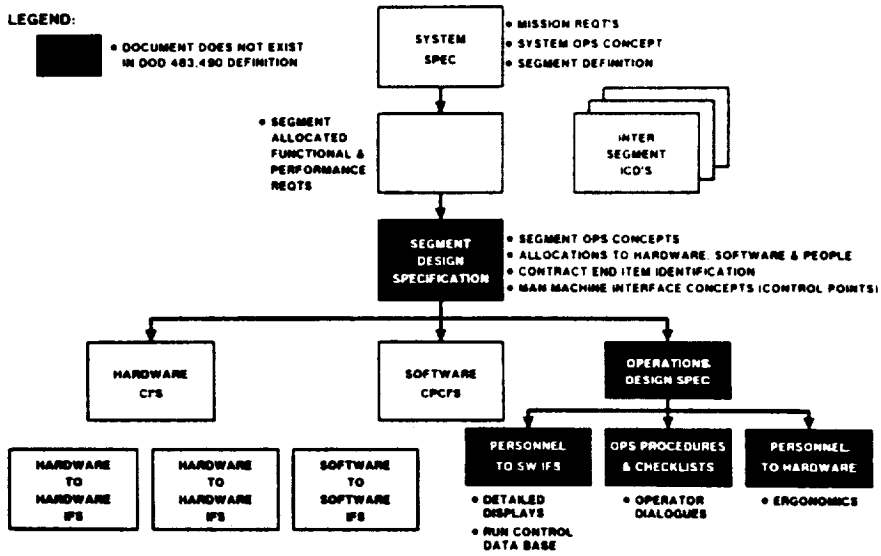
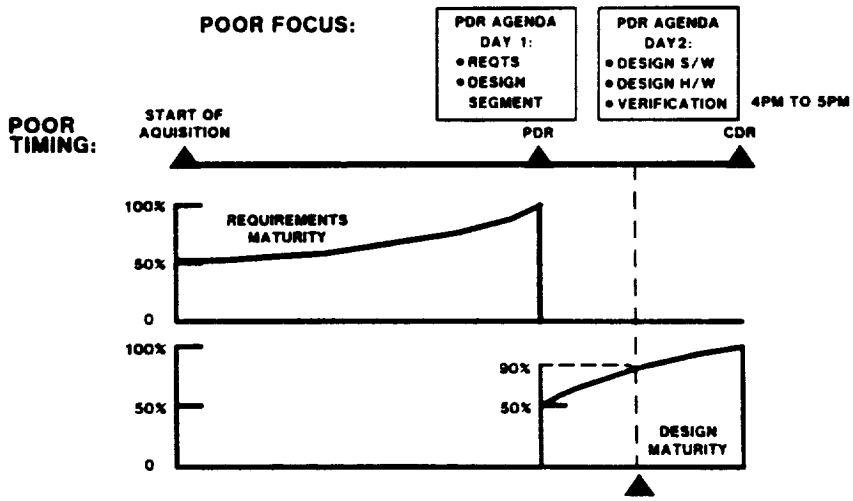


Figure 7. Specification Tree

SOFTWARE DEVELOPMENT - PROBLEMS/SOLUTIONS

VERIFICATION REQUIREMENTS



A BETTER LOCATION FOR A VERIFICATION BASELINE

Figure 8. Verification Requirements

SOFTWARE DEVELOPMENT - PROBLEMS/SOLUTIONS

NAIVETE

- THE SULTAN'S REQUEST
- THE DANGER SIGNALS
  - GENERAL LACK OF MOTIVATION
  - INABILITY TO IDENTIFY PROBLEM CAUSES
  - LACK OF UNDERSTANDING OF PROJECT STRUCTURE AND METHODOLOGIES
  - INABILITY TO SET GOALS & OBJECTIVES
  - OVERALL LACK OF ORGANIZATION AND COMMUNICATIONS

Figure 9. Naivete

SOFTWARE DEVELOPMENT - PROBLEMS/SOLUTIONS

DESIGN PHASE PROBLEMS

LOST

AND FOUND

- DOCUMENTATION MANUALLY GENERATED AND MAINTAINED
  - VOLUMINOUS AND COSTLY
  - MAINTENANCE ACCURACY AND COST
  - COMPLIANCE WITH CONTENT STANDARD
- FLOW CHART PRESENTATION IS OBSOLETE
  - COSTLY TO GENERATE - EVEN MORE COSTLY TO MAINTAIN
  - DIFFICULT TO READ
  - CAN'T MEASURE ACCURACY OR CONSISTENCY
- MEASURING DESIGN QUALITY IS ELUSIVE

Figure 10a. Design Phase Problems

SOFTWARE DEVELOPMENT - PROBLEMS/SOLUTIONS

PDL - PROGRAM DESIGN LANGUAGE

- DESCRIBES PROGRAM DESIGN USING A COMBINATION OF STRUCTURED PROGRAMMING CONSTRUCTS AND REGULAR ENGLISH
- BENEFITS
  - SIMPLE TO LEARN AND EASY TO USE
  - ASSURES PRECISION IN DESIGNING LOGICAL PROCESSES
  - ASSURES STRUCTURED DESIGN
  - AFFORDS BETTER HUMAN COMMUNICATIONS
  - NATURAL TRANSITION TO THE ACTUAL CODE
  - PROVIDES INHERENT DOCUMENTATION - OBVIATES THE NEED FOR FLOW CHARTS
  - SIMPLIFIES MAINTENANCE
- PROVEN EFFECTIVE OVER FLOWCHARTS
  - 45% FEWER ERRORS
  - 20% FEWER RESUBMISSIONS

Figure 10b. Program Design Language

SOFTWARE DEVELOPMENT - PROBLEMS/SOLUTIONS

SDE PHASE I DESIGN TOOLS

PROLOGUE - ASSURES A CURRENT TOP LEVEL DESIGN AND INTERFACE VIEW

- COMMENTS APPEARING AT THE BEGINNING OF EVERY PROGRAM
  - PROGRAM'S PURPOSE OR FUNCTION
  - CALLING ARGUMENTS
  - INPUTS/OUTPUTS
  - DEFINITION OF MAJOR VARIABLES
  - FILES REFERENCED
  - PROGRAM LIMITATIONS
- FORCES A DEGREE OF DOCUMENTATION
  - ASSURES EXISTENCE OF A MINIMUM LEVEL OF DOCUMENTATION FROM INCEPTION
  - STANDARD PRESENTATION OF DATA
  - QA CAN AUTOMATICALLY VERIFY THE PRESENCE OF REQUIRED DATA

CRITICAL DESIGN DATA COMMUNICATED EARLY, COMPLETELY AND ACCURATELY

Figure 10c. Prologue

SOFTWARE DEVELOPMENT - PROBLEMS/SOLUTIONS

DOCUMENTATION SUPPORT SYSTEM

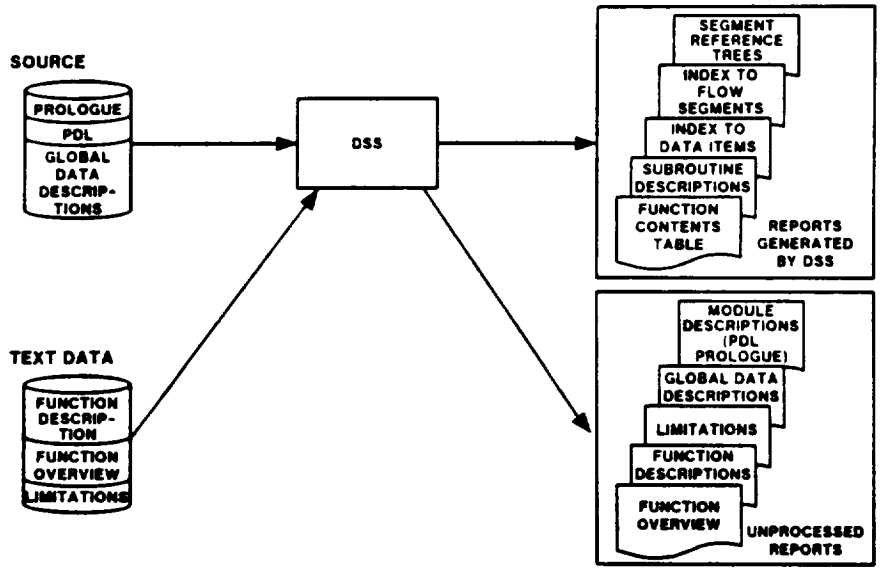


Figure 10d. Documentation Support System

SOFTWARE DEVELOPMENT - PROBLEMS/SOLUTIONS

TOOLS?

- "DRACULA'S DATA GENERATION"
  - SOFTWARE THAT ONLY FUNCTIONS AT NIGHT
  - CONFIGURATION MANAGEMENT

    ⬇

    CONTROLLED LIBRARIES
  - VIRTUAL DATA BASES

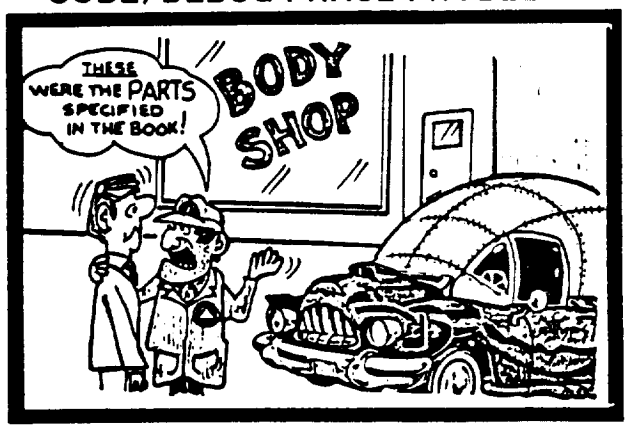
    ⬇

    FLEXIBLE DEVELOPMENT ENVIRONMENT

Figure 11. Tools

SOFTWARE DEVELOPMENT - PROBLEMS/SOLUTIONS

CODE/DEBUG PHASE PROBLEMS

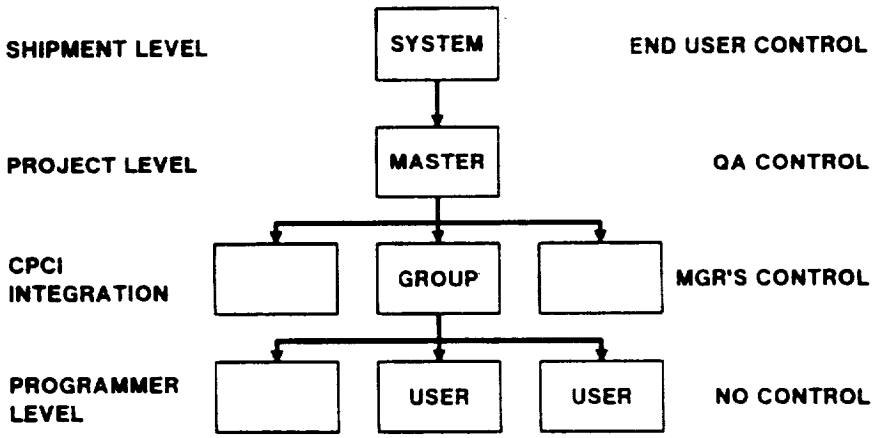


- CONFIGURATION MANAGEMENT IS A MAJOR CHALLENGE
  - CAPABILITY TO DETERMINE THE PRECISE CONFIGURATION OF ALL PROGRAMS AT ALL TIMES
  - PARALLEL DEVELOPMENT OF DIFFERENT PARTS OF THE SYSTEM
  - PERFORM MAINTENANCE IN PARALLEL WITH UPGRADES
  - CAPABILITY TO CONTROL ACCESS TO SOFTWARE FOR CHANGE AUTHORIZATION
- MAKING SURE CODE REPRESENTS THE DESIGN - NO LESS, NO MORE

Figure 11a. Configuration Management

SOFTWARE DEVELOPMENT - PROBLEMS/SOLUTIONS

CONTROLLED LIBRARY STRUCTURE



- PASSWORD PROTECTION
- UPWARD CONCATINATION
- FORMAL BUILD PROCESS: GROUP -- MASTER; MASTER -- SYSTEM

Figure 12. Controlled Library Structure

SOFTWARE DEVELOPMENT - PROBLEMS/SOLUTIONS

LESSONS LEARNED

PEOPLE SENSITIVE PROCESS

- DEDICATION
- INNOVATION

Figure 13. Lessons Learned

SOFTWARE DEVELOPMENT - PROBLEMS/SOLUTIONS

THE TROUBLE WITH WAR STORIES . . .

- THE WAR ISN'T OVER
  - THE PROBLEMS ARE BIGGER
  - THE TOOLS AREN'T THERE YET
  - IT'S STILL LABOR INTENSIVE

Figure 14. War Stories

AN OVERVIEW OF SOFTWARE COST ESTIMATING METHODOLOGY

AT THE NATIONAL SECURITY AGENCY

BY

THOMAS F. MASTERS, II

Currently at the National Security Agency (NSA), we use four major software models: RCA's PRICE S, SLIM, COCOMO, and JENSEN. We have found that because of the four basic methods of cost estimating done at NSA (the Delphi Technique or Expert Opinion, Analogy, Parametric Estimating, and Industrial Engineering Method) and because of the many contractors we have, it is difficult to retrieve data on historical projects. Therefore, the Office of Programs Analysis has been tasked to develop: 1) A Data Item Description (DD-1664) on System Parametric Data Report for our contractors, and 2) An NSA Circular on Cost Breakdown Structure for our systems.

This paper will discuss "The System Parametric Data Report" (DID DD-1664) and the reasons for the establishment and use of this report by our contractors at Initial Operational Capability (IOC). In addition, the report will include a description of NSA's milestones compared with the four major software models that are used in the acquisition process. Experience factors or rules-of-thumb that we apply to these software models will also be described. Finally, an overview of the current automative processes concerning the four stages of NSA analysis will be discussed (Decision Analysis, Parametrics, Program Management and Control Systems (PAC II) and Cost/Schedule Status Reporting (C/SSR)).

"AN OVERVIEW OF SOFTWARE COST ESTIMATING METHODOLOGY AT THE NATIONAL SECURITY AGENCY"

THOMAS F. MASTERS, II
DoD, NSA
Code: N32
Ft. George G. Meade, MD 20755

ABSTRACT

Currently the National Security Agency (NSA) uses four major software models: RCA's PRICE-S, SLIM, COCOMO, and JENSEN. It has been determined that because of the four different cost methods implemented at NSA (the Delphi Technique or Expert Opinion, Analogy, Parametric Estimating, and Industrial Engineering Method) and because of the many contractors we have, it is difficult to retrieve data on historical projects. Therefore, the Office of Programs Analysis is in the development to produce: 1) a Data Item Description (DD-1664) on System Parametric Data Report for our contractors, and 2) an NSA Circular on Cost Breakdown Structure for our systems.

This paper will discuss NSA's Acquisition Milestones and the evolutions of these processes. The NSA Acquistion Milestones will be compared with the four software cost models; reasons for the establishment of "The System Parametric Data Report" will also be presented. In addition, Software Rules-of-Thumb or experience factors which the Agency applies to these software models will be described. Finally, some suggestions for "Monitoring Cost" with a minimum set of "tools" necessary to allow good cost estimating is explained.

Any views expressed in this paper are those of the author. They should not be interpreted as reflecting the views of the Office of Programs Analysis or the official opinion or policy of the National Security Agency.

TABLE OF CONTENTS

		PAGE
I.	INTRODUCTION	1
II.	NSA ACQUISITION MILESTONES	1
III.	SOFTWARE COST ESTIMATING MODELS	4
IV.	SOFTWARE RULES-OF-THUMB	8
V.	MONITORING COST	13
BIOGRAPHY		16
REFERENCES		17

APPENDIX A: Software Life Cycle Milestones

APPENDIX B: System Parametric Data Report (DD1664)

APPENDIX C: Overall Program Methods

I. INTRODUCTION

This paper presents a short overview of the parametric software cost estimating used at NSA. The primary motivation for this paper is to help bring an understanding of the cost consequences of decisions that parametric estimators will make in the software development cycle and applied experience factors to the requirements and maintenance life cycle.

Software estimating is an art not an exact science. Software estimating models have become a management tool but are not a substitute for estimating experience and intelligence. Parametric estimating techniques apply accurately only to some of the development portion of a software life cycle. They should not be represented as total software life cycle, that would be applicable to the requirements or maintenance. For requirements or maintenance estimation accuracy, no present estimating model is of real value. Experience factors can produce a rule-of-thumb percentage for those two phases of the life cycle, and a simple percentage can be included into the total software life cycle.

II. NSA ACQUISITION MILESTONES

First, I should quickly point out that what I call the "NSA process" is a distillation of a number of general standards and MIL-SPECS ("NSA Software Acquisition Manual"), so it is probable that other DoD Organizations follow a very similar process when acquiring software. Another point should be made; most contractors have similar policies, and NSA's process represents an evolution, not a revolution. Many documents from private industry were used as reference material for the "Software Product Standards Manual." The software development policies of the Defense and Space Systems Group at TRW were particularly useful.

The primary DoD references upon which the NSA Software Acquisition Process is based are:

1. DoDD 5000.1, "Major System Acquistions"

2. DoDI 5000.2, "Major System Acquisition Procedures"

3. AFR 800-14, Vol. II, "Acquisition and Support Procedures for Computer Resources in Systems"

4. MIL-S-52779A, "Software Quality Assurance Program Requirements"

5. MIL-STD-483 (USAF), "Configuration Management Practices for Systems, Equipment, Monitors, and Computer Programs"

6. MIL-STD-1521A, "Military Standard Technical Reviews

Appendix A, Figure 1 depicts the total system milestones in relationship with the four software cost models, and what I believe DoD milestones are applicable to these models. Appendix A, Figure 2 depicts the sequence of software development covered in the NSA Software Acquisition Manual (NSAM 81-2). It also identifies documents required by the manual and shows the sequence in which they shall be prepared and reviewed.

In Appendix A, Figure 1, NSA has four System Concept Plans (SCP I through SCP IV) and a System Acquisition Plan (SAP), before the Purchase Description (PD) or solicitaion of Request For Proposal (RFP) is sent out to our contractors. After Contract Award (C/A), four major documents are produced. They are: Development Plan (DP), Configuration Management Plan (CMP), Quality Assurance Plan (QAP), and the Software Requirements Specifications (SRS). Software Requirements Reviews are on going, and a generation of preliminary Software Development Test and Evaluation (DT&E) Plans are provided at this phase.

During the time from Preliminary Design Review (PDR) to Critical Design Review (CDR), the Software Program Specifications (SPS) are generated. Written within these specifications is a description of the overall design of the computer program and its detailed components. An expansion of the system/sub-system specification is essentially conducted to produce a detailed Program Specification that serves as the "coding" basis for the software end products. These specifications provide an audit trail of requirements from design to code and subsequent verification of the software.

At CDR, a review of Program Specification and any critical issues are resolved. An agreement is made on the accuracy of the detailed design and plans, and a commitment may be obtained to proceed into the implementation phase of the project.

At this point, from CDR to DT&E, there are three phases:

1. A unit development period where the software units are coded and tested.

2. A software testing period where software units are integrated and tested and where the final computer program is subjected to Software Development Test and Evaluation (DT&E I).

3. System Build and Integration period whereby the hardware and software are integrated and the system is subjected to DT&E. This step encompasses the fabrication, assembly, and test of the various parts of the system components and verification of the overall system's readness to start formal Development Test and Evaluation (DT&E II).

The Development Test and Evaluation (DT&E I and II) phases are unique versus DoD milestions. These phases are conducted to ascertain whether or not the system meets its technical requirements (both hardware and software) and is ready for Operational Acceptance Test and Evaluation. During the DT&E I, the system is tested at the developer's facility. When satisfactorily completed, the system is available for initial operations and assumption of responsibility for ongoing operations and maintenance. Authorization to acquire additional versions of the system is contingent on achieving an acceptable operational capability.

At DT&E II, all effort is devoted to final system acceptance test and at Initial Operational Capability (IOC), the DD-250 is signed-off.

Appendix A, Figure 1, dipicts these software models not covering all of the Agency's milestones. What they do cover is approximately from the Software Requirements Review (SRR) (which is after the Contract Award), and stops at our first Development Test and Evaluation (DT&E I) phase. Therefore, we must include additional percentages to the Front-End and Back End Development Cost to each model (see IV. Software Rules-of-Thumb).

III. SOFTWARE COST ESTIMATING MODELS

Over the past two years, the Office of Program Analysis has reviewed the state of cost modeling for both hardware and software throughout the DoD and private industry. In regards to software cost estimating models, we have reached the following conclusions:

1. A large number of software cost estimating models have been developed over the last 10 years. The vast majority of these use "lines-of-code" as their primary input.

2. Software parameters to which cost is most sensitive are size, schedule, manpower and difficulty. Because of operational requirements, schedule is the most critical cost parameter.

3. Comparative studies of the most frequently used models conclude that no one model is superior to the rest (see reference 1.).

4. Analogy modeling approaches are most valuable early in the life cycle (e.g., SCP I through SAP), while parametric approaches are most valid during the latter portions of the life cycle (e.g., SRR through DT&E I).

5. No one model contains all of the significant cost drivers for NSA programs (e.g., site preparation, training, etc.).

6. Calibration is the "key word" for any model. To determine if the model is a predictor, one must have the actual cost parameters and the technical characteristics of the completed system. System parameters that are collected and used for subsequent applications of the models must be consistant with the models.

7. Industry models typically are not scoped to encompass all the activities government must estimate (requirements generation, source selection, etc.).

Based on our review of numerous papers and discussions with many knowledgeable cost analysts in DoD and industry, and based on limitations of cost, time, and utility of more models; we have decided to use four of the most widely-used cost models. They are:

a. RCA's PRICE S

b. PUTNAM (SLIM)

c. BOEHM (COCOMO)

d. JENSEN (JS1&2)

Each of the first three models have a different underlying structure, while Jensen is similar in structure to Putnam.

I personally feel that computerized parametrics is the "WAVE OF THE FUTURE." However, there are several basic points that should be addressed. What is a good model? I believe they should possess the following characteristics:

1. Interactive: The model and computer system should be interactive. The user and terminal should provide a mutual or reciprocal action or influence.

2. Conversational: Special terminology should be avoided. The user(s) must be able to communicate with the software without the barrier of specific computer jargon.

3. Limited Inputs: Any computer model should restrict the inputs and provide many output alternatives for decision making.

4. Capable of Providing Internal Cross-Checking: The model should provide internal cross-checking to analyze the input data for consistency and continuity.

5. Easy to Calibrate: A good model allows the user to calibrate, or tailor, the model to their specific organizational needs.

6. Definitional: Terms defined unambiguously.

7. Fidelity: Predictions represent reality plus or minus.

8. Stable: Model can handle singularities and side effects.

9. WHAT-IF PROCESSING: Handles "REAL WORLD" variations in sensitivity analysis.

There are in excess of 400 computer parametric models in the United States and the free world that deal with hardware, cost and price. There are many more dealing with software development, software life cycle, and hardware life cycle. It is my personal view that one must use at least two or more parametric software models. Observe the distortions (if any) in the variations of parameters (e.g., schedule, risk, performance, complexity, etc.). Then, observe the final output, "cost data."

There are many reasons for poor estimates. These are but a few of the reasons:

1. There is lack of agreement on terminology.

2. The nature of the software or system is not well understood.

3. The full scope of the job is not quoted.

4. Estimators assume "ALL WILL GO WELL" and DO NOT connect risk with cost.

5. There is a lack of actual cost data.

6. There are unsophisticated users. Someone who periodically plays with a software model or who knows the basic calculations of any one model and delivers a cost estimate as the final solution to a project has not completed the task correctly. BEWARE!

Reliable cost estimation can be achieved if effort is invested in creating a meaningful estimation methodology (e.g., Hughes - Fullerton reports it can estimate within a 10% precision, as do TRW and many many other firms.). Justifying any investment requires hard data which should be collected routinely.

Estimates are the foundations of project plans. However, nobody likes to make an estimate, because:

1. It could be wrong, and

2. Someone will remind you when it is proven that you were wrong.

Remember - An estimate is your judgment of costs, schedules, resources, etc.; it is NOT a guarantee. To estimate, one has to forecast what the world and his part of it will be in the future. There is no magic formula. Since forecasting is not perfect, adjustments will have to be made as the future becomes clearer.

The government must provide an independent cost estimate and analysis for the total system prior to the solicitation or Request For Proposal (RFP). This is based upon various sources of information within the System Concept Plans. I believe that we have better than good software requirements. However, even though we build a system for the first (and may be the last) time, the technology and development in the conceptional stage is approximately ninety percent unknown.

In my opinion, we have various methodologies for interpreting the inputs of any software model. Likewise the contractor can only interpret, in so many ways, what we are asking for in our RFP's. However, it has become too prevalent among software contractors to use a parametric estimating system output solely to validate the software development bid they already have in mind to submit to the government. In order to obtain the output which is to be used to substantiate their software development bid, some contractors frequently, and often openly, tamper (or "dry lab") with the inputs so as to have the model produce the output which they will submit to the government. This conduct can be and often is responsible for serious cost overruns and schedule slippages for the government and for their company many months later.

As mentioned before, calibration is the "key word" for any model. To determine if the model is a predictor, one must have the actual cost parameters and the technical characteristics of the completed system. System parameters that are collected and used for subsequent applications of the models must be consistant with the four software models. Therefore, the Office of Program Analysis is the process of developing a "System Parametric Data Report/Data Item Description, DID UF-5596-A, DD-1664." See Appendix B. Currently, this DID is a working paper, and it is anticipated that this report will be available by the end of FY83.

Even though this DID is not official yet, our contractors will be required to submit the Parametric Data Report within 60 days after IOC. However, R&D organizations may require this report at various milestones early in the life cycle of the system. For hardware, if the project is naturally defined by one or more Line Replaceable Units (LRU represents the box or equipment level), the Air Force form ASD-218 (Parametric Data Sheet), and NSA's supplement will be filled out for each LRU.

For software, the contractor will fill out the parameter sheets for any developed or modified software that will be integrated into the total system and/or that is defined in the System Design Plan. The contractor will not fill out the software section for any off-the-shelf (OTS) software. The software input sheets are made so there are no one standard input parameter sheet(s) for any one independent model. What we have attempted to do is to consolidate all four software models into a few pages. There will always be enhancements; but for now, we believe we have the necessary essential parameters to build an data base (e.g., actual schedule dates, the contractor staffing level(s), dollars per manyear, etc.).

IV. SOFTWARE RULES-OF-THUMB

The simplest form of cost estimating model is the so-called "rule-of-thumb." It usually takes the form of a multiplicative factor that converts some output/product measure into the effort/dollars required to achieve that output. Examples include:

- lines-of-code per man-month (software "productivity"),
- dollars-per-line-of-code,
- dollars-per-page-of-documentation,

1. Productivity. One of the most widely-used methods of estimating software development costs is to first size the project in terms of number-of-lines-of-code, multiply this by a productivity factor (e.g., man-months per line-of-code), and then adjust the cost based upon parameters to which the cost is sensitive.

There are several major problems associated with this technique:

First, lines-of-code (LOC) estimates are notoriously inaccurate, and the term takes on many different meanings (i.e., unique LOC, no comments, etc.).

Next, productivity values range widely from 12 to 8000 LOC per man-month (taken from the RADC data base on over 400 software projects). The scope of what activities they encompass and what environment they were developed in is often unknown.

The problem with productivity is the wide variability in its definition, specifically in the determination of what to count in the man-months value in the denominator (does it include program management, systems engineering, etc., or just the programmers effort?). Does it span the entire development cycle, or just count the man-months devoted to coding and unit testing?

Finally, most productivity models baseline production rates on the past and not the future. If the Agency is to fund productivity improvement projects for industry, it must be shown payback on its investment. Therefore, productivity improvement must be factored into future system's costs.

Basically, we should view with considerable suspicion any productivity data that we obtain from outside sources, unless we can determine just what it does and does not include.

There are efforts underway to develop a standardized productivity values for types of software developments (both in-house and contracted), but at this time, we have no historical data on which to base an estimate. However, based on a review of a number of in-house studies and opinions of experienced industry software developers, productivity is defined as:

$$P = \frac{\text{total-number-of-derived source lines-of-code (including comments)}}{\text{total man-months to develop system (all software-related functions from C/A to IOC)}}$$

A range of 90 LOC/mm $\leq$ P $\leq$ 360 LOC/mm can be expected, with the lower values occuring for new efforts and/or new personnel and/or high reliability.

The major factors which would affect the ultimate productivity for a given job are:

1. Application (real-time C^2, operating system, business application, etc.)

2. Familiarity of personnel with application.

3. Familiarity of personnel with language.

4. Experience of personnel.

5. Size/complexity of job.

6. Required reliability of software

7. Percent-of-hardware CPU memory/cycles used.

8. Management

9. Risk

10. Degree of modern programming practices employed.

11. Degree of software tools employed.

(Note: Dr. Barry Boehm, TRW, indicated in his book "Software Engineering Economics", with last two items, TRW is to reach 100% productivity by these methods. In 1985, 300% and in 1990, 500% productivity.)

12. Degree of compatability of hardware development.

From a dollar standpoint, the range of productivity varies from $70-per-line-of-code to $150-per-line-of-code. In summary, productivity rules-of-thumb can provide you with order-of-magnitude estimates, but cost estimating errors of 100 to 200% are possible.

PRODUCTIVITY FOR HIGH ORDER LANGUAGE (HOL):

JENSEN :	50-150 DSLOC/mm = $85 - $125/DSLOC/mm
SLIM :	100-150 DSLOC/mm = $70 - $ 90/DSLOC/mm
PRICE S:	100-150 DSLOC/mm = $85 - $150/DSLOC/mm
COCOMO :	UNKNOWN

2. UP-FRONT AND BACK-END DEVELOPMENT COST:

All four software models typically start around our Software Requirements Review (SRR). Therefore, we must add in a percentage into the up-front and back-end development cost. They typically are:

UP-FRONT DEVELOPMENT COST
(Contract Award to SRR) :

JENSEN - 7 to 10%
(Starts at SRR)

SLIM - 7 to 10%
(Starts after SRR)
(man-month * monthly burden uninflated rate)

PRICE S - 6 to 12%
(Starts before SRR)

COCOMO - 6 to 15%
(Starts at SRR - additional calculation of P&R to A&C is needed)

BACK-END DEVELOPMENT COST - Likewise, all software models stop around our first Development Test and Evaluation (DT&E I) phase. A typical percentage from DT&E I to the Initial Operational Capability (IOC) is needed. They are:

(From DT&E I to IOC) : JENSEN - 15 to 25%
(FQT stops around Build/I&T)
SLIM - 10 to 20%
(FOC stops at DT&E I)
PRICE S - 10 to 25%
(I&T can stop at Build/I&T)
COCOMO - 10 to 25%
(S/W ACCEPTANCE stops at Build/I&T)

3. Additional rules-of-thumb when using these models are:

a. CALCULATION OF HOURS/MONTH OR HOURS/MAN-YEAR:

Contractor: 149.7 hours/month or 1796 hours/my

In-house : Civilian - 149 to 151 hrs/mo or 1802 hrs/my
Military - 144 to 146 hrs/mo or 1742 hrs/my

b. FULLY LABOR BURDEN RATES ($/MY):

EAST COAST - $100 to $130k
WEST COAST - $130 to $160k

c. INFLATION (LABOR FY83):

APPROX. EAST COAST - 8.3 to 9%
WEST COAST - 10 to 11%

d. DOCUMENTATION:

SLIM: Look at average number of pages * $120/pg.

PRICE S: For Design, Implementation, I&T for Documentation Cost divide by average number of pages times approx. $130-$160/pg.

JENSEN: UNKNOWN

COCOMO: UNKNOWN

APPENDIX A

SOFTWARE LIFE CYCLE MILESTONES

When planning a software project, Reifer[1] believes the percentage of the cost allocated to the following activities for independent development and contractor development is as follows:

	DEVELOP YOURSELF	NORMALIZE	DEVELOP BY CONTRACTOR	NORMALIZE
° SYSTEMS ANALYSIS	9.1%	4.5%	12.12%	4.6%
° S/W ANALYSIS & DESIGN	24.24%	12.2%	24.24%	9.3%
° CODE & UNIT TEST	12.12%	6.1%	12.12%	4.7%
° S/W TESTING	24.24%	12.2%	24.24%	9.3%
° INTEGRATION & SYSTEM TEST	18.18%*	9%	24.24%*	9.3%
° MAINTENANCE	**	50%	**	50%
° MANAGEMENT	6.06%	3%	18.18%	7%
° DOCUMENTATION	6.06%*	3%	15.16%	5.8%
	100%	100%	130.3%	100%

* Of S/W Development Effort
** Equal to Development Cost

[1] Reference: D.J. Reifer, Reifer Consultants, Inc., May 1983

PAGE 78 INTENTIONALLY BLANK

V. MONITORING COST

There are several points on monitoring cost. First, invest in a standard methodology (e.g., Standard WBS/CBS and a Parametric Reporting DID's can serve as a good basis, also formalize steps and train personnel). Collect data as the norm on all projects. Use it to refine the methodology. Identify cost drivers and bound the risk. The cost variation is a function of the life cycle and the risk can be bound via confidence regions. Use your WBS/CBS and DID's to keep track of cost, and always ask for a "Cost to Complete" estimate.

Second, Reifer[1] comments indicate there are controllables versus noncontrollable variables. They are:

CONTROLLABLES	NONCONTROLLABLES
° Software Deliverables	° Schedule
° Software Standards	° Complexity
° Programming Practices	° Development Personnel Capabilities & Stability
° Management Practices	° Personnel Conflicts
° Management Team Capabilities	° Inexperience
° User Interference	° Hardware Capabilities
° Resource Growth	° Legal and Environmental Factors
° Hardware Configuration	

Third, we can derive a parametric cost estimate if we have:

1. The existence of historical cost/parameter information about similar systems.

2. The ability to predict with some degree of probability or confidence the expected parameters of our future alternatives; (e.g., the correct milestones of each software/hardware model covers, experience level of contractors, etc.).

3. A competent analyst who can tell us if the historical cost of the like systems do vary in some defined way with the chosen parameters of a cost estimating relationship (CER). If they do, he/she will give us the estimating equation.

Finally, DO NOT become so enamoured with an estimating model that you ignore the assumption made in its development, and the reliability of the sample input data. A computer will furnish an impressive and detailed readout, even if the input data is unreliable. Carefully scrutinize sample data, data sources, and assumptions made in developing parametric estimates.

In addition to parametrics, we are complementing our tracking processes with additional automated analysis. Examples of these program analysis include; Decision Analysis in the early stage of our life cycle. At Contract Award, the use of Program Management and Control Systems (PAC II); and Cost/Schedule Status Reporting (C/SSR) or Cost Performance Reporting (CPR). In Appendix C, Overall Program Methods is an brief discussion on these processes.

If one thinks about it for a few seconds, it quickly becomes obvious that the key to accurate cost estimating is having access to a data base (hopefully as current as possible) of cost information on the "equipment(s)" one intends to buy.

In the case of "off-the-shelf" equipment, the "data base" can be equipment catalogs. Unfortunately, when hardware is being built for the first (and maybe the last) time, or new software is being written/modified, no "catalogs" exist per se. Instead, a more generic data base must be used, one that contains fundamental descriptors (parameters) of hardware and software systems.

Given a set of physical or functional parameters which define a system, (e.g., number of lines-of-code, number of printed circuit boards, size, weight, volume, field-site deployed, etc.), the other tool necessary for good cost estimating is one or more parametric models relating the ultimate development cost of the item to the parameters that define the item. Clearly, for any model to have a chance at being a good predictor, it must be based on data from systems already developed, i.e., systems whose parameters and actual costs are known. The process by which a model is tailored to a particular class of systems with the intent of making it a better predictor is called calibration. This process requires calibration of completed projects with actual costs. This is the major reason why we are having our contractors complete the "System Parametric Data Report" at IOC.

In summary, the minimum set of "tools" necessary to allow good cost estimating is:

- An accurate up-to-date analogous data base containing completed systems parameters with risk and cost.
- A collection of cost models that predict development of new systems, given estimates of the parameters of the new systems.
- Planning tools to identify risks.
- A standard methodology to help you make meaningful comparisons.

"THERE ARE NO SIMPLE SOLUTIONS - ONLY INTELLIGENT CHOICES"

REFERENCES:

1. An Analysis of the State-of-the-Art of Parametric Software Cost Estimation Models, D. J. Reifer and D. D. Galorath, Reifer Consultants, Inc., 1981.

2. An Evaluation of Software Cost Models, R. Thibodeau, GRC, May 1981.

3. A Poor Man's Guide to Estimating Software Costs, D. J. Reifer, Reifer Consultants, Inc., 1 February 1982.

4. NSA Cost Estimating Handbook, NSA,OPIN32, NOV 1982.

5. An Overview of Cost Estimating Methodology, T. Masters, NSA, 1982.

6. L. Putnam, President, OSM, May 1983.

7. Dr. B. Boehm, TRW, May 1983.

8. Dr. R. Jensen, Hughes, May 1983.

9. Dr. R. Park, RCA PRICE Systems, May 1983.

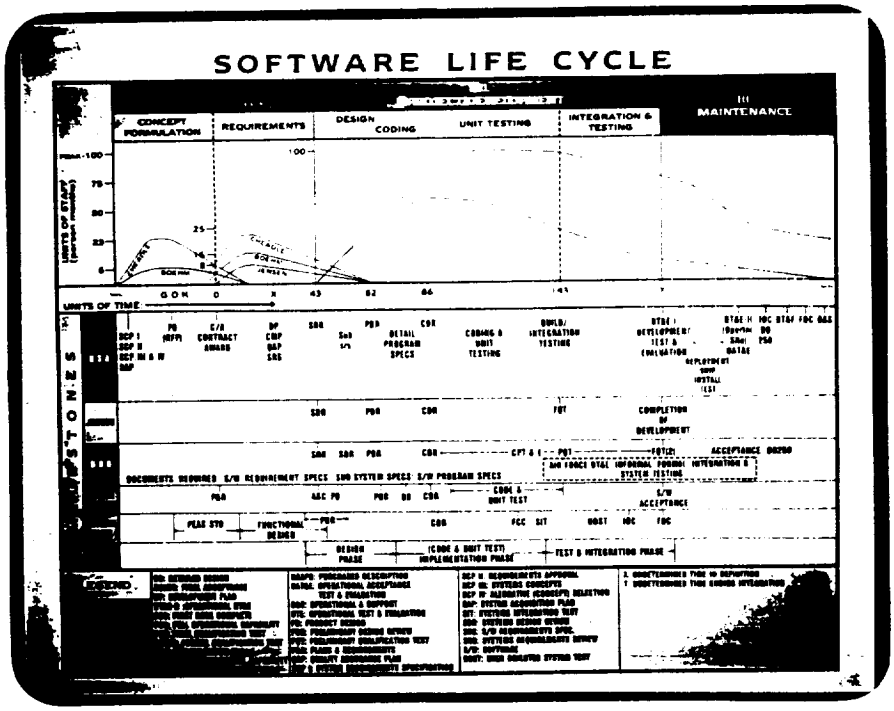


FIGURE 1

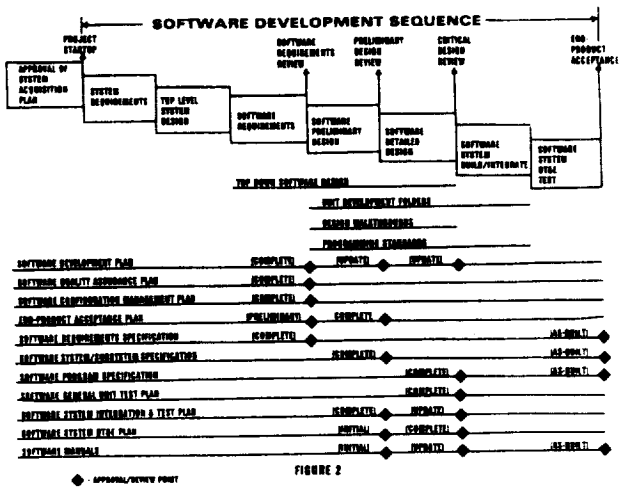


FIGURE 2

DATA ITEM DESCRIPTION	2 IDENTIFICATION NO(S).	
	AGENCY	NUMBER
1. TITLE System Parametric Data Reports	NSA/CSS	U-F-5596-A

3. DESCRIPTION/PURPOSE

3.1 To accumulate descriptive information on the physical and operational characteristics of a system which, when combined with cost data will provide information for effective cost estimation for future systems.

4. APPROVAL DATE

15 April 1982

5. OFFICE OF PRIMARY RESPONSIBILITY

N32

6. DDC REQUIRED

8. APPROVAL LIMITATION

7. APPLICATION/INTERRELATIONSHIP

7.1 This data item provides the data necessary to exercise the PRICE-H, PRICE-S, SLIM, JENSEN, and COCOMO cost estimating models as well as data to create cost estimating relationships for cost elements not included in these models.

7.2 Interrelated Data Items:

DI-F-6010A Cost/Schedule Status Report
DI-F-6000C Cost Performance Report

9. REFERENCES *(Mandatory as cited in block 10)*

MCSL NUMBER(S)

10. PREPARATION INSTRUCTIONS

10.1 The contractor shall prepare the System Parametric Data Reports for system hardware (equipments) and system software. The reports shall conform to the requirements and format in the paragraphs and figures that follow. Each element of data in each report shall be completed as it is applicable to those end products to be developed for delivery. Modified off-the-shelf (OTS) devices and modified software are to be considered as developed items. This report shall consist of two sections, Hardware Parametric Data and Software Parametric Data.

10.1.1 System Hardware Parametric Data Report - for a single conceptual overall system (hardware equipment). Section I, System Hardware Parametric Data Form (page 2-4 of 21) shall be used. If the project is naturally defined by one-or-more Line Replaceable Units (LRU - represents the box or equipment level), ASD218 (Parametric Data Sheet) (Figure 5) and NSA Supplement shall be filled out for each LRU.

10.1.2 System Software Parametric Data Report - Section 11 (pages 5- of) shall be filled out for any developed or modified software that will be integrated into the total system hardware (equipments) and/or that is defined in the System Design Plan. The contractor shall not fill out Section 11 for off-the-shelf (OTS) software.

10.1.3 The contractor will submit the Parametric Data worksheets within 60 days after IOC.

DD FORM 1 JUN 68 1664

PAGE _____ OF _____ PAGES

SOFTWARE COST ESTIMATING PARAMETERS

1. Estimate Lines of Code (LOC) _____ _____ _____ for each system or subsystem.
   Low Avg. High

2. Type of Programming Language(s) and % of System _____ _____ _____

3. System Attributes (Check off)

   _____ Batch
   _____ On-line communication
   _____ Network
   _____ Network
   _____ Distributed
   _____ Local
   _____ Real time command & control/aid time
   _____ Data /Command Lang.

   Applications

   _____ Data storage & retrietal/non time constrained
   _____ Telecommunications
   _____ Interactive operations/time constrained
   _____ Scientific systems/mathematical operations
   _____ String/data manipulation
   _____ Operating system/logical
   _____ DBMS
   _____ Business systems
   _____ Process control

4. System Description

   _____% New systems interfaced with existing systems
   _____% New systems stand alone
   _____% Rebuild system by recoding, integration & interfaces
   _____% Composite system with independent subsystems with considerable overlag
   _____% Composite system with independent subsystems in virtual parallel

5. Programming Practices (<30% minimal, 30% to 60% average, >60% extensive)

   _____% Structured Programming
   _____% Chief Programmer Teams
   _____% Experience with Programming Language
   _____% Design & Code
   _____% Overall Skill & Qualifications
   _____% Experience with similar systems (computer designs and applications)
   _____% Top Down Development
   _____% Experience with development

6. Fully Burdened Labor Rate _____ ($ 1 man year)

7. Response Requirements (must equal 100%)

   _____% Real time code
   _____% On line
   _____% Time constrained
   _____% Non time constrained

8. Anticipate Staff Leading (people)
   _____ Minimum _____ Average _____ High

9. Schedule (Enter start and finish dates - month/yr. - for entire software program. Enter intermediate dates if cases are unusual or gaps exist.)

Start of Contract: _____
Start of detailed Design Phase (PDR): _____
Start of Implementation: _____
Start of Test and Integration: _____
Start of System Design Review (SDR): _____
End of Design Phase (CDR): _____
End of Implementation: _____
End of Test & Integration: _____

10. Operating Environment (Check appropriate box to explain in what environment software is going to operate. Keep in mind this describes not so much the location of S/W, but specifications it must meet.)

NSA in-house internally developed		NSA/SCE Field Site		Non MIL-SPEC Avionics		Unmanned Space	
NSA in-house contracted software		Military Mobile (van or shipboard)		MIL-SPEC Avionics		Manned Space	

11. System Integration and Test (For this subsystem, estimate the relative level of engineering, programming and testing required to integrate the subsystem into the total unified software system.)

Simple _____ Routine _____ Difficult _____ Complex _____

TECHNOLOGY CONSTANT

1. INTERACTIVE FACTOR - MEASURES OF A DEVELOPER'S SOFTWARE PRODUCTION CAPABILITY

   0 = Total Batch Program Submittal

   1 =

   2 =

   3 = 1 terminal per 20 programmers

   4 = 1 terminal per 10 programmers

   5 = 1 terminal per 5 programmers

   6 = 1 terminal per 4 programmers

   7 = 1 terminal per 3 programmers

   8 = 1 terminal per 2 programmers

   9 = 1 terminal per 2 programmers (located in the prog. office)

   10 = 1 terminal per 1 programmer

2. RESOURCE FACTOR

   0 = 1 response per day

   1 = 1.5 responses per day

   2 = 2 responses per day

   3 = 3 responses per day

   4 = 4 responses per day

   5 = 60 minute response time

   6 = 30 minute response time

   7 = 20 minute responses time

   8 = 10 minute response time

   9 = 5 minute response time

   10 = 3 minute response time

3. System Type (Check one)

Production center software -- developed by user

Production center software -- developed by contractor

Satellite ground system with extensive manual checking and intervention

MIL-SPEC ground system

Non-real-time command and control system

Inventory control system

Satellite ground system with major automated functions, little manual checking and intervention

Satellite ground system -- fully automated

Commercial avionic systems

Military avionic systems

Unmanned spacecraft systems

Manned spacecraft systems

4. Documentation Type:

Assign a rating to the Documentation Type envirnomental parameter according to the level or depth of the documentation required for the software development. Select the rating from the following table that is the most consistent with the documentation type required by the development. The amount of documentation required by the project has a pronounced effect on the schedule and cost if the documentation is either inadequate or excessive.

(Check one)

Personal software -- developed by user

Commercial software -- developed by user

Commercial software -- contractor developed with detailed documentation

MIL-SPEC documentation -- complete essential data

MIL-SPEC documentation -- full

Commercial software -- contractor developed, minimal level

MIL-SPEC documentation -- minimal set

High reliability application (public safety requirements)

5. Development Time in Months = _____.

6. Indicate the per-man-year-rating to be used _____.

If none is given, 101K burdened rate will be used for FY83 with a 8.3% inflation rate.

(SRR contract award)	@ Up Front Cost 7% of Development	= _____
	@ Development	= _____
DT&E IOC	@ Back End 10-15% of Development	= _____
	TOTAL	= _____

APPENDIX C

OVERALL PROGRAM METHODS

DECISION ANALYSIS is a computerized aid to problem solving that structures the relationships among the various considerations which must be faced in any decision task. Decision analysis models often include a decision tree or decision diagram. Inputs to such models can include numerical probabilities, that quantify judgments about uncertain future events, as well as numerical assessments that express the decision maker's attitudes, or the organization's policies, as regards the assumption of risk. A model's outputs may include a display of the probabilities of each possible outcome for every action alternative, or a specification of a single course of action to be preferred under the assumptions of the model.

One of the characteristics distinguishing decision analysis from other quantitative tools available to managers is that these other techniques-such as operations research, Bayesian statistics, mathematical programming- embrace much more narrowly applicable classes of models. Specific operations research models, for example, may be useful in selecting warehouse sites, in balancing assembly-line facilities, or in forecasting sales growth; but none of these models will fit the great majority of the large and small decisions faced by managers who must plan and implement business strategies. Decision analysis, on the contrary, can usefully address many problems meriting more than momentary consideration.

Applications have ranged from decisions regarding product development to production facility, pricing, and other marketing and financial problems. This discipline has been uniquely helpful in resolving choices between actions with complex and uncertain future effects-effects that, in themselves, frequently depend in part on subsequent decisions. However, as a systematic approach to making choices, it has contributed valuable insights in many situations that initially seemed barely complicated or substantial enough to merit writing any numbers down on paper at all.

This technique is under evaluation for use during our System Concept Plan II (SCP II), and with the major decision makers participating in a communicative session. The process typically takes about one to two days. A monitor is present and writes the priorities on a board. There is also an analyst who interprets the information and inputs various parameters on an in-house computer.

Complex problems are decomposed into clearly defined components such as options, uncertainties and values. Then these components are used to structure a formal decision model from which a process evolves that can be used to help assign levels of expenditure to different areas or projects; therefore, the greatest overall benefit is gained for the least cost.

PAGE 42 INTENTIONALLY BLANK

PROGRAM MANAGEMENT AND CONTROL SYSTEMS (PAC II)™

This program will start during our conceptional stage, and our contractors will be required to input data. There are two primary capabilities:

1. Information data base which consists of three categories of information:

   a. Project resource information

   b. Project workload and schedule information, and

   c. User defined tables and dictionaries which allow the system to be tailored to local user requirements.

