-PL - IN-63-CR

326643

P 101

JPL Publication 90-21

Spacecraft Health Automated Reasoning Prototype (SHARP)

A Report on SHARP and the Voyager Neptune Encounter

R. G. Martin, Editor
D. J. Atkinson
M. L. James
D. L. Lawson
H. J. Porta

August 15, 1990

NASA

National Aeronautics and
Space Administration

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

JPL Publication 90-21

Spacecraft Health Automated Reasoning Prototype (SHARP)

A Report on SHARP and the Voyager Neptune Encounter

R. G. Martin, Editor
D. J. Atkinson
M. L. James
D. L. Lawson
H. J. Porta

August 15, 1990

NASA
National Aeronautics and
Space Administration

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

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

PREFACE

The purpose of this publication is to provide a report on the development and application of the Spacecraft Health Automated Reasoning Prototype (SHARP) for the operations of the telecommunications system and link analysis functions in Voyager mission operations. The report is intended primarily for a mixed audience of technical managers and system developers in the area of mission operations automation. The report provides an overview of the design and functional description of the SHARP system as it was applied to Voyager. Some of the current problems and motivations for automation in real-time mission operations are discussed, as are the specific solutions that SHARP provides. This report is not a substitute for detailed documentation, which is being prepared separately.

The application of SHARP to Voyager telecommunications had the goal of being a proof-of-capability demonstration of artificial intelligence as applied to the problem of real-time monitoring functions in planetary mission operations. Members of the Artificial Intelligence Group of the Advanced Information Systems Section at Jet Propulsion Laboratory (JPL) sought to demonstrate and evaluate the capability in one of the planetary exploration program's most rigorous environments -- real-time operations during a planetary encounter. The encounter of the Voyager spacecraft with the planet Neptune in August 1989 provided the SHARP team with this opportunity. As part of achieving this central goal, the SHARP application effort was also required to address the issue of the design of an appropriate software system architecture for a ground-based, highly automated spacecraft monitoring system for mission operations, including methods for: (1) embedding a knowledge-based expert system for fault detection, isolation, and recovery (FDIR) within this architecture; (2) acquiring, managing, and fusing the multiple sources of information used by operations personnel; and (3) providing information-rich displays to human operators who need to exercise the capabilities of the automated system. In this regard, SHARP has provided JPL with an excellent example of how advanced artificial intelligence techniques can be smoothly integrated with a variety of conventionally programmed software modules, as well as guidance and solutions for many questions about automation in mission operations.

From a broader, operational perspective, SHARP has shown that a large set of mission operations functions in the area of real-time monitoring of spacecraft health and status can be effectively automated, with significant payoffs in the areas of safety, workforce savings, personnel productivity, and system reliability. These payoffs are discussed in the latter pages of the report. As SHARP migrates into operations and additional applications are prototyped, we expect to continue to refine and quantify the payoffs from artificial intelligence applications to mission operations.

SHARP has received widespread attention in the press, ranging from the mass news media to highly technical journals. News articles about SHARP have appeared in *Information Week Magazine*, *AI Week*, *Computerworld*, *Computers and Physics Journal*, *Federal Computer Week*, *Symbolics Symposium*, *IEEE Expert*, *IEEE Spectrum*, *Government Executive Magazine*, *Insight*, *Aerospace Engineering*, *The Washington Times*, *The New York Times*, *The Los Angeles Times*, *Nikkei Artificial Intelligence* (Japanese publication), and the *Journal of Commerce*. While this level of attention is certainly exciting, it also indicates the importance that both technologists and users are attaching to effective applications of artificial intelligence.

The SHARP application to Voyager telecommunications was conducted as part of a long-term research and development task being conducted for the National Aeronautics and Space Administration (NASA) Office of Aeronautics and Exploration Technology (OAET) Artificial Intelligence Program called "Ground Data Systems Automation". This task, which began in October 1987 has the on-going objective of developing and demonstrating automation technologies that enable and enhance multi-mission capabilities of operations ground systems for planetary spacecraft and for the Deep Space Network (DSN). Additional support for the development of the SHARP system was provided by the NASA Office of Solar System Exploration and by the NASA Telescience Testbed Program, managed by the NASA Ames Research Center.