2. Software programs to manipulate the information from the data base. This allows the manager to:

   a. Plan new projects

   b. Determine impact of new projects on existing ones and adjust resources of schedules to minimize impact.

   c. Develop alternatives for new project schedules based on resource use, project priority network dependencies, and end user needs.

This system provides three major levels of standard project reports:

1. Executive level reports

2. Department level reports

3. Project Manager level reports.

In addition, a report writer facility allows managers to specify precisely what they want in a report to meet unique needs.

COST/SCHEDULE STATUS REPORTING (C/SSR) AND

COST PERFORMANCE REPORT (CPR)

The Cost/Schedule Status Report (C/SSR) is applicable to contracts of $2 million or over (for development and production) and 12 months' duration or more which do not use the Cost Performance Report (CPR). The Cost Performance Report (CPR) is required on selected contracts within major defense system programs. Both reports are contained in DoD Instruction 7000.10 dated August 6, 1974. Software is an automated analysis to generate management reports using the HP9845. The initial software package was obtained from the Army's MICOM Center. The benefits to C/SSR are:

1. Confidance in contractor's internal management system.
2. Objective (rather than subjective) contract status information.
3. Cost impact of know problems.
4. Identification of problems not previously recognized.
5. Capability to trace problems to source.
6. Quantitative measure of schedule deviation in dollars.
7. Measurement against a contract oriented baseline.

The criteria to Cost Performance Report (CPR):

1. Contract cost performance reporting requires effetive integration of work, schedule, budget and cost.
2. Provide standard formats for displaying contract cost performance.
3. Baseline is the key to effective performance measurement.
4. Baseline changes resulting from contract modifications and internal replanning must be controlled.
5. Baseline budgets must sum to the contract target cost if budgeted-cost-of-work-performed-to-date (BCWP) is to provide a meaningful indicator of contract cost performance.
6. When baseline budgets exceed the contract target cost, contract performance reporting is adversely affected.

THE CONSTRUCTIVE COST MODEL (COCOMO)

Barry W. Boehm, TRW

Thomas H. Bauer, TRW

May 1, 1983

Introduction

This paper presents a short overview of the COnstructive COst MOdel (COCOMO) for software cost estimation. A great deal more detail on COCOMO's capabilities and usage is provided in the book, Software Engineering Economics [Boehm, 1981].

The primary motivation for the COCOMO model has been to help people understand the cost consequences of the decisions they will make in commissioning, developing, and supporting a software product. Besides providing a software cost estimation capability, COCOMO also provides a great deal of material which explains exactly what costs the model is estimating, and why it comes up with the estimates it does. Further, it provides capabilities for sensitivity analysis and tradeoff analysis of many of the common software engineering decision issues.

COCOMO is actually a hierarchy of three increasingly detailed models which range from a simple macro estimation scaling model as a function of product size to a micro-estimation model with a three level work breakdown structure and a set of phase-sensitive effort multipliers for each cost driver attribute. To provide a reasonably concise example of a current state of the art cost estimation model, the intermediate level of COCOMO is described below.

Intermediate COCOMO estimates the cost of a proposed software product in the following way:

1. A nominal development effort is estimated as a function of the product's size in Delivered Source Instructions in thousands (KDSI) and the project's development mode.

PAGE 96 INTENTIONALLY BLANK

2. A set of effort multipliers are determined from the product's ratings on a set of 15 cost driver attributes.

3. The estimated development effort is obtained by multiplying the nominal effort estimate by all of the product's effort multipliers.

4. Additional factors can be used to determine dollar costs, development schedules, phase and activity distributions, computer costs, annual maintenance costs, and other cost elements from the development effort estimate.

Step 1. Nominal Effort Estimation

First, Table 1 is used to determine the project's development mode. Organic-mode projects typically come from stable, familiar, relatively unconstrained environments, and were found in the COCOMO data analysis of 63 projects to have a different scaling equation from the more ambitious, unfamiliar, tightly-constrained embedded mode. The resulting scaling equations for each mode are given in Table 2; these are used to determine the nominal development effort for the project in man-months as a function of the product's size in KDSI and the project's development mode.

For example, suppose we are estimating the cost to develop the microprocessor-based communications processing software for a highly ambitious new electronic funds transfer network with high reliability, performance, development schedule, and interface requirements. From Table 1, we determine that these characteristics best fit the profile of an Embedded-mode project.

We next estimate the size of the product as 10,000 Delivered Source Instructions, or 10 KDSI. From Table 2, we then determine that the nominal development effort for this Embedded-mode project is

$$2.8(10)^{1.20} = 44 \text{ man months (MM)}.$$

Step 2. Determine Effort Multipliers

Each of the 15 cost driver attributes in COCOMO has a rating scale and a set of effort multipliers which indicate by how much the nominal effort estimate must be multiplied to account for the project's having to work at its rating level for the attribute.

These cost driver attributes and their corresponding effort multipliers are shown in Table 3. The summary rating scales for each cost driver attribute are shown in Table 4, except for the Complexity rating scale which is shown in Table 5 (Expanded rating scales for the other attributes are provided in [Boehm, 1981]).

The results of applying these tables to our microprocessor communications software example are shown in Table 6. The effect of a software fault in the electronic fund transfer system could be a serious financial loss; therefore, the project's RELY rating from Table 4 is High. Then, from Table 3, the effort multiplier for achieving a High level of required reliability is 1.15, or 15% more effort than it would take to develop the software to a Nominal level of required reliability.

The effort multipliers for the other cost driver attributes are obtained similarly, except for the Complexity attribute, which is obtained via Table 5. Here, we first determine that communications processing is best classified under Device-Dependent Operations (column 3 in Table 5). From this column, we determine that communication line handling typically has a Complexity rating of Very High; from Table 3, then, we determine that its corresponding effort multiplier is 1.30.

Step 3. Estimate Development Effort

We then compute the estimated development effort for the microprocessor communications software as the nominal development effort (44 MM) times the product of the effort multipliers for the 15 cost driver attributes in Table 6 (1.35, in Table 6). The resulting estimated effort for the project is then

$$(44 \text{ MM})\ (1.35) = 59 \text{ MM}.$$

Step 4. Estimate Related Project Factors

COCOMO has additional cost estimating relationships for computing the resulting dollar cost of the project and for the breakdown of cost and effort by life-cycle phase (requirements, design, etc.) and by type of project activity (programming, test planning, management, etc.). Further relationships support the estimation of the project's schedule and its phase distribution. For example, the recommended development schedule can be obtained from the estimated development man-months via the Embedded-mode schedule equation in Table 2:

$$T_{DEV} = 2.5(59)^{0.32} = 9 \text{ months}$$

As mentioned above, COCOMO also supports the most common types of sensitivity analysis and tradeoff analysis involved in scoping a software project. For example, from Tables 3 and 4, we can see that providing the software developers with an interactive computer access (Low turnaround time) reduces the TURN effort multiplier from 1.00 to 0.87, and thus reduces the estimated project effort from 59 MM to

$$(59 \text{ MM})(0.87) = 51 \text{ MM}.$$

The COCOMO model has been validated with respect to a sample of 63 projects representing a wide variety of business, scientific, systems, real-time, and support software projects. For this sample, Intermediate COCOMO estimates come within 20% of the actuals about 68% of the time (see Figure 1). Since the residuals roughly follow a normal distribution, this is equivalent to a standard deviation of roughly 20% of the project actuals. This level of accuracy is representative of the current state of the art in software cost models. One can do somewhat better with the aid of a calibration coefficient (also a COCOMO option), or within a limited applications context, but it is difficult to improve significantly on this level of accuracy while the accuracy of software data collection remains in the "± 20%" range.

A Pascal version of COCOMO is available free of charge from the Wang Institute, under the name WICOMO [Demshki et al, 1982]. It can be obtained by sending a tape to:

Professor James Bouhana
Wang Institute of Graduate Studies
Tyngsboro, MA 01879

with a request for a copy of the WICOMO model and user's manual.

REFERENCES

[Boehm, 1981]. B. W. Boehm, Software Engineering Economics, Prentice-Hall, Englewood Cliffs, NJ, 1981.

[Demshki et al, 1982]. M. Demshki, D. Ligett, B. Linn, G. McCluskey, and R. Miller, "Wang Institute Cost Model (WICOMO) Tool," the Wang Institute, June 1982.

Table 1. COCOMO Software Development Modes

Feature	Mode: Organic	Semidetached	Embedded
Organizational understanding of product objectives	Thorough	Considerable	General
Experience in working with related software systems	Extensive	Considerable	Moderate
Need for software conformance with pre-established requirements	Basic	Considerable	Full
Need for software conformance with external interface specifications	Basic	Considerable	Full
Concurrent development of associated new hardware and operational procedures	Some	Moderate	Extensive
Need for innovative data processing architectures, algorithms	Minimal	Some	Considerable
Premium on early completion	Low	Medium	High
Product size range	< 50 KDSI	< 300 KDSI	All sizes
Examples	Batch data reduction Scientific models Business models Familiar OS computer Simple inventory, production control	Most transaction processing systems New OS, DBMS Ambitious inventory, production control Simple command-control	Large, complex transaction processing systems Ambitious, very large OS Avionics Ambitious command-control

Table 2. COCOMO Nominal Effort and Schedule Equations

DEVELOPMENT MODE	NOMINAL EFFORT	SCHEDULE
Organic	$(MM)_{NOM} = 3.2\,(KDSI)^{1.05}$	$TDEV = 2.5\,(MM_{DEV})^{0.38}$
Semidetached	$(MM)_{NOM} = 3.0\,(KDSI)^{1.12}$	$TDEV = 2.5\,(MM_{DEV})^{0.35}$
Embedded	$(MM)_{NOM} = 2.8\,(KDSI)^{1.20}$	$TDEV = 2.5\,(MM_{DEV})^{0.32}$

(KDSI = thousands of delivered source instructions)

Table 3. Intermediate COCOMO Software Development Effort Multipliers

Cost Drivers	Ratings: Very Low	Low	Nominal	High	Very High	Extra High
Product Attributes						
RELY Required software reliability	0.75	0.88	1.00	1.15	1.40	
DATA Data base size		0.94	1.00	1.08	1.16	
CPLX Product complexity	0.70	0.85	1.00	1.15	1.30	1.65
Computer Attributes						
TIME Execution time constraint			1.00	1.11	1.30	1.66
STOR Main storage constraint			1.00	1.06	1.21	1.56
VIRT Virtual machine volatility*		0.87	1.00	1.15	1.30	
TURN Computer turnaround time		0.87	1.00	1.07	1.15	
Personnel Attributes						
ACAP Analyst capability	1.46	1.19	1.00	0.86	0.71	
AEXP Applications experience	1.29	1.13	1.00	0.91	0.82	
PCAP Programmer capability	1.42	1.17	1.00	0.86	0.70	
VEXP Virtual machine experience*	1.21	1.10	1.00	0.90		
LEXP Programming Language experience	1.14	1.07	1.00	0.95		
Project Attributes						
MODP Use of modern programming practices	1.24	1.10	1.00	0.91	0.82	
TOOL Use of software tools	1.24	1.10	1.00	0.91	0.83	
SCED Required development schedule	1.23	1.08	1.00	1.04	1.10	

* For a given software product, the underlying virtual machine is the complex of hardware and software (OS, DBMS, etc.) it calls on to accomplish its tasks.

Table 4. COCOMO Software Cost Driver Ratings

Cost Driver	Ratings					
	Very Low	Low	Nominal	High	Very High	Extra High
Product attributes						
RELY	Effect: slight inconvenience	Low, easily recoverable losses	Moderate, recoverable losses	High financial loss	Risk to human life	
DATA		DB bytes < 10 Prog. DSI	10 ≤ D < 100 P	100 ≤ D < 1000 P	D ≥ 1000 P	
CPLX	See Table 5-4					
Computer attributes						
TIME			< 50% use of available execution time	70%	85%	95%
STOR			< 50% use of available storage	70%	85%	95%
VIRT		Major change every 12 months Minor 1 month	Major 6 months Minor 2 weeks	Major 2 months Minor 1 week	Major 2 weeks Minor 2 days	
TURN		Interactive	Average turnaround < 4 hours	4-12 hours	> 12 hours	
Personnel Attributes						
ACAP	15th percentile*	35th percentile	55th percentile	75th percentile	90th percentile	
AEXP	< 4 months experience	1 year	3 years	6 years	12 years	
PCAP	15th percentile*	35th percentile	55th percentile	75th percentile	90th percentile	
VEXP	< 1 month experience	4 months	1 year	3 years		
LEXP	< 1 month experience	4 months	1 year	3 years		
Project attributes						
MODP	No use	Beginning use	Some use	General use	Routine use	
TOOL	Basic microprocessor tools	Basic mini tools	Basic midi/maxi tools	Strong maxi programming test tools	Add requirements, design management documentation tools	
SCED	75% of nominal	85%	100%	130%	160%	

*Rating criteria analysts (programmers) ability, efficiency, ability to communicate and cooperate

Table 5. COCOMO Module Complexity Ratings versus Type of Module

Rating	Control Operations	Computational Operations	Device-dependent Operations	Data Management Operations
Very Low	Straightline code with a few non-nested SP* operators: DOs, CASEs, IF-THEN-ELSEs. Simple predicates	Evaluation of simple expressions: e.g., A = B+C * (D-E)	Simple read, write statements with simple formats	Simple arrays in main memory
Low	Straightforward nesting of SP operators. Mostly simple predicates	Evaluation of moderate-level expressions, e.g., D = SQRT (B**2-4. *A*C)	No cognizance needed of particular processor or I/O device characteristics. I/O done at GET/PUT level. No cognizance of overlap	Single file subvolume. No data structure changes, no edits, no intermediate files
Nominal	Mostly simple nesting. Some inter-module control. Decision tables	Use of standard math and statistical routines. Basic matrix/vector operations	I/O processing includes device selection, status checking and error processing	Multi-file input and single file output. Simple structural changes, simple edits
High	Highly nested SP operators with many compound predicates. Queue and stack control. Considerable inter-module control	Basic numerical analysis: multivariate interpolation, ordinary differential equations. Basic truncation, roundoff concerns	Operations at physical I/O level (physical storage address translations; seeks, reads, etc.). Optimized I/O overlap	Special purpose subroutines activated by data stream contents. Complex data restructuring at record level
Very High	Reentrant and recursive coding. Fixed-priority interrupt handling	Difficult but structured N.A.: near-singular matrix equations, partial differential equations	Routines for interrupt diagnosis, servicing, masking. Communication line handling	A generalized, parameter-driven file structuring routine. File building, command processing, search optimization
Extra High	Multiple resource scheduling with dynamically changing priorities. Microcode-level control	Difficult and unstructured N.A.: highly accurate analysis of noisy, stochastic data	Device timing-dependent coding, micro-programmed operations	Highly coupled, dynamic relational structures. Natural language data management

*SP = structured programming

Table 6. COCOMO Cost Driver Ratings: Microprocessor Communications Software

Cost Driver	Situation	Rating	Effort Multiplier
RELY	Serious financial consequences of software faults	High	1.15
DATA	20,000 bytes	Low	0.94
CPLX	Communications processing	Very High	1.30
TIME	Will use 70% of available time	High	1.11
STOR	45K of 64K store (70%)	High	1.06
VIRT	Based on commercial microprocessor hardware	Nominal	1.00
TURN	Two-hour average turnaround time	Nominal	1.00
ACAP	Good senior analysts	High	0.86
AXEP	Three years	Nominal	1.00
PCAP	Good senior programmers	High	0.86
VEXP	Six months	Low	1.10
LEXP	Twelve months	Nominal	1.00
MODP	Most techniques in use over one year	High	0.91
TOOL	At basic minicomputer tool level	Low	1.10
SCED	Nine months	Nominal	1.00
	Effort adjustment factor (product of effort multipliers)		1.35

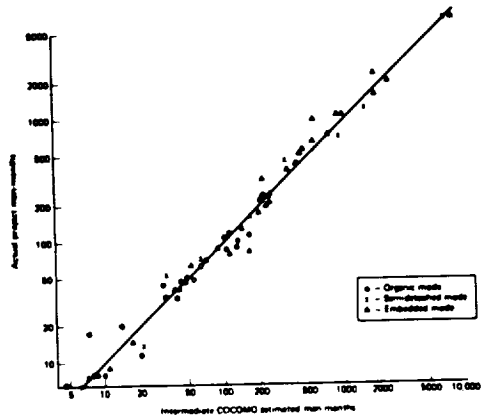


Figure 1. Intermediate COCOMO Estimates versus Project Actuals

A Method To Measure The "Effective Productivity" In Building Software Systems

Lawrence H. Putnam

Douglas T. Putnam

Lauren P. Thayer

If an average programmer can turn out 10 lines of source code per day, then a 10-man project group can turn out 45,000 lines in two years. Right? No, of course not. Twenty five years of overruns have taught the computer industry that this simple productivity-times-time formula does not work reliably.

Perhaps the daily productivity measure is at fault. After all, early in the two-year project most of the team were busy defining specifications, selecting algorithms, and designing the procedures that would ultimately be reduced to source code. Then late in the project they were occupied with inspection and testing -- finding bugs and fixing them. In fact, some studies have revealed that only 10% of project time is spent actually coding.

PAGE 106 INTENTIONALLY BLANK

What we really need is an overall measure of the efficiency with which a software project is carried out. We believe that there is such a measure and we explain what it is and how it can be used to plan software projects more satisfactorily.

Moreover, because finding and fixing bugs takes a lot of time and in this way affects project productivity, we show how the length of the project schedule affects error rates and, in consequence, how a good schedule can reduce errors and increase the "effective productivity" in this way as well.

Simple Productivity Inadequate

First we need to consider what simple productivity is and why, by itself, it is an inadequate measure for use in developing software systems. The traditional definition of software system productivity is the total output divided by the total effort required to produce the system.

In one sense the total output of a software development project embraces specifications, design, source code, object code, test results, and documentation -- the total functionality that has been created. However, this total would be hard to measure. For practical purposes it has been found that the number of lines of source code that were developed (not counting code picked up bodily from earlier work)

is proportional to this functionality. Thus, delivered (or developed) source lines of code -- DSLOC is the numerator in the productivity expression.

Effort is measured in manmonths or manyears of manpower applied. All the work involved in producing the total functionality must be counted.

The ratio of total output/total effort, i.e. source statements/ manmonths, is the system productivity. We shouldn't confuse it with programmer productivity, especially with coding productivity. As we have seen, coding is only a small portion of the work involved in building systems.

Furthermore, system productivity embraces far more than the efficiency of individual professionals. A software development project involves a group of people of varying talents, one or more levels of management of various degrees of competence, a programming language of high or very high order, a variety of software tools, and the use in varying degrees of modern programming practices such as structured design. All together, these variables make up what has come to be termed the software environment. These environments vary widely from one organization to another. It is a matter of common observation, if not heretofore precise measurement, that these environmental features lead to software groups of widely different "effective productivity."

However, the most serious problem with the source statements/ manmonths ratio as an objective measure of this complex environment is that productivity has been found empirically to vary with the length of the project schedule, or development time.

There are a number of different time-effort strategies which a management can use to build a given software system. At one extreme management assigns a very large number or people in order to get the system built in what it hopes will be the shortest possible time. At the other extreme it lets a few people work for a longer time until the job gets done. This situation usually happens when people or money are scarce. Of course, there are a range of strategies between these extremes.

In the first approach, the "unduly short" schedule, we have observed that system productivity is low. In the second approach, the "reasonably long" schedule (with the same people and environment), the productivity is high. This finding runs counter to our intuition -- that driving a project makes it productive.

Schedule time is not explicitly stated in the definition -- source statements/manmonths. The "manmonths" of effort may be expended in an "unduly short" schedule or in a "reasonably long" one. Therefore, it follows that we need an "effective productivity" expression that explicitly takes into account schedule time, or

development time. This expression exists. Lawrence Putnam discovered it seven years ago when he was working for the Army Computer Systems Command at Fort Belvoir. He called it the software equation.

The software equation is represented by:

$$S_s = C_k \cdot K^{1/3} \cdot t_d^{4/3}$$

where,

S_s	=	Executable source statements developed on project	Number
C_k	=	Technology constant (efficiency level)	Number
K	=	Life cycle effort	Manmonths
$\frac{E}{\beta}$	=	$\frac{\text{Development effort}}{\text{Adjusting factor}}$	$\frac{\text{Manmonths}}{\text{Number}}$
t_d	=	Development time (schedule)	Months

In words, the software equation states that a system of a given size in source statements (S_s) can be developed in a period of t_d months using E manmonths of effort at an efficiency level of C_k.

The software equation was derived empirically from a study of historical data covering some 200 systems from a half a dozen organizations. Since then it has been further verified by data from about 600 additional systems from many organizations.

This equation multiplies time and effort together. Thus, a change in the schedule causes a change in the effort required to get the product done to meet that schedule. This effect can be thought of as a software development trade-off law. It permits time to be traded for effort.

Most people familiar with software development agree that there is a time-effort trade-off relationship, even though it runs counter to the intuition based on hundreds of years of experience with simpler work processes. What is seldom appreciated without the software equation to measure it is the magnitude of this trade-off. The equation demonstrates that software development is a process extraordinarily sensitive to time, i.e. schedule time. Because of the exponents in the equation, very small changes in time produce very large changes in effort. A substantial amount of data on completed projects supports the view that this trade-off is close to the inverse fourth power.

Thus, the software equation becomes a very powerful tool for planning and monitoring software development. If an organization deliberately plans to take a little longer, it can reduce the effort

required to produce its system by a very substantial amount. This plan must be adopted before the project starts; management cannot just slip into it. Moreover, a few good people given enough time to build a product right make fewer errors that have to be found and fixed later, at still more schedule expense.

Technology Constant. The technology constant, C_k , is the constant that appeared when the software equation was derived from empirical data. From observation of a number of projects, the constant appeared to represent the overall efficiency of the organization developing the software. This observation seems to be logical -- manpower times time times an efficiency factor normally equals the work produced.

More formally, the technology constant is a macroscopic parameter that measures the aggregate impact of all influences that affect how long it takes and how much effort is required to build a system. In essence, this parameter measures how efficient an organization is in building a certain class of software.

A very efficient producer of management-information-systems software, for example, would have a high value of C_k. This type of environment would typically include on-line development, good software development tools, strong professional skills, stable requirements, etc. A less efficient producer would have a lower value for C_k. For a product of a given size higher values of C_k produce shorter schedules that require less effort.

This outcome is the real objective of system building. The technology constant measures this objective very well. It is a measure of "effective productivity".

The value of C_k is not based on guesswork or rough estimates. For a system that has been recently completed, three terms of the software equation are known: the number of lines of source code developed, the development time, and the development effort. Thus, the fourth term, C_k, can be calculated with the same precision as the known terms.

After collecting and analyzing the technology constants for more than 800 systems, it was evident that there is a set of technology constants and that each level of this set is related to a particular class of development work. The levels appear to be driven primarily by the complexity of each class of work. The same values of the technology constant have been observed in similar organizations building the same type of software during the same time frame, i.e. periods during which the software environment is comparable. In calculating the technology constant for a given organization, one should be careful to include only homogenous classes of work, i.e., all business systems or all real-time embedded systems, etc.

RADC Trend Lines. The data base of software project data collected by the Rome Air Development Center provides the opportunity to examine the relationship between simple productivity and developed source lines of code (Figure 1), project duration and DSLOC (Figure 2) total manmonths and DSLOC (Figure 3), and average number of people and DSLOC (Figure 4). This data base, obtained from a wide variety of applications organizations, is the largest heterogeneous data base ever collected. Richard Nelson of RADC analyzed the data base with respect to these management parameters. The range of the data base is suggested by the extent of the number of source statements per system -- from about 100 to over one million, a range which spans the scope of most systems ever built.

On the figures, because both scales are logarithmic, the relationships plot as straight lines, indicating that the basic functions are power functions. The middle line in each case represents the average best fit of all data points. The upper and lower lines are the plus one and minus one standard deviation (or sigma) lines.

The correlations of total manmonths to source statements (R = .85), average number of people to source statements (R = .80), and project duration to source statements (R = .70) are quite good. Even so, there is still a large variation in the data at any particular system size. For example, at a system size of 10,000

lines the project duration varies from one month up to 1.5 years. It is evident that something more than straightforward correlation is needed if we are to use these relationships for project planning.

Moreover, the correlation of simple productivity to source statements is only .03 -- very poor, indicating essentially no relationship.

Three of the metrics analyzed by Nelson, source statements, total manmonths of effort, and project duration (development time) -- are the same ones that appear in the software equation. His productivity parameter, however, is different from the technology constant that appears in the software equation. Use of the set of technology constants -- one for each class of work -- provides the capability to sort the data on the vertical axis, making the resulting curves more meaningful.

Case Studies

In this section we apply the methodology based on the software equation to the data resulting from the development of five actual software systems. The five systems were built by three well known and respected organizations. The systems fall into three application categories with, as we shall see, three different technology constants:

(1) Real-time embedded software -- a cruise missile;

(2) Systems software -- the EXEC operating system;

(3) Manufacturing support systems --

  (a) Parts No.

  (b) RFM

  (c) Materials

For each project the size (source statements), development time, and development effort are known. From this data the technology constant was calculated.

Since the values of the technology constant have been observed to range from 610 to 121,393 in a series of 22 discrete steps, it is convenient to translate these large numbers into a series of small numbers, running from one to 22.

Another parameter, called the gradient level, was also calculated. The formula for the gradient level is:

$$\text{Gradient} = K/t_d^3$$

i.e., the gradient level is the ratio of life cycle effort to the development time cubed. As in the case of the technology constant, the calculated values are transformed to a linear sequence ranging from one to six.

The gradient level measures how fast management actually applies people to a project. For example, if a system's functional content is well known, management could build the project's staffing to a high level quickly. Such a project would have a relatively high gradient level, perhaps in the range of three, four, five, or six. On the other hand, gradient levels one or two represent a very gradual buildup of manpower. This level seems to match fuzzy, ill-defined,

and largely sequential problems where time compression by means of parallel effort is less feasible.

Table 1 summarizes the known data for these five systems and displays the results of the calculations. The table is in the form of computer printout because the calculations were performed by a computerized implementation of the software equation called SLIM (for Software LIfe Cycle Methodology). In SLIM the Calibrate function calculates values of the technology factor and the gradient level from data of previously developed software projects.

In Figure 5 the technology factors of the five systems are plotted. By means of Figure 6 these technology factors may be contrasted with the technology factors of all the systems presently contained in the SLIM data base.

In certain situations where resources -- people or money -- are constrained, it is possible to force a project to be built in a very sequential way even though high degrees of parallelism would have been possible. The after the fact gradient levels measure how projects were actually built, not how they might have been built. For example, a system that is being rebuilt, which normally falls in gradient level three, may have been built like a new and complicated system, which usually falls in gradient level one, because only seven people were available to work on it.

When a situation like this occurs, it is an indication that the software trade-off law is being exercised. Indeed, four of the five systems in these examples are cases where trade-offs did take place. They are the EXEC operating system and all three of the manufacturing support systems.

Cruise Missile. In this case the software was embedded in a computer contained in a cruise missile. Its purpose was to process a stream of radar information reflecting the terrain below the missile. Because of limited memory, the need to maximize processing speed, and the extreme complexity of the algorithms used, it was necessary to write the 5800 instructions in assembly language.

The resulting technology constant was 754 (technology factor of one), the lowest value that we have observed historically. This low value was determined by the fact that the project was very difficult. The gradient level was also one, a level we would expect for such a complex system.

We have observed that projects involving micro code or ROM-based firmware also tend to be at the low end of the technology factor scale. These two, together with real-time embedded systems, appear to be equally difficult to solve.

EXEC System. This system can be classified as an operating system -- a system executive. It was designed to run on a standalone

computer and handle all data and memory management functions. It was written primarily in Fortran (80-85 percent) with the remaining 15 to 20 percent being written in assembly language in order to optimize certain functions.

The system was done under contract by a large computer corporation. The project manager was adamant that the team used on the main software build be kept small -- not more than 10-12 people. He insisted that he was not going to be talked into trying to "steam-roller" this project with large numbers of people. The main reason for his insistence was his intuitive feel that small groups could communicate much better than large groups. He felt that a small team would be better able to implement the structured, modular approach which he planned to use extensively. Among the tools he used were front-end structured design, PDL (a design language), and code walk-throughs. The ultimate result was a very modular and efficient code.

In the beginning the system was designed and coded around an Interdata 7/32 central processing unit. This machine had some significant memory constraints which somewhat compromised the design. Approximately one-third of the way through the development the software builders switched to a Perkin-Elmer 3242 as the development machine. This CPU did not have the memory problems of the earlier one, but it did permit the overall design to remain consistent with the original concept. (It is not clear in retrospect whether the switch of machines caused a break in the continuity of the project

-- it usually does.) Both development computers supported on-line interactive development.

EXEC consisted of 61,000 lines of executable code. The development time from detailed logic design to full operational capability was 33 months; the effort expended during this period came to 248 manmonths.

Trade-Off Verification. When these data were run through the computerized implementation of the software equation, the result was a technology factor of eight (equivalent to a technology constant of 4181) and a gradient level of one. The gradient level may be thought of as the manpower acceleration rate -- the rate at which manpower is applied to the project. As described by the project manager, however, we would have classified EXEC as a standalone system of gradient level two. Gradient level one, determined by calculation, indicates how the work was actually done.

This discrepancy between the expected and the actual is often a clue to the fact that management has made a trade-off. A system of gradient level one takes longer than a system of gradient level two. In consequence, if the actual time is significantly longer than the minimum time for that type of system, the difference was very likely caused by some management constraint, such as restricted funding or limited manpower. In this case the constraint was the 10-12 people beyond which the project manager was not willing to go.

Taking another approach, a Monte Carlo simulation was run. This simulation is one of the features of the SLIM package. The simulation used the actual size of EXEC (61,000 source statements), and the technology factor (eight) obtained by calibration, but used a gradient level of two, corresponding to the project manager's description of the work. The simulation produced these figures:

Minimum Development Time	25.8 Months
Development Effort	658 Manmonths

These results were encouraging, since the actual time was 7.2 months longer than the simulation output and the actual effort was only 248 manmonths.

We next reasoned that if the manmonths were set to the actual, SLIM would calculate a development-time solution very close to the actual data point, 33 months. SLIM has a Design-To-Cost function which performs this calculation. Sure enough, the outcome was a perfect fit, 33 months. It appears that the fourth power software trade-off law was valid for this case. A 29% stretchout in the development-time schedule had produced a 62% reduction in cost.

Suppose that we had assumed a gradient level one system. Then the minimum time result would have been 28.5 months and 438 manmonths. In this instance an 18% stretchout had produced a 48% reduction in cost.

The actual computer printout of the foregoing calculations is reproduced as Appendix A.

Manufacturing Support Systems. These three systems, RFM, Parts No., and Materials, were built by the same company to support manufacturing operations. Over a four year period (1978 - 1982) this company moved out of a non-IBM environment into a very modern large-scale IBM mainframe one. The software development people using this new environment characterized it as much better for software development than the old one. The utilities that they formerly had to create were now available as part of the normal IBM development tool kit. Additional capabilities included a data base management system and a full capability screen editor. The group also used the ADE tools associated with IDMS. The developing organization is very tool-conscious. It upgrades and assimilates new equipment and software tools on a regular basis. The functional designers made extensive use of the PRIDE structured development methodology.

Futhermore, this organization is aware of the software equation and makes strong use of the trade-off law. It chooses to build software with small groups of people; in other words, its projects are manpower constrained.

Its results are further proof of the efficacy of this methodology. The technology factors for the three systems are 15, 16, and 17 -- in the top 10% of what has been observed in the United States, Europe, and Japan. Moreover, it is interesting to note that this organization moved from a baseline technology factor of 12 in 1978

to a factor of 16 in 1982 in its new environment. This increase appears substantial, but translated to terms of the technology constant, 10,946 to 28,657, it is a phenomenal increase in "effective productivity." Used in this way the technology constant is a good measure of the real increase in efficiency of an organization. Futhermore, this organization has found that it gets a higher quality product using this software development philosophy. This point is further developed in the sidebar.

Materials System. It is worth taking a more detailed look at one of these systems -- Materials. First, it is a very large system, containing 700,000 lines of Cobol. Built in the environment described above, it took 38 months and 384 manmonths. The technology factor was 17, corresponding to a technology constant of 35,422. This is a very high technology factor -- in the top 5% of what has been observed. These figures, and others to be developed, are summarized in Table 2 for convenient reference. The system designers described Materials as a rebuild of an existing system.

The actual data from Materials was run through the computerized implementation of the software equation in order to determine the minimum time that would have been required to complete the project, assuming a gradient level of three. This level was used because it corresponds to the fastest manpower buildup rate possible for a rebuilt system. This gradient level represents a considerably faster

buildup than the one which actually occurred. Calibration of the actual data showed that management worked the problem like a gradient level one.

This difference in gradient levels is a good indication, as we mentioned before, that a trade-off has, in fact, taken place. Running the Monte Carlo simulation in the SLIM model produced the following results:

Minimum Development Time	27 Months
Development Effort	1434 Manmonths

However, to build the system in this minimum time would have required a peak staffing of 82 people, and this organization typically doesn't use this many people on projects. In actuality, Materials took 11 months longer to build than this minimum time, but it used much less effort.

To test the actual results against the fourth power software trade-off law, the Design-To-Cost function in SLIM was used. Putting in the actual recorded manmonths, 384, the Design-To-Cost function predicted a development time of 37.45 months, only two weeks off from what actually happened. This prediction is very close, considering that most people don't record the data more accurately than whole months.

Thus, the eleven month stretchout of the schedule from the minimum-time estimate reduced peak staffing from 82 to 16 people, a reduction of 80%. The increase in development time, from 27 months to 38 months, amounted to 39%. This increase in development time was traded for a reduction of 1050 manmonths in effort, a decrease of 73%. At a burdened labor rate of $50,000/manyear, this reduction represents a cost saving of $4.5 million.

The actual computer printout of the foregoing calculations is reproduced as Appendix B.

Sidebar

Software Error Prediction

Some work in software reliability --or software error prediction-- has also been incorporated in the SLIM computer model. At present this model is backed by a lesser quantity of historical data than the software equation, but its output is useful for comparative purposes, even if it is not precisely correct. The reliability model provides the capability to compare the expected error rate for the minimum-time solution against that for the actual-time development in the case of the Materials system or the Design-To-Cost solution for pending projects. By this means management can establish that there are quality trade-offs, as well as effort trade-offs, between the different development-time approaches.

The model shows that when the Materials system first became operational (on the 37 month schedule), it had a Mean Time to Failure (MTTF) of about one week. In other words, the average time between major system failures that required prompt corrective action was about one week. During these 37 months some 474 errors (range 393-556)would have been identified and fixed out of a total of 500 (26 more) found through the first 20 months of operation. This level is very close to the reliability level experienced in the field.

On the other hand, for the minimum-time schedule (27 months) the reliability model predicts a MTTF of only .05 months (about one work day) and 1776 errors (range 1483-2068)up to full operational capability

out of 1870 errors (94 more) through a 15 month "Work em Out" maintenance period.

It is evident that there is a significant difference in the two approaches to development. This difference seems logical. Because software development is manpower-intensive, people in it spend a great deal of time trying to communicate with one another. The human communication process is ambiguous. Therefore, erroneous human communication is always happening. In consequence, software developments that have large amounts of human communication tend to generate more "noise." This means that large numbers of errors occur in such systems and the mean time between failures is short.

By reducing the human communication on a project, management can also reduce this noise. A few good people given enough time will not commit nearly as many design or coding errors. The computerized printouts show how the quality changes from one solution to the other.

Case Studies Vs. Data Base

Earlier we showed trend lines derived from RADC data base for productivity, total manmonths, average number of people, and project duration -- all plotted against source statements. Now we have obtained comparable data for three classes of software: a real-time embedded system, an operating system, and three manufacturing support systems. The next step is to plot the case data on the trend lines established by the RADC analysis.

Productivity. In Figure 7 the productivity in source statements per manmonth for the five cases have been superimposed on the log-log plots of trend line plus and minus one standard deviation. The productivity of the cruise-missile project falls more than one standard deviation below the average productivity of the over 400 systems in the RADC data base. The software in this case was extremely complex and had a very low technology factor (one on a scale of 1 to 22). The development strategy was minimum time, which corresponds to maximum effort. Not surprisingly, the productivity of the cruise-missile software plots exactly where one would expect, i.e., very low.

The productivity of the EXEC system plots almost exactly on the average trend line. With a technology factor of eight, lower than the average technology factor of all the systems in the RADC data base, we would expect to see slightly lower than average productivity. In

fact, we would have if the minimum schedule (maximum effort) strategy had been employed. Under this strategy productivity would have been 61,000/665, or 93 source statements per manmonth of work. This level of productivity would plot well below the average trend line. In this case, however, we know that the project was constrained to a maximum staffing of 11 people. This manpower constraint caused a 7.2 month increase in the schedule and a substantial reduction in effort. As a result, the productivity for the system as it was actually built is 61,000/248, or 246 source statements per manmonth of work. This level of productivity is much higher than the one associated with the minimum-time strategy. Thus, the more efficient schedule explains why the EXEC system's productivity plots on the average trend line even though a technology factor of eight initially cause us to think it would be lower.

The three manufacturing systems have very high productivity rates, well beyond the plus one standard deviation line. All three of these systems had very high technology factors and all were resource-constrained by the organization's small-team development approach. Consequently, all three of these systems fall in the trade-off region. We see that this organization's capital investment of tools, coupled with its development philosophy, has paid off. In fact, its productivity points plot two to three standard deviations higher than the average of the more than 400 systems.

Project Duration. In Figure 8 project duration is plotted against system size. The duration of the cruise-missile project is more than one standard deviation longer than the average. The very complex nature of this work intuitively leads us to expect this result.

The EXEC system plots close to one standard deviation longer than the average duration. Why? Because the schedule was deliberately stretched out 7.2 months from the minimum-time schedule. The trade-off becomes evident.

The three manufacturing systems all plot very close to the average duration. In these cases the influence of the high technology factors tends to shorten the development schedules even though the schedules were stretchd out to take advantage of the trade-off of time against money. The Materials schedule, for example, was stretched out 11 months, but even with this dramatic time stretchout, it still plots slightly shorter than the average duration. The other two systems were handled in a similar way.

Total Manmonths. Figure 9 represents total manmonths versus system size. Since manmonths are proportional to cost, this graph also represents the expense of building a system. The three manufacturing systems required two to three standard deviations less effort than the average. The great reduction in manmonths means that these systems were very inexpensive, compared to what other organizations have paid historically for similar sized systems. Again, this figure shows clearly that this organization's capital investments and development practices have paid off handsomely.

Total manmonths for the EXEC system are sightly less than the average. This result is attributable to the constrained resource development aproach (trade-off) that was used.

The cruise missile, being more complex, had lower productivity, took longer to build, and required more effort. In its case the effort was more than one standard deviation higher than the average.

Average Manpower. This graph is essentially the same as the previous one, since total manmonths and average manpower are related. As we would expect, the cruise-missile software required a greater than average number of people.

The EXEC system took less than the average number of people -- not surprising because the system was resource-constrained at 11 people.

The manufacturing support system used significantly fewer people. All three systems fall two to three standard deviations below the average.

Path To Effective Productivity

The graphs support the conclusion that the software equation is very close to expressing the way software systems behave with respect to changes in the development schedule and effort. The large number of systems in the RADC data base provides an objective means to measure the performance achieved on a variety of projects. The technology constant derived from the software equation and its index expression as the technology factor supply the capability to measure the relative level of overall efficiency in homogeneous classes of work.

When these metrics are combined, they give a good account of how you really performed. This measure of your efficiency becomes more meaningful as you acquire new tools and develop new methodologies. Measure the old projects. Measure the new projects. Did you really get the payoff that you anticipated? What is the payoff? Are you getting an equal or better quality product in less time, using less effort and fewer people? This software life-cycle methodology gives you numbers that you, as a manager, are used to thinking in.

Using this approach you will be able to capture the long-term dynamics of the software development process. As your environment changes over time, you will be able to assess more effectively the actual capabilities of your organization. You can feed this information back into your estimating process to make your estimates consistent with your organization's current capabilities. But most important, you can enhance the effective productivity of your organization with respect to both resources and quality.

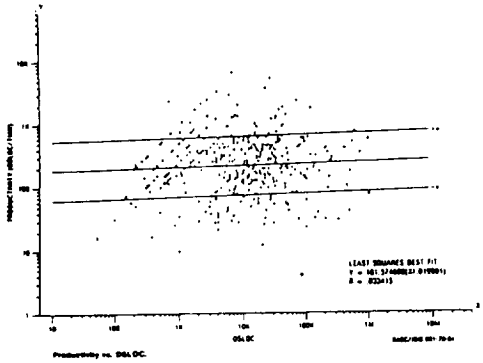


Figure 1. There is very little correlation between simple productivity and developed source lines of code.

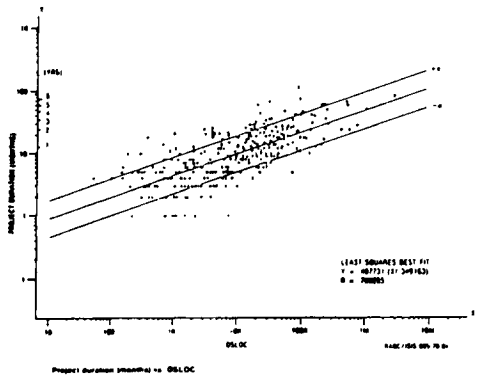


Figure 2. The correlation between project duration and developed source lines of code is much higher.

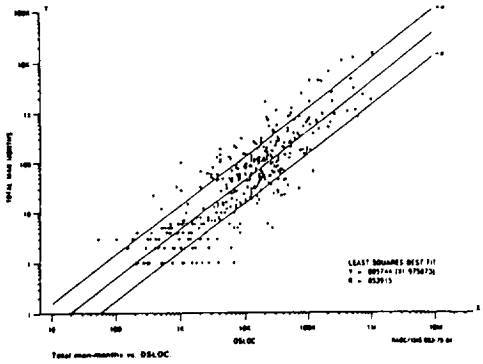


Figure 3. The correlation between manmonths of effort and developed source lines of code is the highest of all the parameters analyzed by Richard Nelson of RADC.

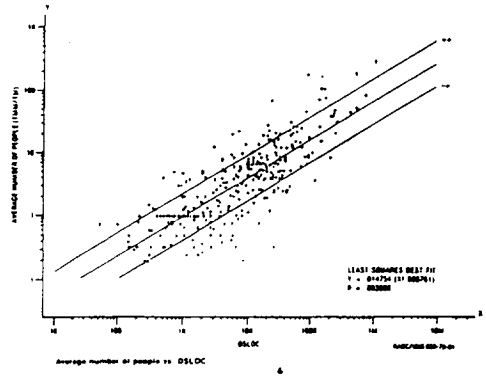


Figure 4. The correlation between average number of people and developed source lines of code is also high. Of course, manmonths and average number of people are closely related.

```
*****************************************************************************
                              CALIBRATE
*****************************************************************************
```

SYSTEM NAME: CRUISE MISSILE	
SIZE (SOURCE STATEMENTS):	5000
DEVELOPMENT TIME:	24 MONTHS
DEVELOPMENT EFFORT:	107 MANMONTHS
GRADIENT LEVEL:	1
TECHNOLOGY FACTOR:	1
C_k = 754	
THIS TECHNOLOGY FACTOR SEEMS UNREASONABLE. PLEASE RECHECK YOUR DATA.	

SYSTEM NAME: BFR	
SIZE (SOURCE STATEMENTS):	100000
DEVELOPMENT TIME:	21 MONTHS
DEVELOPMENT EFFORT:	48 MANMONTHS
GRADIENT LEVEL:	1
TECHNOLOGY FACTOR:	13
C_k = 21,892	

SYSTEM NAME: EXEC SYSTEM	
SIZE (SOURCE STATEMENTS):	61000
DEVELOPMENT TIME:	13 MONTHS
DEVELOPMENT EFFORT:	240 MANMONTHS
GRADIENT LEVEL:	1
TECHNOLOGY FACTOR:	8
C_k = 4181	

SYSTEM NAME: MATERIALS	
SIZE (SOURCE STATEMENTS):	700000
DEVELOPMENT TIME:	30 MONTHS
DEVELOPMENT EFFORT:	304 MANMONTHS
GRADIENT LEVEL:	1
TECHNOLOGY FACTOR:	17
C_k = 35,422	

SYSTEM NAME: PARTS NO.	
SIZE (SOURCE STATEMENTS):	100000
DEVELOPMENT TIME:	21 MONTHS
DEVELOPMENT EFFORT:	25 MANMONTHS
GRADIENT LEVEL:	1
TECHNOLOGY FACTOR:	16
C_k = 28,657	

Table 1. The Calibrate function of the SLIM computer model calculates the technology factor and gradient level when data about completed projects is input to it. The technology factors have also been translated back to Ck values.

CASE STUDY
SOFTWARE DEV DATABASE

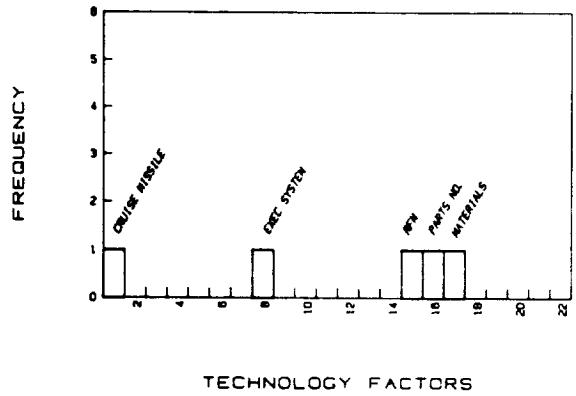
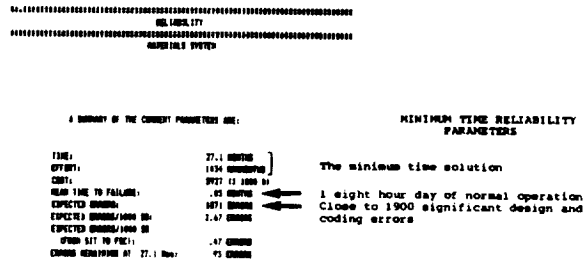


Figure 5. This diagram exhibits the distribution of the technology factors for the five cases studied. If the technology constants were diagramed, the range would be much broader.



SLIM SOFTWARE DATABASE
as of JUNE 82
(includes RADC DB)

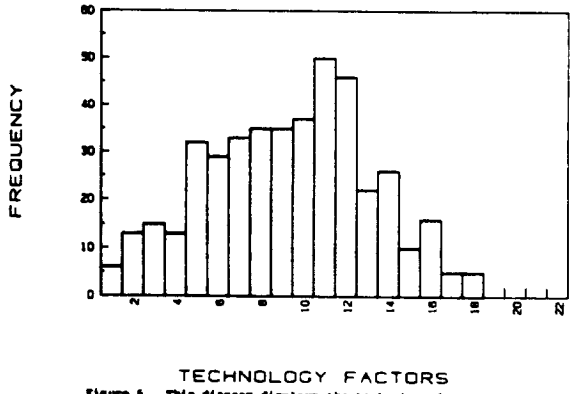


Figure 6. This diagram displays the technology factors of all the systems included in the RADC data base. Most systems have factors along the center of the scale. Only a few systems have very high or very low technology factors.

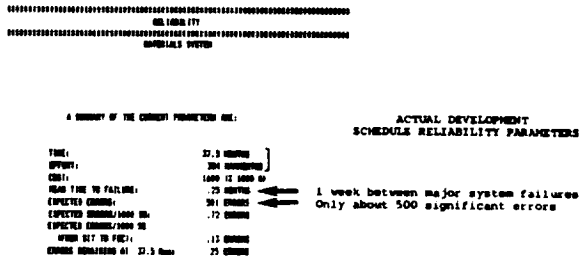


Table 2. COMPARATIVE DATA FOR MATERIALS SYSTEM

Solution	Source Statements	Effort (Manmonths)	DEV Time (Months)	Peak People	Technology Factor	Gradient
Actual	700,000	384	38	16	17	1
Minimum Time	700,000	1434	27	82		3
Design To Cost	700,000	384	37.45	16		

THE TABLE BELOW SHOWS THE EXPECTED ERROR RATE, MEAN TIME TO FAIL-URE (MONTHS), AND EXPECTED CUMULATIVE ERRORS FOR DEVELOPMENT THROUGH THE .999 RELIABILITY LEVEL. THESE VALUES ARE BASED ON A DEVELOPMENT TIME OF 27.1 MONTHS AND A TOTAL DEVELOPMENT EFFORT OF 1134.2 MANMONTHS

MINIMUM TIME ERROR FORCAST

EXPECTED ERRORS

MONTH	MEAN ERROR RATE	ERROR RATE RANGE	EXPECTED CUM ERRORS FIXED	RANGE CUM ERRORS FIXED	MTTF (MONTHS)
JAN 79	15.1	10.7 - 19.7	0	0 - 0	-
FEB 79	36.1	21.4 - 38.0	36	25 - 35	-
MAR 79	44.2	31.7 - 56.7	68	54 - 79	-
APR 79	57.3	41.4 - 73.2	110	99 - 130	-
MAY 79	69.0	50.2 - 87.8	183	152 - 212	-
JUN 79	79.2	58.0 - 100.5	256	214 - 298	-
JUL 79	87.6	64.5 - 110.6	339	284 - 393	-
AUG 79	94.2	69.9 - 118.4	430	360 - 501	-
SEP 79	98.8	73.0 - 121.8	527	440 - 614	-
OCT 79	101.6	76.4 - 126.6	628	524 - 731	-
NOV 79	103.6	77.6 - 127.6	730	610 - 850	-
DEC 79	101.9	77.4 - 126.3	832	695 - 969	-
JAN 80	99.6	76.1 - 123.1	933	779 - 1087	-
FEB 80	96.1	73.0 - 118.4	1031	861 - 1201	-
MAR 80	91.5	70.6 - 112.3	1125	940 - 1310	-
APR 80	86.8	66.6 - 105.3	1214	1014 - 1412	-
MAY 80	79.8	62.2 - 97.5	1296	1083 - 1510	-
JUN 80	73.2	57.3 - 89.2	1373	1147 - 1599	-
JUL 80	66.5	52.2 - 80.7	1443	1205 - 1680	.02
AUG 80	59.7	47.0 - 72.3	1506	1258 - 1754	.02
SEP 80	53.0	41.9 - 64.1	1562	1305 - 1819	.02
OCT 80	46.4	36.9 - 56.2	1612	1347 - 1877	.02
NOV 80	40.3	32.2 - 48.8	1656	1383 - 1928	.02
DEC 80	34.9	27.8 - 41.9	1693	1414 - 1972	.02
JAN 81	29.7	23.8 - 35.7	1725	1441 - 2010	.02
FEB 81	25.1	20.1 - 30.1	1753	1464 - 2041	.04
MAR 81	21.6	16.9 - 25.1	1776	1483 - 2068	.05
APR 81	17.4	14.0 - 20.8	1795	1499 - 2091	.06
MAY 81	14.3	11.5 - 17.0	1811	1513 - 2109	.07
JUN 81	11.5	9.4 - 13.8	1824	1523 - 2124	.09
JUL 81	9.3	7.6 - 11.1	1834	1532 - 2136	.11
AUG 81	7.5	6.0 - 8.9	1842	1539 - 2146	.13
SEP 81	5.9	4.8 - 7.0	1849	1545 - 2154	.17
OCT 81	4.6	3.8 - 5.5	1854	1549 - 2160	.22
NOV 81	3.6	2.9 - 4.3	1858	1552 - 2164	.28
DEC 81	2.8	2.2 - 3.3	1862	1555 - 2168	.36
JAN 82	2.1	1.7 - 2.5	1864	1557 - 2171	.47
FEB 82	1.6	1.3 - 1.9	1866	1559 - 2173	.63
MAR 82	1.2	1.0 - 1.4	1867	1560 - 2175	.84
APR 82	.9	.7 - 1.1	1868	1561 - 2176	1.13
MAY 82	.7	.5 - .8	1869	1561 - 2177	1.53
JUN 82	.5	.4 - .6	1870	1562 - 2177	2.10

No MTTF because we don't have a system yet

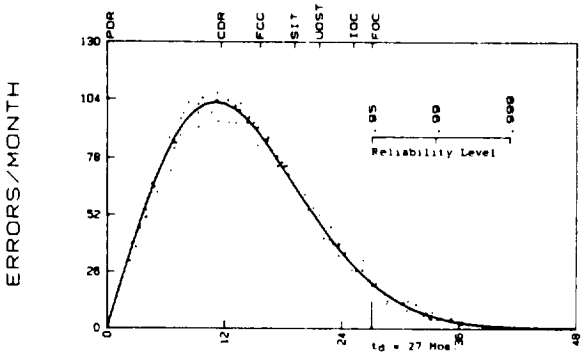
Systems Integration Test starts about here.

MTTF = .05 Mos. when the system will become operational

You will have to continue to test and work an additional 7.5 months to get a MTTF of 1 week. The development cost now will be close to 10 million dollars.

EXPECTED ERROR RATE

Minimum Time Case



MATERIALS SYSTEM
GS PROGRAM
TF 17
LEVEL II

DEV TIME (months)

EXPECTED ERRORS

MONTH	MEAN ERROR RATE	ERROR RATE RANGE	EXPECTED CUM ERRORS FIXED	RANGE CUM ERRORS FIXED	MTTF (MONTHS)
JAN 79	2.1	1.4 - 2.9	1	1 - 1	-
FEB 79	4.2	2.8 - 5.7	4	3 - 5	-
MAR 79	6.3	4.2 - 8.4	10	8 - 11	-
APR 79	8.5	5.6 - 11.0	17	14 - 20	-
MAY 79	10.2	6.9 - 13.4	26	22 - 31	-
JUN 79	11.9	8.2 - 15.6	37	31 - 44	-
JUL 79	13.5	9.3 - 17.7	50	41 - 58	-
AUG 79	14.9	10.4 - 19.5	64	53 - 75	-
SEP 79	16.2	11.3 - 21.1	80	66 - 93	-
OCT 79	17.3	12.2 - 22.4	96	80 - 113	-
NOV 79	18.2	12.9 - 23.5	114	95 - 134	-
DEC 79	18.9	13.4 - 24.4	133	110 - 156	-
JAN 80	19.4	13.9 - 24.9	152	126 - 178	-
FEB 80	19.7	14.2 - 25.3	172	142 - 201	-
MAR 80	19.9	14.1 - 25.4	191	158 - 224	-
APR 80	19.8	14.0 - 25.3	211	175 - 247	-
MAY 80	19.6	14.1 - 25.0	231	191 - 271	-
JUN 80	19.3	14.1 - 24.5	250	207 - 293	-
JUL 80	18.8	13.8 - 23.8	269	223 - 316	-
AUG 80	18.2	13.4 - 23.0	288	239 - 337	-
SEP 80	17.5	13.0 - 22.1	306	253 - 358	-
OCT 80	16.7	12.5 - 21.0	323	268 - 379	-
NOV 80	15.9	11.9 - 19.9	339	281 - 398	-
DEC 80	15.0	11.2 - 18.8	355	294 - 416	-
JAN 81	14.1	10.6 - 17.6	369	306 - 433	-
FEB 81	13.1	9.9 - 16.3	383	317 - 449	.08
MAR 81	12.2	9.2 - 15.1	396	328 - 464	.08
APR 81	11.2	8.5 - 13.9	407	337 - 477	.09
MAY 81	10.3	7.8 - 12.7	418	346 - 490	.10
JUN 81	9.4	7.2 - 11.6	428	354 - 501	.11
JUL 81	8.5	6.5 - 10.5	437	362 - 512	.12
AUG 81	7.7	5.9 - 9.5	445	368 - 521	.13
SEP 81	6.9	5.3 - 8.5	452	374 - 530	.15
OCT 81	6.1	4.7 - 7.4	459	380 - 537	.18
NOV 81	5.3	4.2 - 6.7	464	385 - 544	.20
DEC 81	4.8	3.7 - 5.9	470	389 - 550	.21
JAN 82	4.2	3.3 - 5.2	474	393 - 556	.24
FEB 82	3.7	2.9 - 4.5	478	396 - 560	.27
MAR 82	3.3	2.5 - 3.9	481	399 - 564	.31
APR 82	2.8	2.2 - 3.4	485	401 - 568	.36
MAY 82	2.4	1.9 - 2.9	487	404 - 571	.41
JUN 82	2.1	1.6 - 2.5	489	405 - 574	.48
JUL 82	1.8	1.4 - 2.1	491	407 - 576	.57
AUG 82	1.5	1.2 - 1.8	493	408 - 578	.67
SEP 82	1.3	1.0 - 1.5	494	409 - 579	.79
OCT 82	1.1	.8 - 1.3	496	410 - 581	.94
NOV 82	.9	.7 - 1.1	497	411 - 582	1.12
DEC 82	.7	.6 - .9	497	412 - 583	1.34
JAN 83	.6	.5 - .7	498	412 - 584	1.62
FEB 83	.5	.4 - .6	499	413 - 584	1.96
MAR 83	.4	.3 - .5	499	413 - 585	2.38
APR 83	.3	.3 - .4	499	414 - 585	2.91
MAY 83	.3	.2 - .3	500	414 - 586	3.58
JUN 83	.2	.2 - .3	500	414 - 586	4.42
JUL 83	.2	.1 - .2	500	414 - 586	5.47
AUG 83	.1	.1 - .2	500	414 - 586	6.82
SEP 83	.1	.1 - .1	500	415 - 586	8.53
OCT 83	.1	.1 - .1	501	415 - 587	10.72

THE ACTUAL SCHEDULE ERROR FORCAST

No system yet

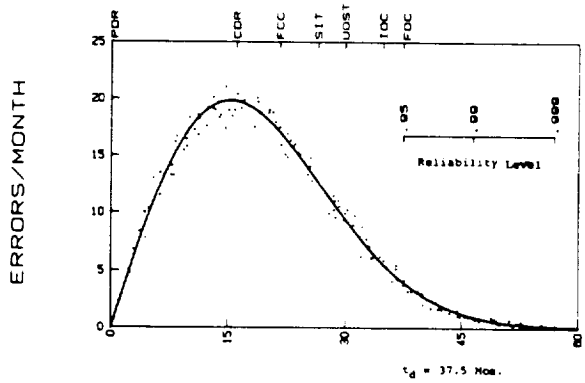
Systems Integration Test

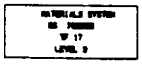
MTTF = 1 week at full operational capability

How things get better as you continue to test and work.

EXPECTED ERROR RATE

For Trade-off Case





PRODUCTIVITY
VS. Ss

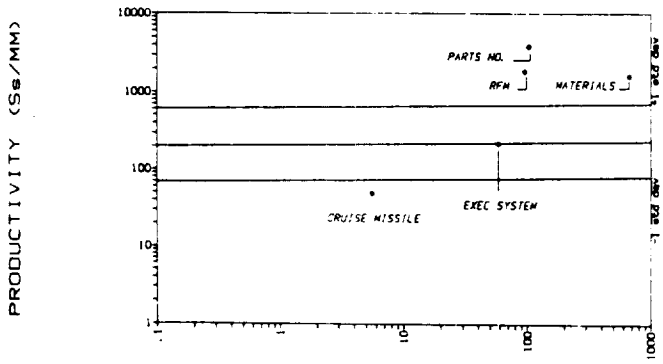


Figure 7. The productivity values of the five cases are superimposed upon the productivity trend lines of the 400 RADC cases. Because the vertical scale is logarithmic, the cruise-missile value is much lower and the three manufacturing support systems are much higher than scale distance by itself implies.

PROJECT DURATION (Mos)
VS. Ss

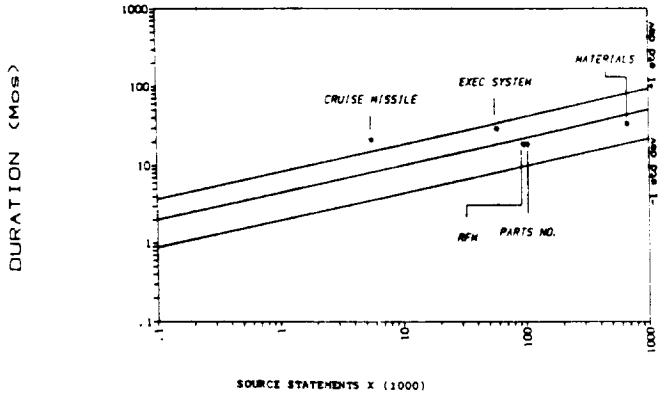


Figure 8. The surprise here is that the project duration of the three manufacturing support systems is only somewhat below average when one might expect that the stretched-out schedules would result in above average development times. The reason for the relatively short durations is the high technology factors of these three systems.

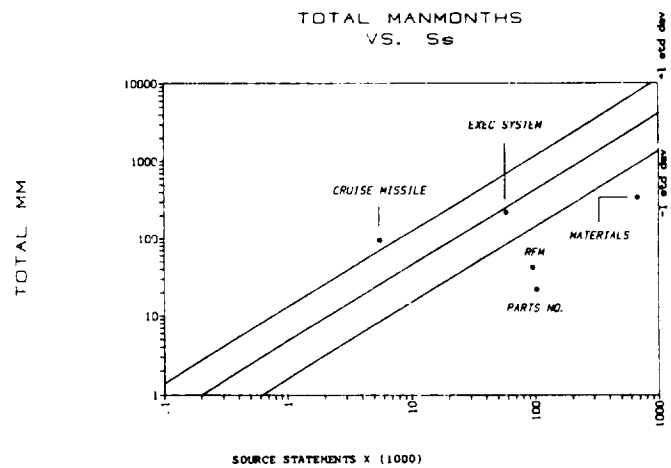


Figure 9. The stretched-out schedules resulted in much below average expenditure of effort in the cases of the three manufacturing support systems. The cruise missile, developed on a fast schedule with much difficulty, used an above average number of manmonths.

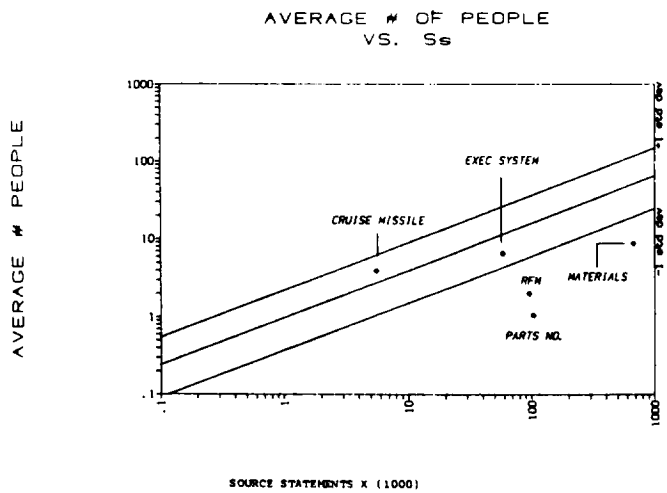


Figure 10. The chart of the average number of people shows a relationship similar to the manmonths chart.

APPENDIX A

EXEC SYSTEM
VERIFICATION OF THE TRADE-OFF LAW

```
SUMMARY OF INPUT PARAMETERS
SYSTEM:  EXEC SYSTEM                                   DATE:  10 AUG 83
                                                       TIME:     13:30

PROJECT START:  8670

COST ELEMENTS
  COST/MY:                75000.          INFLATION RATE:        .100
  MONETARY UNIT:   ($)
  STD DEV (COST/MY):      7500.

ENVIRONMENT
  ONLINE DEVELOPMENT:       1.00          NOL USAGE:             .05
  DEVELOPMENT TIME:         1.00          PRODUCTION TIME:       0.00
  DOMS:                     0.00          REPORT WRITER:         0.00
  LANGUAGE:              FORTRAN

SYSTEM
  TYPE:  OPERATING SYSTEM                  REAL-TIME CODE:        .15
  LEVEL:                    2              UTILIZATION:           .80

MODERN PROGRAMMING PRACTICES
  STRUCTURED PROGRAMMING:  > 75%           DESIGN/CODE INSPECTION:  > 75%
  TOP-DOWN DESIGN:         > 75%           CHIEF PROG TEAM USAGE:   < 25%

EXPERIENCE
  OVERALL:              EXTENSIVE          SYSTEM TYPE:  EXTENSIVE
  LANGUAGE:               AVERAGE                MANPOWER:  MINIMAL

TECHNOLOGY
  FACTOR:                   8

SIZE
  LOWEST:                 51000                  HIGHEST:     71000
```

Gradient level 2 for a standalone system as described by the project manager.

C_k = 418)

Technology factor 8 was calculated from the actual project data.

```
SIMULATION
EXEC SYSTEM
10 AUG 83
13:36

                                        MEAN      STD DEV
SYSTEM SIZE (STATEMENTS)                61000       3313

MINIMUM DEVELOPMENT TIME (MONTHS)        25.8         .8

DEVELOPMENT EFFORT (MANMONTHS)           650.1       70.6

DEVELOPMENT COST ($)
  (UNINFLATED)                           4146        304
  (INFLATED)                             4599        635

PEAK MANPOWER (PEOPLE)                     38
```

This is the fastest time that the EXEC system could have been built in.

To build the system in the minimum time it would have required 38 people at peak staffing.

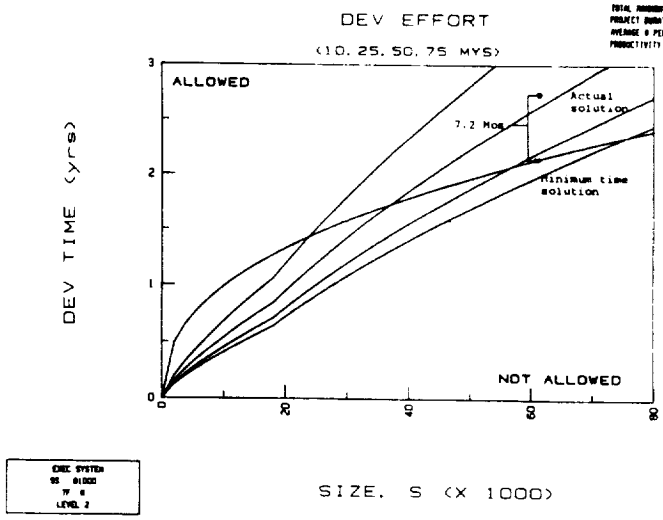
```
SENSITIVITY PROFILE FOR MINIMUM TIME SOLUTION

                 SOURCE STMTS    MONTHS    MANMONTHS    UNINFLATED COST* (X 1000)
                 ------------    ------    ---------    -------------------------
-3 STD DEV        51000.         23.9       510.            3187.
-1 STD DEV        57667.         25.2       614.            3825.
MOST LIKELY       61000.         25.8       650.            4146.
+1 STD DEV        64333.         26.4       718.            4490.
+3 STD DEV        71000.         27.5       824.            5152.

A CONSISTENCY CHECK WITH DATA FROM OTHER SYSTEMS OF THE SAME SIZE SHOWS:

TOTAL MANMONTHS              650        IN NORMAL RANGE
PROJECT DURATION (MONTHS)    25.8       IN NORMAL RANGE
AVERAGE # PEOPLE              25        IN NORMAL RANGE
PRODUCTIVITY (LINES/MM)       93        IN NORMAL RANGE
```

Productivity at the minimum time is 93 Ss per manmonth.



DESIGN TO COST

EXEC SYSTEM

THE BEST ESTIMATES OF THE MINIMUM TIME AND CORRESPONDING EFFORT AND COST TO DEVELOP YOUR SYSTEM ARE:

MINIMUM TIME:	25.05 MONTHS
EFFORT:	450 MANMONTHS
COST:	6146 (X 1000 $)

ENTER DESIRED DEVELOPMENT EFFORT IN MANMONTHS

?

240 ← The actual manmonths of effort

NEW DEVELOPMENT TIME

EXEC SYSTEM

	MEAN	STD DEV
NEW DEV TIME (MONTHS)	33.00	.90
NEW DEV EFFORT (MANMONTHS)	240	37
NEW DEV COST (X 1000)	1550	210

A near perfect fit to the inverse 4th trade-off relationship.

YOUR FILE IS NOW UPDATED WITH THESE NEW PARAMETERS. RUN RUNLOGLOG AND CAMPLAN TO SEE HOW THESE CHANGES CAN BE REALIZED.

A CONSISTENCY CHECK WITH DATA FROM OTHER SYSTEMS OF THE SAME SIZE SHOWS:

TOTAL MANMONTHS	240	IN NORMAL RANGE
PROJECT DURATION (MONTHS)	33.0	IN NORMAL RANGE
AVERAGE # PEOPLE	8	IN NORMAL RANGE
PRODUCTIVITY (LINES/MM)	246	IN NORMAL RANGE

← Productivity has increased from 93/MM to 246/MM..

RUNLOGLOG

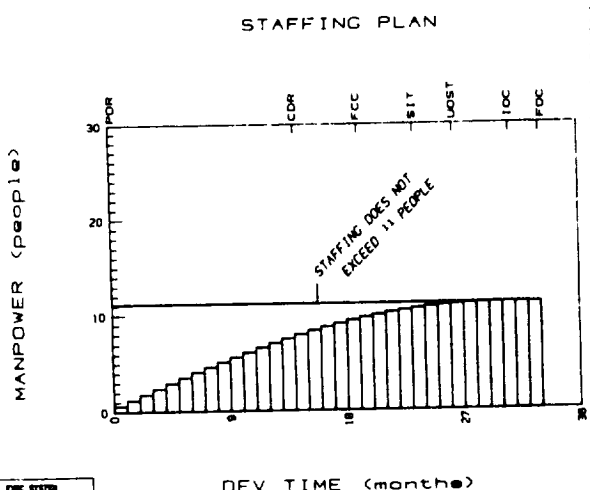
EXEC SYSTEM

THE TABLE BELOW SHOWS THE MEAN PROJECTED EFFORT AND STANDARD DEVIATION EXPENDED PER DEVELOPMENT. THESE VALUES ARE BASED ON A DEVELOPMENT TIME OF 33.0 MONTHS AND A TOTAL DEVELOPMENT EFFORT OF 240.0 MANMONTHS.

STAFFING PLAN

MONTH	PEOPLE/MONTH	STD DEV	CUMULATIVE MANMONTHS	CUM STD DEV
JUN 78		.1		
JUL 78	1	.2	1	
AUG 78	1	.3	3	
SEP 78	2	.4	5	1
OCT 78	3	.5	7	1
NOV 78	3	.6	11	1
DEC 78	4	.7	14	2
JAN 79	4	.8	19	2
FEB 79	5	.9	24	3
MAR 79	5	.9	29	3
APR 79	6	1.0	35	4
MAY 79	6	1.1	41	4
JUN 79	7	1.1	48	5
JUL 79	7	1.2	55	6
AUG 79	8	1.2	63	7
SEP 79	8	1.3	72	8
OCT 79	9	1.3	80	9
NOV 79	9	1.4	89	10
DEC 79	9	1.4	98	11
JAN 80	10	1.4	108	12
FEB 80	10	1.5	118	13
MAR 80	10	1.5	128	14
APR 80	10	1.5	138	15
MAY 80	11	1.5	149	16
JUN 80	11	1.5	159	17
JUL 80	11	1.5	170	18
AUG 80	11	1.5	181	19
SEP 80	11	1.5	192	21
OCT 80	11	1.5	203	22
NOV 80	11	1.5	214	23
DEC 80	11	1.5	226	24
JAN 81	11	1.5	237	25
FEB 81	11	1.5	248	27
AUG 81	6	.7	254	28

Peak staffing does not exceed 11 people (this is the maximum number of people the project manager was willing to use).



EXEC SYSTEM
88 04:00
TF 0
LEVEL 2

APPENDIX B

MATERIALS SYSTEM
VERIFICATION OF THE TRADE-OFF LAW

SUMMARY OF INPUT PARAMETERS

SYSTEM: MATERIALS SYSTEM — DATE: 10 MAR 83 — TIME: 12:26

PROJECT START: 0179

COST ELEMENTS
- COST/MY: 50000. — INFLATION RATE: .100
- MONETARY COST: (0)
- STD DEV (COST/MY): 5000.

ENVIRONMENT
- ONLINE DEVELOPMENT: 1.00 — HOL USAGE: 1.00
- DEVELOPMENT TIME: 1.00 — PRODUCTION TIME: 0.00
- DOMS: .30 — REPORT WRITER: 0.00
- LANGUAGE: COBOL

SYSTEM
- TYPE: BUSINESS APPLICATION — REAL-TIME CODE: 0.00
- LEVEL: 3 ← — UTILIZATION: .30

A rebuild of an existing system.

MODERN PROGRAMMING PRACTICES
- STRUCTURED PROGRAMMING: > 75% — DESIGN/CODE INSPECTION: > 75%
- TOP-DOWN DESIGN: > 75% — CHIEF PROG TEAM USAGE: < 25%

EXPERIENCE
- OVERALL: EXTENSIVE — SYSTEM TYPE: UNKNOWN
- LANGUAGE: EXTENSIVE — HARDWARE: EXTENSIVE

TECHNOLOGY
- FACTOR: 17 ← — Ck = 35422

Exceptionally high. A very good performer.

SIZE
- LOWEST: 600000 — HIGHEST: 800000

SIMULATION
MATERIALS SYSTEM
10 MAR 83
12:27

	MEAN	STD DEV
SYSTEM SIZE (STATEMENTS)	700000	33333
MINIMUM DEVELOPMENT TIME (MONTHS)	27.1	.7
DEVELOPMENT EFFORT (MANMONTHS)	1434.2	143.1
DEVELOPMENT COST ($)		
(UNINFLATED)	5927	812
(INFLATED)	6664	990
PEAK MANPOWER (PEOPLE)	82 ←	

This is the absolute minimum time that this system could have been built.

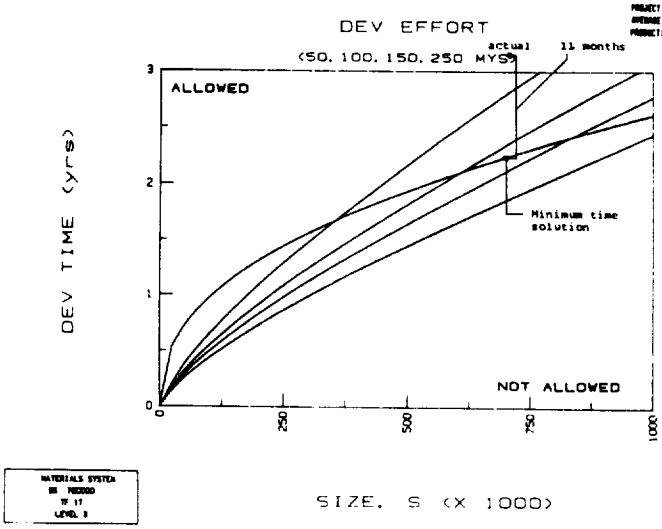
To build the system in the minimum time would require 82 people. This is not the organizational style -- a few good people is their approach.

SENSITIVITY PROFILE FOR MINIMUM TIME SOLUTION

	SOURCE STATS	MONTHS	MANMONTHS	UNINFLATED COST ($ 1000)
-1 STD DEV	600000.	25.2	1170.	4982.
-1 STD DEV	666667.	26.4	1349.	5619.
MOST LIKELY	700000.	27.1	1434.	5927.
+1 STD DEV	733333.	27.5	1524.	6352.
+1 STD DEV	800000.	28.3	1705.	7103.

A CONSISTENCY CHECK WITH DATA FROM OTHER SYSTEMS OF THE SAME SIZE SHOWS:

TOTAL MANMONTHS	1434	IN NORMAL RANGE
PROJECT DURATION (MONTHS)	27.1	IN NORMAL RANGE
AVERAGE # PEOPLE	53	IN NORMAL RANGE
PRODUCTIVITY (LINES/MM)	488	IN NORMAL RANGE



[illegible]

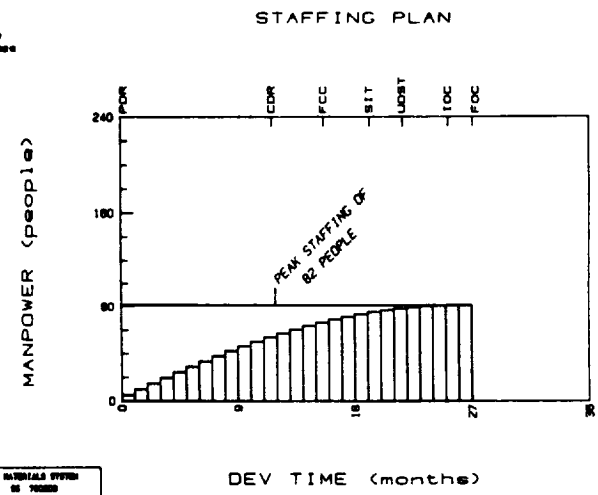
MATERIALS SYSTEM

THE TABLE BELOW SHOWS THE MEAN PROJECTED EFFORT AND STANDARD DEVIATION REQUIRED FOR DEVELOPMENT. THESE VALUES ARE BASED ON A DEVELOPMENT TIME OF 27.1 MONTHS AND A TOTAL DEVELOPMENT EFFORT OF 1434.7 MANMONTHS.

STAFFING PLAN

MONTH	PEOPLE/MONTH	STD DEV	CUMULATIVE MANMONTHS	CUM STD DEV
JAN 79	7	.4	7	
FEB 79	7	1.3	14	1
MAR 79	17	2.1	32	3
APR 79	17	2.9	49	4
MAY 79	22	3.7	62	6
JUN 79	27	4.4	89	9
JUL 79	31	5.0	120	12
AUG 79	36	5.4	155	16
SEP 79	40	5.7	196	20
OCT 79	44	6.7	240	24
NOV 79	48	7.3	288	29
DEC 79	52	7.6	340	34
JAN 80	56	8.0	396	40
FEB 80	59	8.4	455	45
MAR 80	62	8.7	517	52
APR 80	65	9.0	582	58
MAY 80	69	9.3	651	65
JUN 80	71	9.4	722	72
JUL 80	73	9.3	795	79
AUG 80	75	9.7	869	87
SEP 80	76	9.0	945	94
OCT 80	79	9.6	1023	102
NOV 80	79	9.0	1102	110
DEC 80	80	9.0	1182	118
JAN 81	81	9.0	1263	126
FEB 81	81	9.0	1345	134
MAR 81	82	9.7	1426	142
APR 81	41	6.0	1467	150

This is more people than they usually use on a project.



[illegible]

DESIGN TO COST

MATERIALS SYSTEM

THE COST ESTIMATES OF THE REQUIRED TIME AND CORRESPONDING EFFORT AND COST TO DEVELOP YOUR SYSTEM ARE:

REQUIRED TIME: 26.99 MONTHS
EFFORT: 1430 MANMONTHS
COST: 9967 IN 1000 $

GIVEN REDUCED DEVELOPMENT EFFORT IN MANMONTHS
?
384 ←

They actually built the system in 38 months for 384 manmonths.

Let's try these numbers

MATERIALS SYSTEM
DS 700000
SF 17
LEVEL 3

NEW DEVELOPMENT TIME
MATERIALS SYSTEM

	MEAN	STD DEV
NEW DEV TIME (MONTHS)	37.45	1.34
NEW DEV EFFORT (MANMONTHS)	384	49
NEW DEV COST IN 1000$	1600	204

This is within two weeks of what actually happened.

YOUR FILE IS NOW UPDATED WITH THESE NEW PARAMETERS. NEW HORIZONTAL AND CHARTPLOT TO SEE HOW THESE CHANGES CAN BE REALIZED.

A CONSISTENCY CHECK WITH DATA FROM OTHER SYSTEMS OF THE SAME SIZE SHOWS:

TOTAL MANMONTHS	384	< NORMAL RANGE
PROJECT DURATION (MONTHS)	37.5	OK NORMAL RANGE
AVERAGE # PEOPLE	10	< NORMAL RANGE
PRODUCTIVITY (LOC/MONTH)	1023	> NORMAL RANGE

NOTICE THE FLAGS

Less than normal effort

Less than normal # people

Greater than normal productivity

THESE ARE GOOD FLAGS !!

[redacted]

THE TABLE BELOW SHOWS THE MEAN PROJECTED EFFORT AND STANDARD DEVIATION ASSIGNED FOR DEVELOPMENT. THESE VALUES ARE BASED ON A DEVELOPMENT TIME OF [redacted] MONTHS AND A TOTAL DEVELOPMENT EFFORT OF [redacted] MANMONTHS.

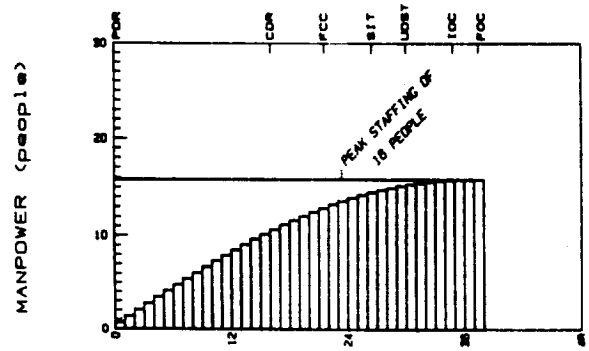
STAFFING PLAN

MONTH	PEOPLE/MONTH	STD DEV	CUMULATIVE MANMONTHS	STD DEV STD DEV
JAN 79		.1		
FEB 79	1	.2	.1	
MAR 79	2	.3	3	
APR 79	3	.5	6	1
MAY 79	3	.6	9	1
JUN 79	4	.7	13	1
JUL 79	4	.8	17	2
AUG 79	5	1.0	22	2
SEP 79	6	1.1	28	3
OCT 79	6	1.2	34	4
NOV 79	7	1.3	41	4
DEC 79	8	1.3	49	5
JAN 80	8	1.4	57	6
FEB 80	9	1.5	66	7
MAR 80	9	1.6	75	8
APR 80	10	1.7	85	9
MAY 80	10	1.7	95	10
JUN 80	11	1.8	106	11
JUL 80	11	1.8	118	12
AUG 80	12	1.9	130	14
SEP 80	12	1.9	142	15
OCT 80	13	2.0	155	16
NOV 80	13	2.0	168	17
DEC 80	13	2.0	181	19
JAN 81	14	2.1	195	20
FEB 81	14	2.1	209	22
MAR 81	14	2.1	223	23
APR 81	15	2.1	238	25
MAY 81	15	2.2	253	26
JUN 81	15	2.2	268	28
JUL 81	15	2.2	283	29
AUG 81	15	2.2	298	31
SEP 81	16	2.2	314	32
OCT 81	16	2.2	330	34
NOV 81	16	2.2	345	35
DEC 81	16	2.2	361	36
JAN 82	16	2.1	377	37
FEB 82	0	1.1	392	41

The actual staffing

This plan peaks at 16 people. (This is the typical organizational style for this company)

STAFFING PLAN



MATERIALS SYSTEM
SW TRADED
W 17
LEVEL 3

ABSTRACT

PRICE S

PETER B. KORDA
RCA/PRICE SYSTEMS

This paper describes PRICE S - a parametric cost estimating model for software development and maintenance.

PRICE S is an interactive model, accessed on commercial time-sharing computers via telephone lines from standard office terminals. Immediately following the entry of project descriptors, the model derives and displays projected costs for each of three development phases: Design, Implementation, and Test Integration.

In addition to costs, the model computes typical schedules for the size, type and difficulty of the project described. If desired, manpower and scheduling contraints which apply to the software development effort can be specified. The consequences of these constraints are examined internally, and costs are adjusted to account for apparent accelerations, stretch-outs and phase transition inefficiencies.

RCA

PRICE S

Programmed Review of Information for Costing and Evaluation

Software Model

P44BP00

TABLE II INSTRUCTION MIX

APPLICATION TYPE	WEIGHT	IDENTIFYING CHARACTERISTICS
OPERATING SYSTEMS	10.95	Task management. Memory management. Heavy hardware interface. Many interactions. High reliability and error timing requirements.
INTERACTIVE OPERATIONS	10.95	Real time man/machine interfaces. Human engineering considerations and error protection very important.
REAL TIME COMMAND AND CONTROL	8.46	Machine to machine communications under tight timing constraints. Queuing not practicable. Heavy hardware interface. Strict protocol requirements.
ON-LINE COMMUNICATIONS	5.16	Machine to machine communications with queuing allowed. Timing restrictions not as restrictive as with real time command and control.
DATA STORAGE AND RETRIEVAL	4.10	Operation of data storage devices. Data base management. Secondary storage handling. Data blocking and deblocking. Hashing techniques. Hardware oriented.
STRING MANIPULATION	2.31	Routine applications with no overriding constraints. Not oriented toward mathematics. Typified by language compilers, sorting, formatting, buffer manipulation, etc.
MATHEMATICAL OPERATIONS	.88	Routine mathematical applications with no overriding constraints.

RCA
PRICE Systems

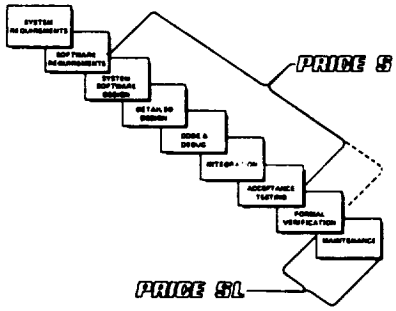


TABLE III TYPICAL PLATFORM VALUES

Operating Environment	Platform
Production Center Internally Developed S/W	6 — 8
Production Center Contracted S/W	1.0
MIL-Spec Ground	1.2
Military Mobile (Van or Shipboard)	1.4
Commercial Avionics	1.7
MIL-Spec Avionics	1.8
Unmanned Space	2.0
Manned Space	2.5

RCA
PRICE Systems

PRICE S - Principal Inputs

- How big?
  INSTRUCTIONS
- Who does it?
  RESOURCE
- Where used?
  PLATFORM
- What character?
  APPLICATION/MIX
- How difficult?
  COMPLEXITY
- What hardware constraints?
  UTILIZATION

TABLE IV Typical Complexity Adjustments

Personnel	
Outstanding crew, among best in industry	- 2
Extensive experience, some top talent	- 1
Normal crew, experience	0
Mixed experience, some new hires	+ 1
Relatively inexperienced, many new hires	+ 2
Product Familiarity	
Old hat, redo of previous work	- 2
Familiar type of project	- 1
Normal new project, normal line of business	0
New line of business	+ 2
Complicating Factors	
New hardware	+ 1
New language	+ 1
More than one location/organization	+ 2
Hardware developed in parallel	+ 3
Many changing requirements	+ 3

--- PRICE SOFTWARE MODEL ---

DATE 01/16/78 TIME 14:32:55

SAMPLE CASE — MOBILE RADAR

INPUT DATA

FILENAME: SAMPLE — DATED: 07/23/77

DESCRIPTORS					
INSTRUCTIONS	34000	APPLICATION	0.0	RESOURCE	3.500
FUNCTIONS	0	STRUCTURE	0.0	LEVEL	2.000

APPLICATION CATEGORIES	MIX	NEW DEVELOPMENT DESIGN	CODE	SYSTEM CONFIGURATION TYPES	QUANTITY
DATA S/S	0.0	1.00	1.00	0	0
REAL TIME COMM	0.00	1.00	1.00	1	1
REALTIME CSC	0.00	1.00	1.00	2	2
INTERACTIVE	0.23	1.00	1.00	3	2
MATHEMATICAL	0.00	0.50	0.75	***	***
STRING MANIP	0.25	1.00	1.00	***	***
OPR SYSTEMS	0.07	1.00	1.00	***	***

SCHEDULE					
COMPLEXITY	1.250				
DESIGN START	OCT 77	IMPL START	0	TEI START	0
DESIGN END	0	IMPL END	0	TEI END	0

SUPPLEMENTAL INFORMATION					
YEAR	1977	ESCALATION	0.0	TECH IMP	1.00
MULTIPLIER	1.000	PLATFORM	1.4	UTILIZATION	0.80

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	340.	14.	293.	977.
PROGRAMMING	40.	63.	104.	210.
CONFIGURATION CONTROL	71.	19.	158.	248.
DOCUMENTATION	38.	6.	63.	101.
PROGRAM MANAGEMENT	29.	6.	33.	68.
TOTAL	500.	110.	611.	1222.

ADDITIONAL DATA

DESCRIPTORS					
INSTRUCTIONS	34000	APPLICATION	0.299	RESOURCE	3.500
FUNCTIONS	400	STRUCTURE	0.941	LEVEL	2.000

SCHEDULE					
COMPLEXITY	1.250				
DESIGN START	OCT 77	IMPL START	FEB 78	TEI START	JUN 78
DESIGN END	AUG 78	IMPL END	DEC 78	TEI END	AUG 79

SCHEDULE GRAPH

OCT 77 — AUG 79

```
************* DESIGN ************
            ************ IMPLEMENT ************
                        ************** TEST & INTEGRATE ****************
```

--- PRICE SOFTWARE MODEL ---

DATE 01/16/78 TIME 14:32:55

SAMPLE CASE — MOBILE RADAR

MONTH	% COMPLETED DESIGN	IMPL	T & I	% EXPENDED THIS MONTH	TOTAL	% EXPENDED THIS MONTH	TOTAL
OCT 77	2.4	0.0	0.0	18.2	18.2	0.0	0.0
NOV 77	15.1	0.0	0.0	65.3	73.4	5.3	5.2
DEC 77	34.1	0.0	0.0	95.3	170.7	7.8	13.0
JAN 78	54.0	0.0	0.0	99.3	270.1	8.1	22.1
FEB 78	71.4	0.2	0.0	87.2	357.3	7.1	29.3
MAR 78	84.4	3.4	0.0	69.7	427.0	5.7	35.0
APR 78	93.1	11.8	0.0	52.0	479.0	4.3	39.3
MAY 78	96.0	25.7	0.0	38.5	516.5	3.2	42.5
JUN 78	99.7	47.3	0.0	28.2	544.6	2.3	44.7
JUL 78	100.0	62.0	0.3	23.4	570.2	1.9	46.7
AUG 78	100.0	78.7	1.3	24.5	594.7	2.0	48.7
SEP 78	100.0	91.2	3.4	27.0	621.6	2.2	50.9
OCT 78	100.0	98.0	7.2	30.7	652.3	2.5	53.4
NOV 78	100.0	100.0	13.0	37.1	689.4	3.0	56.4
DEC 78	100.0	100.0	20.0	48.1	737.6	3.9	60.4
JAN 79	100.0	100.0	30.0	61.2	798.8	5.0	65.4
FEB 79	100.0	100.0	42.0	72.9	871.6	6.0	71.4
MAR 79	100.0	100.0	54.1	81.3	952.9	6.7	78.0
APR 79	100.0	100.0	69.9	84.4	1037.4	6.9	84.9
MAY 79	100.0	100.0	82.9	79.8	1117.2	6.5	91.5
JUN 79	100.0	100.0	93.5	64.8	1182.0	5.3	96.8
JUL 79	100.0	100.0	99.5	36.4	1218.5	3.0	99.7
AUG 79	100.0	100.0	100.0	3.1	1221.5	0.3	100.0

	DESIGN	IMPL	T & I
ALPHA	0.02	0.0	0.0
BETA	0.18	1.00	0.15
PEAK/AV	1.93	1.00	1.93

FOR PROFILE GRAPHS RESPOND OK : ?

--- PRICE SOFTWARE MODEL ---

DATE 01/17/78 TIME 11:02:34

SAMPLE CASE — MOBILE RADAR

SENSITIVITY DATA

COMPLEXITY

RESOURCE	1.150	1.250	1.350
3.400	COST 1085. MONTHS 19.8	COST 1170. MONTHS 21.8	COST 1248. MONTHS 23.7
3.500	COST 1132. MONTHS 19.9	COST 1222. MONTHS 21.9	COST 1315. MONTHS 23.8
3.600	COST 1180. MONTHS 20.0	COST 1274. MONTHS 22.0	COST 1371. MONTHS 23.9

--- PRICE SOFTWARE MODEL ---

DATE 01/17/78 TIME 10:50:42

SAMPLE CASE — MOBILE RADAR

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

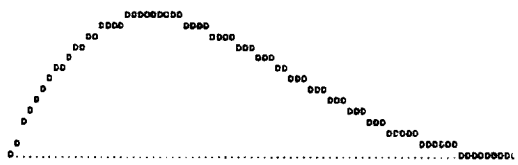
COMPLEXITY = 1.250	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE (OVERLAP)	7.0 (4.0)	8.0 (5.0)	9.0	19.0
TYPICAL SCHEDULE (OVERLAP)	9.0 (5.7)	10.0 (5.9)	13.9	21.9

DEVELOPMENT COSTS

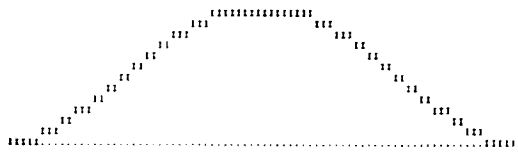
COMPLEXITY = 1.250	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	572.	122.	781.	1475.
TYPICAL SCHEDULE	500.	110.	611.	1222.
ESTIMATED PENALTY	72.	12.	170.	253.

RESOURCE ALLOCATION PROFILES

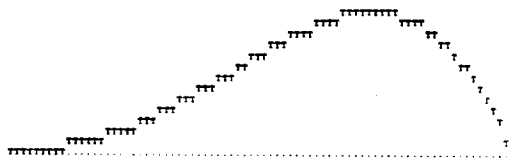
DESIGN — ALPHA= 0.02 BETA= 0.18 PEAK/AVERAGE RESOURCE: 1.93



IMPLEMENTATION — ALPHA= 0.0 BETA= 1.00 PEAK/AVERAGE RESOURCE: 1.00



TEST AND INTEGRATION — ALPHA= 0.0 BETA= 0.14 PEAK/AVERAGE RESOURCE: 1.93



ADJUSTING SOFTWARE ESTIMATING MODELS TO OBTAIN COMPARATIVE RESULTS: A PROGRESS REPORT

By: Bernard Rudwick
Chief Management Scientist
PRC Government Information Systems

1. Introduction

As software parametric estimating models increase in use, model users are faced with two challenges. First, an increasing number of automated software estimating models are becoming available through lease or purchase. These include RCA PRICE S, SLIM, COCOMO, Jensen JS-1, and SOFTCOST. In addition, some users are developing their own models since they feel they more accurately represent their work process. So the user needs to determine which model(s) best serve his/her organization. A second challenge is that more and more users now have access to more than one model. As one example, a software developer may wish to obtain higher confidence in the estimates and hence checks the results of one model against another before presenting the model results to the client, as illustrated in the top part of figure 1. A second situation involves the government (or other) client being presented with two cost proposals for the same system. In this case the client needs to compare the results from two or more vendors as shown in the middle part of figure 1. Finally, a vendor may state that he used a particular model for making the cost estimate, and the government client may choose to validate the cost estimate using a different estimating model.

On one hand it would appear that the use of more than one model has definite benefits. Since each model uses its own set of algorithms and unique inputs, each model "looks" at the software development process in a somewhat different way. Hence, the set of different outputs obtained yields a range of uncertainty bounding the "correct" cost and completion time, as illustrated in figure 1. On the other hand, unless the models have been used in a consistent fashion they can actually describe different, inconsistent sets of conditions, and the range of uncertainty obtained will be wider than it need be. Thus it behooves the recipient of the two (or more) sets of results to become familiar with each of the models used, and be able to validate that the set of conditions postulated are consistent, if not identical. The objective of this paper is to provide the user with certain guidelines for doing so.

I was first confronted with the problem under discussion when working on a joint project between PRC and Martin Marietta. Using the SLIM model I provided the project with cost estimates that differed considerably from those obtained by two Martin analysts using the RCA PRICE S model. In fact each of these analysts came out with different results while using the same model. A fourth (different) set of results was obtained by an analyst using an internal Martin Marietta model. Obviously, it was important that we resolve this problem prior to our meeting with the client where we were asked to present the results obtained from each model.

In attempting to correlate these different results, several points became obvious. First, the two analysts using PRICE S used different inputs which obviously produced different outputs. The second point is that PRICE S defines "Development Costs" and "Development Time" differently from SLIM, and hence an adjustment of the SLIM results had to be made to be consistent with the PRICE S results.

As Chairman of the Software Subgroup of the SSCAG (Space Systems Cost Analysis Group), I thought this problem important enough to organize a working group which would identify structural differences which might exist among the models, and devise a way of introducing more consistency into the estimating process. For ease of meeting an East Coast committee was formed to grapple with this process.

Members of the committee were Bob Park (RCA PRICE S) and Larry Putnam (SLIM), representing two model builders; Chris Mankiewich (MITRE), and myself (PRC), representing model users; LtCol Frank Hall (OSD P&A) and Steve Gross (USN), representing DoD; and Tom Masters (NSA), Jack Murray (NSA), representing both users and DoD. Since the membership was drawn from both SSCAG and ISPA members, Ed Tilford (Sperry Univac) and I agreed to co-chair a Joint SSCAG/ISPA Software Parametrics Committee to handle projects such as this.

The objective of this paper is to present an overall discussion of the problem, and a progress report of the results to date.

2. Project Boundary Conditions

To appropriately manage this project certain ground rules were established. First, as indicated previously, since each of the software models are different in both inputs required and algorithms used, it was obvious that the results obtained from the different models are bound to be different. Hence the objective of the effort was not to compare the accuracy of the models. Rather it was to analyze each of the models to see if there is any guidance that could be furnished to users of these different models to enable them to produce results which, while not identical, would be at least be consistent with one another. Second, to test the methodology to be used, the initial phase of the effort would confine the scope of the problem as follows:

- Compare PRICE S to SLIM, including other models in the next phase.
- Focus on the major outputs of: 1) development time; 2) development effort required [since labor cost is equal to the product of effort and labor rate, both can be analyzed under effort]. Manpower staffing schedule, milestones and the other model outputs would be included in the next phase.

Third, after the initial phase was completed the study would be extended to include: 1) other outputs; and 2) other models of interest. Both Randy Jensen (Jensen model) and Barry Boehm (COCOMO) have already been contacted and have indicated an interest in participating in this project.

With this introduction in mind, we shall now focus on each of the factors which must be considered if the model outputs are to be made as equivalent as possible.

3. Analyzing The Models

Let us assume the scenario of a user preparing to apply each of the models to the same problem. As shown in figure 1, the set of input characteristics which are used for both PRICE S and SLIM may be characterized into these major classes:

- Specified development time
- Number of delivered source statements
- Technical and environmental parameters

Each of the models will now be considered for consistency with respect to each of these characteristics.