Currently, under joint sponsorship of the NASA OAET Artificial Intelligence Program and JPL Flight Projects Support Office, a SHARP "shell" is being readied for transfer to the Space Flight Operations Center (SFOC) in the summer of 1990, where it will be used beginning in August 1990 for the Magellan spacecraft's telecommunications operations. At that time, the SHARP shell will be fully compatible with the SFOC environment, meet SFOC user interface and data interface requirements, and be in compliance with JPL software standards for flight software. This shell will be a major component of the Multimission Spacecraft Analysis System (MSAS) tool set which the JPL Flight Projects Support Office is developing. Additionally, the JPL DSN Office of Engineering is applying SHARP technology to telecommunications link performance analysis as part of the Network Operations Control Center (NOCC) upgrade, scheduled for operations in 1991.

With the success of the application for Voyager telecommunications, and follow-on tasks under way which will carry the technology forward into operational systems, we feel that the major objectives set for SHARP have been achieved and that there are now opportunities for multiple applications of artificial intelligence in planetary mission operations. This would not have been possible without the tireless efforts of the SHARP development team and the strong, continuing support we have had from individuals throughout JPL and NASA. In addition to assisting me with management of the Ground Data Systems Automation task, the SHARP development team was lead by Mark James, whose unswerving devotion to detail, the highest qualities of software workmanship, and personal programming fortitude are at the core of SHARP's success. The SHARP software developers, Dr. Harry Porta, Gaius Martin, Denise Lawson, Erann Gat, and Bruce Elgin, displayed high competence and creativity. Denise Lawson did an superb job as the lead knowledge engineer for SHARP. Harry Porta's user-interface graphics for SHARP have been especially effective in helping Telecom users operate the system. Special acknowledgement is deserved by Boyd Madsen, the supervisor of telecommunications subsystem operations for Voyager and other spacecraft, who had the vision to see what was needed, the commitment to make SHARP happen, and the patience to deal with stubborn technologists. John Carnakis, Lieu Nguyen, Ray Stagner, Ken Atkins, and Ted Bahrami made early contributions to the SHARP effort which educated us about the world of flight projects and helped us get on the right track towards a reasonable, and desirable demonstration target. Giulio Varsi, Wayne Schober, Art Zygielbaum, Dave Nichols, Mike Sander, Dave Linick, and Norm Haynes helped secure the funding for SHARP and have provided essential support and representation for us to JPL and NASA management. The entire SHARP team is also grateful to our line management, Charlie Beswick and Tom Thornton, for their rock solid support throughout the planning and execution of this task. Finally, SHARP would not have come to pass without the confidence and vision of our sponsors in NASA, Mel Montemerlo, Joe Bredekamp, and Ann Merwarth, and the support of the NASA OAET Artificial Intelligence Intercenter Working Group.

David J. Atkinson
JPL, April 3,1990

ABSTRACT

This publication provides a report on the development and application of the Spacecraft Health Automated Reasoning Prototype (SHARP) for the operations of the telecommunications system and link analysis functions in Voyager mission operations. The report provides an overview of the design and functional description of the SHARP system as it was applied to Voyager. Some of the current problems and motivations for automation in real-time mission operations are discussed, as are the specific solutions that SHARP provides.

The application of SHARP to Voyager telecommunications had the goal of being a proof-of-capability demonstration of artificial intelligence as applied to the problem of real-time monitoring functions in planetary mission operations. As part of achieving this central goal, the SHARP application effort was also required to address the issue of the design of an appropriate software system architecture for a ground-based, highly automated spacecraft monitoring system for mission operations, including methods for: (1) embedding a knowledge-based expert system for fault detection, isolation, and recovery (FDIR) within this architecture; (2) acquiring, managing, and fusing the multiple sources of information used by operations personnel; and (3) providing information-rich displays to human operators who need to exercise the capabilities of the automated system. In this regard, SHARP has provided the Jet Propulsion Laboratory with an excellent example of how advanced artificial intelligence techniques can be smoothly integrated with a variety of conventionally programmed software modules, as well as guidance and solutions for many questions about automation in mission operations.

CONTENTS

1. Introduction

The Voyager 1 and Voyager 2 spacecraft were launched from Cape Canaveral, Florida, in 1977. The technology to monitor the health and status of these probes was designed and developed in the early 1970s. This now antiquated technology, coupled with the heroic efforts of many JPL personnel over the last 13 years, has carried Voyager 2 through near-fatal catastrophic events to four of the outer planets of our solar system. Despite the spacecraft's failed radio receiver, sunlight damage to the photopolarimeter scientific instrument, and partially paralyzed scan platform (which houses Voyager's imaging system), JPL engineers have kept Voyager operational, enabling the capture and transmission of vast amounts of invaluable information and images of the Jovian, Saturnian, Uranian, and Neptunian systems.