3.1 Specified Development Time

Many times the client specifies to the user in an RFP (Request For Proposal) a required development time, and this value should be used in estimating the effort required to develop the system. As indicated previously, the PRICE S and SLIM models differ in their definition of the phases to be included in the "development process", so care must be taken to make certain that the same "equivalent" development time is used for each model. Figure 2 illustrates the various phases as defined in each model. Perhaps the most explicit definition of the total software development process is described by Barry Boehm in his book, "Software Engineering Economics". Hence the software life cycle as describe in this book was used as the baseline for comparison. Boehm defines five phases up to IOC (Initial Operational Capability) in his book, as illustrated in Figure 2:

- o Plans and Requirements
- o Product Design
- o Detailed Design
- o Code and Unit Test
- o Integration and Test, ending at IOC when acceptably tested code is delivered to the client.

As shown, the COCOMO and SLIM models both assume that development begins with PDR. SLIM's Development phase corresponds to COCOMO's Detailed Design, Code and Unit Test, and Integration and Test phases, ending at IOC. SLIM's Development phase is not explicitly subdivided into the three corresponding COCOMO phases shown in Figure 2. Instead, SLIM specifies a staffing pattern following the Norden-Rayleigh distribution. In addition SLIM provides the values of certain milestones which occur during development; [The comparative analysis of milestones will be analyzed in the next phase of this project].

SLIM also generates estimates for two pre-development, front-end analytical phases: Feasibility Study (which corresponds to Plans and Requirements) and Functional Design (which corresponds to Product Design).

There is one exception for SLIM which was noted by Larry Putnam. Some SLIM users end development with what they call "Pilot Test", in which the code has been acceptably tested on the development computer rather than the operational computer. Most users, however, continue development by continuing operational testing on the final operational hardware. Client acceptance at the end of this test phase is defined as true IOC. The same SLIM model is used for both of these situations. However, the appropriate calibration factor must be used in each case. That is the model must be calibrated using historical data corresponding to the appropriate situation for the new project. Consistency is the major rule!

As shown more clearly in Figure 3, the structure of the PRICE S model differs considerable from COCOMO and SLIM in a number of ways. First, unlike SLIM, the basic PRICE S run provides an output report for each separate software subsystem being considered (three subsystems are considered in figure 3). Each report provides estimates of time and effort for three development phases:

- o Design
- o Implement
- o Test and Integration

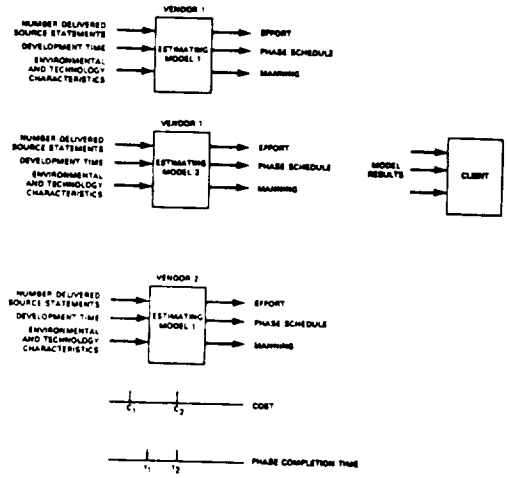


Figure 1 Scenarios Involved

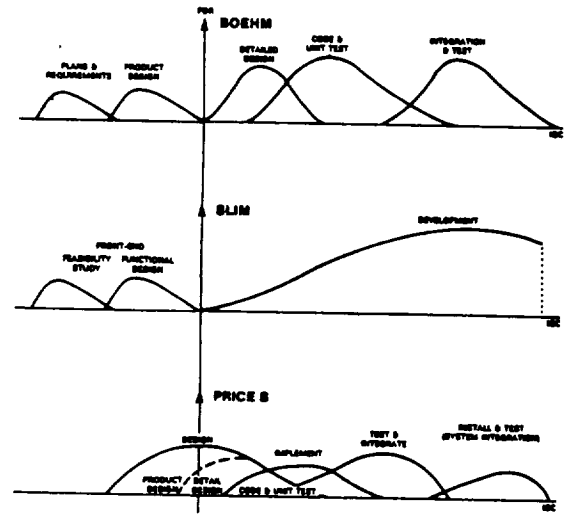


Figure 2 Phases of Effort As Defined in Various Models

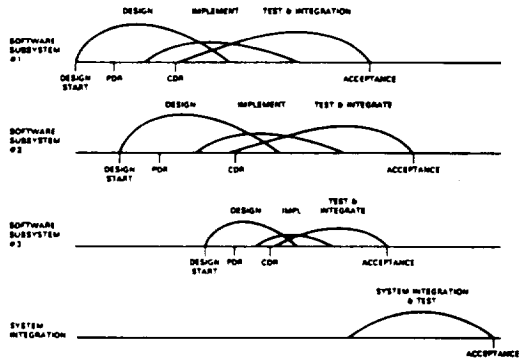


Figure 3 PRICE S Multiple Subsystem Analysis

Second, the start of the Design phase in the PRICE S model occurs before PDR and corresponds to the start of Product Design in COCOMO. Thus the PRICE S Design phase corresponds to the sum of COCOMO's Product Design and Detailed Design phases. While these three phases comprise what PRICE S calls development, SLIM acceptance (of the subsystem) appears to correspond with SLIM's end of Pilot Test. Hence PRICE S Implement and Test & Integrate phases seem to correspond to COCOMO's Code and Unit Test. To extend the development activity to true IOC another phase, called System Integration and Test, is required in which Operational Testing on the final operational hardware is performed. Client acceptance at the end of this phase corresponds to IOC. Estimates for this System Integration and Test phase can be obtained by a second PRICE S run using previous values of the input parameters plus the Integration parameter.

Thus comparability of outputs between the PRICE S and SLIM can be obtained by realizing that the PRICE S Development phase plus System Integration and Test phase is equivalent to SLIM's Functional Design phase and Development phase. Thus it must be determined if the client's specified development time begins with PDR (as with SLIM) or begins with SLIM's Functional Design phase. Appropriate input values to the different models can then be made.

In closing this analysis of the PRICE S development phases it should be noted that Bob Park indicated that some PRICE S users operate the model by defining IOC to be at the end of the first PRICE S run (ending at Test & Integrate), and do not make the second run of Install and Test. As in the discussion of SLIM, the model can be used this way if it is calibrated accordingly, using past historical data defined in the same way. Again, consistency is the rule!

One other major difference between the SLIM and PRICE S models with respect to schedule should be stated. SLIM initially calculates the minimum development time and the effort for this. The user can then insert a larger required development time, and the new value of effort will be calculated. This value will be reduced as the fourth power of the ratio of the longer scheduled time to the minimum time previously calculated. PRICE S also calculates two development times. First it provides the <u>nominal</u> time based on a typical schedule (remember that this time and effort also includes the front end systems design, equivalent to the Functional Design phase of SLIM). When a specified schedule is stated as an input, PRICE S increases the effort (whether this specified schedule is greater or less than the typical schedule), but not to the same degree as SLIM.

3.2 Number of delivered source statements

In the case of SLIM, these are the equivalent number which must be developed or reworked, specified in the units of the predominant language used. For PRICE S they are in equivalent units of delivered, executable, machine-level statements. Thus an acceptable conversion factor must be used to produce equivalent values of this parameter for both models.

3.3 Other Input Parameters

The remaining input characteristics consist of a set of parameters which further describe the characteristics of the system requirements, the capabilities of the available personnel, and the development tools they will employ. The COCOMO and SLIM models provide a calibration parameter which reflects the past experience of similar type projects in the form of a "Productivity Index". PRICE S actually

uses two calibration parameters. The first, called RESOURCE, permits the user to insert data relating to past experience (delivered lines of code, development time, and effort), which allows PRICE S to calculate the value of RESOURCE associated with past projects. This value plus the complexity parameter are then used to "fine tune" the PRICE S model to accommodate future developments of the same type of projects. One word of caution regarding this type of calibration is in order. The calibration factor is based on the same set of conditions as the past project(s) (unless you specifically tell the model otherwise). This means that the same mix of personnel are assumed. (Are secretaries and other support personnel included in the effort, or carried as overhead?). As described previously, the same definition of completion time is implicitly assumed. This may mean a true IOC with a given acceptability standard. Or it may mean development ends with "Pilot Test" and a lower acceptability standard. SLIM operates in a similar fashion, in which the calibrate function generates a "Technology Factor" which is then used in all further calculations. Thus, it must be emphasized that all models need to be calibrated using data from certain past projects that are most closely related to the new proposed project being estimated. To the extent that the new project differs from these past projects the analyst may then modify the estimation procedure to reflect these differences as follows:

- Change the calibration factor (or other parameter values) to reflect the new method of implementing the development process.

- Adjust (add or subtract) the model outputs of time and cost to account for: 1) all cost elements which should be included in the final estimate; and 2) total completion of Development to the true IOC as defined by the client; 3) any differences in the clients acceptability standards of the software product delivered.

4. Summary of Approach

Having described the analysis used, we shall now summarize the key points involved in obtaining comparable results from different parametric models:

a. The situation described is the use of different models for estimating the development cost of the same software system.

b. Formal ground rules or guidelines should be issued to all model users so that they may understand the potential pitfalls which prevent obtaining equivalent estimates, and hence can avoid these pitfalls.

c. To assure uniformity of deliverables (between separate vendors generating the system design and their estimates), two definitions of terms must be made ahead of time:

- The term "Software Development" needs to be defined in either of two ways: 1) As in the case of SLIM and COCOMO (i.e. starting with detailed design at PDR); or 2) as in the case of PRICE S to include Product or Functional Design. This ties down the definition of the start of development.

- Proper definition of IOC, in terms of meeting the acceptance standards of the delivered system on the operational hardware system. This ties down the

completion of development.

Based on these definitions, the definition of development schedule will be understood.

d. The objective of the estimating exercise will be to calculate the effort required to perform the software development at a stated level of acceptability and meeting the required development schedule.

e. Calculate the input value of development time to be used based on the following factors:

- o Specified development schedule
- o Does this time include Functional Design phase or not?
- o Does the model to be used include the Functional Design phase as part of development time (PRICE S) or not (SLIM)?
- o Do projects from which the Calibration Factor was derived include going to true IOC or not? If not, an estimate of the additional time required to go to true IOC must be made (and then subtracted from specified development time).

f. Decide upon the values of the other input parameters used by the model. If more than one model is to be used by a user the value of each set of inputs should be generated by the same individual(s) so that relative equivalence can be obtained, minimizing input errors.

g. From the projects used for calibration, decide if any additional cost elements must be added to the value of effort calculated by the model, to include:

- o Additional effort required from the time associated with the end of development (as calibrated for the model) to true IOC.
- o Additional effort required for development but not included by the calibration factor.

5. Conclusions and Recommendations

Having produced this progress report, here are the tasks to be performed in completing this project:

a. Circulate this progress report to all members of the Software Committee, as well as the model builders (Bob Park, Larry Putnam, Barry Boehm and Randy Jensen) to obtain their validation of the analysis and findings. In addition, Barry Boehm and Randy Jensen will be requested to add data related to their models to enlarge the scope of the report.

b. The scope of effort will further be enlarged by consideration and correlation of other factors of interest, such as the additional model outputs (milestones, staffing pattern, code production, pages of documentation, front end plans and requirements phase etc.

c. Recommendations from model builders and users of a common set of terms to be used.

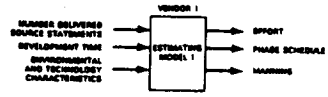
'. Publication of progress and final reports in the ISPA Journal.

ADJUSTING SOFTWARE ESTIMATING MODELS TO OBTAIN COMPARABLE RESULTS

by

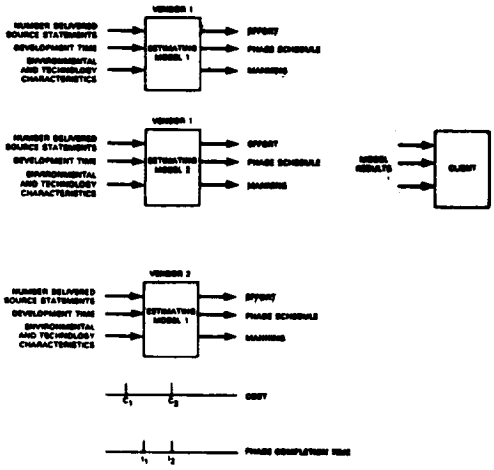
Bernard H. Rudwicz, P.E.
Chief Management Scientist
Planning Research Corporation
PRC Government Information Systems
McLean, VA.

PROJECT BOUNDARY CONDITIONS

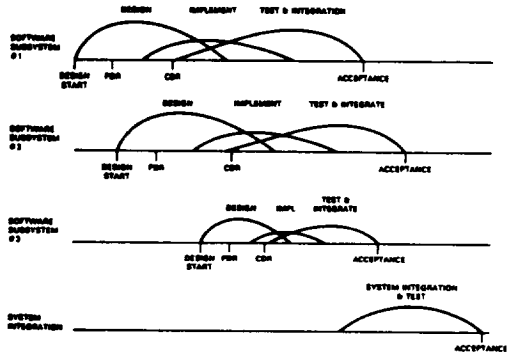


- Results will differ since models differ
- Objective is consistency of results
- Initially compare PRICE S to SLIM
- Focus on major outputs:
  - Development time
  - Development effort/cost
- Later analysis:
  - Other outputs
  - Other models

SCENARIOS INVOLVED

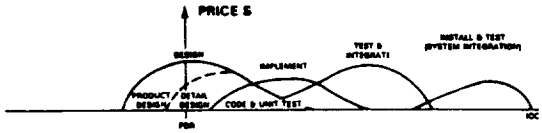


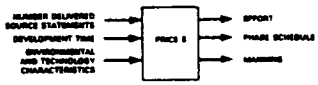
PRICE-S MULTIPLE SUBSYSTEM ANALYSIS

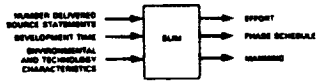


PAGE 158 INTENTIONALLY BLANK

ANALYZING MODELS USED

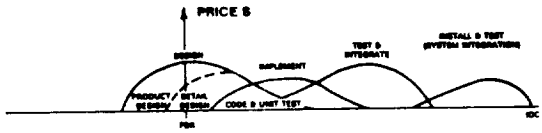






- Need consistency on:
  - Number of delivered source statements
  - Specified development time
  - What is meant by "development"?
  - Technical and environment parameters
- Need for calibration
  - Calibrate to related programs
- Adjust output results for total cost and time

SUMMARY OF APPROACH



- Situation: How to use different models to obtain comparable costs and time for developing same software system
- Recognize that models may define "development" in different ways:
  - May start with product design
  - May end before IOC
- Need to use equivalent values of related input parameters
- This requires an understanding of each model being used
- May need to modify output results based on:
  - Differences with projects used in calibration
  - Add other cost elements
  - Add additional time and effort

Software Estimating and Scheduling Using Parametric Models

William G. Cheadle

Martin Marietta Denver Aerospace

Introduction

The June 1983 Aerospace Corporation's Third Mission Assurance Conference objective is to examine key elements and factors critical to space mission success and to identify tools, techniques and methods to achieve mission assurance. The Software Cost Estimating Workshop will consist of members of the software cost model development community and model users. All of these people have extensive experience with software cost estimating and scheduling parametric models and have been invited to share this vast experience with the other members of the workshop.

Martin Marietta's Parametric Model

As a member of this invited group, I have been privileged in that I am a user of several of the models that will be discussed, as well as being the developer of the Martin Marietta Software Parametric Cost Estimating Model (PCEM). Being a user as well as a developer, I relate to both activities and want to share with you some of my experiences, which I hope will be of some benefit to all of you. Martin Marietta Denver Aerospace decided in 1975 to establish a study and analysis effort tasked to investigating the software development process and to learning how to plan, schedule, size and estimate software. During a period of several years we performed an extensive literature survey and evaluation of past and state-of-the-art software cost estimating models and methods. Figure #1 lists some of the models that were evaluated.

The outcome of this evaluation was that pieces of these models were determined to be useful within our company environment and were studied in detail, but none was felt to be adequate for our use. So, Martin Marietta decided to develop their own company peculiar parametric software cost estimating and software scheduling model which came to be known as the PCEM, Parametric Cost Estimating Model. The model is designed to run on a VAX 11/780 computer and

was developed using actual software development data collected from past completed software projects. Cost Estimating Relationships (CERs) were created, project and mix complexity factors were established and independent variables were identified. Software estimating equations were developed for flight and ground operating environments, for assembly and for high order language software. These equations and our resulting manual and computerized parametric models have been validated by comparing sizing and labor actuals with our estimates and by comparing projected schedules to actual software development span times on completed projects.

Martin Marietta's computerized software parametric cost estimating model has provided our estimating and engineering people with a cost estimating, sizing, scheduling and risk assessment tool. This new tool has greatly aided us during proposal periods and with statusing on-going projects by providing "to go" cost and schedule estimates. Having a "company peculiar" model which reflects the specific way Martin Marietta does business gives us a distinct advantage in costing and scheduling our software development activities.

The Software Development Life Cycle

Figure #2 reflects the software development phases, subphases and design reviews used by Martin Marietta. These phases and design reviews are required by our software development standards. The PCEM model estimates budget for all the phases and subphases reflected in Figure #2.

Figure #3 compares the phases and subphases of software development covered by our Martin Marietta model, PCEM, and the phases and subphases covered by some of the other widely used software cost estimating and scheduling models.

The Martin Marietta Parametric Cost Estimating Model (PCEM) estimates software development costs and provides a software development schedule for three organizations: Systems for software development, Test for software

development and Software Engineering for the complete software development life cycle. As reflected in Figure #3, our PCEM model schedules and estimates the entire software development life cycle. Design reviews are used as milestone points to monitor how the software development is progressing. There are certain documents that must be completed before a design review can be conducted. Often times front end activities and documents have already been prepared and our PCEM model identifies these documents and allows credit to be taken. The PCEM model reflects the Martin Marietta way of doing things and expects that the company software standards will be followed and that certain documentation and procedures will be provided and adhered to. For instance, the PCEM model questions how requirements are to be defined. The requirements definition is the most important part of software development, and we have taken a structured systems analysis approach to defining requirements and require that logical requirements and design be provided prior to defining a physical design. One tool and technique we have found to be helpful for requirements and design definition is the use of data flow diagrams. When data flow diagrams are used during the planning phase (prior to contract award) they greatly aid us in determining the costs and schedules required for the software development activity.

Data Flow Diagrams and Flow Charts

Data flow analysis (data flow diagrams) and definition methodology consist of tools and techniques that build a logical model of a proposed system using graphical techniques which enable users, system analysts and designers to get a clear and common picture of the system and how its components fit together prior to the allocation of requirements to hardware and software. There has been a great temptation in the past to sketch a physical design (flow chart) of a new system before we have a full understanding of all the

logical requirements. In the software world, for instance, the very act of drawing a flow chart means that a decision has already been made as to whether the input will be on cards or through a CRT, which files will be on tape and which will be on disk, and which programs will produce output. Figure #4 reflects the common conventional flow chart symbols.

The use of data flow diagram analysis tools to aid in defining requirements and the design enables the systems analyst to understand the customer requirements, and to identify logical requirements without dictating a physical implementation. This front-end analysis identifies early those logical requirements which will require trade study activities before they can be allocated to either hardware or software.

Data flow diagrams show the sources and destinations of data, identify and name the logical functions, identify and name the group of data elements that connect functions together and identify data files which they access. A complete system (hardware and software) data flow diagram can be constructed using only three basic symbols plus arrows that specify the direction of the data flow. Figure #5 shows the four basic symbols used in logical data flow diagrams.

The Logical Functional Specification

When the data flow diagrams plus the other documents which make up the logical data flow concept are prepared, we have a logical functional specification. This provides a detailed statement of what the system is supposed to do, free from physical considerations of how it will be implemented. This logical functional specification represents a firm baseline of hardware/software requirements from which to design the system.

The logical functional specification is included as a part of the program Systems Requirements Specification. This systems specification reflects customer, derived and operational requirements which will be allocated to hardware, software or other means for solving the customer's overall problem. Figure #6 is an example of an Input/Process/Output Logical Data Flow Diagram.

Architecture

The selection of a hardware, software architecture follows the data flow diagram activity. The selection of a specific brand of computers and peripherals follows the determination of whether the design concept will feature a centralized (information processing) or a distributed (network processing) system. Figure #7 shows an example of a centralized and of a distributed system.

Successful Software Estimating

The amount of quality activity that goes into the front end process of any software development activity determines how successful that software development will be. Successful software scheduling and cost estimating parametric models must be able to evaluate these quality activities and must be able to penalize those projects that do not adequately define requirements. New techniques and tools such as the use of data flow diagrams must be taken into consideration and the parametric model must be able to evaluate its influence. Another important criteria of a successful software parametric estimating model is how it handles lines of code sizing. Source lines of code vs. object lines of code (machine instructions). Assembly language vs. high order language. Many parametric models do not take into account the difference that the language makes in determining cost estimates. We have found at Martin Marietta that the language affects both costs and schedule.

Figure #8 shows sizing in assembly language and Fortran.

Software Development Schedules

The schedule selected can greatly affect the software development costs and parametric models must be able to accurately determine the optimum schedule. This schedule must reflect the whole software development activity and not just a part of it. Penalty factors must be built into the models to penalize a schedule that is either too short or one that is too long.

Software Parametric Models

The new software development methodologies and software parametric cost and schedule models have greatly influenced the way software is being developed today. Software engineers in the past were skilled in only one or two of the software disciplines. They were skilled in defining requirements or design or they were programmers or test engineers; very few had all of these software development talents. For this reason, the engineers' "bottoms up" estimate of what the software development activity would cost and how long it would take usually was incomplete, fragmented, or was just a partial estimate. The integration of all of these separate and fragmented pieces was often overlooked. Also, the engineers tended to think of the software development process as a "straight line" approach where the requirements phase was followed by the design, code and test phases. What we found out was that "real life" software development projects were proceeding "iteratively", meaning that some requirements were provided, then a little design, then back for more detailed requirements, then more design, then some coding, then more design, and so on. Figure #9 displays a straight line and iterative approach to software development. The path of such a project moved back and forth through

the four phases of software development or, as software parametric modelers have discovered, there were large overlaps between the phases. Management control over this kind of activity has been difficult. Interim status reporting, which required the analyst or engineer to estimate the degree of completion of an activity (such as design being 40% complete or requirements being 90% done) meant very little. Management control cannot be based on activities, as it has in the past, but must now be based on documentation. What is really being produced in a software project is a set of deliverables (documentation and code listings) of increasing refinement leading up to the delivery of each document and version of the document. All of these deliverables must be identified in the Software Development Plan (SDP) and they must all be statused and reviewed at scheduled design reviews. Thus, the ground rules of management control of a project have changed. Today we must put the emphasis on the delivery of identified products, rather than the completion of specific activities. Parametric models must identify with and relate to these software development products.

Summary

In summary, the way we develop software and the methodology we have been using have been changing. Software parametric models will have to reflect these new improved procedures. The front end requirements activities are becoming more and more important, and the percentages of time spent for each of the four phases of software development has already shifted. Documentation is being created early and reviewed at scheduled design reviews. New requirements and design tools are being developed and new languages are changing the way we have done things in the past. Because of this upheaval, software estimating and scheduling parametric models will have to consider and evaluate each new change. Data bases will have to be updated and new inputs will have to be made to Cost Estimating Relationships to see what

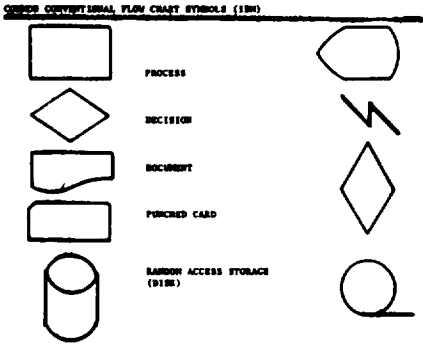
the effects might be. Past research on the software development process and the life cycle will have to be re-evaluated to see if it is still applicable today. Even though there have been changes for the better, software developers still are experiencing problems with developing large real time command and control software systems. Additional new tools must be developed and software developers must continue to be retrained on the use of these new methods and ideas. Both the software development community and the parametric model builders must realize that this is a new era of change, and both groups must communicate to make sure that the models and the methodology are staying together.

SOFTWARE ESTIMATING MODELS — FIGURE #1

- BOEING COMPUTER SERVICES, INC.
- TRW/R. W. WOLVERTON MODEL
- AEROSPACE CORPORATION/V. SCHNEIDER
- IBM/C. E. WALSTON AND C. P. FELIX MODEL
- SYSTEM DEVELOPMENT CORPORATION MODEL
- GENERAL RESEARCH CORPORATION MODEL
- RADC/DOTY ASSOCIATES
- TRW/B. W. BOEHM COCOMO MODEL
- UNIVERSITY OF MARYLAND/N. ZELKOWITZ
- FARR AND ZAGORSKI MODEL
- NAVAL AIR DEVELOPMENT CENTER MODEL
- TECOLOTE RESEARCH INC. MODEL
- IBM/J. D. ARON MODEL
- LAWRENCE H. PUTNAM SLIM MODEL
- RCA PRICE S SOFTWARE MODEL
- THE JENSEN MODEL

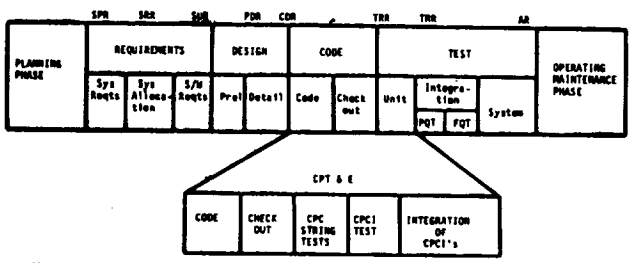
MARTIN MARIETTA

CERTAIN CONVENTIONAL FLOW CHART SYMBOLS (IBM) — FIGURE #4



MARTIN MARIETTA

SOFTWARE SIZING AND ESTIMA NS — FIGURE #2



SPR System Planning Review
SRR System Requirements Review
SDR System Design Review
PDR Preliminary Design Review
CDR Critical Design Review
TRR Test Readiness Review
AR Acceptance Review
PQT Preliminary Qualification Test
FQT Formal Qualification Test
CPC Computer Program Component
CPCI Computer Program Configuration Item
CPT&E Computer Program Test & Evaluation

MARTIN MARIETTA

LOGICAL DATA FLOW DIAGRAM SYMBOLS — FIGURE #5

THIS SYMBOL DESCRIBES EXTERNAL SOURCES AND DESTINATIONS OF THE DATA.

THIS SYMBOL DESCRIBES IN ORDINARY TERMS THE PROCESS TO BE PERFORMED.

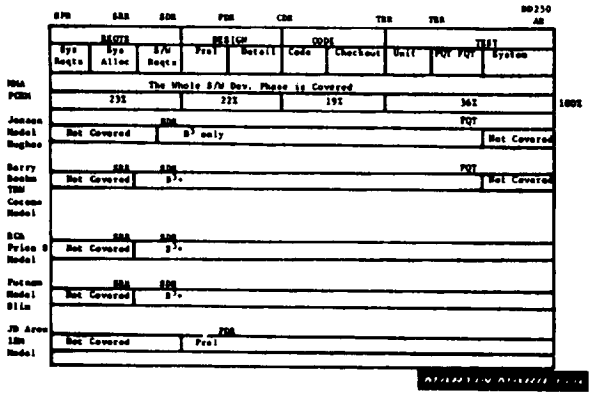
THE SYMBOL IDENTIFIES PLACES WHERE DATA IS BEING STORED BETWEEN PROCESSES.

FLOW OF DATA

MARTIN MARIETTA

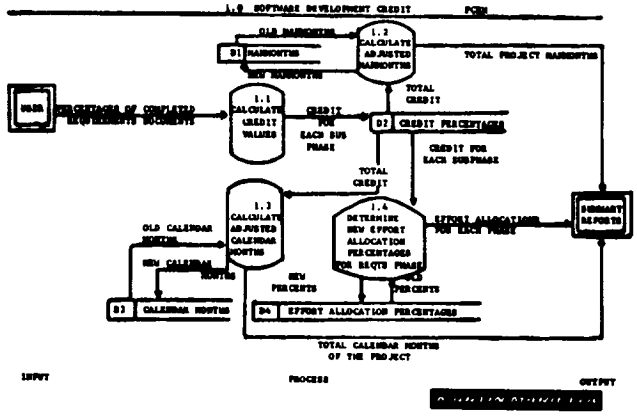
FIGURE #3

SOFTWARE DEVELOPMENT PHASE, SUBPHASES & DESIGN REVIEWS COVERED BY PCEM AND BY SOME OF THE OTHER WIDELY USED SOFTWARE COST ESTIMATING AND SCHEDULING MODELS



MARTIN MARIETTA

LOGICAL DATA FLOW DIAGRAM — FIGURE #6



MARTIN MARIETTA

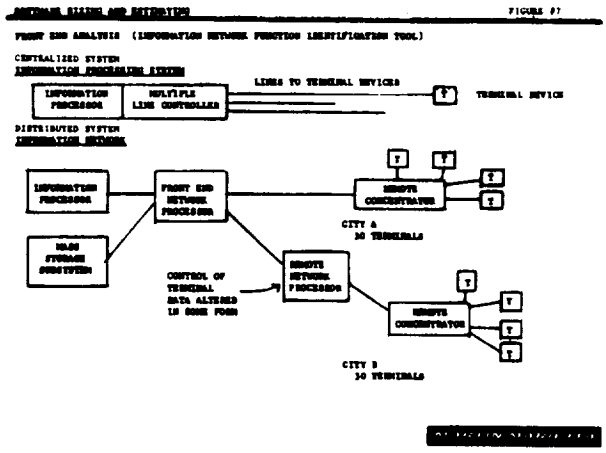
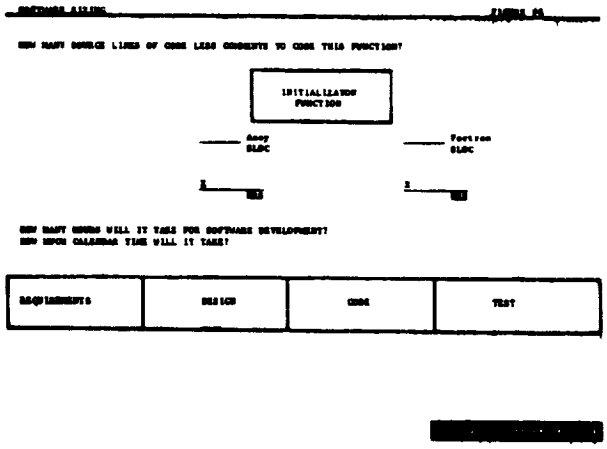


FIGURE 87
FRONT END ANALYSIS (INFORMATION NETWORK FUNCTION IDENTIFICATION TOOL)
CENTRALIZED SYSTEM
INFORMATION PROCESSING SYSTEM
INFORMATION PROCESSOR
MULTIPLE LINE CONTROLLER
LINES TO TERMINAL DEVICES
TERMINAL DEVICE
DISTRIBUTED SYSTEM
INFORMATION NETWORK
INFORMATION PROCESSOR
FRONT END NETWORK PROCESSOR
REMOTE CONCENTRATOR
CITY A
30 TERMINALS
MASS STORAGE SUBSYSTEM
CONTROL OF TERMINAL DATA ALTERED IN SOME FORM
REMOTE NETWORK PROCESSORS
REMOTE CONCENTRATOR
CITY B
30 TERMINALS


SOFTWARE SIZING
FIGURE 88
HOW MANY SOURCE LINES OF CODE LESS COMMENTS TO CODE THIS FUNCTION?
INITIALIZATION FUNCTION
Assy SLOC
Fortran SLOC
HOW MANY HOURS WILL IT TAKE FOR SOFTWARE DEVELOPMENT?
HOW MUCH CALENDAR TIME WILL IT TAKE?
REQUIREMENTS
DESIGN
CODE
TEST

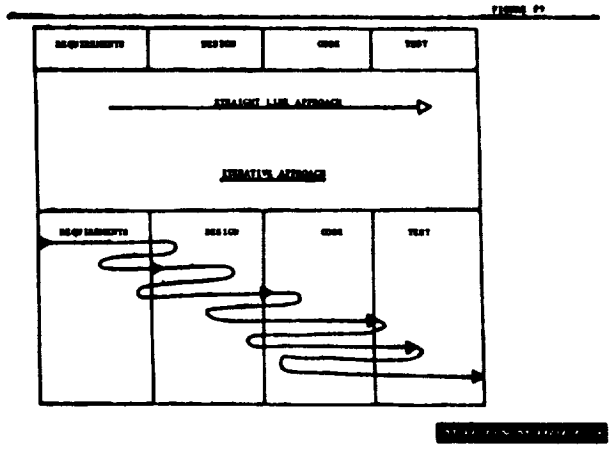


FIGURE 89
REQUIREMENTS
DESIGN
CODE
TEST
STRAIGHT LINE APPROACH
ITERATIVE APPROACH
REQUIREMENTS
DESIGN
CODE
TEST

SUMMARY OF ISSUES AND RECOMMENDATIONS

1. Software (Instruction Set Architectures)

Issue

MIL-STD-1750A ISA Executive

Recommendations

Most of the requires MIL-STD-1750A support software is in place; however there is no standard, commonly used real time executive. It is recommended that such an executive be developed and made available.

Suggested OPR

ECSPO, STARS, Industry

2. Software (Instruction Set Architectures)

Issue

Ada Compiler for MIL-STD-1750A Computers

Recommendations

The development of Ada Compilers targeted to MIL-STD-1750A computers is lagging. It is recommended that 1750A targets be added to the Air Force Ada Integrated Environment and/or the Army Ada Language System as soon as possible.

Suggested OPR

Issue

AJPO, ECSPO, STARS

3. Software (Ada Transition)

Issue

Measurement is critical to better software development (in general) and maximum use of Ada specifically. Data collection necessary for measurement is a cost very seldom funded, hence an activity performed infrequently.

Recommendations

Develop and support a mechanism that would motivate contractors to collect and make measurement data available. There is some mention of this in the STARS program plan and International Society of Parametric Analysts (ISPA). The Space System Cost Analysis Group (SSCAG) has been addressing this for cost data.

Suggested OPR

STARS Program Office, Air Force Space Division, SSCAG, NASA International Society of Parametric Analysts (ISPA)

4. Software (Ada Transition)

Issue

For successful use of Ada (i.e., maximum benefit - reuseability, portability) significant training will be required - some estimate 35 to 40 days in a 6 month period.

Recommendations

Government should fund development of innovative training programs fostering the use of technology (e.g., teletraining, computer-based education (CBE), computer-aided education (CAI), video disc).

Suggested OPR

STARS Program Office, Ada Joint Program Office

5. Software (Ada Transition)

Issue

Early research has shown that design methodologies different from those used today will have to be adopted to achieve the stated benefits of Ada (e.g., portability) and to effectively use many of its features (e.g., packaging).

Recommendations

Commence evaluation of known design methodologies as part of a phased approach to identifying and developing (if necessary) design methodologies best suited for Ada.

Suggested OPR

AJPO, Industry

6. Software (Ada Transition)

Issue

Ada will impact the validity of existing models and paradigms used for software costing and describing the software life cycle. The mandatory use of Ada may temporarily set back the gains in the use of models to estimate and manage software costs.

Recommendations

a) Current mode developers and users and the developers and potential users of Ada should work closely together during the current development and future transition of Ada.

b) Fund the development of engineering examples, paradigms, and models that reflect using Ada for software development.

Suggested OPR

Space Systems Cost Analysis Group (SSCAG), Government, International Society of Parametric Analysts (ISPA)

7. Software (Cost Estimating)

Issue

Proper use of cast models for system acquisition and cost management often requires the use of more than one model. The productivity of models (over 400) and the lack of formal comparative guidelines causes problems for consistent use across government and industry.

Recommendations

Model users must continue their collective and objective evaluation of available models to assure consistent parametric definitions and common application methods, and to develop formal comparative guidelines for the results of the various models. Recent efforts by the International Society of Parametric Analysts (ISPA) and Space Systems Cost Analysis Group (SSCAG) are commended.

Suggested OPR

ISPA, SSCAG

8. Software (Cost Estimating)

Issue

Both the software development community and the parametric model builders must realize that this is an era of change, perhaps rapid change. Both groups must communicate to ensure that the models and methodologies stay linked.

Recommendations

More joint conferences and workshops bringing together modelers, software development practitioners, and software technology developers.

Suggested OPR

NSIA, ISPA, SSCAG

9. Software (Cost Estimating)

Issue

Cost estimating models are highly dependent on new language development (e.g., Ada), new methodologies and techniques, (e.g., reuseable code), and changing technologies (e.g., tools). As these change rapidly, the models may be invalid or lose their accuracy.

Recommendations

Cost modelers need to work closely with new language developers and prospective users in order to adapt current models to changing productivities, new langugage complexities, and peculiar schedule profiles. An implementation mechanism could be sponsored symposia and workshops bringing these groups (modelers new language developers/users together).

Suggested OPR

NSIA, SSCAG, ISPA

10. Software (Cost Estimating)

Issue

All software cost models depend on lines of code as the basic input parameter. Industry needs better methods to estimate lines of code early in the concept phase.

Recommendations

Continue existing efforts to collect and analyze data on program size to determine its correlation with other parameters that are available with greater definition at the concept phase.

Suggested OPR

Model developers, International Society of Parametric Analysts (ISPA), Space Systems Cost Analysis Group (SSCAG)

11. Software (Management)

Issue

Contracts for software development are established before requirements are adequately defined.

Recommendations

a) Establish software prototyping prior to PDR, encourage competitive prototyping of software (software shootout), and provide adequate funding for this activity.

b) Contractually separate the validation phase from the development phase.

Suggested OPR

Government Contracting Agencies

12. Software (Management)

Issue

Proper and sufficient management leverage points have not been established for software development. Software is being force fit into the traditional hardware methodology.

Recommendations

The operational concepts aspects of the project must be established as part of the functional baseline.

The pre-PDR phase must receive the highest management attention and focus to clear all TBDs and to clarify design requirements.

Suggested OPR

Software Development Managers

13. Software (Management)

Issue

Software development must be planned and managed to the total life cycle of the program rather than just to the acquisition process.

Recommendations

Government must review and assess the budget allocation process to ensure that the O&M aspects of the project are adequately factored into the acquisition process.

Suggested OPR

Government Acquisition Agencies

14. Software (Management)

Issue

A lack of software expertise is evident at the system/project management level within both the government and contractor organizations.

Recommendations

Broaden the exposure of top management personnel within government and contractor organizations to the particular needs of software development. Augment the project/program management staff with software professionals. Elevate the software management responsibility within these organizations to the program level.

Suggested OPR

Industry Executives and Government Management (NASA, SD)

WORKSHOP J

TEST

Chairmen:

Dr. Alfred Krausz
Manager, Technology Test and
Field Operations Division
TRW

Les Jepsen
Technical Requirements Specialist
USAF Space Division

Coordinator:

Frank Refalo
LMSC

Maj. Ron Dodge
USAF Space Division

WORKSHOP PRESENTATIONS

Effective Spacecraft Test Programs	Don Hill, LSMC
Quantitative Airlink Satellite Testing	Neil K. Shirk, General Electric
Testability Technical Overview	Bill Keiner, NSWC
On-Orbit Test Using Shuttle and Space Station	E. R. Medler, TRW

EFFECTIVE SPACECRAFT TEST PROGRAMS

By: D. C. Hill, Lockheed Missiles & Space Company

INTRODUCTION

This paper describes significant test changes at Lockheed Missiles & Space Company (LMSC) since the late fifties and how those changes have affected spacecraft reliability.

Figure 1 describes the LMSC Spacecraft flight history, a 20 spacecraft moving average as a function of success and calendar year. The major increase in successful satellite operation from the problems of the early sixties has been due to improved design techniques, more thorough design reviews, and better corrective action. Improved test programs since the testing of the Discoverer Satellite in 1958 were a major contributor and several major test "milestones" are noted, the bands shown in Figure 1 being due to the tests being implemented at different times due to LMSC's many programs and customers.

As a result of these programs, LMSC has had only one spacecraft placed in orbit that has not exceeded its mission contract life in the last 13 years, over 60 successful "long life" spacecraft, one failure, and even that spacecraft accomplished its prime mission technical objectives.

DISCUSSION

As a result of the early flight failures, LMSC investigated test techniques to improve the reliability of its spacecraft and launch system. System level thermal-vacuum cycling was introduced in 1962 and System Level Acoustic testing was introduced in about 1968. These tests were designed to reproduce, or slightly exceed worst case flight conditions (funnel concept) and they helped as Figure 1 shows, although the cost benefits of Acoustic Testing are still a controversial subject.

TWENTY SPACECRAFT MOVING AVERAGE

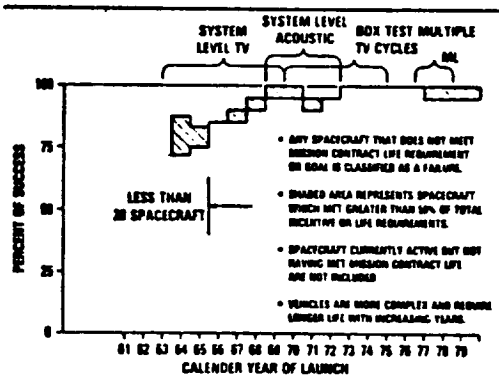


Figure 1

In the late sixties, LMSC's concerns about system test and flight failures and the more complex and longer life spacecraft being contracted, led to a multi-approach attack to increase system reliability. The approach included parts programs, box testing, and systems testing. A study of System test failures revealed data similar to that in Figure 2 which showed that high percentages of failures were due to workmanship and to parts.

SYSTEM TEST FAILURES BY CATEGORY

MAJOR DEFECT CATEGORIES	NO. OF FAILURES	% OF TOTAL	% IN SETS 9 & 10
WORKMANSHIP	277	37.6	27.0
PARTS	191	26.0	28.5
DESIGN	154	20.9	28.5
VERIFIED UNKNOWN	114	15.5	16.0
TOTALS	736	100.0	100.0

BASED UPON DATA SETS 4 THROUGH 16

Figure 2

Data sets refers to groupings of vehicles by time periods, set 10 being the most recent. Data for Sets 1 thru 8 in Figures 2 and 10 to 13 are from Reference 1.

A 1969 study on a major program showed that many complex electronic hardware units failed the box level ATP (Acceptance Test Procedure) temperature cycle or the System Level Thermal Vacuum Testing. Forty-two boxes were selected and an additional three ATP temperature cycles performed.

Over one-third of the boxes failed the test, but the net result was that flight and system test failures were reduced and essentially eliminated on two problem box types. The test program was later summarized in an LMSC report (Reference 2). The study resulted in a series of Transponder tests starting in 1971 (Figure 3)

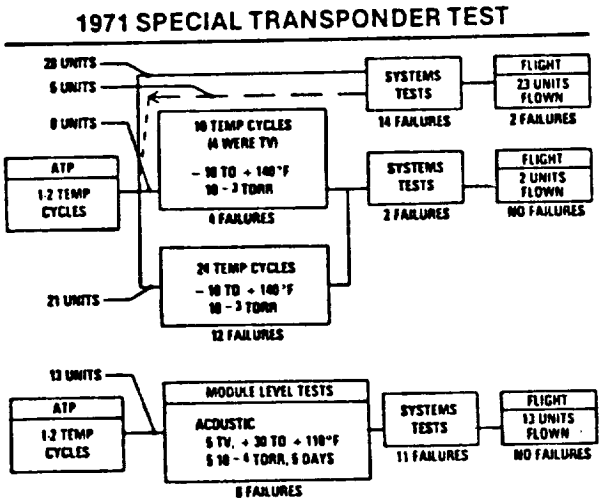


Figure 3

which showed a major reduction in system test and flight failures after additional thermal cycles were conducted (Reference 3). As a result of these studies, the LMSC Space System Division adopted a standard of eight temperature cycles for electronic boxes which was included in its 1972 test baseline. The baseline formed the centerpiece for the First Aerospace Testing Seminar held in Palo Alto, California in 1973 which was the forerunner of MIL-STD-1540. Much of this temperature cycling data was included in Figure 4 from a 1972 Martin Marietta report prepared under NASA Contract (Reference 4).

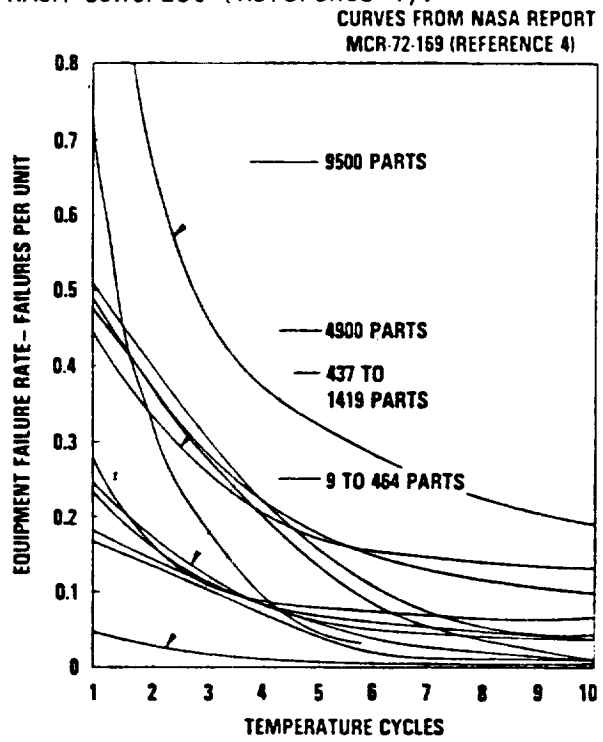


Figure 4

Failures caused by piece parts were also attacked by an improved parts screening program and by ensuring the parts were fabricated correctly. The latter took the form of Destructive Physical Analysis (DPA) or Delidding where small quantities of devices were procured. DPA is an engineering evaluation and consists of opening part samples and examining them for defects under magnification, including use of a scanning electron microscope. In addition, since many of the major part failures experienced were semiconductor devices, LMSC devised a program in 1972 to attack the predominant modes of failures described in Figure 5.

LMSG MONITORED LINE PARTS PROGRAM

SEMICONDUCTOR PARTS PROBLEMS	ACTION TAKEN
• DIE SURFACE PROBLEMS - SCRATCHED & UNREMOVED METALLIZATION - OXIDE PIN HOLES OVER JUNCTION - WIRE BONDS EXCESSIVELY OFF PADS - SURFACE CONTAMINATION	- SEM AT WAFER LEVEL - PRE CAP INSPECTION 100X MINIMUM BY VENDOR AND LMSC
• WIRE BOND PROBLEMS - OPEN OR LOW PULL STRENGTH BONDS - IMPROPERLY FORMED	- CONTROLLED WIRE BONDING/TESTING
• CONDUCTIVE PARTICLE PROBLEMS - SHORTS - INTERMITTENTS	- PROTECTED DIE SURFACE (GLASSIVATION) - RADIOGRAPHIC INSPECTION PIND TEST (NEW) - DELIDDED SAMPLES BEFORE ACCEPTANCE - DELIDDED INVENTORY SAMPLES SELECTIVELY - PIND ADDED IN 1980
• ELECTRICAL TEST AND DATA PROBLEMS - MIXED AFTER ELECTRICAL TEST - TEST CONTACT PROBLEMS - PASS ROOM, FAIL HIGH OR LOW TEMP - EXCESSIVE PARAMETER DRIFT (WITHIN SPECIFICATION)	- POWER BURN IN 250 HOURS - SCREEN PARAMETERS AT TEMPERATURE - RECORDED PARAMETER DRIFT CRITERIA - DIAGNOSED CATASTROPHIC SCREENING FAILURES - BURN-IN LOT REJECTION CRITERIA - IMPROVED TRACEABILITY

Figure 5

To ensure the parts were fabricated correctly, Resident Quality and Reliability engineers were stationed at certain high volume semiconductor suppliers and performed the activities noted in Figure 6.

LMSC MONITORED LINE RESIDENT ACTIVITIES

- ASSEMBLY MONITORING
  - SEM AUDIT & APPROVE PHOTOGRAPHS LOT JEOPARDY (L,I)
  - WIRE BOND STRENGTH SAMPLING (L,I)
  - 100% PRE CAP VISUAL INSPECTION
  - CONFIGURATION CONTROL
- TEST MONITOR (AUDIT)
  - WITNESS STATION PROOFING
  - VERIFY EQUIPMENT CALIBRATION
  - VERIFY BURN IN BOARDS
  - VERIFY TEST TAPE
- PRE SHIP ACTIVITIES
  - DESTRUCTIVE PHYSICAL ANALYSIS (L,I)
  - EXTERNAL VISUAL SAMPLE
  - RADIOGRAPHIC FILM REVIEW
  - TRACEABILITY VERIFICATION
  - PART PACKAGE VERIFICATION
  - VERIFICATION REPORT
  - DATA PACKAGE VERIFICATION
- ADDITIONAL SPECIFICATION REQUIREMENTS
  - DIRECT FAILURE ANALYSIS OF SELECTED SCREENING REJECTS
  - REVIEW PDA LOT FAILURES AND DIRECT CORRECTIVE ACTION
  - MANAGE CONFIGURATION CHANGE REQUEST TESTING
- EVALUATE CONFIGURATION CHANGE REQUESTS
- REPORT SIGNIFICANT IN PROCESS YIELD LOSSES WITH SCHEDULE IMPACT
- ANALYZE DISCREPANT HARDWARE CONDITION, WHEN REQUESTED, FOR USER TO ASSESS
- MONITOR REPORT ASSEMBLY/TEST PROGRESS ON SELECTED "HOT" LOTS
- OTHER ACTIVITIES (NON INTERFERENCE)
  - MONITOR PROCESS VERIFICATION TESTING
  - DIRECT SPECIAL LOT TESTING
  - PROVIDE TRACEABILITY INFORMATION

Figure 6

The Monitored Line (ML) approach, which consists of controls, audits, and testing on a standard manufacturing line, was deemed to be more economical than a captive line approach since LMSC would have small quantities of parts on order at times.

Figure 7 indicates a failure rate improvement exceeding 3:1 has been experienced.

COMPARISON OF MONITORED LINE VS NON-MONITORED LINE SEMICONDUCTORS (ONE PROGRAM)

- PARTS PURCHASED IN SAME TIME FRAME
- PARTS USED IN SAME EQUIPMENT
- TRANSISTORS, AND DIGITAL, LINEAR AND HYBRID IC's

	QUANTITY PARTS	NUMBER FAILURES	% FAILURES	RELIABILITY GAIN FACTOR	
INTEGRATED CIRCUITS					
MONITORED	37863	68*	.127	3.1X	2X
NON MONITORED	5412	21	.388		
TRANSISTORS					
MONITORED	22863	4	.018	2X	
NON MONITORED	2884	1	.035		

* 52 FAILURES WERE DUE TO A METALLIC CONTAMINATION PROBLEM (DESIGN CHANGED)

Figure 7

Over 420,000 monitored line parts have been delivered to LMSC and its subcontractors, and there has been only one system test and no flight ML failures. The purchase price of ML parts may exceed that of standard "Hi-Rel" parts, but the cost of monitoring is less than itinerant plus receiving inspection (including in-house DPA), and the cost of failures is significantly less due to less ML failures. The result is greater productivity because the cost of ownership is less and there are fewer schedule problems. As a result, the ML parts are presently specified in MIL-STD-1546 as substitutes for MIL-STD 38510 Class S.

How did these improvements at box and part level affect systems test? Figure 8 shows the major test changes of several major LMSC programs broken up into three major time periods. These are

MAJOR BOX AND PARTS PROGRAM CHANGES

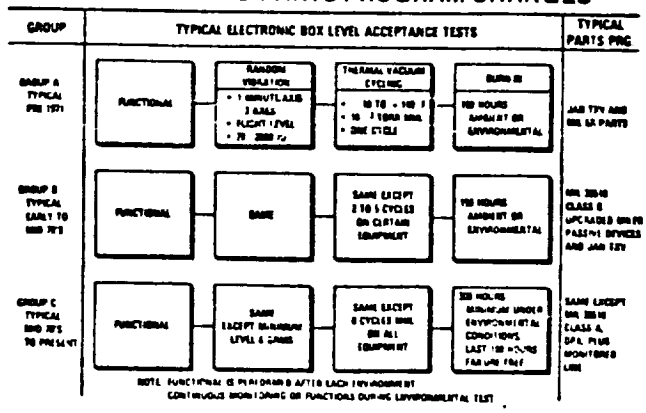


Figure 8

overlap due to the phase-in of the new testing techniques on different programs. The techniques are also not exactly the same on each program. However, in gross terms, there are reasonable separations or breakdowns as related to the significant test program changes. All of the spacecraft were subjected to system level testing (typically shown in Figure 9) which is essentially that in MIL-STD-1540.

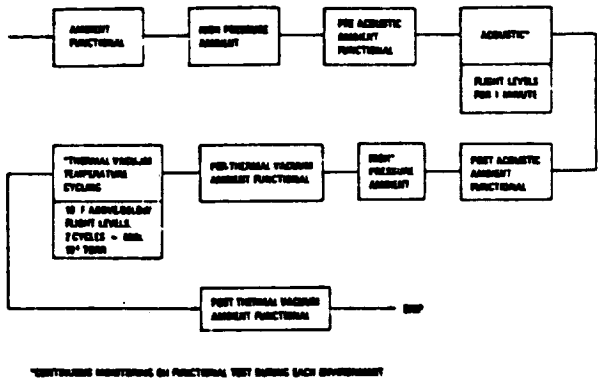
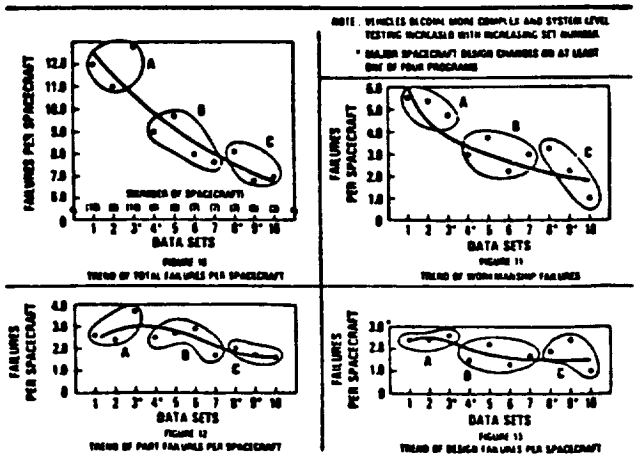


Figure 9

Figures 10, 11, 12, and 13 include Systems Test data from several major LMSC Spacecraft and Second Stage Booster test programs. Figure 11 shows the decreasing failures in system test on these programs as a function of these major test groupings. Figures 11, 12, and 13 give the failure history broken down by types of

SYSTEM TEST FAILURES PER SPACECRAFT



verified failures. The system level workmanship and part failures which were attacked by the improved test program showed major improvements in the frequency of failures. The improvement is even more significant when it is recognized that the system testing has significantly expanded in later years and that the average vehicle in group B is roughly 10 percent more complex in terms of electrical parts count than in group A, while group C is roughly one-third more complex than those in group A. Both these factors result in more opportunity for failures.

This raises the question of module level testing (that is several boxes mounted together into an assembly or subassembly). We are fortunate to have data available on one program which had module level testing over the entire span on Figure 10 covering groups A, B, and C. This data is summarized in Figure 14. Each of the

MODULE TEST HISTORY

NO. OF MODULES	TYPICAL PARTS PROGRAM	BOX TESTS	MODULE TESTS	MODULE FAILURES	SYSTEMS TEST FAILURES PER SPACECRAFT
24 GROUP A	JAN-TX. TXV & MIL-ER	VIBRATION. ONE MIN/AXIS. ONE TV CYCLE – 10 TO 140°F. 100 HOURS MINIMUM ROOM AMBIENT	ACOUSTIC. ONE TEMP CYCLE 0 TO 140°F	110	14
24 GROUP B	UPGRADED JAN TXV. MIL-ER AND MIL 38510 CLASS B	VIBRATION. 3 MINUTE/AXIS. ONE TV CYCLE – 10 TO + 140°F. 150 HRS./MIN AT 130°F	SAME EXCEPT: 4 TEMP CYCLES ~5 DAYS	57	9.3
23 GROUP C	SAME EXCEPT MONITORED LINE & ADDED SCREENING	SAME EXCEPT 2 TV CYCLES PRECEDES VIBRATION. 5 ADDITIONAL TEMP CYCLES (7 TOTAL TV OR T CYCLES). 500 HRS./ MINIMUM ALL ENVIRONMENTS	SAME EXCEPT 6 TEMP CYCLES ~ 10 DAYS	25	6.25

Figure 14

groupings of modules are for the same program but a different spacecraft group (with major design changes between each group). An examination of the data reveals no major "learning curve," certainly within groups B and C. The number of

failures at the system level for each group, for example, are spread fairly evenly across the spacecraft in that group. The data shows a significant decrease in module level and spacecraft level failures with an improved parts program and box level test. The data in Figure 3 shows transponders that were assembled into module tests after receiving only 1 and 2 temperature cycles versus those that went directly to Systems test after an additional 10 or 24 temperature cycles. A comparison of this limited data indicates it is more effective to do a comprehensive test at the box level than at the module. Note, however, that there were fewer temperature cycles and a narrower temperature range at the module level (the latter being due to limitations of certain equipment in the modules).

There are several additional key items that need to be addressed including more details of the flight results and what would have happened if the tests had not been conducted. A partial answer was found in a review of system test data from an LMSC study of 49 spacecraft over a 12-year period (Reference 5). The study included many of the spacecraft of Figure 1 and most of those in Figure 10. The spacecraft were typically subjected to the system level tests in Figure 9. The study included data on what the spacecraft flight results would have been if the system level environmental testing had been eliminated.

This was accomplished by preparing Figure 15 type charts for each of the 49 spacecraft analyzed. These became the basis of analysis for determining the actual flight and ground operating time in days. The system acceptance test operating time spans as shown in Figure 15 were obtained from the systems test

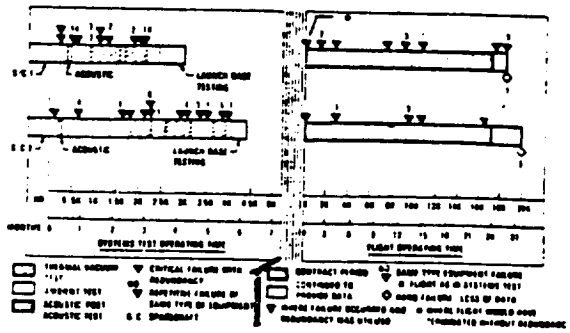


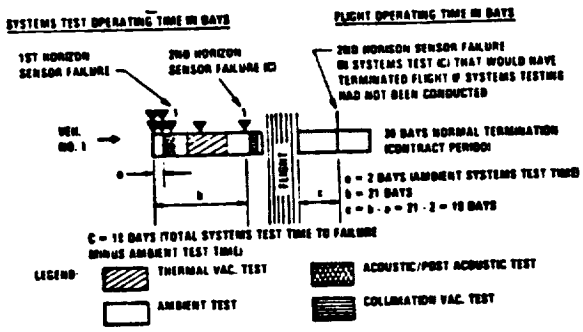
Figure 15

SYSTEMS ACCEPTANCE TEST AND FLIGHT FAILURE DATA

data available for each spacecraft, i.e., test operating records, log books, systems test summary reports, test conductor records, etc. The time spans shown represented as closely as possible the systems test power-on hours for the completely assembled spacecraft. Not included were the box and/or module level acceptance or troubleshooting hours. The flight operating time spans for the vehicles analyzed were also shown in Figure 15 to the right of the systems acceptance test operating time. The shaded areas represent the contract period. The space flight operating time in hours, months, and days were shown. The flight data was acquired from spacecraft flight summary reports or equivalent. The types of environmental tests were coded on each chart and were identified in the Legend; i.e., Ambient, Test, Acoustic/Post Acoustic Test, Thermal Vacuum Test, etc. Critical equipment flight and system test functional failures for each vehicle were shown in the order and at the time in which they occurred during system acceptance testing. Included were those failures which would have seriously degraded the mission or which would have resulted in mission failure if not backed up by a redundant

unit and the Legend included the nomenclature which identified the type of failures appearing on the charts. Also included in the study were cases where orbital workarounds were accomplished and where failures occurred but equipment operated in a degraded mode.

The charts then summarized the conditions which actually occurred. For the case of no environmental tests, all of the time accumulated after the initial ambient functionals was assumed to have occurred in flight. This was accomplished as shown in Figure 16. Failures occurring during this period were also assumed to have occurred at the same time in flight as they had occurred in the system level environmental test.



METHOD OF COUNTING SPACE FLIGHT OPERATING TIME IN DAYS

Figure 16

Figure 16 shows that both horizon sensor failures 1 and 2 occurred after the initial ambient functionals noted by (a). The time to the second failure is (b). The time (b) -(a) is 19 days and indicates that both primary and redundant units would have failed in flight prior to achieving the 30-day orbital contract life period. This vehicle would then be classified as a failure. The results of all 49 spacecraft are summarized in Figure 17.

$$\frac{\text{Orbit time without systems environmental acceptance tests}}{\text{Orbit time with systems environmental acceptance tests (actual}} = \frac{3773 \text{ days}}{6929 \text{ days}} = 55\%$$

$$\text{Actual system acceptance test effectiveness} = \frac{312 \text{ System Test Failures}}{76 \text{ Orbital Failures} + 312} = 80\%$$

$$\text{Systems Acceptance test effectiveness without environmental tests} = \frac{106 \text{ System Test Failures}}{282 \text{ Orbital Failures} + 106} = 22\%$$

SYSTEMS ENVIRONMENTAL TEST EFFECTIVENESS

Figure 17

If the acceptance tests for the system level environment were deleted, the total number of orbital failures would have increased from 76 to 282 (system test failures would decrease to 106). There was actually one failure of a spacecraft in flight prior to contract life. If the spacecraft environmental tests had been deleted, 17 additional, or a total of 18 spacecraft would have failed prior to the required contract life. Note that the actual improvement in orbital life is even greater than that shown since some of the spacecraft were and still are in operation. NOTE: Reference 6, an update of certain parts of Reference 5 covering 67 spacecraft, indicates that the effect of system level acceptance is even more pronounced.

Since the systems environmental acceptance tests are conducted at worst case flight levels, the tests are somewhat more severe than typical or average orbital conditions. Even taking this into consideration, the data indicates that the system level testing as performed is cost effective. Reference 5 estimated that on the average, if systems level environmental tests were eliminated, a spacecraft would have to be delivered at slightly over 50 percent of its current cost to achieve the same cost effectiveness (considering the cost of the satellite only).

A further examination of the acceptance data at box test, system test, and flight reveals a typical test effectiveness of 70 to 80 percent at the box level, 70 to 90 percent at the systems test level, with an overall box/system test effectiveness typically of 96 to 98 percent. This indicates that you cannot eliminate all of the potential failures when stressing only the box or only the system. A comprehensive program is required at both levels.

CONCLUSIONS

The LMSC test history on several major programs has been reviewed indicating that a comprehensive test program on parts, boxes, and systems is required to effectively reduce flight failures. In addition, effective early test programs have a positive effect on maintaining spacecraft schedules. Preserving schedules will become even more critical now that the Space Shuttle has progressed to operational status. A good parts program, including monitoring, and a comprehensive box test program is effective in eliminating system and flight problems.

An effective program of failure analysis and corrective action is an essential and integral part of a successful test program. Three hundred and twelve potential space flight failures occurred of which 129 were of redundant hardware (failures of the same type hardware). These failures would have had significant detrimental effects on the missions if they had not been detected during systems acceptance testing and eliminated through effective corrective action. None of the causes of the 312 failures repeated in space flight.

REFERENCES

1. P. F. Brown, Evaluation of Spacecraft System Acceptance Testing, Fourth Aerospace Testing Seminar Proceedings, March 1978.

2. D. C. Hill, Equipment Level Temperature Cycling, LMSC Space Systems Division Reliability Bulletin 12, 7 July 1971.

3. P. F. Brown, Analysis of Transponder Testing, LMSC Report D327838 dated May 14, 1975.

4. Volume IV, Long Life Assurance Study for Manned Spacecraft Long Life Hardware, MCR-72-169, Contract NAS 9-12359, dated September 1972.

5. D. A. Smith, Systems Environmental Testing and Redundancy versus Shuttle On Orbit Repair/Satellite Retrieval, 6th Aerospace Testing Seminar Proceedings, March 1981.

6. V. S. Bollman and R. Blockinger, Spacecraft Redundancy and Environmental Tests, An Historic Evaluation, Satellite Services Workshop, NASA JSC, June 1982.

QUALITATIVE AIRLINK SATELLITE TESTING

Neil K. Shirk, General Electric - Space Division, Valley Forge, PA

Abstract

Traditionally, satellite factory testing hasl been performed via direct connection of the transmitter output RF power to the test equipment, thereby disrupting the satellite configuration between the TWTA's (Traveling Wave Tube Amplifiers) and the transmit antennas. In the past, some members of industry have used airlink testing for baseline information (not qualitative information) just prior to launch. This paper will describe the qualitative airlink measurement techniques used in performing electrical system tests, RF compatibility, and intermodulation products on a communication system using X-Band, S-Band, and UHF frequencies to meet stringent subsystem and system specifications. This improved test technique has allowed us to maintain satellite hardware integrity through environmental testing (acoustic vibration, shock, thermal vacuum, etc.) without disturbing the flight configuration.

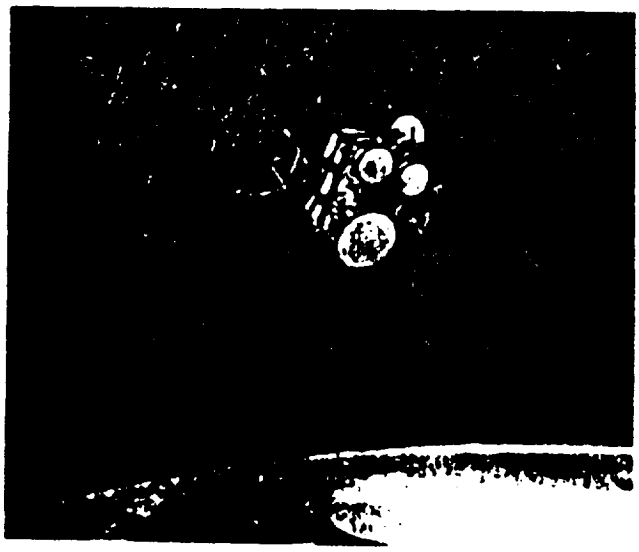


FIGURE #1

MISSION DESCRIPTION

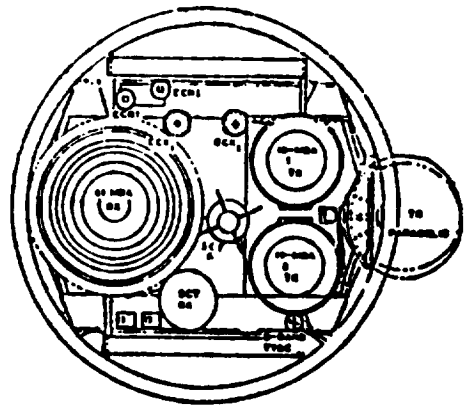
The goal of the DSCS III program is to provide a network of four operational communication satellites and two on-orbit spares in synchronous orbit. Each satellite is designed to have a life expectancy of ten years and the capability of being launched from a Titan booster or the Space Shuttle.

The six channel communications payload has anti-jam capability and nuclear survivability provisions The primary system utilizes SHF (X-Band) frequencies to serve the Department of Defense World Wide Military Command and Control Communications System (WWMCCS), the ground mobile forces, the Defense Communications System (DCS), the White House Communications Agency, and the Diplomatic Telecommunication System. A secondary communications payload is an Air Force Satellite Communications Single Channel Transponder which provides service to Air Force elements.

I. SATELLITE DESCRIPTION

The Defense Satellite Communications System (DSCS III) is a third generation communication satellite designed and built by the General Electric Company for use by the Department of Defense. The satellite R F payload is capable of operation at UHF, S-band and X-band frequencies. There are twelve antennas on the spacecraft; one UHF receive, one UHF transmit, two S-band omni-directional, two X-band earth coverage receive, one 61 element X-band multi-beam receive, two X-band earth coverage transmit, two 19 element X-band multi-beam transmit and one X-band gimballed high gain transmit reflector.

The frequency spectrum covered by the communications payload is 7900 to 8400 MHZ for X-band receive and 7250 to 7750 MHZ for X-band transmit. This spectrum is divided into 6 channels with transmit power provided by two 40 watt and four 10 watt Traveling Wave Tube Amplifiers. Two standard SGLS channels are allocated for S-band TT&C control. Also, onboard is a single channel UHF transponder which communicates with all AFSAT Comm equipped force elements.



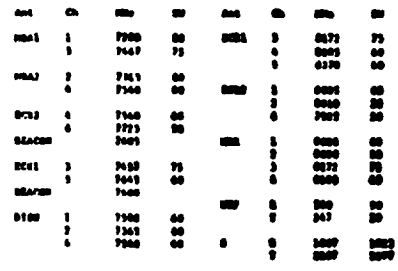


Figure #2

SATELLITE RF TESTING

The primary focus of the RF testing is the X-band communications transponder payload. It is for this subsystem that the "Antenna Coupling Equipment" (HAT) was designed, built, and implemented as a test facility. All qualitative performance tests of the communications subsystem are performed under the "HAT." Communication Satellite measurements taken include: channel gain, gain flatness, gain slope, power output, power transfer characteristics, gain match between redundant elements, spurious phase modulation, noise power, total phase shift (AM to PM), phase linearity, small signal suppression, and active and passive intermodulation products. RF compatibility has also been successfully performed under the "HAT." All performance test data is compared to data taken at the subsystem level, as well as to previous satellite data to verify satellite performance before and after environmental testing and to identify any possible degradation within the satellite.

The spacecraft performance specifications are sufficiently stringent to warrant an elaborate "HAT"/spacecraft alignment and characterization technique. Since the spacecraft is removed from the "HAT" for performance of environmental testing (acoustic, thermal vacuum, etc.), repeatability of the "HAT"/spacecraft alignment is critical. The spacecraft is repositioned under the "HAT" with a tolerance of ± 0.15 inches in height and ± 0.30 inches laterally from its original alignment. The spacecraft and its associated support fixture are placed on a special ball bearing dolly located under the "HAT" to facilitate this alignment.

The "HAT" consists of a box fourteen feet long, ten feet wide and six feet high open only at the bottom. (See Figure #3 - Satellite Under "HAT"). The box is made of 1 3/4 inch plywood and is supported by a steel structure on the exterior. The interior of the box is lined with aluminum sheeting 0.020 inches thick. Over this aluminum sheet is a layer of RF absorber material to provide a "quiet" atmosphere for testing. Imbedded in this absorber are the test equipment transmit and receive elements (See Figure #4). This construction provides approximately 50 dB of isolation from external sources, and reduces reflections from the spacecraft by about 100 dB.

TESTING UNDER THE HAT

The first step of performing spacecraft tests under the "HAT" is to characterize the "HAT" and associated signal routing equipment. The characterization consists of measuring the insertion loss from the test equipment through the antennas in the spacecraft to the input of the low noise amplifier in the spacecraft for the uplink paths. See Figure #5. The downlink paths are similarly measured through the spacecraft antennas through the "HAT" and into the signal routing equipment. A sixth order polynomial is fitted to this data on a per channel basis. The coefficients of these polynomials are stored in the HP1000 computer for use during qualitative performance testing. Offsets are entered into the loss polynomial to correct absolute power measurements to power of within ± 0.1 dB of those taken during subsystem level testing. An initial set of measurements of the saturated power output and of the 3 dB backoff power point are made to give an indication of the accuracy of the "HAT" characterization; and also serve as a gross check of the health of the spacecraft.

At this point, actual Electrical System RF performance testing may begin (See Block Diagram in Figure #5 for test set-up). All amplitude measurements are made using an HP 8566A spectrum analyzer. However, the measurement system reference is an HP8484A

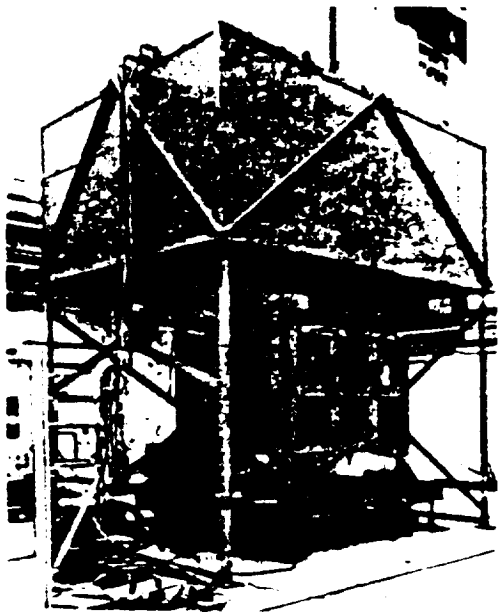


Figure #3

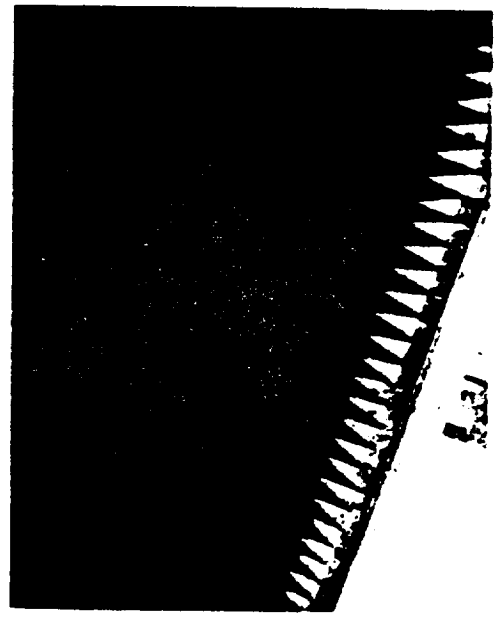


Figure #4

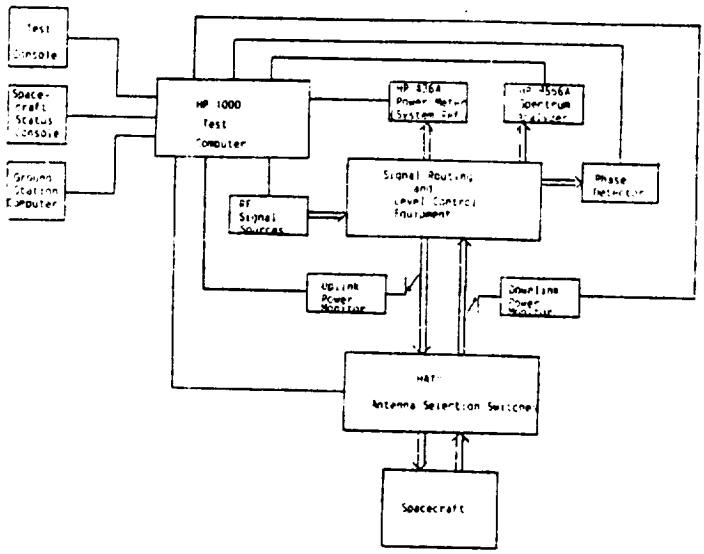
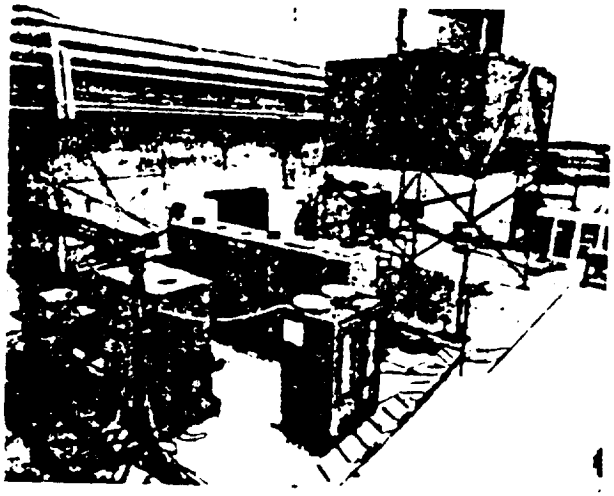


Figure #5

high sensitivity power sensor connected to an HP436A power meter. The power sensor is used as the system reference since it has a much greater temperature and time stability than does the spectrum analyzer. The narrow bandwidth and large measurement range of the spectrum analyzer, however, makes it ideal for performing the actual measurement. Since the power meter is the system reference, an elaborate analyzer to power meter calibration must be performed. This calibration measures things such as IF Gain, preselector peaking, IF bandwidth, dynamic range, and frequency. All of these factors are applied to every measurement taken with the spectrum analyzer to obtain an equivalent power meter reading. A spectrum analyzer calibration check is run automatically



by the computer at the beginning of each automated test. If this spot check shows that the calibration has drifted by more than ± 0.15 dB from the last calibration, the currently executing test program is aborted and a complete calibration must be performed before testing may resume. A side benefit of this calibration check has been to identify faulty or failing test equipment. When all of the equipment is functioning properly, a spectrum analyzer calibration will typically hold for eight hours, however, experience has shown that it is best to run through a complete spectrum analyzer calibration every four hours.

TEST RESULTS:

The data that has been taken with the spacecraft under the "HAT" has exceeded the design requirements as well as the users' expectations. The variation between comparable amplitude measurements is less than the design goal of 0.5 dB, and the variation seen in phase measurements is well within the allowable tolerances. The error contribution associated with the alignment of the spacecraft under the "HAT" has been shown to be less than 0.2 dB. Repeatability over the complete factory test period (approximately nine months) is within the 0.5 dB tolerance, this includes measurement equipment error and spacecraft to "HAT" alignment errors.

LESSONS LEARNED:

Even though the "HAT" was designed to isolate the spacecraft from external signals, invariably, some do get to the spacecraft. In all cases these spurious signals, that have appeared in the data, were identified as coming from some external source (other spacecraft under test, faulty or leaky test equipment, etc.), none were identified to come from the spacecraft. The side panels of the "HAT" reduce this type of interference and play an

important part in obtained repeatable RF measurements. The errors associated with the sixth-order polynomial curve fit of the path loss from the test equipment to the spacecraft have been reduced substantially by minimizing the amount of coaxial cable in the path. Presently only about fifteen percent of the entire path length consists of coaxial cable and coaxial switches.

The errors associated with the alignment of the spacecraft under the "HAT" are not as severe as was originally expected. As a result, the current alignment specifications are considerably more lenient than the original specifications.

Experience has shown that nearly all measurement anomalies have been a direct result of test equipment failures of one sort or another, e.g., waveguide switches that do not switch completely, coaxial switches that do not activate, resistive contacts on coaxial switches, intermittent mechanical drivers for waveguide vane attenuators, etc. To detect these various test equipment "quirks", couplers have been added at various points along the signal path between the test equipment and spacecraft. These couplers are monitored by the test computer at various points during the test program to provide a health check on the test equipment.

Perhaps the most insidious problem, was the inherent or apparent drift of the measurement equipment, most specifically of the spectrum analyzer and to a lesser extent the frequency synthesizers used as signal sources.

The analyzer measurements showed a measurement variation of 0.2 dB to 0.4 dB depending on the setting of the preselector filter control digital to analog converter in the spectrum analyzer, and the last frequency to which the analyzer was tuned. This was corrected by building a Table stored in the computer, of preselector settings during each spectrum analyzer to power meter calibration. This Table is referenced each time an analyzer measurement is made and the analyzer is commanded to the appropriate setting. Also, after each measurement is taken, the analyzer is retuned to 7945 MHz. This frequency was determined by trial and error to provide the greatest amount of measurement stability over the frequency test range.

Another subtle problem was related to the frequency synthesizers. Typically, the first several measurements taken if the synthesizer had not been used for a while would show a variation in power level, this usually would occur during the analyzer to power meter calibration. The solution to this was to leave the synthesizer turned-on at a very low power level (-110 dBm) and tuned to a frequency outside of the DSCS band.

LOOKING TO THE FUTURE

Currently, General Electric is planning to perform mechanical, as well as electrical and RF testing, under the "HAT." This will enable the maximum amount of parallel testing possible, and will also decrease the number of times the spacecraft must be moved. The spacecraft will then have only four test locations, under the "HAT", Acoustic Chamber, Thermal Vacuum Chamber, and Mass Property Test Area.

Also currently under development are means to more fully automate testing at all levels for all subsystems.

CONCLUSIONS:

On very complex spacecraft, such as DSCS III, airlink RF testing has been found to be advantageous for several reasons. From a reliability viewpoint the fewer times you are required to remove flight waveguide, coax cables, etc. to connect test equipment, the better your reliability. With the use of the Airlink RF testing "HAT", no flight/test equipment connections are required during the factory test cycle. Two additional "HATS" are used by the General Electric Company in spacecraft testing. One "HAT" is used for baseline testing at the conclusion of factory testing and is also used at the Cape to verify no RF degradation during shipment of the Spacecraft to the Launch Site. The Third "HAT" is used inside the 54 foot Thermal Vacuum Facility to transmit/receive RF signals from the spacecraft during Thermal Vacuum testing. Therefore, through the use of the three "HATS" for qualitative testing in ambient, testing in Thermal Vacuum, and Baseline Testing, no disassembly is required to implement RF test equipment set-ups and the highest spacecraft reliability possible is retained.

From a cost analysis consideration, time has been saved in the manufacturing cycle since the time involved in aligning the spacecraft under the "HAT" is only a few hours compared to days to remove waveguide between the six transponders and the twelve antennas and tie in the test waveguide for each of the two electrical system tests, RF compatibility Test, three intermodulation compatibility tests, Thermal Vacuum Test, and two Baseline Tests. The half million dollar investment for the three "HATS" has been determined to be very cost effective. With this overwhelming success achieved through RF Technical repeatibility by the characterization methods indicated previously, General Electric is committed to the highest reliability and cost effective system testing which includes the qualitative airlink satellite testing via the "HAT."

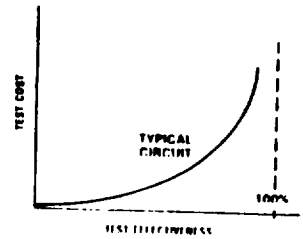
TESTABILITY

TECHNICAL OVERVIEW

BILL KEINER

NAVAL SURFACE WEAPONS CENTER

TEST DEVELOPMENT COSTS



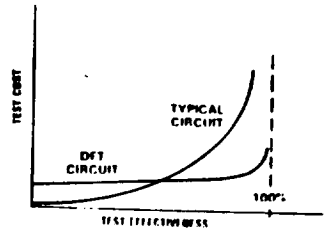
TESTING OF NAVY ELECTRONIC SYSTEMS

EXPERIENCING POOR TEST EFFECTIVENESS
- UNDETECTED FAILURES
- LARGE AMBIGUITY GROUPS
- HIGH FALSE ALARM RATES

INCURRING HIGH TEST COSTS
- TEST EQUIPMENT
- TEST PROGRAMS
- TRAINING

IMPACTING OPERATIONAL READINESS

TEST DEVELOPMENT COSTS



DESIGN FOR TESTABILITY

- ENHANCED PARTITIONING
  - PHYSICAL
  - ELECTRICAL
  - FUNCTIONAL
- INCREASED TEST CONTROL
- INCREASED TEST ACCESS
- IMPROVED BIT
- DECREASED COSTS
  - DEVELOPMENT
  - SUPPORT

TESTABILITY BENEFITS

SYSTEM DEVELOPMENT

REDUCED TPS DEVELOPMENT TIME AND COST

REDUCED BIT DEVELOPMENT TIME AND COST

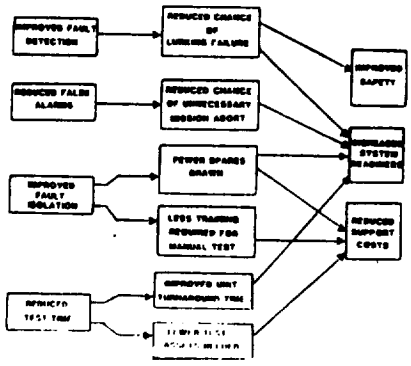
REDUCED FACTORY TEST TIME AND COST

OPERATION AND SUPPORT

HIGH QUALITY MAINTENANCE TEST CAPABILITY

HIGH QUALITY OPERATIONAL MONITORING CAPABILITY

IMPACT OF IMPROVED TEST EFFECTIVENESS



TESTABILITY DEFINITION

A DESIGN CHARACTERISTIC WHICH ALLOWS THE STATUS OF A UNIT AND THE LOCATION OF ANY FAULTS WITHIN THE UNIT TO BE CONFIDENTLY DETERMINED IN A TIMELY FASHION.

STATUS :	UNIT :
OPERABLE	DEVICE
INOPERABLE	BOARD
DEGRADED	EQUIPMENT
	SYSTEM

THE TRENDS

ELECTRONIC SYSTEMS ARE BECOMING:

- MORE COMPLEX (HIGH SPEED, HIGH DENSITY CIRCUITS)
- MORE CRITICAL TO MISSION SUCCESS

TESTING MUST BECOME:

- MORE AUTOMATED
- MORE COMPREHENSIVE
- LESS EXPENSIVE PER FUNCTION TESTED

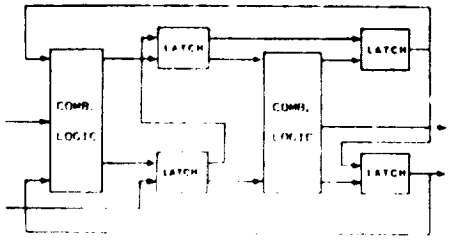
THE LSI REVOLUTION

- COMPONENTS, SUCH AS LARGE SCALE INTEGRATED CIRCUITS, ARE BECOMING MORE CAPABLE AND COMPLEX
- SYSTEM DESIGN IS DICTATED BY THE LSI COMPONENTS WHICH ARE AVAILABLE
- A SYSTEM MAY BE MADE TESTABLE ONLY IF ITS INDIVIDUAL COMPONENTS ARE TESTABLE

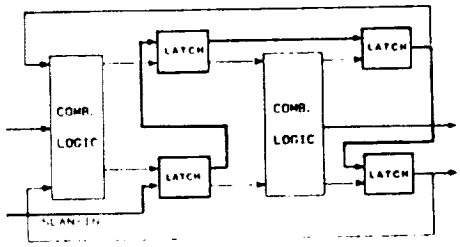
APPLYING TESTABILITY TO CUSTOM LSI CIRCUITS

- IBM EXAMPLE (LEVEL SENSITIVE SCAN DESIGN)
  - IBM CONCLUDED THAT AD HOC LSI DESIGNS COULD NOT BE ADEQUATELY TESTED:
    - HIGH TEST GENERATION COSTS
    - POOR TEST COVERAGE
    - DESIGN ENGINEER WORKLOAD
  - DEPOSED STRICT DESIGN CONSTRAINTS ON CHIP AND BOARD DESIGNERS (LSSD)
  - LSSD ENSURES CHIP AND BOARD IS 100% TESTABLE
  - TEST GENERATION COSTS CUT GREATLY
  - LSSD PENALTY 4 TO 20% CHIP AREA
    1 TO 4 PINS

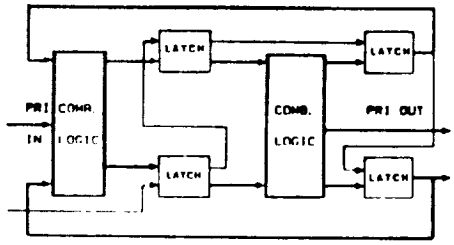
LSSD TEST PROCESS
(EXAMPLE)



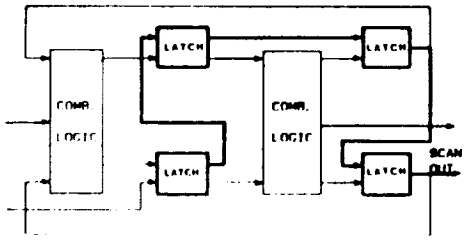
SERIALLY SHIFT TEST PATTERN
INTO SRL's USING SCAN-IN INPUT



TEST PATTERN PROPAGATION
THROUGH COMBINATIONAL LOGIC
OUTPUTS ARE CLOCKED INTO SRL's



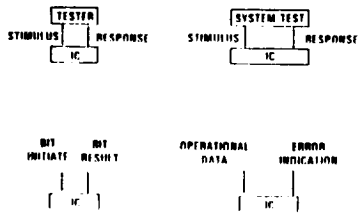
SERIALLY SHIFT TEST RESPONSE
OUT OF SRL's
USING SCAN-OUT OUTPUT



NEW TECHNOLOGY EXAMPLE

- VERY HIGH SPEED INTEGRATED CIRCUIT (VHSIC) PROGRAM
  - 100,000 EQUIVALENT GATES ON A SINGLE CHIP
  - 100 MHz CLOCK RATES
  - EXTENSIVE USE OF ON CHIP BUILT IN TEST

TESTING ENVIRONMENTS



AUTOMATIC TEST PROGRAM GENERATION (ATPG) SOFTWARE

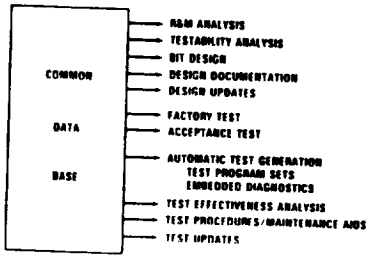
- ACCEPTS MODEL OF SYSTEM/UUT HARDWARE DESCRIPTION
- EXAMINES MODEL STRUCTURE, DEVELOPS CANDIDATE TESTS
- "INSERTS" FAULTS INTO MODEL AND APPLIES TEST
- SIMULATES RESPONSE OF SYSTEM/UUT IN PRESENCE OF FAULTS
- "GRADES" TESTS WRT FAULT DETECTION, FAULT ISOLATION

FAULT TOLERANCE

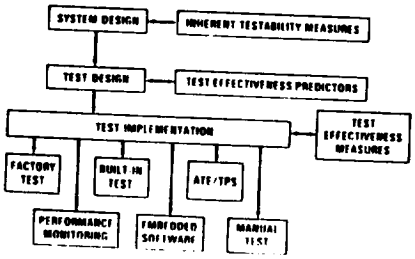
USE ON-CHIP REDUNDANCY ONLY IF IT FITS ON EXISTING CHIP AREA:

OTHERWISE, USE FULL-CHIP REDUNDANCY:

CAD/CAM/CAT INTEGRATION



MEASURES



JOINT LOGISTIC COMMANDERS
PANEL ON AUTOMATIC TESTING
TESTABILITY SUBGROUP

MEASURES

- INHERENT TESTABILITY MEASURES
  - RULE CHECKERS FOR STRUCTURED TESTABILITY
  - CHECK LISTS FOR AD HOC TESTABILITY
  - OBSERVABILITY/CONTROLLABILITY MEASURES
- TEST EFFECTIVENESS PREDICTORS
  - FAULT DETECTION
  - FAULT ISOLATION
  - TEST TIMES
- TEST EFFECTIVENESS MEASURES
  - CND's/RTOK's/FALSE REMOVALS
  - REPAIR TIME
  - SKILL LEVEL

JLC PANEL ON AUTOMATIC TESTING SUBGROUPS

- POLICY AND PROCEDURES
- TPS/SOFTWARE
- TESTABILITY
- NEW TECHNOLOGY
- OFF-LINE ATE
- COMMUNICATION/EDUCATION
- MACHINERY TESTING

JOINT SERVICE ELECTRONIC DESIGN FOR TESTABILITY COURSE

- THREE DAY COURSE
- FOR PRIME EQUIPMENT DESIGNERS AND MANAGERS
- EMPHASIS ON MILITARY NEEDS

TESTABILITY PROGRAM PLAN

- TASKS TO BE ACCOMPLISHED
- PROGRAM SCHEDULE
- RESPONSIBLE CONTRACTOR ORGANIZATIONAL ELEMENT
- ORGANIZATIONAL INTERFACES

SYSTEM ENGINEERING	RELIABILITY	LOGISTICS SUPPORT
DESIGN ENGINEERING	MAINTAINABILITY	TECH MANUALS
TEST ENGINEERING	TRAINING	PRODUCTION
SOFTWARE		

- CONTROL OVER SUBCONTRACTOR PROGRAMS

DFT COURSE OBJECTIVES

- CREATE AWARENESS OF TESTABILITY NEEDS, COSTS AND BENEFITS
- PRESENT WIDELY APPLICABLE TESTABILITY DESIGN TECHNIQUES
- PRESENT MANAGEMENT FRAMEWORK FOR INCORPORATING TESTABILITY WITHIN THE DoD SYSTEM DEVELOPMENT PROCESS
- IDENTIFY PUBLISHED MATERIAL RELEVANT TO TESTABILITY

DESIGN ENGINEER/TEST ENGINEER INTERFACE

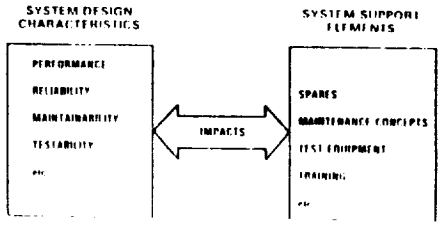
SEVERAL OPTIONS POSSIBLE

- SAME PERSON RESPONSIBLE FOR DESIGN AND TEST
- TEST ENGINEER HAS SIGN-OFF ON DESIGN
- DESIGN ENGINEER CONSTRAINED BY TESTABILITY RULES
- TESTABILITY ENGINEER SERVES AS MIDDLEMAN

TECHNICAL SESSIONS

- DIGITAL ANALOG
  - TESTING
  - TESTABILITY TECHNIQUES
  - BUILT IN TEST
  - TESTABILITY MEASURES
  - SYSTEM EXAMPLES
- COMMON THREADS
  - COMPATIBILITY
  - OBSERVABILITY, CONTROLLABILITY
  - PARTITIONING

SUPPORTABILITY ANALYSIS



TESTABILITY DESIGN PROCESS

MILESTONE	PHASE	
	CONCEPTUAL	· IDENTIFY CRITICAL TESTABILITY PARAMETERS · ESTIMATE WHAT LEVEL OF TESTABILITY IS ACHIEVABLE · IDENTIFY TEST EQUIPMENT TO BE DEVELOPED
1		
	DEMONSTRATION & VALIDATION	· CONDUCT TRADE-OFFS · ESTABLISH TESTABILITY GOALS AND THRESHOLDS
2		
	FULL SCALE DEVELOPMENT	· INCORPORATE TESTABILITY FEATURES · DEVELOP TEST PROGRAM SETS
3		· DEMONSTRATE TESTABILITY
	PRODUCTION & DEPLOYMENT	· FOLLOW-ON TESTABILITY EVALUATION · TESTABILITY GROWTH

ON-ORBIT TEST USING SHUTTLE AND SPACE STATION

By E.H. Medler, TRW Space & Technology Group

ABSTRACT

Performance of certain testing operations in space using the shuttle or the space station appears to be advantageous both for performance and cost benefit. To optimize the benefit, the testing operations must be closely coupled with evolving servicing and maintenance operations in space and accommodations in the orbiter and space station are needed. Standards need to be established to minimize non-recurring design and training costs. Some special concepts for a spacecraft design for assembly and test on-orbit may be applicable. These topics are discussed in exploring the use of the space transportation system (STS) orbiter or a space station as test facilities for on-orbit test.

INTRODUCTION

Advocates exist in substantial numbers stating, "There is a valid argument for certain testing operations to occur in space." Non-advocates, in contrast, point out many arguments discouraging substitution of on-orbit testing for ground testing. This paper explores a few ramifications of the question, "Why the shuttle or a space station as a testing facility in-orbit?" The key topics which are discussed and from which some conclusions are drawn are:

a) Some concepts for testing in space

b) On-orbit servicing - a companion to testing

c) Precursor technology development effort

d) Challenges of using the shuttle or space station as a test facility

WHY TEST ON-ORBIT?

Some of the reasons for testing on orbit as an alternative to ground testing are:

a) Environments not easily duplicated on earth are available, e.g., low g, high vacuum, the real sun

b) Some testing is practical only in space (e.g., large structure dynamics)

c) Final test and checkout is placed closer to the end use point

d) Man's capability is made available for planned and contingency work

e) Testing is companion to and beneficiary of on-orbit servicing technology

There are environments in space which are not easily simulated on the ground. Most obvious is perhaps low gravity. Other environments

include vacuum, the sink of black space, the real sun. Exploiting these environments and controlling exposure to them requires special test design considerations. If, for example, low "g" means 10^{-3}g or less, continuously, then careful control of all motion on the testing platform is a requirement.

Large, flexible structures cannot be fully or adequately tested on the ground. Verification of computer models for large structures with natural frequencies well below one hertz is probably best done in low-g environments. Similarly, verification of control algorithms with large, flexible structures in the control loop benefit from testing in the space environment.

Testing in-orbit, prior to final release from the shuttle or at the space station as a "way station" for a spacecraft enroute to placement in its final orbit, is an opportunity for a last check after exposure to the launch environment of the shuttle or expendable launch vehicle during boost to orbit. Testing at this point allows another go-no-go decision point. While some thrust is toward reducing or protecting space-bound equipment from this environment, the future probably will see many spacecraft still at risk due to this boost environment.

The orbiter already carries a crew which could support on-orbit testing; we also assume the space station will be man-occupied. Therefore, the skills available through man in space are available to support on-orbit testing. On-orbit testing generally, and use of crew skills particularly, couple closely with concepts for on-orbit servicing. Worthwhile use of man in space is already perceived for many planned and contingency operations. Examples are to service and maintain spacecraft and attached payloads; operate laboratories in space; use and maintain orbital transfer vehicles (OTVs) and teleoperator maneuvering systems (TMSs) and attached payloads. Testing on-orbit makes few demands not already planned. Indeed, adding testing on-orbit is just one additional justification of the need for man in space.