1.1. Spacecraft Monitoring Operations Problems

During critical periods of the Voyager mission, up to 40 real-time operators are required to monitor the spacecraft's 10 subsystems on a 24-hour, 7-day-per-week schedule. This does not include the numerous subsystem and scientific instrument specialists who must constantly be available on call to handle emergencies.

Unlike the 1980s, when JPL mission operations could focus on the two Voyager spacecraft, in the coming decade there will be an increasing number of planetary exploration spacecraft flying at the same time. In addition to the Voyagers, the Galileo and Magellan spacecraft have been launched in the past year and are now on their way to Jupiter and Venus, respectively. The Ulysses, CRAF (Comet Rendezvous Asteroid Flyby), Mars Observer, and other spacecraft will follow in the next few years. To accommodate the increasing load on mission operations, JPL has established a Space Flight Operations Center (SFOC) to replace the individual mission control teams and spacecraft teams for each mission. A single, multimission flight team will operate all of the spacecraft. As more spacecraft are launched and begin to carry out their missions, SFOC will require significant advances in automation technology to support the increasing workload on operations personnel and to ensure the safety of the spacecraft.

1.2. Voyager Telecommunications Monitoring Operations

Telecommunication with the Voyager 2 spacecraft suffers from frequent anomalies and requires coordination of efforts of monitoring and diagnosis of both the spacecraft and ground telecommunications systems.

Due to cumbersome and time-consuming manual processes and obsolete technology which will be discussed in later sections, severe limitations exist on the current methods of analyzing Voyager telecommunications data. Even with the substantial improvement in computing support which is part of the new SFOC, the telecommunications area is both an operations area sorely in need of automation as well as one of the most challenging to automate.

As noted earlier, each spacecraft is monitored on a continuous basis. To enable the receipt and collection of spacecraft engineering data, JPL operates three complexes of antennas located around the world. These complexes comprise NASA's Deep Space Network (DSN). With the exception of occultations and a short gap between two of the stations (Canberra and Madrid), a spacecraft is always in view from one of these Deep Space Stations (DSS), as the complexes are called.

Four of the most important functions that are part of analysis of the telecommunications link between the spacecraft, DSN, and ground system computers at JPL are (1) the numerical estimation of telecommunications subsystem and link performance, (2) the monitoring of real-time telecommunications activity, (3) the detection of failures or degraded performance, and (4) the diagnosis, isolation, and recovery from these problems. To accomplish each of these functions, a wide variety of information must be accessed and processed manually by an operator. This information is described in later sections.

As a result of the amount of data needing to be manually processed and analyzed, there remains the potential that serious errors could occur in the monitoring of the spacecraft. This could occur because of the lack of appropriate data or the lack of a timely response to a critical situation.

In telecommunications as in other areas, the ultimate diagnosis, isolation, and recovery from failures, anomalous conditions, or degraded system performance often requires the intervention of experts who have years of specialized experience operating spacecraft subsystems (e.g., power, thermal, telecommunications). One of the most serious limitations on the current method of mission operations is the dependency upon the critical flight skills of these experts. These specialists must be on call at any time, and are frequently consulted on a daily basis. The timeliness of an expert response to a problem can be critical in saving a spacecraft. Furthermore, when the experts retire, their critical skills are lost to mission operations. The Voyager 2 spacecraft has already been flying for almost 13 years, and is expected to operate until 2018. Many future spacecraft are expected to have similar longevity. The accumulated expertise of mission operations personnel is a critical resource which should be preserved, and not recreated every time a senior engineer leaves the flight project.

1.3. The SHARP Response

To assist the flight operation teams to preserve the safety of spacecraft, to increase the productivity of the team members, and to maintain the knowledge acquired through years of flight experience, it is necessary to automate many of the current procedures and to capture the decision making process of the experts. Major challenges for these are: the elimination of manual data processing, assistance in data interpretation, and the automated real-time anomaly detection and analysis.

The Spacecraft Health Automated Reasoning Prototype (SHARP) was developed as part of an ongoing effort to apply artificial intelligence (AI) techniques to mission operations automation. The primary task for an operational SHARP system will be multi-mission monitoring and diagnosis of spacecraft and ground systems in the Space Flight Operations Center.

As tools such as SHARP are developed, they are demonstrated and evaluated in tough, operational settings to prove their performance. The Voyager 2 spacecraft was targeted for the initial demonstration of the SHARP system. The spacecraft's August 1989 encounter with the planet Neptune afforded an excellent opportunity to evaluate SHARP in a rigorous environment. The monitoring and troubleshooting of the telecommunications subsystem on board Voyager 2 and the process of real-time telecommunications link analysis were selected as the initial operations functions to be automated.