In summary then, some examples of tests which might be carried out on-orbit are:

a) Short final spacecraft checkout
b) Dynamic tests - final trimming of control laws, particularly on larger, flexible structures
c) Test bed development testing of large structures
d) Final verification and trimming of thermal subsystems
e) Assembly of modules (subsystem, payloads, deployables) and final integrated testing
f) Propellant system final assembly, fueling, and testing

g) Alignment and alignment checks - thermal distortion tests and adjustments - particularly on larger structures

h) Testing of and actual deployment of appendages and arrays

What are some of the disadvantages cited for testing on-orbit? A few are:

a) Time and manhours in space are expensive

b) Automation and robotics to improve time and man-availability constraints may be costly

c) Use of man in space (e.g., EVA) involves risk

d) It simply duplicates tests which are already done on the ground

Pros and cons not withstanding, we found a substantial interest in the space industry for testing on-orbit. As might be expected this interest was coupled with comparable interest in service and maintenance on-orbit. As part of TRW's contract with NASA Headquarters (Contract NASW-3681 Space Station Needs, Attributes and Architectural Options Study) we surveyed seven U.S. and seven European organizations. These organizations were not those directly under contract with NASA in the parallel space station studies; rather, they were other companies who are spacecraft manufacturers and/or satellite operators. The purpose of the survey was to ascertain what benefit, if any, they perceived from on-orbit testing and on-orbit service and maintenance. Of the nine respondents, all foresaw benefits. Confirming the on-orbit test examples listed above, these spacecraft manufacturers perceived the greatest benefits to be as follows:

a) Test of deployables

b) Install/replace/service ORUs

c) Some actual assembly and test operations

d) Initial and post-service checkout

e) Load propellants

ON-ORBIT SPACECRAFT ASSEMBLY AND TEST

One of the areas which we at TRW have specifically investigated is the benefit of performing some spacecraft assembly and test operations on-orbit i.e., assembly and test of a spacecraft upon its delivery to orbit and prior to its initial on-orbit release. The premises for such an investigation include assumptions that packaging and testing provisions are made to be suitable for space operations during the next decade or so. In other words, a testing operation currently done in a high bay or an environmental test facility cannot be transferred step-by-step without modification to a space environment. Bridging to suit what is feasible in space in the near term is necessary. Some design provisions include:

1) packaging into large, easily re-

placed modules, 2) creation of relatively standard, common physical and electrical interfaces, 3) design in self-test capability at the module level. With provisions such as these the payoff of doing the final assembly and test of a spacecraft after it arrives on orbit may be substantial. In the following paragraphs we will briefly look at some of the potential benefits of doing some final assembly and test on-orbit and a conceptual design for a spacecraft for such assembly and test on-orbit.

Some of the benefits of design such as the concept shown in Figure 1 are:

a) Reduces STS transportation costs via lighter more compact structure and more efficient use of cargo space
b) Deployment of appendages by manned operations reduces the need for automated mechanisms
c) Enables on-orbit fueling
d) Allows spacecraft to be of larger diameter, greater length and/or greater weight than the STS cargo bay handles
e) Design and test of modules for STS environment is not required (modules protected by the special carrier)
f) Thermal subsystem can be trimmed on-orbit, requiring less ground test
g) Spares can be carried on the initial launch, for immediate use if required, or stored on the space station for later use

Figure 1 shows a concept for a spacecraft carried to orbit with assembly incomplete. A "carrier module," designed as flight support equipment (FSE) is a cargo container in which are packed units to be assembled on-orbit to the basic module. The carrier module protects the units from the shuttle environment. (This is analogous to the "pack-to-protect from" concept used in most everyday shipments by common carriers). The basic module is assumed to house the propellant tanks. The tanks may be empty if the further assumption is made that the propellant is available on-orbit for an on-orbit fueling operation. What are some of the benefits of such a packaging and transporting concept as listed in (a) and (d) above? The structure can be lighter weight and more compact; design and testing of components for the boost environment can be reduced and perhaps even eliminated.

Figure 2 shows a prediction of possible economic benefits with some assembly and test on-orbit. The data is illustrative and is selected from a recent cost/benefits analysis. The savings are in 1984 dollars. The time frame for the savings is the 1990 to 2000 decade and a manned space station is assumed. The scenarios are for various space station system configurations and growth options. The mis-

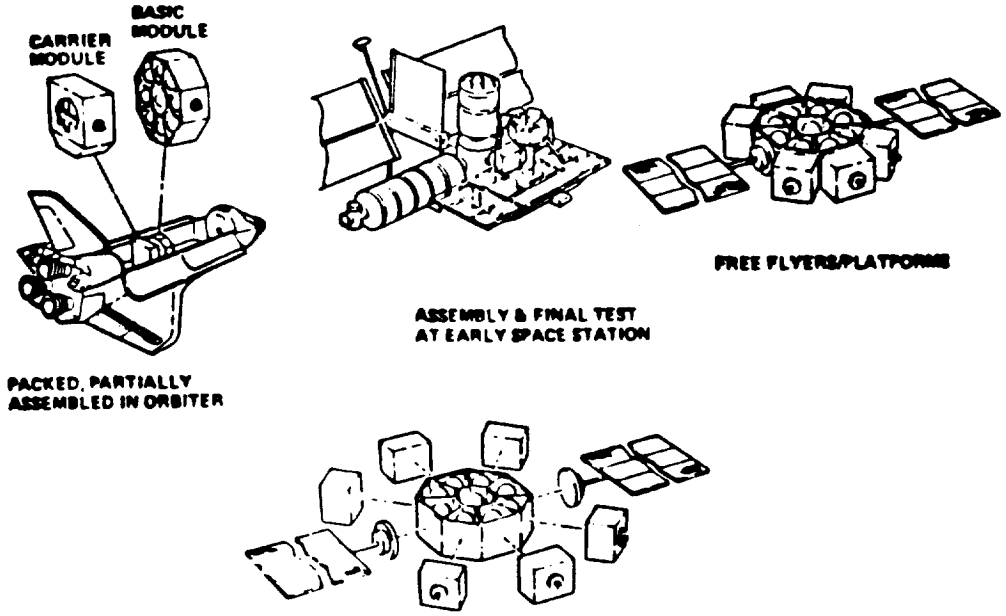


Figure 1. Spacecraft Design for Assembly and Test On-Orbit

	SAVINGS (BENEFIT) - $(M)		
	SCENARIO #2	SCENARIO #3	SCENARIOS 4 & 5
ASSEMBLY AND TEST ON-ORBIT - USING LARGE ORUS AND CONVENTIONAL COMPONENT PACKAGING	490	385	504
STS TRANSPORTATION COST SAVINGS	51	40	52
VALUE ATTRIBUTED TO SPARES CARRIED TO ORBIT AND USED IN INITIAL CHECKOUT	179	141	185
VALUE ATTRIBUTED TO CHANGES IN COMPONENT PACKAGING	221	174	228
TOTAL	941	740	969

- THIS IS AN EXAMPLE FROM SOME TRW STUDIES SHOWING THAT POTENTIAL SAVINGS ARE SIGNIFICANT

Figure 2. Summary - Potential Savings (Benefits) With Some Assembly and Test On-Orbit

sion model is that constructed by TRW for the Space Station Needs, Attributes and Architectural Options Study previously mentioned for the last decade of this century. About 1/3 of the total mission model was assumed to use some standardized assembly and test techniques on-orbit and thus produce the benefits shown. (Offsetting the benefits are, of course, costs ascribed for services provided by the space station; STS transportation savings are based on a flight charge of about $86 M).

PRECURSORS TO TEST OPERATIONS ON-ORBIT

On-orbit servicing and maintenance technology is closely coupled with on-orbit test. In fact, it is a precursor. Provision for planned service and maintenance operations are designed into Space Telescope (ST) and will be part of the design for Gamma Ray Observatory (GRO) and the Advanced X-ray Astrophysics Facility (AXAF). Contingency service is best illustrated by the plan to retrieve and repair the Solar Maximum Mission (SMM) -- see Figure 3. The figure shows the SMM after recovery and placement in the Flight Support System (FSS). Two astronauts in foot restraints (one on the remote manipulator system) prepare to remove and replace a module of the Multimission Modular Spacecraft type. Figure 4 shows a 1/20 scale model of a ser-

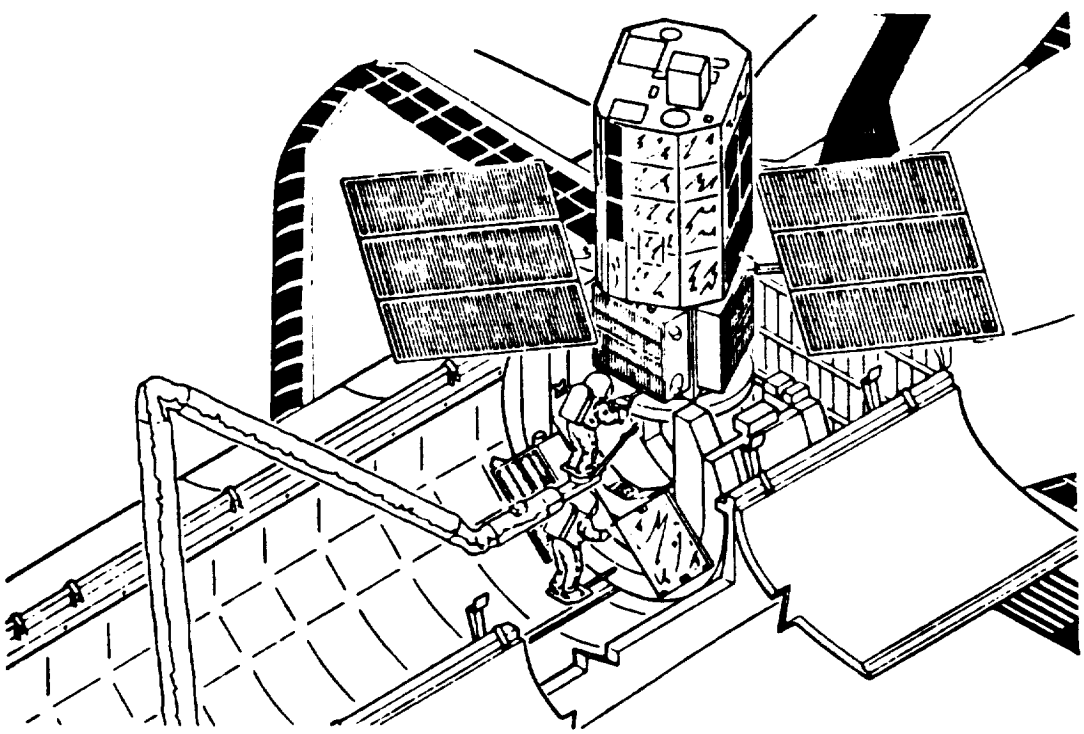


Figure 3. Solar Maximum Repair Mission

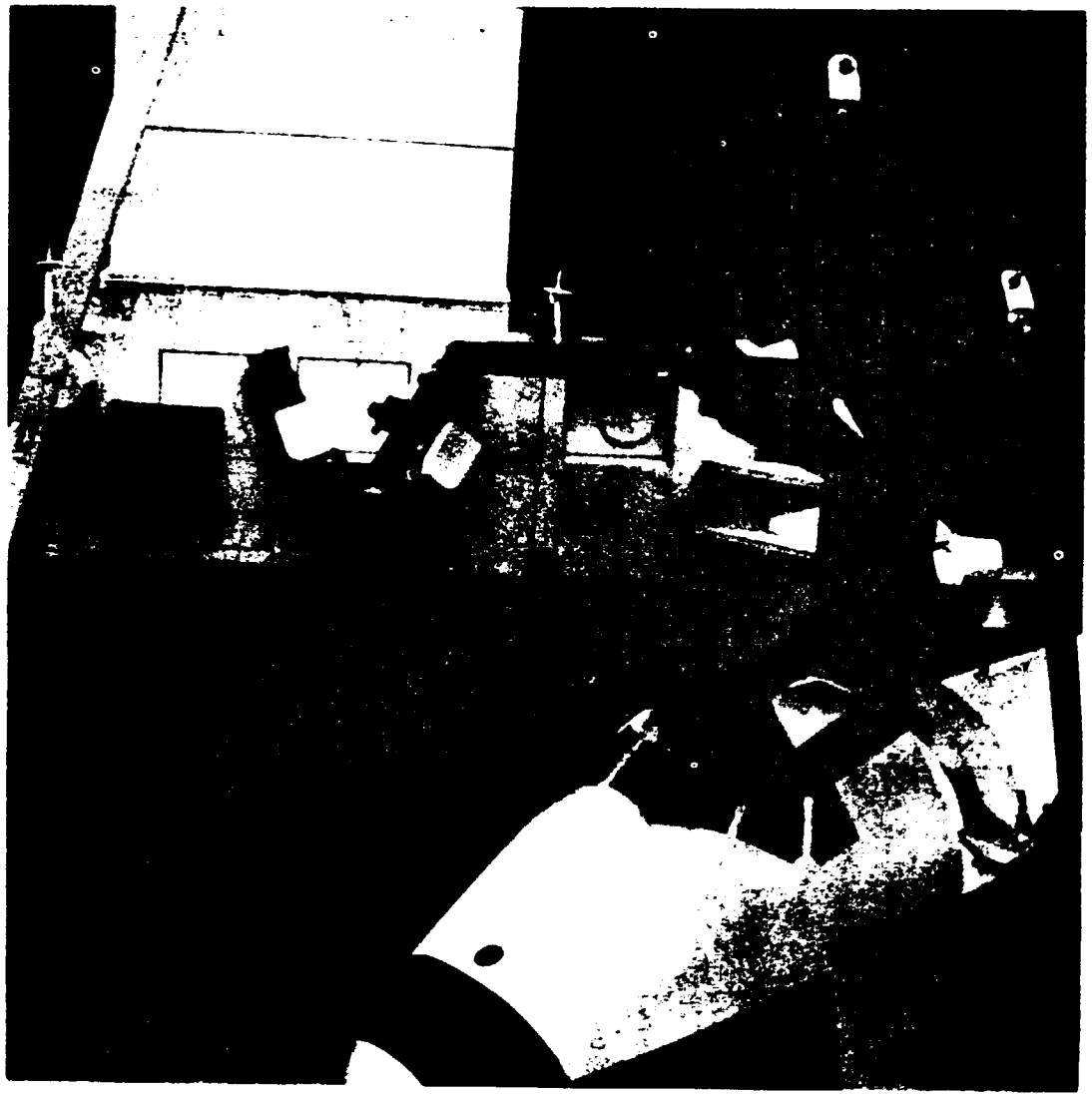


Figure 4. Port Solar Array Jettision or Replacement

vicing operation studied for a Space Platform. All of these programs use a combination of extra vehicular activity (EVA), intervehicular activity (IVA) and ground operations which are directly applicable to other space operations such as test. NASA/MSFC via four interrelated contracts are studying technology development missions which could be used to develop and demonstrate methodology applicable to on-orbit service with much direct applicability to test on-orbit.

Service and maintenance operations on-orbit were just briefly summarized as precursors for on-orbit test operations. Even preceding these are ground test precursors and special development tests using the STS. Some of this advance work is done in ground facilities such as the Neutral Buoyancy Simulator at NASA/MSFC (Figure 5) and the Weightless Environment Training Facility at NASA/JSC. Some of the experiments planned for STS flights are (1) fluid transfer and storage - mid-deck cargo bay experiments for non-cryogens as well as cryogens management experiments, (2) 15 meter antenna experiment on STS and (3) extravehicular activities on many flights. Information system and data handling system test beds - and man-machine interface development and test beds are in the offing.

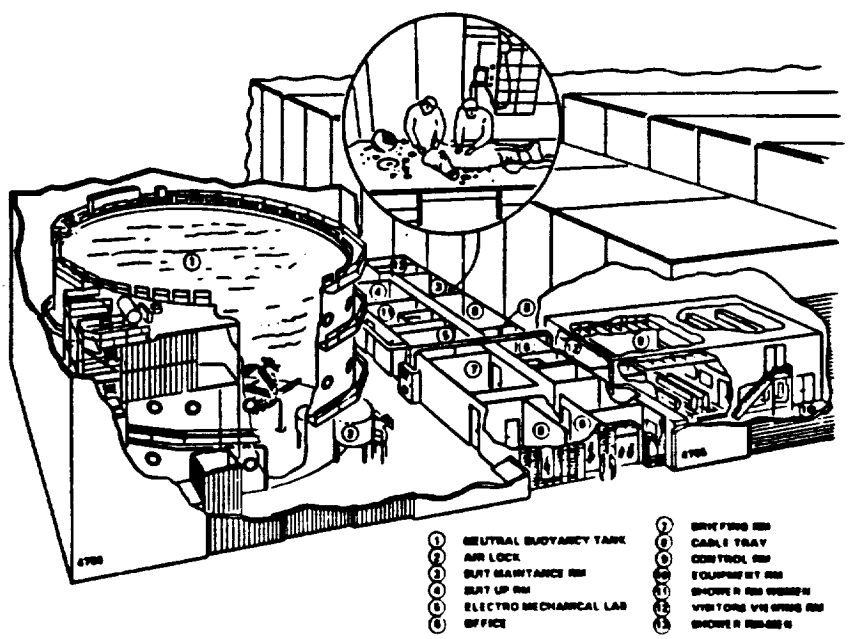


Figure 5. NASA/MSFC Neutral Buoyancy Simulator

One of the current examples of development work involves large, modular orbital replacement units (ORUs) of the type that would be useful for the spacecraft concept of Figure 1 or the large platform concept of Figure 4. This ORU (Figure 6) was recently tested in the Neutral Buoyancy Simulator at NASA/MSFC. Figure 7 shows other devices under test in the facility.

CHALLENGES OF TEST ON-ORBIT

Assuming that test in space is viable, then, some basic steps must be taken to assure its success. Some challenges exists which must be met. These challenges include:

a) Making spacecraft easy to work on in space

b) Standardizing designs and interfaces

c) Developing realistic techniques for minimizing test time (compared to earth)
   - Optimized use of man
   - Automation, fault detection/isolation, built-in test
   - Robotics
   - Man-machine interfaces (MMI)

d) Providing shuttle/space station accommodations
   - Improved space suits, maneuvering units, manipulators
   - Propellant storage and loading
   - External work areas - hangars, platforms
   - Test stations and nominal test equipment
   - Perhaps some shop-like internal (pressurized) capability

e) Evolve meaningful mission assurance methodologies

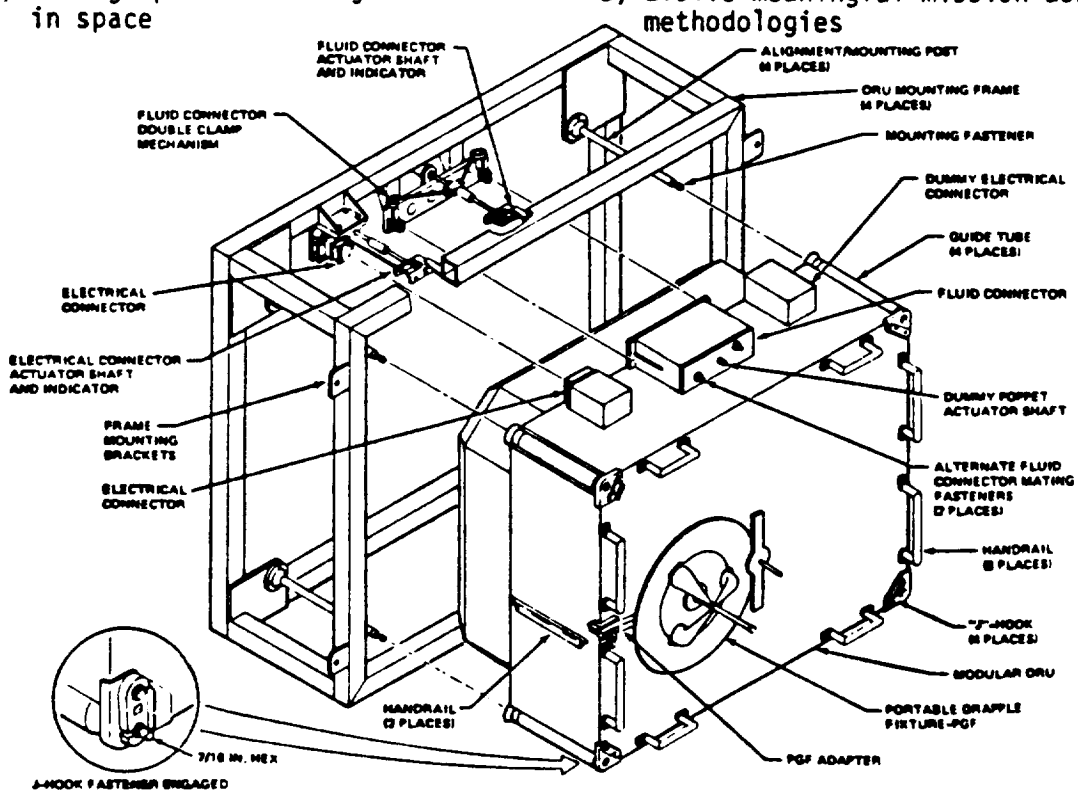


Figure 6. Modular ORU and ORU Mounting Frame

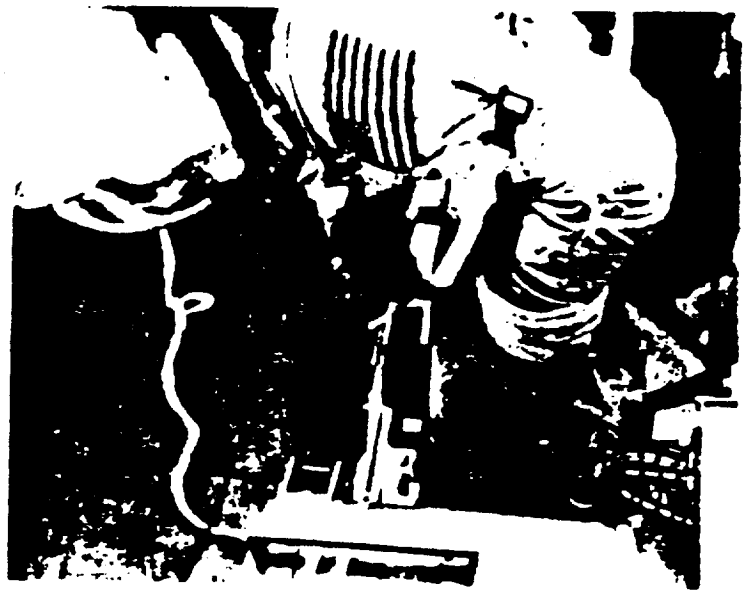


Figure 7. Changeout of Stored Solar Array Assembly in Neutral Buoyancy Simulator

Designs to make things possible to work on in space is not virgin territory. Much study has been done. In space, operations have existed starting with Gemini and through Apollo, Skylab, and into the shuttle era. Extra vehicular operations particularly have produced standards in areas such as tools, foot restraints, maneuvering devices, tethers, handholds, etc. To maximize exploitation of test operations in space, the idea of standardization must be extended to spacecraft in features such as (1) physical interface to the orbiter or space station (2) electrical interface for power, signal, etc. (3) replaceable modules (ORU), (4) safety provisions for proximity operations, and (5) propellant transfer methods and provisions. Standardization in these areas and continuing to use EVA standards will minimize non-recurring engineering and maximize the cost benefits. As robotics, man-machine interface concepts, use of heads up display by astronauts, telepresence come "on line" in space operations they will provide further enhancement. Space stations and shuttle accommodations, as listed in (d) above, must also be made for test operations in orbit. Finally, some thought must be given to assurance methodologies. Obviously, such an activity is still needed, but current ground-based concepts for inspection and monitoring are not applicable.

CONCLUSIONS

Some broad conclusions can be reached regarding use of the shuttle or space station for on-orbit test and include the following:

a) Some needed testing can only be done on-orbit

b) Testing on-orbit is closely coupled with on-orbit servicing for which economic benefit does exist

c) Testing on-orbit potentially has both economic and performance benefit

d) Viability of testing on-orbit needs acceptance so that the required support functions can be put in place

e) Standards are necessary to assure broadest application of on-orbit testing

SUMMARY OF ISSUES AND RECOMMENDATIONS

SUMMARY OF
ISSUES AND RECOMMENDATIONS

1. Test Requirements and Practices

Issue

Requirements for ground testing of space vehicles and their subsystems and components are defined by MIL-STD-1540B. Tailored application of these requirements must reflect the performance history of recent spacecraft and be cost effective.

Recommendations

a) Re-examine and evaluate the perceptiveness of specified test procedures for prototype and production vehicles, their components and subsystems in the light of actual test cost and results.

b) Develop a statistically significant data base which will enable rational decisions to be made regarding requirement tailoring and revisions to test procedures and sequences.

c) Define common categories for failures and discrepancies which will support test effectiveness analysis.

2. Testability

Issue

Increasing spacecraft complexity and short development, fabrication and test schedules require special design provisions to allow rapid and effective testing. Program managers and spacecraft designers must ensure that test considerations are factored into front-end design.

Recommendations

a) Strengthen knowledge of testability techniques for space application.

Industry and SD to schedule the tri-service testability course

NSIA ATC to sponsor local testability workshops and pursue dialog with other professional societies.

b) Determine methods to quantify cost and benefits from specific testability provisions throughout the life cycle of a program.

c) Endorse testability improvements in satellite development and modification.

Include specific testability requirements in prime item specifications and SOWs.

Define standard testability assessment tools.

d) Review the proposed testability standard and determine how it can be best applied to space systems.

3. ATE and Software Standardization

Issue

Lack of an adequate high level satellite test control language continues to impact test cost, schedules, resources, and effectiveness. Satellite complexity and the importance of automated test systems strengthens the need for a standard control language.

Recommendations

a) Continue to support on-going work with the IEEE Atlas Committee and NSIA to finalize System Control and Operations Language (SCOL) requirements; evaluate candidate languages and define a suitable industry standard.

b) Strengthen participation by spacecraft industry and the government in (SCOL) definition.

Suggested OPR

Industry lead, government support

4. ATE and Software Standardization

Issue

The methods and extent of automated test system, test program and procedure verification and certification frequently is not clearly defined. This condition leads to increased cost and schedule delays, or to curtailment of validation/certification activities, with increased risk to mission success.

Recommendations

Review validation and certification methods employed for automated test system, test program and procedures; and the methods used to impose these requirements. Determine and implement methods to clarify these requirements.

Suggested OPR

Space Division

5. ATE and Software Standardization

Issue

The Air Force Modular Automatic Test Equipment (MATE) standardization initiatives may have a potential to improve cost effectiveness of spacecraft automated test system development and support.

Recommendations

Assess the MATE system, identify highest return on investment features and the most suitable method of implementing these features in in-plant test systems.

Suggested OPR

Space Division, industry

6. On-Orbit Test Facilities

Issue

The shuttle orbiter and future space platforms make it possible to conduct a variety of development tests in space instead of in the laboratory or factory. At present, there appears to be a general lack of knowledge among program managers and test engineers regarding planned and current capabilities for conducting experiments and demonstrations in space.

Recommendations

a) Publish current and planned flight experiment/development opportunities (i.e., identify carrier programs including on-orbit services, availability, costs, and schedules).

b) Promote early planning and use of space-flight experiments/demonstrations as integral elements of development phases for verification of analytic modeling, life tests, on-orbit assembly, and optimized design for space environment.

c) Develop general standards to facilitate procedural, interface, and test requirements.

SPECIAL REPORT

JOSEPH S. HEYMAN
NASA LANGLEY RESEARCH CENTER
AND
ENGMIN J. CHERN
COLLEGE OF WILLIAM AND MARY

Joseph S. Heyman[1] and Engmin J. Chern[2]

ULTRASONIC MEASUREMENTS OF AXIAL STRESS

REFERENCE: Heyman, J. S. and Chern, E. J., "Ultrasonic Measurements of Axial Stress," Ultrasonic Measurements of Stress Symposia, ASTM STP , T. Proctor and J. S. Heyman Ed.

ABSTRACT: Several applications of ultrasonic techniques are gaining acceptance for accurate preload determination of critical fasteners. In this presentation one class of techniques based on continuous wave ultrasonics is examined in detail with benefits and limitations discussed as they apply to accurate determination of fastener preload. The theory of acoustic propagation in strained media is reviewed as it applies to bolt geometries. Propagation measurements in these geometries require an understanding of guided-wave effects which can influence stress determination. However, studies of the acoustic spectra of rods identify diffraction and resulting mode conversion as the main artifact for such a measurement. The acoustic spectra, obtained with a novel tone burst spectroscopy method, clearly show mode conversion effects at low frequencies to be absent at high frequencies.

Changes in the acoustic spectra are shown to accompany applied stress and suggest an instrument technique for measuring changes in fastener strain. The method, called a Reflection Oscillator Ultrasonic Spectrometer or ROUS, is discussed with an improved technique called Pseudo Continuous Wave method (PCW) described. Data presented are obtained with the PCW for strain in several engineering materials.

Bolt geometry is shown to have an influence on the stress measurements. In particular, percent fastener length under load is evaluated for its effect on material strain measurements. A computer model predicting a linear dependence on the percent of fastener under load is verified using the PCW technique. Other aspects of bolt geometry are discussed including poor geometry bolts, bolts with transverse holes, and bolt loading anisotropy. Experimental techniques for performing measurements on fasteners with poor geometry are discussed.

KEY WORDS: Ultrasonic measurements of stress, axial stress, residual stress, continuous wave techniques, pulsed phase locked loop system, residual stress and attenuation, phase insensitive transducers.

INTRODUCTION

Threaded fasteners play a significant role in establishing general integrity. Under proper preload, a joint can be designed to represent a problem-free area of a structure. However, for most installations, fastener preload is only known at best to $\pm 20\%$ with many variations beyond an acceptable design limit. Engineers have recognized these problems for many years and have used strain-gaged bolts or oversized fasteners in critical applications. Use of mechanical extensiometers (micrometers, dial indicators, etc.) has increased preload repeatability as well. However, the difficulty and expense of these methods has limited their use to highly critical areas. Of greater concern are the cases where fastener extension cannot be mechanically measured or where the nominal $300 cost of a strain-gaged bolt is impractical.

Several acoustic techniques have been developed which measure fastener strain. The most widely used ultrasonic technique for general nondestructive testing is pulse echo (PE). A special type of PE system has been applied to bolt stress by several sources. In theory, the time domain PE measurement should be equivalent of a frequency domain continuous wave (CW) measurement. A pulse, however, contains a broadband of frequencies all with a different propagation property. The main effect is diffraction, but attenuation effects also play a role. A leading edge time-of-flight measurement must set a trigger threshold which will affect the measurement time as the pulse amplitude changes. Furthermore, it is difficult to measure $\Delta T/T$ to parts in 10^7 (10^{-12} seconds out of 10 microseconds) while it is easy to measure $\Delta F/F$ to parts in 10^7 (1 Hz out of 10 MHz). For these reasons, the CW approach will be the basis of this study.

In this paper, we examine the theory of acoustic propagation in strained media with particular emphasis on rod (bolt) geometries. The theory is developed from the standpoint of a frequency domain analysis to obtain a single relationship between applied stress and normalized frequency shift. Experimental results are presented in the frequency domain which agree with theory when transducer bandwidth and mode conversion effects are included. A continuous wave (CW) instrument is described which can precisely determine changes in applied stress from measurements of frequency shifts of acoustic resonance modes. Several other instruments are discussed which are more practical and able to function in fasteners of poor geometry. Data obtained with these pseudo-CW techniques are presented showing variations in stress acoustic constants in materials and within one material as a function of percent load length. Finally, data are presented suggesting that the variations in ultrasonic attenuation resulting from residual stress may in fact be an artifact of the measurement transducers. The attenuation variation is shown to vanish where a true power transducer is used thus eliminating phase cancellation.

THEORY OF ACOUSTIC PROPAGATION IN STRESSED RODS

We assume that a fastener is a homogeneous rod of length $a/2$ with a velocity of sound v. The rod resonance frequencies are:

$$F_j = jv/a \tag{1}$$

where rod diameter $d \gg \lambda$, the acoustic wavelength, and j a harmonic number. For this condition, we may consider the acoustic propagation as a plane wave and use the material bulk velocity for v. The lineshape of the mechanical resonance is generally Lawrencian and the complex particle displacement can be expressed as [1,2]:

$$A(t) = \frac{1}{a}^{i\omega t} \Big/ \left[1 - \exp - (\alpha a + ika)\right] \tag{2}$$

where α is the attenuation, $\omega = 2\pi f$, and $k = (\omega/v)$. The rate of change of $|A|$ with k is an important lineshape in that at the center of the mechanical resonance, $d|A|/dK = 0$. In general we can write [1,2]:

$$\frac{d|A|}{dk} = \frac{a \exp(\alpha a/2) \sin ka}{2\sqrt{2}\,(\cosh \alpha a - \cos ka)^{3/2}} \tag{3}$$

The lineshapes of $|A|$ and $d|A|/dk$ are shown in Fig. 1 as a function of

[1]Head, Laboratory for Ultrasonics, NASA Langley Research Center, Hampton, VA 23665

[2]Research Associate, NASA Langley Research Center, Hampton, VA 23665

frequency. The change in sign of $d|A|/dk$ at $F = F_j$ has been used to lock ultrasonic spectrometers accurately to F_j [3].

If we now concentrate on the frequency domain when the rod is strained we find a strain derivative [4]:

$$\frac{dF_j}{da} = \frac{j}{a}\frac{dv}{da} - \frac{jv}{a^2} \tag{4}$$

or

$$\frac{\Delta F_j}{F_j} = \left[\frac{1}{v}\frac{dv}{da} - \frac{1}{a}\right]\Delta a \tag{5}$$

for small strains. An alternate elastic range expression can be written for the stress using the relationship:

$$\frac{\Delta a}{a} = \frac{\Delta\sigma}{E} \tag{6}$$

and

$$\frac{dv}{da} = \frac{dv}{d\sigma}\frac{d\sigma}{da} \tag{7}$$

where σ is the stress and E the Young's modulus, resulting in a stress derivative:

$$\frac{\Delta F_j}{F_j} = \left[\frac{1}{v}\frac{dv}{d} - \frac{1}{E}\right]\Delta\sigma \tag{8}$$

Equation (5) can be written in completely normalized form as:

$$\frac{\Delta F_j}{F_j} = \frac{\Delta v}{v} - \frac{\Delta a}{a} \tag{9}$$

and represents an important material parameter directly linked to the thermodynamic Grüneisen parameter [5].

The bracketted expressions in equations (5) and (8) represent the material strain and stress acoustic constants. The value of these constants depends on the acoustic wave type (compressional or shear), propagation direction with respect to the load, and for anisotropic materials also depends on wave polarization [6].

For the simple case of a longitudinal wave propagating in the direction of the applied load we find that $dv/d\sigma$ may be expressed as [7]:

$$\frac{dv}{d\sigma} = \frac{-1}{6\rho\,(\lambda + 2\mu/3)\,v}\left[2l + \lambda + \frac{\lambda + \mu}{\mu}(4m + 4l + 10\mu)\right] \tag{10}$$

where λ and μ are the second order elastic constants of Lamé and l and m are the third order elastic constants of Murnaghan [8], and ρ is the density.

An important feature of these equations (5, 8) is that for small strains, they represent a linear relationship between $\Delta F/F$ and the applied load. The relative magnitude of $\Delta v/v$ to $\Delta a/a$ is nominally 2 to 1 for many of the metals typically used for fasteners. As an example, for mild steel, the ratio of $\Delta v/v$ to $\Delta a/a$ is 2.62 while for aluminum (2024) the ratio is 2.52.

EXPERIMENTAL VERIFICATION

The concepts of the previous section indicate that the acoustic response of a threaded fastener should appear as a series of mechanical resonances. An acoustic technique based on the tone burst spectroscopy (TBS) method [9] was used to obtain the frequency domain response of the fastener. The basic block diagram of the TBS system is shown in Fig. 2. The output of an oscillator locked in frequency to the center frequency of a spectrum analyzer's bandwidth is gated onto a transducer. Once equilibrium

conditions are established in the transducer/fastener composite resonator, the output of the oscillator is gated off and a receiver gated on. The output of the receiver enters the spectrum analyzer which produces the desired frequency domain response of the transducer/fastener system.

A typical response shown in Fig. 3 covers the frequency range from 3.5-7.5 MHz and shows about 130 mechanical resonances of the tested fastener (a 30.5-cm aluminum stud). Convolved with the stud response is the transducer bandwidth centered nominally at 6 MHz. An additional modulation more pronounced at the lower frequencies occurring every nine harmonics will be discussed later. The mechanical resonances are shown more clearly in Fig. 4 which expands the frequency scale to cover from 5.5 ~ 5.7 MHz. The mechanical resonances are plotted for four load levels in a fastener. As the fastener is stressed, all the mechanical resonances shift down in frequency as expected from equation (8).

Some care must be exercised in a true CW [1], SCW, [10], or TBS measurement. There are modulations in the spectral response shown in Fig. 3 that have not yet been accounted for. These result from the mode converted longitudinal (L) to shear (S) to (L) waves. Fig. 5 identifies the time domain breakup of a single L wave propagating in a rod. The single L wave packet separates into a series of waves as identified in the figure. The time separation of the wave packets is determined by the diameter of the rod, the shear wave velocity, and the angle, θ, at which the shear wave propagates. θ is dictated by the "Snell's Law" relationship for the grazing longitudinal waves incident at the side walls. The angle is shown to be [11]:

$$\theta = \sin^{-1}(v_s/v_l) \tag{11}$$

and the time of arrival:

$$t = 2\frac{md}{v_L} + nd\frac{\left(v_L^2 - v_S^2\right)^{1/2}}{v_L\,v_S} \tag{12}$$

where $m = 1, 2, 3, \ldots$ and $n = 0, 1, 2 \ldots$ and d is the fastener diameter. In the frequency domain, the effect of the mode converted wave is the observed modulation of the spectra of Fig. 3. Whenever the L and LSL waves are π out of phase, cancellation occurs. These frequencies are:

$$F_p = \frac{(2p + 1)\,v_L\,v_S}{nd\left(v_L^2 - v_S^2\right)^{1/2}} \tag{13}$$

where $p = 0, 1, 2 \ldots$. CW measurements of frequency shifts near the cancellation points are subject to error caused by the superposition of the mode converted waves and thus should be avoided.

The pronounced modulation of the spectra of Fig. 3 resulting from mode conversion nearly disappears at the higher frequencies. For mode conversion to occur, the L wave must be incident at a grazing angle to the sample side wall. The far field acoustic diffraction angle, $\phi = \sin^{-1}(1.22\lambda/D)$, depends on frequency with the angle decreasing as frequency increases. Therefore, less of the acoustic wave energy converts at the higher frequencies.

CW BOLT STRAIN MONITOR

The observed spectral shifts of stressed fasteners suggests that a CW approach is feasible for a bolt stress instrument. Such a device based on an acoustic marginal oscillator [12,13] has been reported [4,14] which locks an instrument oscillator to a high acoustic harmonic of a fastener. The

instrument is a Reflection Oscillator Ultrasonic Spectrometer [15] (ROUS) and is shown in Fig. 6. The device is quite simple electronically and has a high inherent resolution (parts in 10^7) since the sample plays the role of a phase feedback path. Furthermore, the nature of the measurement takes advantage of the sensitivity enhancement [1,2] present in CW acoustic systems.

As the fastener is strained, the increased phase shift alters the total feedback. For marginal oscillation to occur, the loop phase shift must be an integral number of 2π and the loop gain must equal the loop loss. By maintaining the marginality of oscillation, as is done in typical NMR systems, the ROUS tracks in frequency any change in the fastener.

A limitation of the ROUS method occurs when applied to long fasteners of poor acoustic lineshape width. The separation between mechanical resonances is $F_j - F_{j\pm1} = v/a$ and for a 10-cm-long aluminum rod is 31 kHz. For a flat and parallel sample, the resonance width measured at half power points is 2 kHz. For an off-the-shelf fastener, the width might increase to 10 kHz. Under these conditions it is possible for the ROUS to jump to a new harmonic terminating the measurement. To prevent this, a high Q electronically tunable filter was placed in the feedback loop that kept the ROUS on the desired harmonic. Details of the locking scheme are presented elsewhere [14].

The ROUS transducer must provide adequate isolation (typically greater than 60 dB) between transmitter and receiver section to prevent electromagnetic crosstalk [16]. Fig. 7a is one design of a concentric highly isolated transducer compatible with the marginal oscillator system. Figure 7b shows the necessary ground plane transducer mount for high isolation. The transducer complexity and fragility can cause some problems for general field use. Other transducer geometries are possible, such as side-by-side piezoelectric elements, but they do not represent a cylindrically symmetric acoustic source. Lack of source symmetry can lead to positioning artifacts.

PSEUDO-CW BOLT STRAIN MONITOR

A solution to the transducer complexity and the somewhat delicate operation of the marginal oscillator was developed which continued the CW approach to bolt strain measurement. The device was called a Pseudo-CW Bolt Monitor [17] (PCW) which is a modification of some of the pioneering work of Miller and Bolef [1].

A block diagram of a PCW is shown in Fig. 8. Although the electronics are more complex than the ROUS, the PCW can use ordinary commercial transducers and requires less user training.

In this system, a voltage controlled oscillator (VCO) is gated onto a transducer until equilibrium conditions exist (power lost by attenuation equals input power). The drive power is momentarily interrupted and a receiver is connected to the transducer. The RF amplitude is detected, sampled and held, and the amplitude modulation (AM) phase is compared to the frequency modulation (FM) phase of the VCO. At the peak of a mechanical resonance, there is a 180 degree phase shift between the AM and FM as can be seen from the $d|A|/dk$ curve in Fig. 1. The phase signal is used in a feedback loop to control the center frequency of the VCO thus locking the system to a mechanical resonance peak.

The transmitter off time is short compared to the decay time in the sample. Therefore, the sample is at near equilibrium acoustic conditions at all times. By gating the transmitter off during the time of the measurement, the high isolation necessary in the ROUS system has been incorporated in the electronics of the instrument. Furthermore, full advantage of sensitivity enhancement is still maintained.

For fasteners of good geometry, the PCW system is nearly ideal. The operator need only apply the transducer to the fastener, lock the feedback loop, and record the initial lock frequency. As the fastener is tensioned, the change in frequency, $\Delta F/F$, indicates the preload For fasteners of poor geometry, the clean "picket fence" spectral lineshapes are replaced with complex spectra. The PCW, like the ROUS, locks to any spectral peak. For complex spectra which may change shape during preloading, the CW systems may lead to a measurement error as the spectra varies. For such fasteners, a different acoustic approach was necessary.

SPOILED "Q" BOLT MONITOR

Poor geometry fasteners have spectra such as is shown in Fig. 9. Near the peak of a mechanical resonance, the frequency domain and time domain responses appear nearly normal (Lawrencian and exponential, respectively). Examination of Fig. 9 away from the central peak, however, shows that frequency peak splitting can nearly destroy the normal exponential time domain decay and the corresponding sensitivity enhancement lineshapes in the frequency domain. Locking in the frequency domain to such a spectra is meaningless.

If the detail of the spectra could be lost yet maintain the general spectral lineshape, locking to even poor spectra is possible. There are several methods to spoil the "Q" of a resonance. One example of a spoiled "Q" spectra for the same fastener of Fig. 9 is shown in Fig. 10. This spoiled "Q" system allows only one standing wave overlap to occur in the sample instead of the PCW equilibrium condition. The spectral detail is lost while the main periodic harmonic response is preserved. This spectra can be used to measure stress in a fashion similar to the FM/AM phase locking of the PCW.

THE PULSE PHASE LOCKED LOOP

The last system to be discussed here is a mixture of [illegible] pulse technology. The first system of this nature was reported by Blume [18] with an improved version disclosed recently [19] for fastener tension. A simplified block diagram of the pulse phase locked loop (P^2L^2) is shown in Fig. 11.

A small portion of the RF from a VCO is gated to a transducer. The signal is received by a second transducer (or by the transmitter transducer) and the phase of the acoustic wave is compared to that of the VCO. Should the phase be out of quadrature, a control voltage adjusts the frequency of the VCO to establish the fixed phase condition. Any change in acoustic phase will produce a corresponding change in VCO frequency.

Mathematically, this is setting any gradient of phase equal to zero. For example, consider a strain derivation of the phase ϕ:

$$\frac{d}{da}\left(\phi - \frac{2\pi a v}{V}\right) = 0 \tag{14}$$

This frequency can be written:

$$\frac{\Delta F}{F} = \frac{\Delta v}{v} - \frac{\Delta a}{a} \tag{15}$$

which is identical to equation (9).

Therefore, the P^2L^2 operates as if it were a CW system with $\Delta F/F$ representing a fundamental material property change similar to the ROUS and PCW. The data presented in the next section was obtained with a P^2L^2 device. This technique is somewhat less critical of fastener geometry than the spoiled "Q" method and has higher resolution.

An acoustic spectrum is shown in Fig. 12 for a poor geometry fastener having symmetric raised lettering and a hole through the fastener normal to the axis. The complexity of the spectra is indicative of the bolt geometry. Fig. 13 shows stress/$\Delta F/F$ data for the fastener of Fig. 12 obtained with a PCW operating in a spoiled "Q" mode and for the P^2L^2. Although there is some structure on the P^2L^2 data, it generally represents a clear advancement in bolt stress measurement capability by producing a linear plot for such a complex sample geometry.

AXIAL STRESS MEASUREMENTS

The measurement of axial stress using ultrasonic techniques involves many factors. From equation (8) we may define an inverse stress acoustic constant:

$$H^{-1} = (\Delta\sigma)/(\Delta F/F) = \left[\frac{1}{v}\frac{dv}{d\sigma} - \frac{1}{E}\right]^{-1} \tag{16}$$

which when multiplied by $\Delta F/F$ produces the change in stress. The constant contains elastic properties of the stressed material and varies from material to material.

Figure 14 shows a typical range for engineering materials as measured with the P^2L^2 system. The inverse slopes of these curves represent H^{-1} and are determined from this figure as: H^{-1} aluminum = 1.86×10^4 MPa; H^{-1} stainless steel = 4.29×10^4 MPa; H^{-1} titanium = 4.78×10^4 MPa; and H^{-1} mild steel = 6.29×10^4 MPa.

A second factor which influences ultrasonic bolt stress measurements is the percentage of fastener under load. The effect of load length on H is identical to the effect load length plays on strain. For uniform strain abruptly going to zero at the fastener/nut interface, the effective $H_{eff} = HX/a$ where a is the fastener length and X is the load length. The slope of $H_{eff}(X)$ should be linear and is shown in Fig. 15 for a 25-cm fastener as X was varied. The excellent linearity is gratifying but obscures the fact that the extended line does not go through zero. The cause of this is related to the nonuniform stress distribution in the nut and head of the bolt. The stress continues in the bolt head and actually becomes compressional for a bolt under tension. For this reason, short bolts should be calibrated (true load cell vs. $\Delta F/F$) to obtain an H_{eff}^{-1} rather than using values of the material elastic constants in equation (16).

Another factor which plays some role in determining H^{-1} is the elastic/plastic zone effect. For a virgin mild steel bolt, the acoustic response becomes nonlinear at less than 50 percent ultimate as shown in Fig. 16. The lower curve was the first preloading with an initial starting frequency of 4.99536 MHz. After the first loading cycle, the initial starting frequency dropped to 4.98919 MHz for a change of 6.17 kHz. If we assume no residual

stress, the change represents a permanent strain of 1.2×10^{-3}. For no strain, the change in frequency corresponds to a residual compressive stress of 90 MPa. The true situation lies somewhere between the two extreme assumptions. Further loading cycles show little if any nonlinearity up to 270 MPa with the initial frequencies remaining nearly fixed. This is a classic case of work hardening but measured in a new more fundamental fashion related to the material Grüneisen constant.

EFFECTS OF TEMPERATURE

Temperature and stress have quite similar effects on a material from the standpoint of frequency measurements. Hydrostatic pressure and a decrease in sample temperature both lead to a volume contraction and an increase in sound velocity for most materials. For a uniaxial measurement, changes in temperature also produce corresponding strains as well as velocity shifts as shown in Fig. 17. These data were obtained for a 30-cm-long 2024 aluminum sample. The ratio of $\Delta v/v$ to $\Delta a/a$ for this case (thermal) is 7.5 compared to 2.5 for stress data shown in Fig. 18 for the same sample. From Figs. 17 and 18 it is apparent that a small change in temperature such as 5°C will produce a corresponding $\Delta F/F$ of 10^{-3} or about 20 MPa effective stress. Therefore, measurements obtained at different temperatures, such as a recertification of preload at a later date, must correct for temperature changes.

A more subtle temperature effect involves the temperature rise due to fastener friction as well as the compensating decrease in temperature resulting from the thermodynamic volume expansion. The second effect has not been reported in this context but is similar to the simple expansion of a gas and its subsequent cooling. In Fig. 19 data are presented for temperature, strain, and $\Delta F/F$ as a function of time for a step loading/unloading of the sample of Figs. 17 and 18. The load applied was about 125 MPa or about 50 percent of ultimate for that sample. It is important to appreciate this subtility if one wishes to make full use of the high resolution capability of the P^2L^2 for measurements of joint relaxation, creep, etc.

OTHER ULTRASONIC MEASUREMENTS OF STRESS

Many other acoustic techniques are applicable in stress measurements. Acoustic birefringence, for one, has demonstrated great potential where material texture can be determined. Under such conditions, absolute determination of stress is possible. Another recent technique has been associated with residual stress. This uses the apparent increase in attenuation which accompanies residual stress. This is a complex phenomenon which to date involves a correlation study rather than direct physical modeling.

Complex residual stresses imply complex velocity profiles. An initially plane wave acoustic propagation in such a medium will be altered to a complex phase front. Conventional transducers will output a reduced electrical amplitude for such situations. The reduction, however, is to a great extent an artifact of the nature of the piezoelectric device. Since all conventional transducers are phase sensitive, phase cancellation occurs producing phantom measurements [20]. These effects can be extreme at ultrasonic frequencies [21-23] and can lead to misinterpreted data. Fig. 20 represents an extreme case of phase cancellation.