2. SHARP Overview

The Spacecraft Health Automated Reasoning Prototype (SHARP) introduces automation and artificial intelligence technologies to the process of monitoring spacecraft operations. One of the goals of SHARP is the elimination of much of the mundane processing and tedious analysis currently required of operations personnel. Another goal is to provide faster and more reliable identification of errors that occur during a spacecraft mission than is currently available. The major automated functions provided by the SHARP system include:

- Real-time anomaly detection and diagnosis;
- Visualization of channelized data and system status;
- Acquisition and centralization of operations data in a single workstation, including real-time spacecraft and ground system engineering data, sequence of events, and alarm tables;
- Real-time analysis of spacecraft performance predictions;
- Integration with specialized numerical analysis software, e.g., Fast Fourier Transforms for determining spacecraft antenna pointing accuracy.

The SHARP prototype was developed for use in the Voyager telecommunications (telecom) monitoring area. The SHARP system provides telecom personnel with an environment that allows them to have a more complete understanding of how the telecommunications link is functioning between a spacecraft and the Deep Space Stations (DSS). Deep Space Station sites are located at Goldstone, California, Madrid, Spain, and Canberra, Australia, and collectively form NASA's Deep Space Network (DSN).

The SHARP environment contains the necessary data to allow SHARP to oversee the expected behavior of the spacecraft and DSS it is monitoring. It also receives real-time data which reveal how these systems actually are performing. If the real-time data fails to correlate to the expected behavior, SHARP informs the operator responsible for the monitoring operation that an alarm condition exists. It also lists the potential causes for this anomaly and suggests what actions to take to respond to the alarm condition.

In SHARP, the automation of fault detection and diagnosis is accomplished through the use of artificial intelligence programming techniques. Artificial intelligence techniques are distributed throughout all components of the SHARP system. Artificial intelligence programming methodologies have enabled more effective automation and thorough analysis by SHARP functions.

In addition to having complete access to all of the relevant data which allow SHARP to perform its necessary analysis functions, the SHARP system contains an extensive collection of graphical displays. These displays give the operations personnel a comprehensive view of the status and dynamics of the systems that they are monitoring.

Figure 2-1 illustrates a top-level view of the SHARP system. Shown are the individual modules that comprise the system, as well as relevant components that are external to the Voyager application of SHARP. SHARP is implemented in Common LISP on a Symbolics 3650 color LISP Machine. The system is currently being ported to a Sun workstation, also running Common LISP. SHARP relies extensively on an expert system building language called STAR*TOOL, developed at JPL.