A bolt is put under anisotropic loading to produce a shifted phase front. Data are shown for a conventional transducer and for a new ultrasonic

power transducer [24,25] which is phase insensitive. The four figures show an increase in load producing a reduction in amplitude of the piezoelectric signal while the power detector (Acousto-Electric Transducer or AET) shows no appreciable change. Thus, there is no increase in acoustic absorption or even attenuation in the fastener, just a perceived change which depends on the complex strain field, the acoustic frequency, and the transducer directivity. It is therefore difficult to interpret this type of "attenuation" without careful analysis.

CONCLUSIONS

The use of CW and pseudo-CW ultrasonic techniques have been shown to produce reliable measurements of axial stress. Factors which may influence a CW measurement are discussed in detail including propagation effects, mode conversion, frequency, material properties, and geometry. Data are presented for poor geometry fasteners which demonstrate the benefits of the various techniques. Measurements of acoustic attenuation which have been correlated to residual stress fields are shown to possibly involve transducer phase cancellation. Thus the measured drop in signal is an artifact of the transducer directivity and not an actual decrease of the acoustic power.

ACKNOWLEDGEMENT

The authors wish to thank Dr. John Cantrell for many helpful discussions, F. Dale Stone for outstanding electronics support, and Bill J. McCabe for assistance with the measurements.

FIGURE CAPTIONS

FIG. 1-Acoustic lineshapes for particle velocity and sensitivity enhancement factor for attenuation and dispersion.

FIG. 2-Block diagram of Tone Burst Spectroscopy (TBS) method used for spectral analysis.

FIG. 3-Spectral plot obtained for a 30.5-cm-long aluminum stud showing about 130 mechanical resonances.

FIG. 4-Expanded scale of Fig. 3 with spectra obtained at four different strain levels.

FIG. 5-Time domain representation of mode converted acoustic propagation of a 2 MHz tone burst propagating in an aluminum rod.

FIG. 6-Block diagram of Reflection Oscillator Ultrasonic Spectrometer.

FIG. 7a-ROUS piezoelectric transducer having high electrical isolation between transmitting and receiving sections.

FIG. 7b-ROUS transducer ground plane housing.

FIG. 8-Block diagram of Pseudo-Continuous Wave (PCW) spectrometer.

FIG. 9-Frequency spectra and corresponding time domain decays obtained at the indicated frequency points for a poor geometry resonator.

FIG. 10-Spoiled "Q" spectra of the same resonator shown in Fig. 9.

FIG. 11-Block diagram of pulse phase locked loop ultrasonic system (F^2L^2).

FIG. 12-Acoustic spectra of a poor geometry fastener having raised lettering and a transverse hole.

FIG. 13-Ultrasonic strain data obtained from a spoiled "Q" and a F^2L^2 for the fastener of Fig. 12.

FIG. 14-Stress acoustic constants determined from the slope $(\Delta F/F)/\Delta\sigma$ are shown for several common fastener materials.

FIG. 15-Effect of percent load length on stress acoustic constant.

FIG. 16-Normalized frequency shift for a mild steel bolt loaded 5 times showing a nonlinearity in the first load cycle.

FIG. 17-Normalized frequency shift and strain for an unloaded aluminum sample as a function of temperature.

FIG. 18-Normalized frequency shift and strain for the aluminum sample of Fig. 17 as a function of stress.

FIG. 19-Square wave stress loading of the sample of Fig. 17 illustrating the thermodynamic cooling of a solid.

FIG. 20-Effect of phase cancellation on acoustic amplitudes for anisotropically loaded sample. Apparent changes in attenuation in residual stress samples may result from this effect and therefore be a measurement artifact.

REFERENCES

[1] Bolef, D. I. and Miller, J. G., Physical Acoustics, (W. P. Mason and R. N. Thurston, eds.) Vol. 8, pp. 96-201, Academic Press, New York (1971).

[2] Heyman, J. S. and Miller, J. G., "Verification of Sensitivity Enhancement Factors for CW Ultrasonic Resonators," Journal of Applied Physics, 44, pp. 3398-3400 (1973).

[3] Melcher, R. L., Bolef, D. I., and Merry, J. B., "Frequency Modulation CW Technique for the Measurement of Very Small Changes in Ultrasonic Velocity," Review of Scientific Instruments, 39, pp. 1618-1620 (1968).

[4] Heyman, J. S., "A CW Ultrasonic Bolt Strain Monitor," Experimental Mechanics, 17, pp. 183-187 (1977).

[5] Cantrell, J. H., Jr., "Generalized Grüneisen Tensor from Solid Nonlinearity Parameters," Physics Review, 21, pp. 4191-4195 (1980).

[6] Cantrell, J. H., Jr., "Anharmonic Properties of Solids from Measurements of the Stress Acoustic Constant," ASTM STP , (T. Proctor and J. S. Heyman, eds.) at Ultrasonic Measurements of Stress Symposia, April 1981.

[7] Hughes, D. S. and Kelly, J. L., "Second Order Elastic Deformation of Solids," Physics Review, 92, pp. 1145-1149 (1953).

[8] Murnaghan, F. D., "Finite Deformation of an Elastic Solid," Dover Pub., Inc., New York (1967).

[9] Cantrell, J. H., Jr. and Heyman, J. S., "Ultrasonic Spectrum Analysis Using Frequency-Tracked Gated RF Pulses," Journal of Acoustical Society of America, 67, pp. 1623-1626 (1980).

[10] Miller, J. G. and Bolef, D. I., "A Sampled-Continuous Wave Ultrasonic Technique and Spectrometer," Review of Scientific Instruments, 40, pp. 915-920 (1969).

[11] Hughes, D. S., Pondrom, W. L., and Mims, R. L., Physics Review, 75, p. 1552 (1949).

[12] Bolef, D. I. and Menes, M., Physics Review, 114, p. 1441 (1959).

[13] Conradi, M. S., Miller, J. G., and Heyman, J. S., "A Transmission Oscillator Ultrasonic Spectrometer," Review of Scientific Instruments, 45, pp. 358-360 (1974).

[14] Heyman, J. S., "CW Ultrasonic Bolt Tensioning Monitor," United States Patent # 4,062,227 (1977).

[15] Heyman, J. S., "A Self-Exciting Ultrasonic Reflection CW Instrument," Proceedings IEEE Ultrasonic Symposium Cat. # 76 CH1120-5SU (1976).

[16] Miller, J. G., "Electromagnetic Leakage in High-Frequency Continuous-Wave Ultrasonics," Journal of Acoustical Society of America, 53, pp. 710-713 (1973).

[17] Heyman, J. S., "Pseudo Continuous Wave Instrument," United States Patent # 4,117,731 (1978).

[18] Blume, R. J., Review of Scientific Instruments, 34, p. 1400 (1963).

[19] Heyman, J. S., "Pulsed Phase Locked Loop Strain Monitor," NASA Patent Disclosure LAR 12772-1 (1980).

[20] Heyman, J. S. and Cantrell, J. H., Jr., "Application of an Ultrasonic Phase Insensitive Receiver to Materials Measurements," Proceedings IEEE Ultrasonics Symposium, Cat. # 77CH1264-1SU, pp. 124-128 (1977).

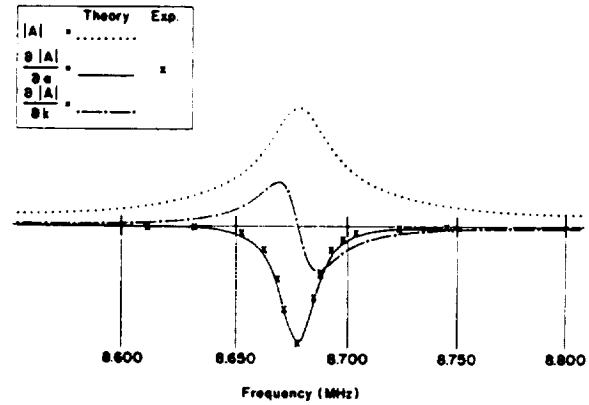
[21] Breazeale, M. A., Cantrell, J. H., Jr., and Heyman, J. S., "Ultrasonic Wave Velocity and Attenuation Measurements," Methods of Experimental Physics, Edmonds, P. D., Ed., Academic Press, New York (1981).

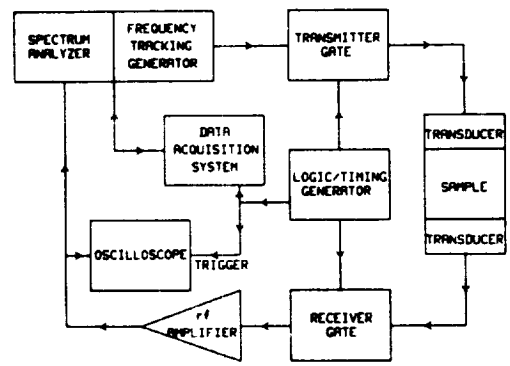
[22] Heyman, J. S., "Ultrasonic Coupling to Optically Generated Charge Carriers in CdS: Physical Phenomena and Applications," Ph. D. Thesis, Washington University, 1975, also NASA TMX-68732.

[23] Heyman, J. S. and Cantrell, J. H., Jr., "Effects of Material Inhomogeneities on Ultrasonic Measurements: The Problem and a Solution, Nondestructive Evaluation and Flaw Criticality for Composite Materials, ASTM STP 696, R. B. Pipes, Ed., pp. 45-56 (1979).

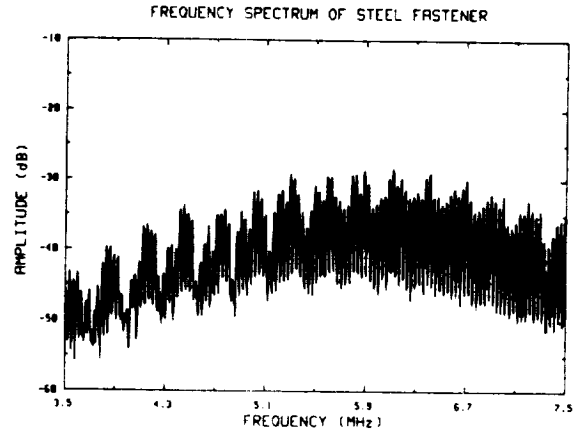
[24] Southgate, P. D., Journal of Acoustical Society of America, 39, p. 480 (1966).

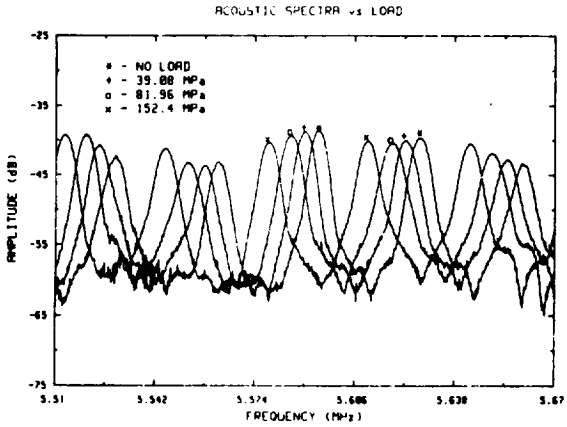
[25] Heyman, J. S., "Phase Insensitive Acoustoelectric Transducer," Journal of Acoustical Society of America, 64, pp. 243-249 (1978).

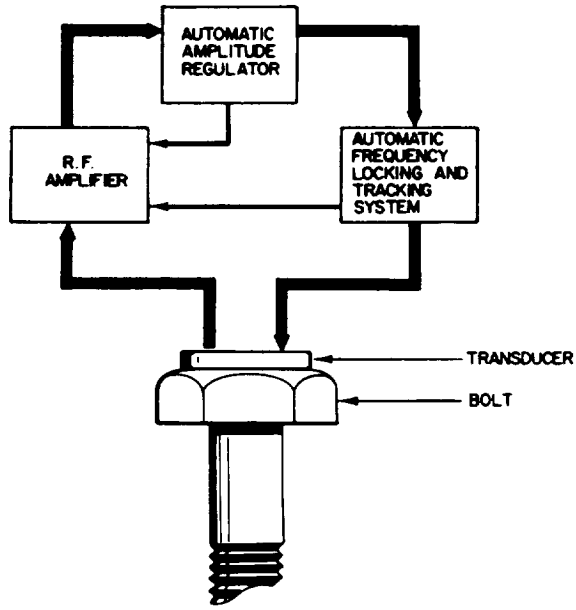


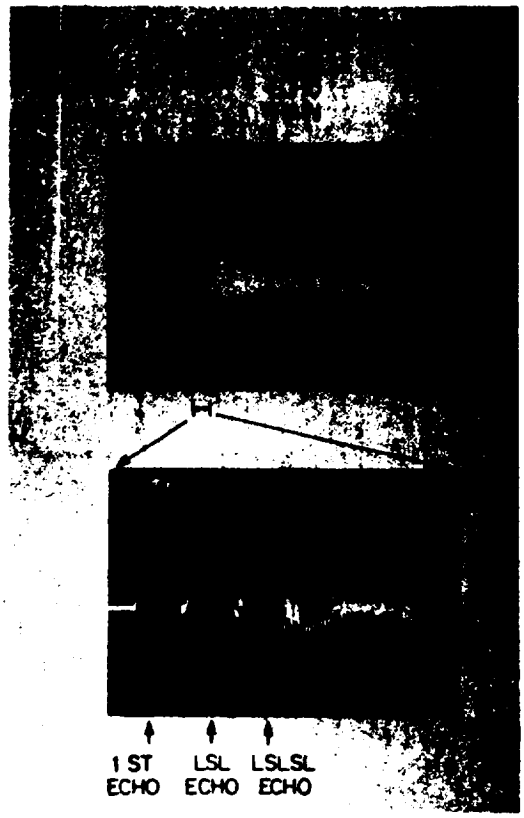


BLOCK DIAGRAM OF SCW MEASUREMENT SYSTEM

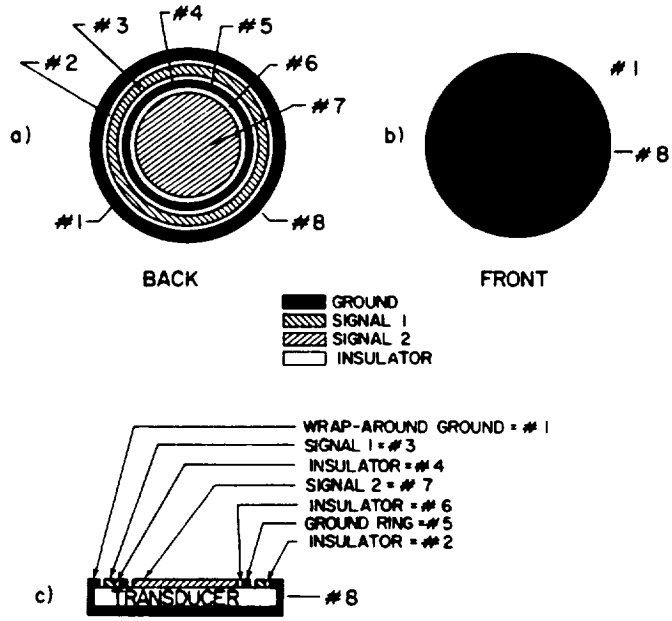


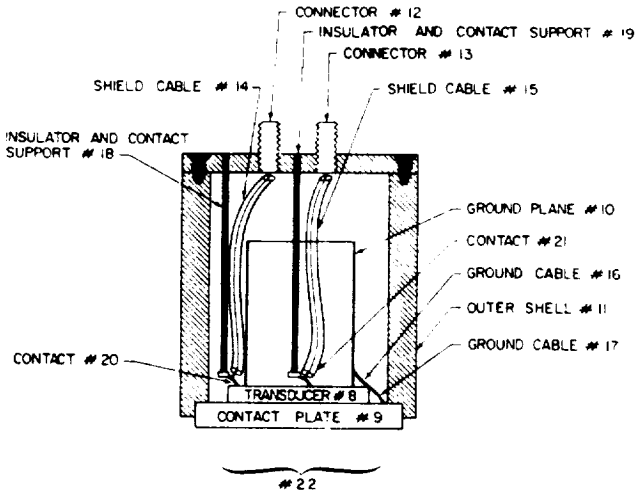




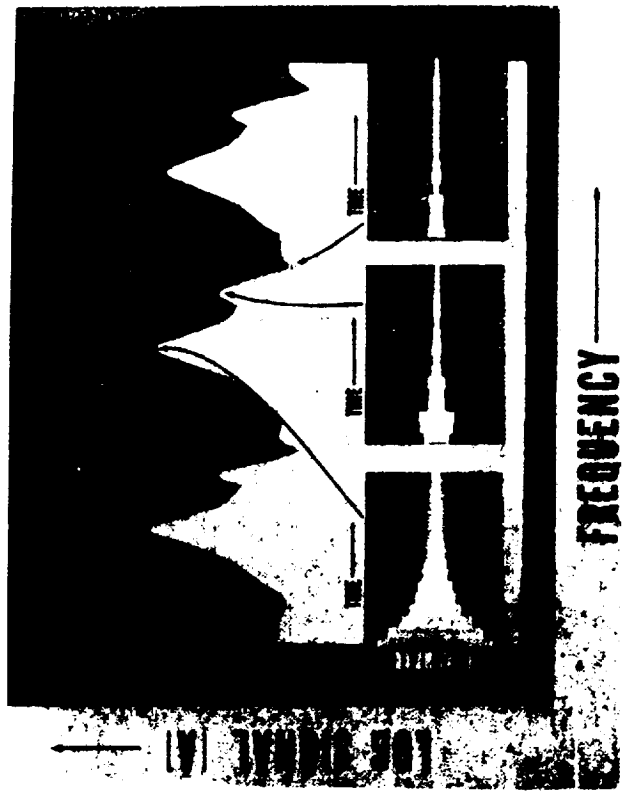


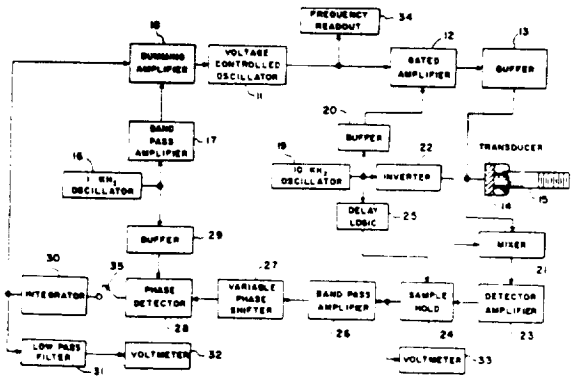
REFLECTION OSCILLATOR ULTRASONIC SPECTROMETER (ROUS) TRANSDUCER

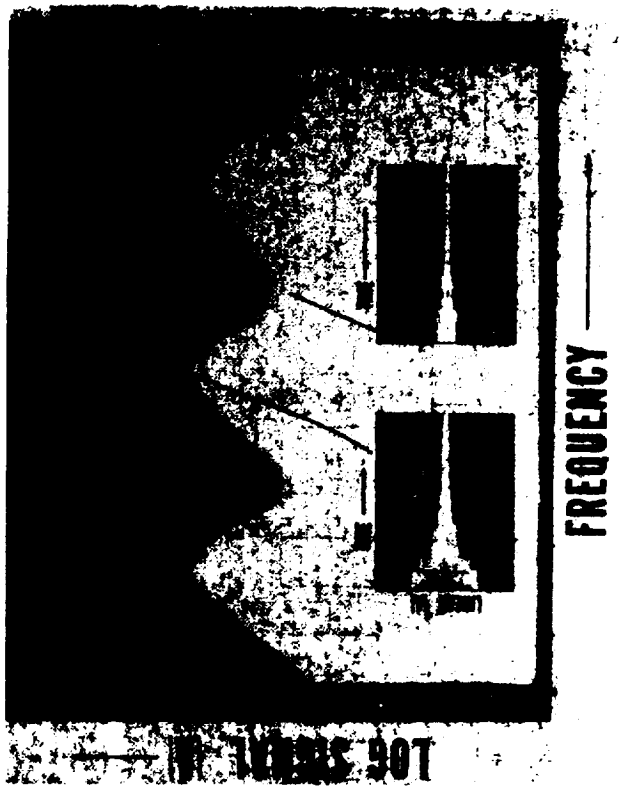


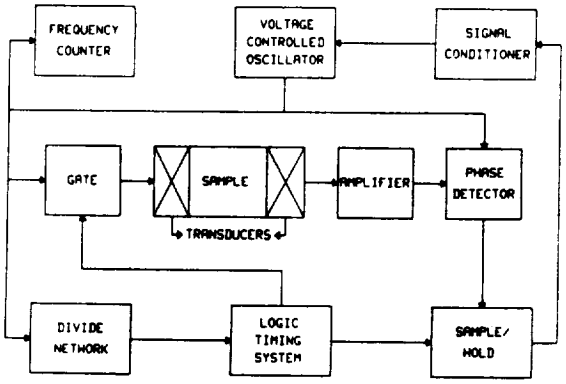


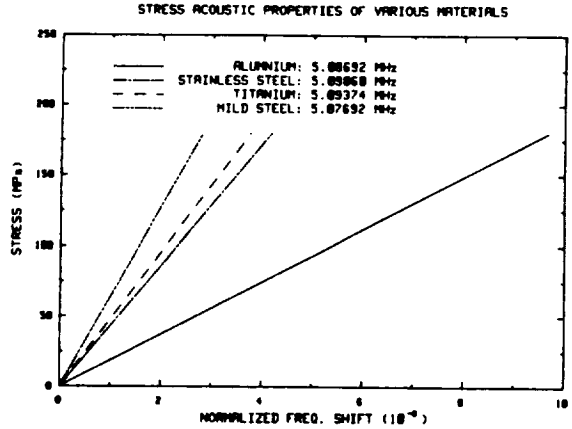
ROUS TRANSDUCER ASSEMBLY

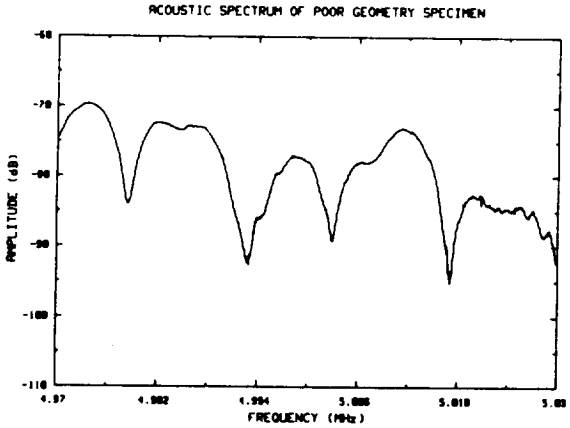


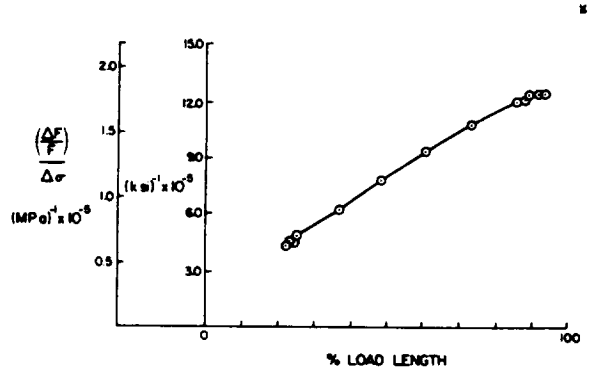


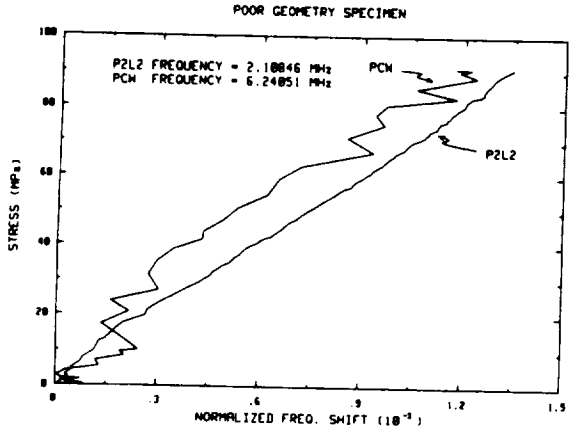


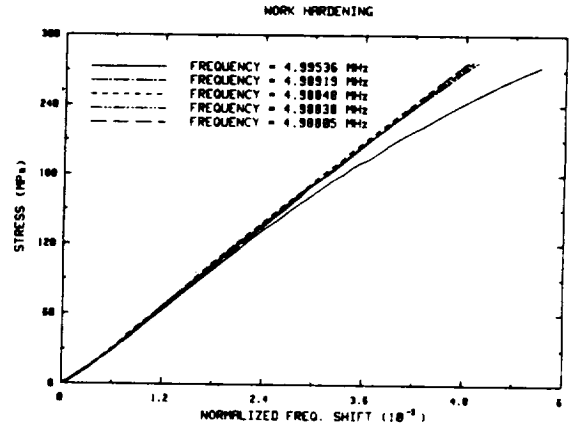

FREQUENCY COUNTER
VOLTAGE CONTROLLED OSCILLATOR
SIGNAL CONDITIONER
GATE
SAMPLE
AMPLIFIER
PHASE DETECTOR
TRANSDUCERS
DIVIDE NETWORK
LOGIC TIMING SYSTEM
SAMPLE/ HOLD

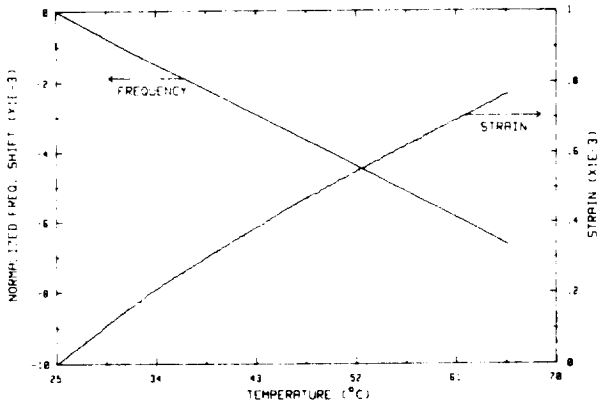

STRESS ACOUSTIC PROPERTIES OF VARIOUS MATERIALS
ALUMINIUM: 5.08692 MHz
STAINLESS STEEL: 5.09060 MHz
TITANIUM: 5.09374 MHz
MILD STEEL: 5.07692 MHz
STRESS (MPa)
NORMALIZED FREQ. SHIFT

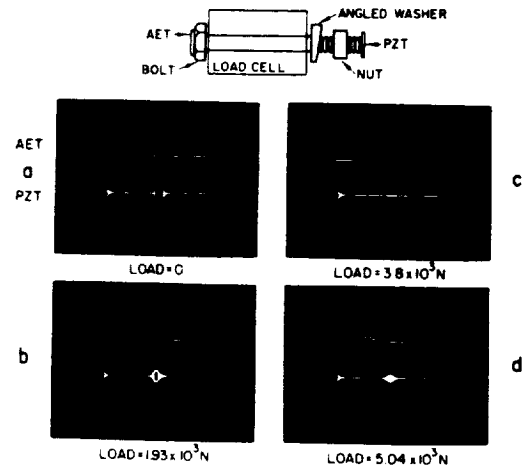

ACOUSTIC SPECTRUM OF POOR GEOMETRY SPECIMEN
AMPLITUDE (dB)
FREQUENCY (MHz)

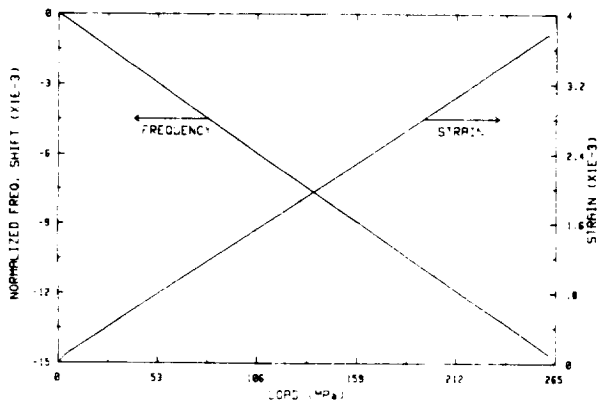

% LOAD LENGTH

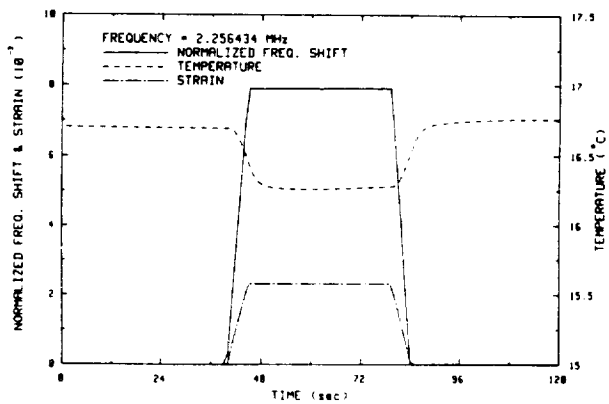

POOR GEOMETRY SPECIMEN
P2L2 FREQUENCY = 2.10846 MHz
PCW FREQUENCY = 6.24051 MHz
PCW
P2L2
STRESS (MPa)
NORMALIZED FREQ. SHIFT


WORK HARDENING
FREQUENCY = 4.99536 MHz
FREQUENCY = 4.99919 MHz
FREQUENCY = 4.99848 MHz
FREQUENCY = 4.99838 MHz
FREQUENCY = 4.99885 MHz
STRESS (MPa)
NORMALIZED FREQ. SHIFT


FREQUENCY
STRAIN
TEMPERATURE (°C)
STRAIN (XIE-3)


ANGLED WASHER
AET
PZT
BOLT
LOAD CELL
NUT
AET
a
PZT
c
b
d
LOAD=0
LOAD=3.8x10^3N
LOAD=1.93x10^3N
LOAD=5.04x10^3N


FREQUENCY
STRAIN
NORMALIZED FREQ. SHIFT (X1E-3)
STRAIN (X1E-3)


FREQUENCY = 2.256434 MHz
NORMALIZED FREQ. SHIFT
TEMPERATURE
STRAIN
NORMALIZED FREQ. SHIFT & STRAIN
TEMPERATURE (°C)
TIME (sec)

REGISTRANTS

INDUSTRY/SPACE DIVISION/NASA MISSION ASSURANCE
CONFERENCE/WORKSHOP
HYATT HOUSE HOTEL, LOS ANGELES INTERNATIONAL AIRPORT
LOS ANGELES, CA
7-9 JUNE 1983

REGISTRANTS

Aaront, Bruce
Vice President, Product Assurance
SPACECOM
1300 Quince Orchard Blvd.
Gaithersburg, MD 20878

Abbey, Harold E.
Manager, Process Control &NDT
Vought Corporation
P.O. Box 225907
Dallas, TX 75229

Ackerman, Arthur W., Jr.
Senior Reliability Engineer/Failure Review
Board Chairman
Ford Aerospace & Communications Corporation
3939 Fabian Way
Palo Alto, CA 94303

Adams, Martin L.
Division Director, Design & Product Assurance
Rockwell International
12214 Lakewood Blvd.
Downey, CA 90805

Adams, Thomas J.
Chief, Quality Assurance Division
NASA Johnson Space Center
Houston, TX 77058

Adelman, Walter B.
Assistant Program Manager
TRW
One Space Park MS TW/1893
Redondo Beach, CA 90278

Akans, Robert E.
Senior Metallurgist
Kaiser Aluminum - Erie Worsk
1015 E. 12th Street
Erie, PA 16512

Allenbaugh, Richard N.
Sr. Quality Engineer
McDonnell Douglas Astronautics Co.
5301 Bolsa Avenue
Huntington Beach, CA 92647

Alley, D.J.
Assistant Director
Lockheed Missiles & Space Company
P.O. Box 504
Sunnyvale, CA 95070

Anderson, Herbert E.
Systems Effectiveness
The Aerospace Corporation
2350 East El Segundo
El Segundo, CA 90245

Anderson, Jerry E.
Director, Quality Assurance
Honeywell Inc.
10901 Malcolm McKinley Drive
Tampa, FL 33612

Anderson, Patricia A.
Associate Principal Director
The Aerospace Corporation
2350 E. El Segundo Blvd., M5/617
El Segundo, CA 90245

Arthur, Russell J.
Coordinator, Motivation Programs Dept. 3020
Martin Marietta Aerospace
13800 Old Gentilly Road
New Orleans, LA 70189

Austin, David J.
Mgr., SR&QA for ET, SRB, SSME & Launch Operations
NASA Headquarters
Code: DP
Washington, D.C. 20546

Baldwin, Howard B.
Quality Assurance Specialist
DCASR-LA
11099 S. La Cienega Blvd.
Los Angeles, CA 92708

Barefield, Road
P.E.
The Aerospace Corporation
P.O. Box 92957
Los Angeles, CA 90009

Barlow, George C.
Director, Reliability & Product Assurance
United Space Boosters Inc./UTC
W. Station, P.O. Box 1626
Huntsville, AL 35807

Barnes, Lynwood S.
Program Manager
Boeing Company/System Assurance Organization
P.O. Box 5440
Vandenberg AFB, CA 93437

Battey, Robert V.
Manager, Customer Service Center
NASA Johnson Space Center
Houston, TX 77058

Bauer, Jerry
MTS
JPL/NASA
4800 Oak Grove
Pasadena, CA 91101

Bauer, Tom H.
Manager, Office of Software Cost Estimation
TRW-DSSG
One Space Park, R2/1076
Redondo Beach, CA 90278

Baugher, Lawrence A.
QA Manager
Harris Gov't Aerospace Systems Division
P.O. Box 37
Melbourne, FL 32901

Bavuso, Salvatore J.
Aerospace Technologist
NASA-Langley Research Center
Hampton, VA 23665

Bay, Michael
SMRM Project Engineer
Fairchild Industries Inc.
20301 Century Blvd.
Germantown, MD 20874

Beardon, Michael
Systems Engineer
GT&T Industries
705A Lakefield Road
Westlake Village, CA 91361

Beck, Wayne E.
Director, Quality Assurance
Ball Aerospace Systems Division
P.O. Box 1062
Boulder, CO 80306

Beheim, Milton A.
Chief Engieer
NASA Lewis Research Center
21000 Brookpark Road
Cleveland, OH 44135

Beland, Russell R.
Chief, Quality Assurance Division
U.S. Air Force/AFPRO Det 10/Martin Marietta
Denver Aerospace
P.O. Box 179
Denver, CO 80201

Benninger, Joseph E.
Marketing Manager
Bendix Test Systems Division
117 E. Providencia Avenue
Burbank, CA 91503

Bennett, William S.
Project Engineer
The Aerospace Corporation
1320 Orleans Drive
Sunnyvale, CA 94086

Berri, Robert E.
MTS
Aerospace Corporation
P.O. Box 92957
Los Angeles, CA 90009

Beyer, Major Jon N.
System Safety Manager
Space Division/SD/SE
Box 92960
Los Angeles, CA 90009

Bienstock, Bernard J.
Project Manager
Hughes Aircraft Company
Space & Communications Group
P.O. Box 92919
Los Angeles, CA 90009

Bincaewski, George J.
Metallurgical Manager
Kaiser Aluminum
300 Lakeside Drive
Oakland, CA 94643

Binczewski, George J.
Metallurgical Manager
Kaiser Aluminum & Chemical Corporation
300 Lakeside Drive
Oakland, CA 94643

Bithell, Wayne
Manager, Systems Effectiveness
Martin Marietta
P.O. Box 1681 MD70B
Vandenberg AFB, CA 93437

Blair, Robert J.
Business Development Manager
TDC Defense Systems Group
624 Six Flags Drive
Arlington, TX 76011

Block, Phillip A.
Manufacturing Manager - ASAT
Vought Corporation
P.O. Box 225907
Dallas, TX 75265

Bloom, Victor A.
Director ILS Design Engineering
Grumman Aerospace Corporation
Mail Station B08-35
Bethpage, NY 11714

Bloomquist, Charles E.
Principal
PRC Systems Services
10960 Wilshire Blvd., #2320
Los Angeles, CA 90024

Boardman, Allan J.
General Manager, Electronics & Optics Division
The Aerospace Corporation
2350 E. El Segundo Blvd.
El Segundo, CA 90245

Borland, Robert W.
Software Development Manager
General Electric, Space Division
P.O. Box 8555
Philadelphia, PA 19101

Bowles, Thomas C.
ILS Program Manager
Magnavox Advanced Products
2829 Maricopa Street
Torrance, CA 90503

Craig, Larry G.
Launch Vehicle Integration Engineer
Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, CA 91101

Curistian, Robert E.
Manager, Management Systems
Hughes Aircraft Company
2400 E. Imperial Highway
El Segundo, CA 90266

Dagonese, Frank
Sr. Quality Engineer
TRW - DSSG
P.O. Box 1310
San Bernardino, CA 92402

Dalebout, Ronald A.
Manager, Satellite Engineering
Southern Pacific Satellite Company
1700 Old Meadow Road
McLean, VA 22102

Dannessa, Angelo T.
Requirements Officer-Networks
NASA, Goddard Space Flight Center
Greenbelt, MD 20771

Darnell, Gregory L.
Software Engineering Manager
Boeing Aerospace Corporation
29145 23 Place S.
Federal Way, WA 98003

Dausch, Robert J.
Senior Staff Engineer, Software Program Planning
McDonnell Douglas Astronautics Co.
5301 Bolsa Avenue, MC A3-370-13-3
Huntington Beach, CA 92647

Davis, Al
Manager, Systems Evaluation & Compliance
Aerojet Strategic Propulsion Company
P.O. Box 15699C
Sacramento, CA 95813

Davis, Bill E.
Supervisor Design Support Reliability
Ford Aerospace
3939 Fabian Way
Palo Alto, CA 94303

Davis, Charles L.
Department Head
Bell Laboratories

Holmdel, NJ 07760

Deatherage, Carolyn M.
Contracts Manager
Hughes Aircraft Company
1700 E. Imperial Hwy.
El Segundo, CA 90045

DeFeo, Pio V.
Senior Principal Engineer
HR Textron
2485 McCabe Way
Irvine, CA 92714

Denman, Gary L.
Chief, Manufacturing Technology Division
Materials Laboratory
AFWAL/MLT
WPAFB, OH 45433

Depp, Robert E.
Chief, Qualification Division
Defense Electronics Supply Center
1105 Wilmington Pike
Dayton, OH 45427

Dergarabedian, Paul
Systems Director
The Aerospace Corporation
P.O. Box 92957 M3-370
Los Angeles, CA 90274

Dick, Paul
Manager System Effectiveness
General Electric Space Systems
P.O. Box 8555 (Upper 2450)
Philadelphia, PA 19101

DiGregorio, Concezio
Senior Scientist
Hughes Aircraft Company
P.O. Box 92919 Bldg. 533, MS C346
Los Angeles, CA 90009

Dionne, Ron J.
Inspection Manager
Lockheed Missiles & Space Co.
1111 Lockheed Way
Sunnyvale, CA 94088

Doennebrink, Fred
Senior Staff Engineer
Martin Marietta Corporation
P.O. Box 1681 MD70B
Vandenberg AFB, CA 93437

Donald, Charlie H.
Manager, Quality Assurance
Boeing Aerospace Company
P.O. Box 3999
Seattle, WA 98124

Donati, John A.
Director of Product Assurance
Honeywell, Space & Strategic Avionics Div.
13350 U.S. Highway 19, South
Clearwater, FL 33516

Dowling, John E.
Lockheed Space Operations Company
1100 West
Lompoc, CA 93436

Driscoll, Kevin R.
Research Scientist
Honeywell Systems and Research
2600 Ridgway Parkway
Minneapolis, MN 55413

Brabson, Col. G. Dana
Dean, Department of Research & Information
Defense Systems Management College
Ft. Belvoir, VA 22060

Braun, Ronald L.
Sr. Software Engineer
Ford Aerospace & Communications Corp./WDL
3939 Fabian Way
Palo Alto, CA 94303

Bredall, C.H.
Associate Director, Defense Communications
Aerospace Corporation
2350 E. El Segundo, Blvd.
El Segundo, CA 90009

Briskman, Robert D.
Vice President, Systems Implementation
Communications Satellite General
950 L'Enfant Plaza, S.W.
Washington, D.C. 20024

Brody, Margaret
HQ Air Force Systems Command
ALXP
Andrews AFB, Washington, D.C. 20334

Brown, Stan L.
AFCMD ECR Focal Point
HQ AFCMD/KRR
Kirtland AFB, NM 87117

Brubaker, Daniel R.
Program Control Manager - Space Shuttle Contracts
Sundstrand Corporation
4747 Harrison Avenue
Rockford, IL 61101

Burley, Thomas G.
Committee Executive
National Security Industrial Association
1015 15th Street, N.W., Suite 901
Washington, D.C. 20005

Burns, Ronald L.
Director, Central Manufacturing Operations
Harris Corporation, Gov't Systems Sector Opns
P.O. Box 37
Melbourne, FL 32901

Burr, William R.
Software Quality Assurance Engineering Manager
Boeing Aerospace Company
P.O. Box 3999 M.S. 8A-14
Seattle, WA 98124

Burrowbridge, Donald R.
Sr. Program Manager
Fairchild Space Company
20301 Century Blvd.
Germantown, MD 20874-1181

Butler, Michael R.
Manager, Quality Assurance, Special Space Programs
McDonnell Douglas Astronautics Corporation
5301 Bolsa Avenue (Dept. 729, M/S 13-2)
Huntington Beach, CA 92647

Calkins, Thomas C.
Quality Assurance Engineer
USAF Western Space & Missile Center
Vandenberg AFB, CA 93437

Cairns, John L.
Manager, of CAD/CAM
TRW
1 Space Park
Redondo Beach, CA 90278

Campbell, S. Steve
IUS E/E PM&P and Contamination Manager
Boeing Aerospace Company/IUS System Assurance
2238 216 Avenue SE (2-3948)
Issaquah, WA 98027

Campbell, William
Systems Safety Engineer
Pratt & Whitney Aircraft
P.O. Box 2691
West Palm Beach, FL 33402

Carder, Ronald E.
Q&RA Manager
Westinghouse Electric Corporation
Ft. Meade Road
Baltimore, MD 21302

Carlan, Alan J.
Sr. Engr. Spec.
Aerospace Corporation
P.O. Box 92957
Los Angeles, CA 90009

Carnahan, Charles E.
Vice President, Operations
Martin Marietta Aerospace
P.O. Box 179
Denver, CO 80201

Carpenter, Sharon A.
Manager, Automatic Subassembely Test
Rockwell International, Collins Gov't Avionics
400 Collins Road, NE
Cedar Rapids, IA 52498

Carpenter, C. Skip L.
Project Administrator/Software Quality Assurance
General Dynamics/Convair
5001 Kearney Villa Road
San Diego, CA 92123

Carr, Don D.
Manager, Product Assurance
Hughes Aircraft Company-Electron Dynamics Div.
3100 W. Lomita Blvd.
Torrance, CA 90509

Carter, R.J.
Senior QA Electronics Engineer
Vought Corporation
P.O. Box 225907
Dallas, TX 75265

Carter, Roger V.
Chief, Metals Technology
Boeing Commercial Airplane Company
P.O. Box 3707 M.S. 73-43
Seattle, WA 98124

Casey, John L.
Director of Mission Success
Martin Marietta-Orlando Aerospace
P.O. Box 5837 MP3
Orlando, FL 32855

Cassen, Quent C.
Director, Microelectronics Department
The Aerospace Corporation
P.O. Box 92957, M4/904
Los Angeles, CA 90009

Chiachierri, Joe
Contracts Administrator
Hughes Aircraft Company
1700 E. Imperial Highway
El Segundo, CA 90045

Chotkevys, Gerald P.
Milstar System Effectiveness Manager
Aerospace Corporation
P.O. Box 92957 M/S M5/669
Los Angeles, CA 90009

Christel, Clifford R.
Quality Engineer Supervisor
Santa Barbare Research Labs., General Motors
6767 Hollister Avenue
Goleta, CA 93117

Churchill, Col. Howard A.
Director of Policy & Programs, DCS/Acq Logistics
HQ AFSC/ALX
Andrews AFB, Washington, D.C. 20334

Clark, Alan R.
Manager
Martin Marietta Orlando Aerospace
P.O. Box 5837
Orlando, FL 32855

Clauser, Robert E.
Principal Director, Mission Assurance Directorate
The Aerospace Corporation
2350 E. El Segundo Blvd.
El Segundo, CA 90245

Clemens, Darlo J.
Chief
General Dynamics Convair
P.O. Box 85357
San Diego, CA 92138

Clymer, Benson
Manager, Product Assurance Systems
NASA-Lewis Research Center
21000 Brookpark Road
Cleveland, OH 44135

Cocchi, Jack L.
Quality Assurance Manager
United Space Boosters Inc.
P.O. Box 5177
Vandenberg AFB, CA 93437

Cohen, Sherman W.
System Effectiveness Project Engineer M5-694
Aerospace Corporation
P.O. Box 92957
Los Angeles, CA 90009

Coker, Walter D.
Quality Assurance Manager
ALCOA
P.O. Box 7500
Lafayette, IN 47902

Cole, Philip J.
Director, National Activities
National Security Industrial Association
1015 15th Street, N.W., Suite 901
Washington, D.C. 20005

Colgan, John C.
Manager, Automated Test Development
Northrop Electronics Division - Orgn. 3500/N5
2301 W. 120th Street
Hawthorne, CA 90250

Colonna, Richard A.
Dep. Mgr., Space Shuttle Projects
NASA-JSC
Houston, TX 77058

Combs, Robert E.
Member Technical Staff
The Aerospace Corporation
2350 E. El Segundo Blvd.
El Segundo, CA 90245

Conaway, Jack M.
Manager, Western Region Program Development
Computer Sciences Corporation
2100 E. Grand Avenue
El Segundo, CA 90245

Corey, Blakely L.
Quality Assurance Supt.
U.S. Air Force/Space Division/YGC
2400 El Segundo Blvd.
El Segundo, CA 90009

Corey, Raymond R.
Chief, Edcuation & Awareness Branch
NASA
Code: PA-EAB
Kennedy Space Center, FL 32899

Cosgrove, Richard W.
Advisory Engineer
IBM Corporation
1299 Orleans Drive
Sunnyvale, CA 94089

Courtright, William E.
Corporate Manager
Hughes Aircraft Company
P.O. Box 1042 Bdlg. C-2, MS B197
El Segundo, CA 90245

Covey, Robert E.
Manager, Electronic Parts Reliability
Jet Propulsion Lab (NASA)
4800 Oak Grove Drive
Pasadena, CA 91109

Cramer, William C.
Industrial Engineer
Air Force Plant Representatives Office
Detachment 9, The Boeing Company
Seattle, WA 98109

Duarte, James E.
Section Manager Statistical Services
Reynolds Metals Company
6601 West Broad Street
Richmond, VA 23261

Dufoe, Gene E.
Chairman Advanced Assurance Technology
McDonnell Douglas Corporation
P.O. Box 516
St. Louis, MO 63166

Dunaetz, Robert A.
Manager, Materials & Processes Engineering Dept.
Hughes Aircraft Company
P.O. Box 902 Bldg. E1/C186
El Segundo, CA 90245

Ebersold, Pauline M.
Acting Director, Product Assurance
U.S. Air Force/Space Division/YGC
2400 El Segundo Blvd.
El Segundo, CA 90009

Edwards, R.M.
Manager, Policy & Review
Martin Marietta Corporation
P.O. Box 179
Denver, CO 80201

Egelston, Richard W.
MTS
Aerospace Corporation
P.O. Box 92957
Los Angeles, CA 90009

Ehl, James H.
Chief, Towing Applications Branch
NASA-MSFC
MFSC, AL 35812

Elliott, George
Chief of Engineering
Martin Marietta Corporation
P.O. Box 1681 MD70B
Vandenberg AFB, CA 93437

Ells, Maynard A.
Senior Staff Engineer
TRW
One Space Park
Redondo Beach, CA 90278

Enz, Ralph E.
M.T.S.
The Aerospace Corporation
P.O. Box 92957
El Segundo, CA 90009

Erickson, Carl G.
M.T.S.
The Aerospace Corporation MS M4/993
2350 E. El Segundo, Blvd.
El Segundo, CA 90245

Erickson, Daniel E.
Technical Group Supervisor
Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, CA 91103

Ernst, Richard B.
Director, Quality Assurance & Reliability
Rockwell International
3370 Miraloma Avenue
Anaheim, CA 92806

Everline, Robert T.
Manager, Flight Manifest Office
NASA/JSC
Houston, TX 77058

Ewalt, Dean C.
Senior Engineer
IBM/FSD
18100 Frederick Pike
Gaithersberg, MD 20879

Evans, Bruce B.
Department Manager, Components Engineering
TRW Electronics & Defense, Operations &
Support Group
One Space Park
Redondo Beach, CA 90278

Evans, Michael W.
Vice President
ICE
2672 Bayshore Frontage
Mountain View, CA 94043

Farschman, George E.
Chief, Quality Engineering
Martin Marietta Aerospace
P.O. Box 1681
Vandenberg AFB, CA 93437

Fear, Reginald E.
Deputy
DCASMA Inglewood
9920 S. La Cienega
Inglewood, CA 90301

Feldmeier, Wilmer O.
Manager Quality Laboratories
McDonnell Douglas Corporation
P.O. Box 516
St. Louis, MO 63166

Fenoglio, William C.
Manager, Quality Records
McDonnell Douglas Aircraft
3855 Lakewood Blvd.
Long Beach, CA 90846

Fernandez, Manuel
Director, Advanced Digital Systems
Litton Guidance & Control Systems Division
5500 Canoga Avenue
Woodland Hills, CA 91365

Fiedlier, George A.
Chief, RSS&QE Branch
NASA-JSC
Shuttle Resident Office
Houston, TX 77058

Fischer, David J.
Manager, Product Assurance Laboratories
Martin Marietta Aerospace
P.O. Box 29304
New Orleans, LA 70127

Fleming, Richard C.
Engineering Specialist
The Aerospace Corporation
P.O. Box 92957
Los Angeles, CA 90009

Forlifer, William R.
Deputy Director-Office of Flight Assurance
NASA-Goddard Space Flight Center
Code 300
Greenbelt, MD 20771

Forsberg, Kevin J.
Program Manager
Lockheed
P.O. Box 504
Sunnyvale, CA 94086

Fort, T. Richard
Coordinator, Systems Refinement Teams
Martin Marietta Corporation
MMC-9
Kennedy Space Center, FL 32899

Fotos, Charles
GTE SSD Quality Assurance Manager
GTE Communications Products Corporation
1 Research Road
Westborough, MA 01581

Frederick, Herbert E.
Systems Manager, Systems Support Office
The Aerospace Corporation
P.O. Box 5068
Vandenberg AFB, CA 93437

Freitag, Robert E.
Deputy Director Space Station Task Force
NASA HQ Code: MFA-13
600 Indenpendence Avenue, S.W., Room 536
Washington, D.C. 20546

Fujiyoshi, Tom
Director of Mission Assurance
Martin Marietta
P.O. Box 1681 MD70B
Vandenberg AFB, CA 93437

Furman, Irwin L.
Staff Engineer
TRW
1 Space Park 120/1080
Redondo Beach, CA 90278

Garrocq, Carlos A.
Project Engineer
Aerospace Corporation
2350 El Segundo Blvd.
El Segundo, CA 90245

Gasery, J. Larry
Manager, QA & S
NASA/NSTL
1100 Rue Corton
Slidell, LA 70458

Geer, Dwight A.
Staff Engineer
Jet Propulsion Lab 198-136
4800 Oak Grove Drive
Pasadena, VA 91103

Geller, Warren
Group Engineer
Lockheed Missiles & Space Company, Inc.
1111 Lockheed Way
Sunnyvale, CA 94086

Geoffrion, R.R.
Associate Group Leader Quality Assurance
Los Alamos National Laboratory
Box 1663
Los Alamos, NM 87545

Gindorf, Thomas E.
Manager, Reliability Engineering Section 513
Jet Propulsion Laboratory Mail Stop 144-218
4800 Oak Grove Drive
Pasadena, CA 91109

Gerke, Paul D.
Manager, Missions Office
Space Station Project Office, Code KA2
NASA Johnson Space Center
Houston, TX 77058

Gibson, Tyrone
Electronic Engineer
HQ AFSC
Washington, D.C. 20001

Gist, Marilyn E.
Management Analyst
NASA
Greenbelt, MD 20771

Ginsberg, Lee J.
Contracts Manager
Martin Marietta Corporation
443 S. Niagara Street
Denver, CO 80224

Gloor, Richard D.
Director, Design Integrity and Productivity
TRW Space & Technology Group
One Space Park, E1-3006
Redondo Beach, CA 90278

Godsey, Jack C.
Director, Quality
Morton Thiokol Inc.
Huntsville Division
Huntsville, AL 35807-7501

Gomberg, Louis
Director, Satellite Programs
RCA Astro-Electronics
P.O. Box 800
Princeton, NJ 08540

Gonsalves, Andy
Manager, Product Assurance Department
Martin Marietta
P.O. Box 1681 MD70B
Vandenberg AFB, CA 93437

Goodwin, Paul F.
Principal Reliability Engineer
Fairchild Space Company
20301 Century Blvd.
Germantown, MD 20874-1181

Goudy, Joseph R.
Program Assurance Rep.
NASA Resident Office-JPL (180/801)
4800 Oak Grove Drive
Pasadena, CA 91109

Gould, R.E. (Ed)
MTS Systems Support Office
The Aerospace Corporation
P.O. Box 5068
Vandenberg AFB, CA 93437

Green, Allen T.
Technical Director
Acoustic Emission Technology Corporation
1812J. Tribute Road
Sacramento, CA 95815

Griggs, John
Chief, Safety Integration
Martin Marietta Corporation
P.O. Box 1681 MD70B
Vandenberg AFB, CA 93437

Griswold, David M.
Manager, Gov't Market Development
RCA Solid State Division
4827 N. Sepulveda Blvd.
Sherman Oaks, CA 91403

Gross, Frederick C.
Chemist/Materials Engineer
NASA Goddard Space Flight Center
Greenbelt Road Code: 313
Greenbelt, MD 20771

Haag, Dwight W.
Director, Product Assurance
Aerojet Liquid Rocket Company
9575 Oakleaf Way
Roseville, CA 95678

Hale, Clovis R.
Branch Chief
McDonnell Douglas Astronautics Co.
P.O. Box 516
St. Louis, MO 63166

Halper, Bill
Marketing Manager, Electronic CAD
Prime Computer Inc.
500 Old Connecticut Path
Framingham, MA 01701

Hamiter, Leon
Vice President
Parts Technology Inc.
211 Homewood Drive
Huntsville, AL 35801

Hamilton, Lowell R.
Manager, Data Systems
NASA-JSC
Houston, TX 77571

Hankins, Joe R.
Supervisory Aerospace Engineer
NASA-Kennedy Space Center - SF-ENG
Kennedy Space Center, FL 32899

Hannum, Richard E.
Division Manager, Product Assurance
Martin Marietta-Denver Aerospace
P.O. Box 179 Mail No. 8003
Denver, Co 80201

Harms, Robert R.
Section Head
TRW Electronics & Defense
One Space Park, M1/1409
Redondo Beach, CA 90278

Harrison, Lloyd E.
Director, Product Assurance, Space & Communications Group
Hughes Aircraft Company
P.O. Box 92919, S41/B341
Los Angeles, CA 90009

Hawkins, Gary F.
Section Manager
The Aerospace Corporation
P.O. Box 92957 M2/250
Los Angeles, CA 90009

Haycraft, Logan Jr.
Project Manager, Safety, Reliability and Quality Assurance
Boeing Aerospace Company
1300 Bay Area Blvd.
Houston, TX 77058

Heimerdinger, M. William
MTS
Rockwell International
201 N. Douglas Street
El Segundo, CA 90245

Heindl, Joseph C.
Staff Manager, Design Integrity & Standards
TRW-Electronic Systems Integrity
One Space Park Bldg. E1, Room 3055
Redondo Beach, CA 90278

Heitkamp, Douglas O.
Project Manager
Hughes Aircraft Company
P.O. Box 92919
Los Angeles, CA 90009

Henson, Harrold
QA Manager
Vought Corporation
P.O. Box 225907
Dallas, TX 75265

Herkes, Leon H.
Chief, Performance Evaluation
Martin Marietta Aerospace - Denver Div., Michoud Operations
P.O. Box 29304 Dept. 3742
New Orleans, LA 70189

Herr, James C.
Chief of Process Control
General Dynamics
P.O. Box 748
Ft. Worth, TX 76102

Hess, Robert W.
Manager, Contracts Aerospace Group
General Electric
P.O. Box 8555
Philadelphia, PA 19101

Heyman, Joe S.
Head, MCIS
NASA-Langley
Hampton, VA 23665

Hill, Donald C.
Manager
Lockheed Missiles & Space Company
1111 Lockheed Way, 0/62-90, B/104
Sunnyvale, CA 94086

Hnatek, Eugene R.
Vice President & Technical Director
Viking Laboratories Inc.
440 Bernardo Avenue
Mountain View, CA 94043

Ho, Van H.
Manager, Task Analysis Office
Aerospace Corporation, STS Mission Development
2350 E. El Segundo Blvd.
El Segundo, CA 90245

Hoag, Gerald J.
Materials Engineer
AFCMD Detachmant 36
200 North Sepulveda
El Segundo, CA 92463

Hoggatt, John T.
Manager M&P
Boeing Aerospace Company
P.O. Box 3999 M.S. 8C-42
Seattle, WA 98124

Hohimer, James R.
QA Manager
Sperry Corporation
P.O. Box 21111
Phoenix, AZ 85036

Holden, Kenneth G.
MTS
The Aerospace Corporation
2350 E. El Segundo Blvd.
El Segundo, CA 90009

Holden, Robert
Group Head, Mat'ls Engineer
Grumman Aerospace
New South Road P1 12
Bethpage, NY 11714

Holness, Lenward E.
Manager, Components & Materials Engineering Lab.
Hughes Aircraft Company
P.O. Box 902 - Bldg. E1/C188
El Segundo, CA 90245

Holt, Jim W.
Manager, Quality Assurance
Pan Am World Services Inc.
National Space Technology Laboratories
NSTL Station, MS 39529

Hoover, Ron D.
Staff Engineer
TRW/E&D
2601 E. Victoria SP 405
Rancho Dominguez, CA 90220

Horn, James A.
AWACS Program Planning Manager
Boeing Aerospace Company
P.O. Box 3999 M/S 21-42
Seattle, WA 98124

Hoskinson, Charles T.
Memeber of the Technical Staff
The Aerospace Corporation
2350 E. El Segundo Blvd.
El Segundo, CA 90245

Houghton, William W.
Chief, Corrective Action Control Center
Martin Marietta Aerospace
P.O. Box 1681 Mail No. MX60X
Vandenberg AFB, CA 93437

Howard, Donald D.
Manager, Quality Assurance
Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, CA 91109

Hughey, Kenneth R.
Senior Project Engineer
Hughes Aircraft Company Space & Communications Group
P.O. Box 92919 Bldg. S-33 MS/C373
Los Angeles, CA 90278

Hurley, Harv C.
Manager, Quality Assurance
IBM Corporation
Bodle Hill Road
Owego, NY 13827

Hutchens, Jr. Spencer
Senior Vice President
Intertek Services Corporation
930 Indian Peak Road
Rolling Hills, CA 90274

Huseonica, William F.
Manager, Shuttle Ops Analysis
NASA/JFK Space Center
1004 Seminole Drive
Rockledge, FL 32955

Hybner, Daniel P.
Director, SSD Quality Assurance
Aerojet Electro Systems
1100 W. Hollyvale Street
Azusa, CA 91702

Jenkins, Jon E.
Senior Engineer
Northrop
4326 W. 142nd Street
Hawthorne, CA 90250

Jensen, Randall W.
Chief Scientist Lab
Hughes Aircraft Company
1700 E. Imperial Highway, S50/X342
El Segundo, CA 90254

Jones, Harold C.
Senior Advisory Engineer
Westinghouse Electric Corporation
P.O. Box 1524
Baltimore, MD 21203

Jones, Robert W.
Manager, Advanced Programs
Hughes Aircraft Company, Technology Support Div.
2000 E. El Segundo Blvd., EO El/C184
El Segundo, CA 90245

Johnson, John E.
Director, Central Product Assurance
Harris Gov't Systems
P.O. Box 37
Melbourne, FL 32901

Johnston, Larry J.
Manager - QA Management Services
Vought Corporation
P.O. Box 225907
Dallas, TX 75265

Johnstone, Harry M.
Special Assistant to Project Manager
Pan Am World Services Inc.
National Space Technology Laboratories
NSTL Station, MS 39529

Jorgensen, James L.
Division Supervisor
Sandia National Laboratories
P.O. Box 5800 - Div 2115
Albuquerque, NM 87185

Juergens, Raymond J.
Branch Chief - Technology
McDonnell Aircraft Company
P.O. Box 516
St. Louis, MO 63166

Karpen, Robert C.
Manager, Parts and Materials
National Aeronautics & Space Administration
600 Independence Avenue, S.W. (Code DR)
Washington, D.C. 20546

Kazerman, Stanley
Product Assurance Manager
GTE Strategic Systems Division
1 Research Drive
Westborough, MA 01581

Keeney, Jackson H.
IUS System Assurance Manager
Boeing Aerospace Company/IUS System Assurance
(2-3948)
2825 94th Avenue NE
Bellevue, WA 98004

Kegarise, Ronald J.
Quality Assurance Manager
ALCOA
P.O. Box 3567
Davenport, IA 52808

Keiner, William L.
Head, Test Technology Office
Naval Surface Weapons Center
Code F305
Dahlgren, VA 22448

Kelton, George T.
Advisory Programmer
IBM/FSD
1299 Orleans Drive
Sunnyvale, CA 94086

Keyes, Ralph E. (Gene)
Safety Chairman Manned Flight Awareness Panel
NASA, Johnson Space Center
Code: NS-3
Houston, TX 77058

Kilburg, Richard F.
Engineering Specialist, Sr.
General Dynamics
P.O. Box 748, MZ 6449
Ft. Worth, TX 76116

Klehman, Barney
Senior Contrac Reviewer
USAF Space Division
P.O. Box 92960 Worldway Postal Center
Los Angeles, CA 90503

Kline, Richard L.
Director, Space Station Programs
Grumman Aerospace Corporation
A09-25
Bethpage, NY 11714

Kopania, Alvin A.
Director
The Aerospace Corporation
P.O. Box 92957
Los Angeles, CA 90009

Kordsmeier, Jr., Norbert H.
Manager, Materials & Processes Engineering
Lockheed Missiles & Space Company
Dept. 62-92 Bldg. 104
P.O. Box 504
Sunnyvale, CA 94022

Kosztyo, Robert L.
Developmetn Engineering Manager
IBM Corporation
Bodle Hill Road
Owego, NY 13827

Kozak, Joseph J.
Manager, DSCS Manufacturing
General Electric Company
P.O. Box 8555
Philadelphia, PA 19101

Krausz, Alfred
Manager for Technology
TRW Space & Technology Group
One Space Park Drive (R4/2090)
Redondo Beach, CA 90278

Krueger, Joe
Procurement Analyst
The Aerospace Corporation
P.O. Box 92960 Worldway Postal Center
Los Angeles, CA 90009

Kuzma, Joseph M.
Quality Control Manager
Hemet Casting Company
760 W. Acacia Avenue
Hemet, CA 92343

Ladyman, W. "Bill" W.
PMP Manager
Vought Corporation
P.O. Box 225907
Dallas, TX 75265

Lam, Albert Y.
Senior Engineer
Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, CA 91109

Lambert, Richard B.
Manager, Quality/Reliability Assurance
Signercs
4130 S. Market Court
Sacramento, CA 95834

Landers, Ray
Hardware Functional Manager
TRW
One Space Park
Redondo Beach, CA 90278

Lange, Henry N.
Director, Quality Assurance
McDonnell Douglas Astro Company
5301 Bolsa Avenue
Huntington Beach, CA 92647

LaRose, Domenic V.
Manager, Quality Management Systems
Avco Systems Division
201 Lowell Street
Wilmington, MA 01887

Laube, Robert B.
MTS
Aerospace Corporation
P.O. Box 92957 MS M4/907
Los Angeles, CA 90266

Lawton, J.W.
Senior QA Engineer
Vought Corporation
P.O. Box 225907
Dallas, TX 75265

Lekven, Carl M.
Project Engineer
The Aerospace Corporation
2350 East El Segundo Blvd.
El Segundo, CA 90245

Leonard, Peter A.
Project Manager Mail Code: D/344, 041, F093
Rockwell International
12214 Lakewood Blvd.
Downey, CA 90241

Letsch, Ernst R.
Systems Director, System Effectiveness
The Aerospace Corporation
2350 E. El Segundo Blvd.
El Segundo, CA 90009

Levine, Joseph H.
Chief, Reliability Division
NASA/JSC
Mail Code NB
Houston, TX 77058

Licker, S. Robert
Manager, Quality Control
ITT Gilfillan
7821 Orion Avenue
Van Nuys, CA 91409

Lieggi, Nickolas
Sr. Staff Engineer
Hughes Aircraft Company
P.O. Box 902
El Segundo, CA 90245

Lindsey, E. Glen
Project Product Integrity Manager
TRW Electroncis & Defense Operations Support
Group
One Space Park
Redondo Beach, CA 90278

Linn, Robert D.
Manager, Q&RA Multi-Division Operations
Westinghouse Defense
P.O. Box 746 MS T-152
Baltimore, MD 21203

Lloyd, Bette J.
Division Productivity Coordinator
Ford Aerospace & Communicaitons Corporation
3939 Fabian Way
Palo Alto, CA 94303

Lloyd, John T.
Director, Acquisition Management
Lockheed Missiles & Space Co., Inc.
P.O. Box 504
Sunnyvale, CA 94086

Longway III, Jay
Chief, Software Quality Engineering
Martin Marietta
P.O. Box 1681 MD70B
Vandenberg AFB, CA 93437

Loveless, Richard M.
ATS Advanced Design Manager
Lockheed-California Company-MIL Design Support
Eng. Division
D/77-40, Bldg. 90-2
P.O. Box 551
Burbank, CA 91520

Low, Henry
Director, Electronic Systems Integrity
TRW Inc., Electronic Systems Group
One Space Park
Redondo Beach, CA 90278

Lubofsky, Marvin
Senior Engineering Specialist
Aerospace Corporation
2350 E. El Segundo Blvd.
El Segundo, CA 90245

Lutolf, Robert E.
General Manager-California Division
Syscon Corporation
440 Santa Clara Street
Vallejo, CA 94590

Lutz, Gerald E.
Vice President, Quality Assurance
Fairchild Communications & Electronics Company
20301 Century Blvd.
Germantown, MD 20874-1182

McAdams, Jerry L.
Vice President, Planning and Analysis
Maritz Inc.
1355 North Highway Drive
Fenton, MO 63026

McCann, Ray
Senior Software Engineer
Prime Computer Inc.
500 Old Connecticut Path
Farmingham, MA 01701

McCoy, Thomas J.
Account Manager
Maritz Motivation Company
12720 Hillcrest Road, #501
Dallas, TX 75230

McDermott, John F.
Vice President, Product Integrity
Pratt & Whitney Aircraft-Gov't Products Div.
P.O. Box 2691
West Palm Beach, FL 33402

McDermott, Thomas C.
Vice President, Quality & Reliability Assurance
Rockwell International Corporation
2230 East Imperial Highway
El Segundo, CA 90245

McGee, George W.
Director, Product Assurance
Martin Marietta Aerospace
P.O. Box 179, Mail No. 9600
Denver, CO 80235

McGowan, M.A.
Member of Technical Staff
The Aerospace Corporation
P.O. Box 92957 M/C M4-996
Los Angeles, CA 90009

McGuirk, Joseph B.
Components Applications Engineer
Honeywell Inc., Space & Strategic Avionics Div.
13350 U.S. Highway 19, South M.S. 813-4A
Clearwater, FL 33546

McKissick, Jr., John
Manager, Computer Software Reliability & Q.A.
General Electric Company
Bldg. 2, Room 14 Court Street Plant
Syracuse, NY 13221

McLaughlin, Clyde W.
Product Assurance Engineer
Hughes Aircraft Company
P.O. Box 3310
Fullerton, CA 92634

McMahon, Paul E.
Manager, Advanced Market Programs
Celanese
26 Main Street
Chatham, NJ 07092

MacDonald, Bruce A.
Division Manager
Lockheed Missiles & Space Company
P.O. Box 504
Sunnyvale, CA 94086

Macias, Armondo A.
Member Technical Staff
The Aerospace Corporation
P.O. Box 21205
Kennedy Space Center, FL 32815

Mahr, Erwin P.
Project Engineer
Aerospace Corporation
P.O. Box 92957
Los Angeles, CA 90009

Mahurin, Walker M.
Director, Western Region Office
NSIA
6033 West Century Blvd., Suite 400
Los Angeles, CA 90045

Malouf, Jubert G.
Engineer-Specialist
Northrop Corporation
2301 W. 120th Street
Hawthorne, CA 90250

Mancino, Vincent J.
Manager, Reliability Engineering
RCA Astro-Electronics
P.O. Box 800
Princeton, NJ 08540

Mara, Paul V.
Vice President, Technical
The Aluminum Association Inc.
818 Connecticut Avenue, N.W.
Washington, D.C. 20006

Marcus, Stanley I.
Manager
Rockwell International Corporation
12214 Lakewood Blvd.
Downey, CA 90241

Martin, A.D.
BAC Director of Quality Assurance
Boeing Aerospace Company
P.O. Box 3999 Mail Stop IF-34
Seattle, WA 98124

Martin, James P.
Mil/Arms Marketing Manager
National Semiconductor
2900 Semiconductor Drive
Santa Clara, CA 95066

Martindale, Richard A.
Test & Evaluation Manager
Shuttle Activation Task Force
Space & Missile Test Organization
Vandenberg AFB, CA 93437

Martinez, Joe
QA Program Manager
Vought Corporation
P.O. Box 225907
Dallas, TX 75265

Mason, Perry R.
Head, Parts Branch
NASA/Goddard Space Flight Center
Code 311, Greenbelt Road
Greenbelt, MD 20771

Masters, Tom F.
Computer Systems S.
DoD, NSA
ATTN: N32
Ft. George G. Meade, MD 20755

Masterson, Dennis P.
Manager, Metallurgy and Quality Assurance
Alcoa Forging Division-Cleveland Works
1600 Harvard Avenue
Cleveland, OH 44105

Mayben, James E.
Director Product Assurance
General Dynamics Fort Worth Division
P.O. Box 748 MZ 6610
Ft. Worth, TX 76101

Mazzanti, Edward E.
Deputy Director, Directorate of Quality Assurance
Air Force Contract Management Division
Air Force Systems Command
Kirtland AFB, NM 87117

McCurry, James W.
Project Manager
TRW Space & Technology Group
One Space Park (SNTG/1476)
Redondo Beach, CA 90278

Medina, Darlene L.
Motivation Administrator
Mail Code: D/340, 041, FC02
Rockwell International-Shuttle Orbiter Division
12214 Lakewood Blvd.
Downey, CA 90241

Medina, Jack A.
Director, Quality Assurance
Lefiell Mfg. Company
13700 Firestone Blvd.
Santa Fe Springs, CA 90670

Medler, E. H.
Staff Engineer
TRW
One Space Park
Redondo Beach, CA 90278

Melcher, John H.
Manager, Systems Verification Dept.
TRW
One Space Park
Redondo Beach, CA 90278

Migneault, Gerard E.
AST
NASA/Langley Research Center
Hampton, VA 23665

Mignot, John A.
Staff Development & Education Coordinator
The Aerospace Corporation
P.O. Box 92957
Los Angeles, CA 90009

Miller, Gordon R.
Manager, Spacecraft
Aerospace Corporation
2350 E. El Segundo Blvd.
El Segundo, CA 92957

Miller, Herbert
Senior Staff Engineer
Hughes Aircraft Company
909 N. Sepulveda
Los Angeles, CA 90009

Miller, Joseph A., Jr.
Quality Assurance Supt.
U.S. Air Force/Space Division/YGC
2400 El Segundo Blvd.
El Segundo, CA 90009

Miller, Robert
Chief Engineer
RCA-Astro
P.O. Box 800
Princeton, NJ 08540

Moke, Richard A.
Manager, Payload Integration
NASA Johnson Space Center
NASA Road 1
Houston, TX 77058

Monshaw, Val R.
Manager, Product Assurance
RCA, Astro-El-ctronics
P.O. Box 800, MS 55
Princeton, NJ 08540

Moore, Don E.
Dep. Executive Director
DLA
6301 Stevenson Avenue, #701
Alexandria, VA 22304

Morey, John C.
Corporate Manager Quality Assurance
Hughes Aircraft Company
P.O. Box 92426 Bldg. 335 M/S 362
Los Angeles, CA 90009

Morrison, K.L.
Martin Marietta Aerospace
888 N. Sepulveda Blvd., Suite 400
El Segundo, CA 90245

Murine, Gerald E.
President
Metriqs, Inc.
390 Oak Avenue, Suite G
Carlsbad, CA 92008

Murphy, Thomas P.
Manager, Software Design & Development
RCA/Astro Electronics
P.O. Box 800
Princeton, NJ 08540

Murray, Richard P.
SRT Facilitator
Aircraft Hydro-Forming
P.O. Box 2310
Gardens, CA 90248

Myers, William A.
Vice President
Science Applications
8016 Falcon Park Street
Long Beach, CA 90808

Mynar, Wilbert
Director, Reliability & Safety
Rockwell International
12214 Lakewood Blvd.
Downey, CA 90241

Nelson, Arlyn G.
Assistant Director, Acq. Management
Lockheed
P.O. Box 504
Sunnyvale, CA 94086

Nelson, Carl E.
Manager, Systems Test Laboratory. Gov't Programs
Hughes Aircraft
901 Nash Street, Bldg. S-1, M/S D-304
El Segundo, CA 90245

Nelson, William N.
IUS-Systems Effectiveness Manager
Delco Electronics Div., GMC
6767 Hollister Avenue
Goleta, CA 93117

Netzband, Gerald O.
Director Microelectronics
Martin Marietta
1605 Lando Lane
Orlando, FL 32806

Newberry, L.M.
Manager, CSOC Systems Integration
The Aerospace Corporation
2350 E. El Segundo Blvd.
El Segundo, CA 90245

Newman, Edward H.
Consultant
E.H. Newman Consulting Services
P.O. Box 1049
Capitola, CA 95010

Nicholson, Richard H.
Director QA Liaison
General Dynamics Electronics
5308 Pacific Highway
San Diego, CA 92138

Nielsen, Norman W.
General Manager, Metallurgy Quality Assurance
Alcoa
Alcoa Bldg.
Pittsburgh, PA 15219

Nix, Claude F.
Sr. Quality Engineer
Lockheed Georgia Company
86 Cobb Drive
Marietta, GA 30067

Nunemaker, Robert R.
NASA/AMES Research Center
Moffett Field, CA 94035

Oertel, John G.
Space Station Ground Operations Manager
NASA/KSC/PT
John F. Kennedy Space Center
Kennedy Space Center, FL 32899

O'Leary, Brian T.
Senior Scientist
Science Applications Inc.
2615 Pacific Coast Highway, #300
Hermosa Beach, CA 90254

Ona, John B.
Manager, PM&P
The Aerospace Corporation
2350 E. El Segundo Blvd.
El Segundo, CA 90009

Otera, John M.
Laboratory Manager
TRW Electronics & Defense
One Space Park, M1/1334
Redondo Beach, CA 90278

Pacesas, Joseph
Quality Assurance
Sunnyvale Air Force Station
1080 Lockheed Way
Sunnyvale, CA 94086

Papke, David
Quality Engineer
Douglas Aircraft
9503 S. Normandie
Torrance, CA 90205

Parker, Allen
Assistant Project Manager
TRW DSSG
One Space Park
Redondo Beach, CA 90278

Paulson, Lee C.
H3500 Product Manager
Honeywell Inc.
5100 Gamble Drive M/S MN54-4015
St. Louis Park, MN 55416

Pena, Antonio
Project Engineer
The Aerospace Corporation
2350 E. El Segundo Blvd.
El Segundo, CA 90009

Perry, Fred E.
Hardware Design Manager, IUS
Boeing Aerospace Company
22907 19th Avenue South
Seattle, WA 98188

Peri, David E.
Marketing Rep.
Martin Marietta
8100 E. Maplewood
Denver, CO 80111

Pesuth, Frank X.
Manager, Human Resource Development & Recognition
Honeywell
13350 US Highway 19
Clearwater, FL 33546

Peterson, Col. Peter E.
Director of Plans
USAF
Sunnyvale AFS
Sunnyvale, CA 94086

Phillipp, Thomas J.
Manager, Quality Engineering
General Dynamics, Convair
5001 Kearney Villa Road
San Diego, CA 92123

Pianin, Albert
Manager, Quality Control, Environmental & Space Systems
Hamilton Standard/UTC
Bradley Field Road
Windsor Locks, CT 06096

Pletz, James A.
Director, Quality Assurance
Morton Thiokol Inc.
P.O. Box 241
Elkton, MA 21921

Polaski, Louis J.
NASA, Ames Research Center
Mail Stop 244-16
Moffett Field, CA 94035

Poyer, Thomas H.
Manager, Test & Evaluation Rel. & QA
General Electric Company
French Road
Utica, NY 13502

Portanova, Peter L.
Systems Director, Shuttle Act & Ops Requirements
Aerospace Corporation
2350 E. El Segundo Blvd.
El Segundo, CA 90245

Porter, William E.
Manager, Data Systems Resource Management
General Electric Company
P.O. Box 8555 M1108
Philadelphia, PA 19101

Powell, Robert M.
Vice President & Asst. General Manager
Space Systems Division
Lockheed Missiles & Space Company
P.O. Box 504
Sunnyvale, CA 94086

Pritchard, E. Brian
Special Asst. Director for Space
NASA
Langley Research Center
Hampton, VA 23665

Profumo, Victor J.
Member of the Technical Staff
Aerospace Corporation MS M1-120
P.O. Box 92957
Los Angeles, CA 90009

Quong, Harry
Manager, Reliability & Quality Assurance
NASA
600 Independence Avenue, SW (Code DR)
Washington, D.C. 20546

Rajcic, Richard A.
Manager, Quality Assurance
Honeywell Inc.
2600 Ridgway Parkway MN17-1565
Minneapolis, MN 55440

Ralph, Kris
Account Manager
Narmaco Materials Inc.
1440 N, Kramer
Anaheim, CA 92806

Ramroth, William G.
Vice President, Quality Assurance & Safety
Morton Thiokol Inc.
P.O. Box 524 Mail Stop 900
Brigham City, UT 84302

Rassa, Robert C.
Manager, Programs
Westinghouse Electric Corp., Electrical Systems Division
1111 Schilling Road Mail Stop 7885
Hunt Valley, MD 21031

Redwine, Samuel T.
Member of Technical Staff
Mitre Corporation
1820 Dolley Madison Blvd.
McLean, VA 22101

Reese, Edward H.
Manager, Ground System Engineering
General Electric Company
Bldg. A Room 23A24
King of Prussia, PA 19406

Refalo, Frank
Lockheed Missiles & Space Company
1111 Lockheed Way
Sunnyvale, CA 94086

Reinke, John W.
Program Manager
ELDEC Corporation
P.O. Box 100
Lynnwood, WA 98036

Reynolds, William R.
Associate Director, Management Science & Engineering Directorate
NASA/MSFC
2607 Pentolope Drive
Huntsville, AL 35803

Rickman, Frank K.
Manager, Advance Program Labs. for Data Processing
Hughes Aircraft Company
1700 East Imperial Highway, S50/X338
El Segundo, CA 90254

Rideout, Walter C.
Director, Gov't Practices
IBM Corporation

Bethesda, MD

Roberts, James A.
QA Manager
JPL
4800 Oak Grove Drive
Pasadena, CA 91103

Roberts, Larry H.
President
Roberts Consultant Service
4324 A. Via Marina
Marina Del Rey, CA 90291

Roberts, Roy W.
Manager, Electronic Development
Lockheed Space Systems
1111 Lockheed Way
Sunnyvale, CA 94086

Rodney, George A.
Director, Mission Success
Martin Marietta Aerospace
P.O. Box 29304
New Orleans, LA 70189

Rogers, Norris
Manager, QA Engineering
Vought Corporation
P.O. Box 225907
Dallas, TX 75265

Romero, James M.
Program Management
NASA HQ
600 Independence Avenue, S.W.
Washington, D.C. 20546

Rosenblatt, Nathan
Engineering Specialist
The Aerospace Corporation
P.O. Box 92957 M1-113
Los Angeles, CA 90009

Rumpza, William R.
Part Materials and Pricess Deputy Manager
Boeing Company
P.O. Box 3999
Seattle, WA 98124

Rucker, John L.
Project Engineer
The Aerospace Corporation
2350 E. El Segundo Blvd.
El Segundo, CA 90245

Ruck, Robert L.
Senior Engineer
Martin Marietta Corporation
P.O. Box 1681 MD70B
Vandenberg AFB, CA 93437

Russell, William L.
Staff Engineer, Sr.
Lockheed Missiles & Space Company
P.O. Box 504
Sunnyvale, CA 94086

Saemisch, M.K.
Martin Marietta Aerospace
P.O. Box 1681 MD70B
Vandenberg AFB, CA 93437

Sanborn, Morgan W.
Director, Programs Dev.
Rockwell
P.O. Box 2476
Seal Beach, CA 90740

Sandifer, J. Paul
Research Scientist
Lockheed California Company
P.O. Box 551 D/74-71 B/204 P/2
Burbank, CA 91520

Sapp, Richard S.
Director, Reliability & Quality Assurance Div.
NASA
600 Independence Avenue, SW (Code DR)
Washington, D.C. 20546

Scandlyn, Paul R.
Sr. Contracts Negotiator
Hughes Aircraft Company SCG
P.O. Box 92919
Los Angeles, CA 90009

Schisano, William C.
Manager, Qaulity Engineering
Chemical Systems/UTC
P.O. Box 358
Sunnyvale, CA 95037

Schooler, Lanny D.
Parts Control Program Manager
Martin Marietta Aerospace
P.O. Box 179
Denver, CO 80201

Schrock, Sherman R.
Manager Space Stations
Martin Marietta Aerospace
P.O. Box 179
Denver, CO 80201

Scott, W. Richard
Technical Group Supervisor
NASA/JPL
4800 Oak Grove Drive M/S T-1180
Pasadena, CA 91109

Seabourn, Melvin C.
Staff Member
Los Alamos National Laboratory
P.O. Box 1663 Group/MST-5 M/S G730
Los Alamos, NM 87545

Seldmeier, Wilmer O.
McDonnell Douglas Corporation
P.O. Box 516, Dept. 852B, Bldg. 220
St. Louis, MO 63166

Seta, 1LT Don S.
Parts Manager
Space Division, USAF
Los Angeles AFS
Los Angeles, CA 90009

Sharp, Richard A.
Staff Engineer
Hughes SECG
901 Nash Street
El Segundo, CA 90245

Sharples, Robert E.
Manager, Space Station Project
TRW
One Space Park
Redondo Beach, CA 90278

Shaw, Wilfred H.
Manager, Assurance Systems
TRW Operations & Support Group
One Space Park
Redondo Beach, CA 90278

Shirk, Neil K.
Manager of Systems Test
General Electric Space Division
Philadelphia, PA 19101

Shipley, William S.
Manager, QA&R
JPL
4800 Oak Grove Drive
La Canada, CA 91011

Shrader, Melvin C.
Member, Technical Staff
Aerospace Corporation
2350 E. El Segundo Blvd.
El Segundo, CA 90245

Simmons, Mark W.
Administrator, Product Assurance
Martin Marietta Corporation
P.O. Box 29304 Dept. 3700
New Orleans, LA 70189

Sipe, Richard A.
IUS Operations Manager
Boeing Aerospace Company
P.O. Box 3999
Seattle, WA 98124

Sisk, Thomas J.
Manager, NASA Program Development
Martin Marietta Aerospace
P.O. Box 179
Denver, CO 80201

Sloan, James H.
Satellite Support Systems Engineer
USAF
P.O. Box 430 Sunnyvale AFS
Sunnyvale, CA 94086

Smith, Claude D.
Quality Engineer Specialist, Sr.
Aerojet Electro Systems
1100 W. Hollyvale Street
Azusa, CA 91702

Smith, Dorsey B.
Sr. Staff Engineer
Lockheed Space Operations Company
1100 West Laurel Street
Lompoc, CA 93436

Somers, Mark J.
Software R&QA Engineer
General Electric Company
P.O. Box 4840, FRP-1-D1
Syracuse, NY 13221

Sonnemann, Harry
Deputy Chief Engineer (Sys. Eng. & Tech.)
NASA
600 Independence Avenue, S.W.
Washington, D.C. 20546

Sparks, Clyde K.
Manufacturing Engineering Manager, IUS Program
Boeing Aerospace Company
P.O. Box 3999
Seattle, WA 98124

Sparks, Richard H.
Manager, Systems Design & Verification
TRW Inc.
One Space Park
Redondo Beach, CA 90278

Spehar, LTC Michael A.
Chief, Mission Assurance SATAF
USAF
SATAF
Vandenberg AFB, CA 93437

Stahl, Robert C.
MTS, System Support Office
The Aerospace Corporation
P.O. Box 5068
Vandenberg AFB, CA 93437

Steinbronn, Otto O.
Program Manager, Space Station
General Dynamics Donvair
5001 Kearny Villa Road
San Diego, CA 92123

Steinhauer, Louis
Project Manager
TRW Space & Technology Group
One Space Park (SNTG/1476)
Redondo Beach, CA 90278

Stephan, Emile S.
NASA
Kennedy Space Center, FL 32899

Storm, Richard E.
NASA
600 Independence Avenue, S.W.
Washington, D.C. 20546

St. Pierre, J.A.
Inspection Manager
Vought Corporation
P.O. Box 225907
Dallas, TX 75265

Struble, Glenn E.
Technical Advisor
WSMC/QA
Vandenberg AFB, CA 93437

Stuckey, Wayne K.
Head, Materials Analysis Dept.
The Aerospace Corporation
P.O. Box 92957 MS M2/250
Los Angeles, CA 90009

Sunlin, William M.
Operating Center Manager
SYSCON Corporation
540 Weddell, Suite 1
Sunnyvale, CA 94086

Sutorus, Arthur E.
Qaulity Assurance Manager
USAF Ballistic Missile Office
BMO/AWQ
Norton AFB, CA 92409

Swanson, Thomas R.
President
OMNI Technology
48881 Kato Road
Fremont, CA 94539

Talley, Dorsey J.
Vice President, Quality Assurance
General Dynamics
P.O. Box 748
Ft. Worth, TX 76101

Taschler, Frank J.
Qaulity Assurance Manager
USAF SATAF
516 N. Lupine
Lompoc, CA 93436

Tedeman, Lars G.
Product Assurance & Safety Manager
European Space Agency
Noordwujk
The Netherlands

Tendick, Kedrick H.
Manager, Components & Materials Laboratories
Hughes Aircraft Company/Technology Support Div.
P.O. Box 902 Bldg. E-1, M/S C-182
El Segundo, CA 90245

Teresi, James M.
Director, Reliability & Components Data Center
The Aerospace Corporation
P.O. Box 92957
Los Angeles, CA 90016

Thiobodeau, Lionel E.
Manager, CSOC System Engineering & Planning
The Aerospace Corporation
2350 E. El Segundo Blvd. M5/611
El Segundo, CA 90245

Tilton, John H.
Manager
Lockheed Missiles & Space Company
P.O. Box 504
Sunnyvale, CA 94086

Thomas, Martin
Manager, Aerospace Division
ILC Technology
399 Java Drive
Sunnyvale, CA 94086

Tomei, Richard L.
Director, ESG Product Assurance
TRW Electronic Systems Group
One Space Park
Redondo Beach, CA 90278

Tokita, Yasuo G.
Marketing
Hercules/Graphite Fibers Dept.
P.O. Box 98
Magna, UT 84044

Tollison, Peggy S.
Supervisor Quality Circles
Avco Aerostructures
Vultee Blvd.
Nashville, TN 37202

Thompson, Philip C.
Coordinator, Systems Refinement Teams
Martin Marietta Aerospace
13800 Old Gentilly Road
New Orleans, LA 70189

Torgersen, Robert H.
System Engineer
Automation Industries
20327 Nordhoff
Chatsworth, CA 91311

Torres, Arthur V.
QA Branch Chief
NASA/JSC Downey Cal.
NASA Oakso c/o Rockwell International
12214 Lakewood Blvd.
Downey, CA 90805

Tortorich, Ralph J.
Administrator, Motivation Programs
Martin Marietta Aerospace
13800 Old Gentilly Road
New Orleans, LA 70189

Tracey, Richard E.
Corporate Manager, Reliability Assurance
Hughes Aircraft Company
P.O. Box 1042
CT/C2/B193
El Segundo, CA 90245

Trattner, Steve
Director, Control Systems
The Aerospace Corporation
2530 E. El Segundo Blvd. M5/565
El Segundo, CA 90245

Tungett, Linda L.
Sr. Personnel Representative
USBI-Huntsville UTC
P.O. Box 1626
Huntsville, AL 35807

Vary, Alex
Reseqrch Engineer
NASA-Lewis Research Center
21000 Brookpark Road
Cleveland, OH 44135

Vassallo, John R.
Manager, Program Quality Assurance
Honeywell Inc., Avionics Division
13350 U.S. Highway 19 S
Clearwater, FL 33516

Voit, Velma A.
MTS
The Aerospace Corporation
2350 E. El Segundo Blvd. M5/611
El Segundo, CA 90245

Volansky, Saul A.
Manager, Data Processing Program Office
Hughes Aircraft Company
1700 E. Imperial Highway, S50/X318
El Segundo, CA 90254

Vrountas, Theodore H.
Director of Operations
GTE-Strategic Systems Division
1 Research Drive
Westborough, MA 01581

Wade, Ernie
Principal Engineering Specialist, Systems Support Office
The Aerospace Corporation
2350 E. El Segundo Blvd.
El Segundo, CA 90245

Walker, Richard K.
Engineering Specialist
Northrop Corporation, Aircraft Division
Orgn 3521/83
One Northrop Avenue
Hawthorne, CA 90250

Weage, Frank C.
Department Manager
TRW
One Space Park, Bldg. 105/2486
Redondo Beach, CA 90274

Weathers, Shelby N.
Safety Spec.
NASA/MSFC
CS 21
Marshall Space Flight Center, AL 35812

Weebe, Robert K.E.
M.T.S.
The Aerospace Corporation
P.O. Box 92957 M.S. M4-991
Los Angeles, CA 90009

Weinrich, Albert W.
RCA Astro Electronics
318 Burd Street
Pennington, NJ 08534

Weiss, Stanley I., Dr.
Chief Engineer
NASA
600 Independence Avenue, S.W.
Washington, D.C.20546

Welch, Arthur L.
Director, Product Assurance
Martin Marietta Aerospace
P.O. Box 29304, Dept. 3700
New Orleans, LA 70189

Welch, R. Kevin
Coordinator, Mission Success
Martin Marietta Aerospace
13800 Old Gentilly Road
New Orleans, LA 70189

Wengert, David P.
Dept. Head
Product Assurance
Santa Barbara Research Center
75 Coromar Drive
Goleta, CA 93117

Westcott, Edmund J.
Assistant for Product Assurance
AFSC
Andrews AFB, MD 20331

Whilden, Richard D.C.
Vice President & General Manager
Test & Field Operations Division, Space Technology Group
TRW Electronics & Defense
One Space Park R4/2098
Redondo Beach, CA 90278

White, Kyle E.
Manager, Systems Support Office
The Aerospace Corporation
P.O. Box 5068
Vandenberg AFB, CA 93437

Wiesner, James F.
Manager, Components Engineering
Aerospace Corporation
P.O. Box 92957
Los Angeles, CA 90009

Wilker, Stanley E.
Division Contracts Manager
TRW Inc.
One Space Park
Redondo Beach, CA 90278

Williams, Edward R.
Administrator Employee Motivation Programs
Mouton/Thiokol Wasatch Div.
P.O. Box 524
Brigham City, UT 84302

Wilson, William M.
Manager, Software Engineering Assurance
NASA
600 Independence Avenue, S.W. (Code DR)
Washington, D.C. 20546

Winslow, Roger G.
Survivable Systems Development Manager
Ford Aerospace
1260 Crossman Avenue
Sunnyvale, CA 94086

Witcofski, Robert D.
Aerospace Engineer
NASA-Langley Research Center
Mail Stop 365
Hampton, VA 23665

Wolbers, Harry L.
McDonnell Douglas Astronautics Company
5301 Bolsa Avenue
Huntington Beach, CA 92647

Wolcott, John L.
Sr. Associate Principal Engineer
Harris GSSO
Box 37
Melbourne, FL 32901

Wolf, Frank J.
Assistant Division Manager
Hughes Aircraft Company-Space & Comm. Group
Bldg. S50/ M/S-X338
1700 E. Imperial Highway
El Segundo, CA 90245

Wolfe, Gene R.
Manager, Space Quality Products
Texas Instruments Inc.
P.O. Box 225621, MS470
Dallas, TX 75265

Wong, N.
Member of Technical Staff
Aerospace Corporation
2350 E. El Segundo Blvd.
El Segundo, CA 90245

Wood, Philip D.
Sr. Staff Engineer, Mission Success
Martin Marietta Aerospace
P.O. Box 29304
New Orleans, LA 70189

Woodcock, Gordon R.
Manager, Large Space Systems
Boeing Aerospace Company
P.O. Box 3999 Mail Stop 84-06
Seattle, WA 98124

Wyett, Leroy C.
Chief QA Division
DCASMA Van Nuys
6230 Van Nuys Blvd.
Van Nuys, CA 91408

Yakura, James K.
Director STS Utilization Planning
Aerospace Corporation
2350 E. El Segundo Blvd.
El Segundo, CA 90245

Yoakum, Victor E.
Senior Project Engineer
Hughes Aircraft Company/Space & Comm Group
P.O. Box 92919 Airport Station
Bldg. S33/C361
Los Angeles, CA 90009

Yolken, H. Thomas
Chief, Office of Nondestructive Evaluation
National Bureau of Standards
Physics B312
Washington, D.C. 20234

Young, Alvin F.
Member of the Technical Staff
The Aerospace Corporation
2350 E. El Segundo Blvd.
El Segundo, CA 90245

Young, Will W.
Senior Staff Engineer
Bendix
Dept. 8011 M/S 4/9
Teterboro, NJ 07608

Younger, Elmer W.
Flight Assignments Engineer
The Aerospace Corporation
2350 E. El Segundo Blvd.
El Segundo, CA 90009

Yu, Ping Y.
Engineer
NASA/KSC
DD-FED-34/ Kennedy Space Center, FL 32899

Zampino, Francis J.
Systems Director, Data Systems Modernization
Aerospace Corporation
P.O. Box 430
Sunnyvale, CA 94086

**********OUT OF SEQUENCE**********

Palmer, Russell E.
Director, Directorate of Quality Assurance
DoD, DLA, DCASR Atlanta
805 Walker Street
Marietta, GA 30060

Sullivan, James F.
Manager, Space & Defense Programs
Ford Aerospace & Communications Corporation
1260 Crossman Avenue
Sunnyvale, CA 94086

SPACE DIVISION ATTENDEES TO MISSION ASSURANCE WORKSHOP

Giovanni Bargero
Computer Specialist
U.S. Air Force/Space Division
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Donald L. Behrendt
Parts Engineer
U.S. Air Force/Space Division
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Henry S. Black
General Engineer
U.S. Air Force/Space Division
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

William R. Briggs
Director of Productivity
U.S. Air Force/Space Division
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Col. Frank S. Christy
Deputy for Contracting
U.S. Air Force/Space Division/SD/PM
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Capt. Raymond S. Cox
Test & Evaluation Engineer
U.S. Air Force/Space Division/SD/ALT
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Major Ronald B. Dodge
Test & Evaluation Engineer
U.S. Air Force/Space Division/SD/ALT
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Capt. James P. Dwyer
Manufacturing Officer
U.S. Air Force/Space Division/SD/PDM
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Richard D. Ebert
General Engineer
U.S. Air Force/Space Division
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Lt. Col. Norman R. Foley
Deputy Director of Productivity
U.S. Air Force/Space Division/SD/PD
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Major John D. Graziadei
Chief, EMC, HFE & T&E Division
U.S. Air Force/Space Division/SD/ALT
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Col. Jerry H. Hatfield
Asst. Deputy for Acquisition Logistics
U.S. Air Force/Space Division/SD/AL
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Edward S. Houston
Chief, Productivity Management Division
U.S. Air Force/Space Division
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Les Jepsen
Technical Requirement Specialist
U.S. Air Force/Space Division
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Ashraf H. Khan
General Engineer
U.S. Air Force/Space Division
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Major William Koss
Director, Computer Resources
U.S. Air Force/Space Division/SD/ALR
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

John A. Lasco
Configuration Mgmt Specialist
U.S. Air Force/Space Division
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

James C. Mahoney
Director of Operations
U.S. Air Force/Space Division
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Louis C. McKee
General Engineer
U.S. Air Force/Space Division
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

George P. Musser
Systems Engineer Mgmt Specialist
U.S. Air Force/Space Division
Worldway Postal Center
Los Angeles, CA 90009

Jack Nelson
Supv. Contract Price Analyst
U.S. Air Force/Space Division
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Jack D. Reberry
Quality Assurance Specialist
U.S. Air Force/Space Division
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Gerald D. Robinson
Actg Chief, Product Assurance Div.
U.S. Air Force/Space Division
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Lt. Col. David P. Schmarje
Chief, Manufacturing Operations Div.
U.S. Air Force/Space Division/SD/PDM
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

John B. Shook
Director, Acquisition Management
U.S. Air Force/Space Division
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Steve Soderquist
Quality Assurance Specialist
U.S. Air Force/Space Division
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Richard L. Spahn
Industrial Engineer
U.S. Air Force/Space Division
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Fred Spiegl
General Engineer
U.S. Air Force/Space Division
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Benjamin N. Theall
Reliability Engineer
U.S. Air Force/Space Division
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

Duane H. Wenberg
Electronic Engineer
U.S. Air Force/Space Division
Worldway Postal Center
P.O. Box 92960
Los Angeles, CA 90009

* U.S. GOVERNMENT PRINTING OFFICE: 1984-426-597