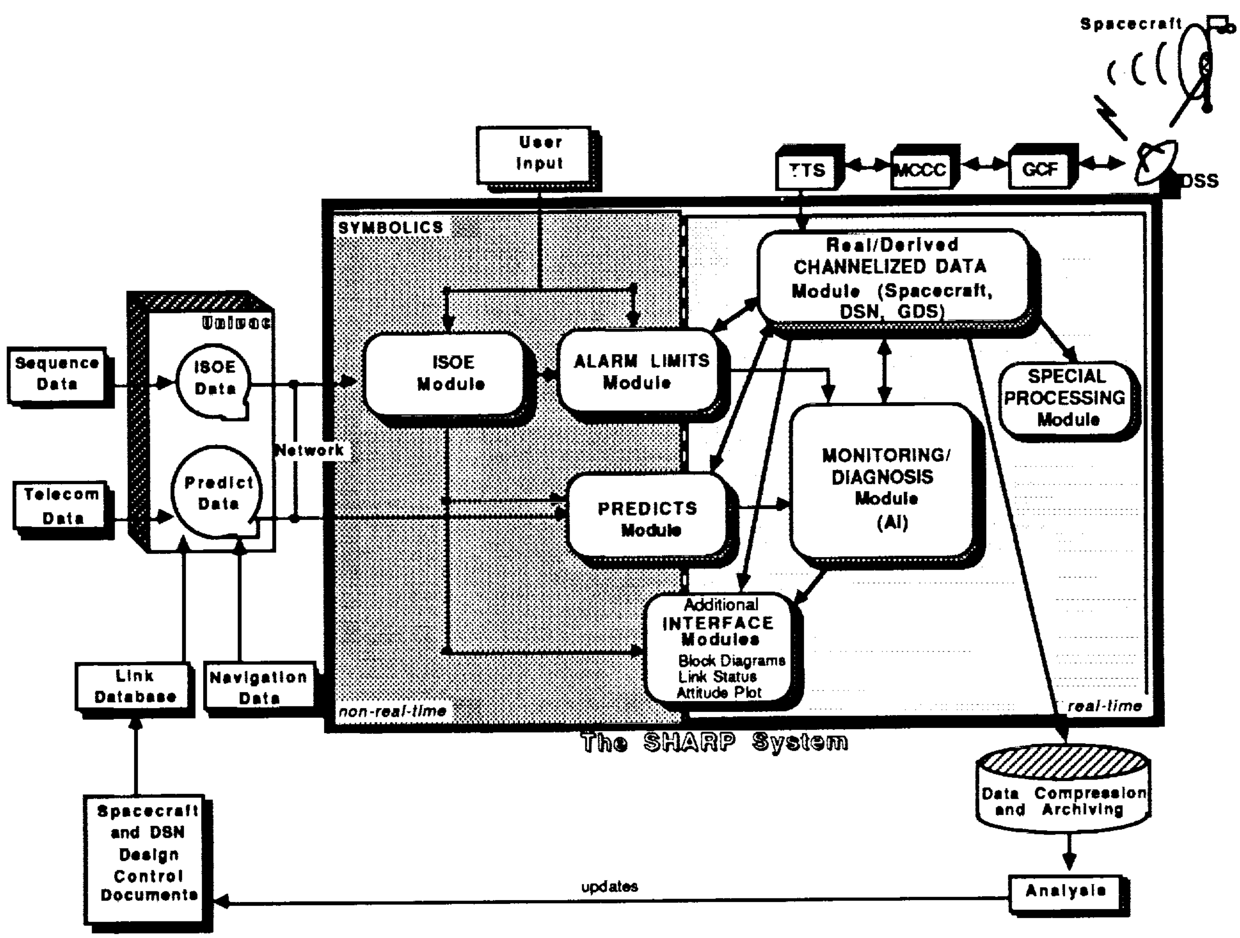


Figure 2-1 SHARP System Overview

2.1. Inputs

In order for the SHARP prototype to analyze the telecommunications link from the spacecraft through the Deep Space Network and ultimately to the computers at JPL, a wide variety of information and data must be accessed and processed. Some data and knowledge are acquired before operational use and are stored in databases or are encoded within the SHARP program. Other data are collected in real time during operational use.

The following gives a short description of the data. The procedures used for acquiring, processing, and storing the data are covered in Section 3. Figure 2-2 gives a summary the source of each type of data, where it is encoded in SHARP, and who has the capability to modify the data.

Data	Source	Location in Sharp	Modifiable by
Predicts	Univac Text File	Database	N/A
ISOE	Univac Text File	Database	User
Channelized Data	Real Time	Database	N/A
Alarm Limits	Domain Expert	Data Tables	User
Rules	Domain Expert	Code	Programmer

Figure 2-2 SHARP Input Data

Predicts

These are predictions of values for spacecraft and Deep Space Network station parameters based on numerical models of system performance.

Integrated Sequence of Events (ISOE)

This is a time-ordered sequence of scheduled spacecraft and DSS activity. Examples of these include: specifications regarding timing and the transmitter power and frequency at the DSS for uplink commands being sent to the spacecraft, when the spacecraft is performing maneuvers, the state of the receivers and transmitters on the spacecraft, etc. Last minute corrections, additions, or deletions to these events can be made by the operator.

Channelized Data

This is real-time engineering data that contain information regarding the status of systems both on board the spacecraft and at the DSSs.

Alarm Limits

These are minimum and maximum values that specify the boundaries of the nominal values for engineering data. Error conditions, i.e., alarm states, occur when the engineering data exceed these limits. Changes to these limits can be made by the operator in response to planned spacecraft state changes as well as unforeseen changes in the behavior of the systems being monitored.

Rules

These contain knowledge of how the various systems that SHARP is monitoring should behave and interact. These rules use that information and the data described above to determine the existence of either simple or complex error conditions, and to give hypotheses regarding the causes of these errors.

2.2. Processing

There are several kinds of processing which occur within the SHARP system. These are:

- Capturing the real-time channelized data and storing it in a database,
- Using conventional and artificial intelligence techniques to analyze the real-time data for the occurrence of alarm conditions,
- Determining probable causes and responses to alarm conditions, and
- Displaying data and management of the various displays.

SHARP runs in a multi-processing environment with interactions among the different kinds of processing.

The processing is covered in more detail in Section 4.

2.3. Outputs

The primary purposes of the SHARP system are to provide a better operational environment for monitoring various spacecraft and DSS systems, to warn an operator if there is an alarm condition in any of the systems that SHARP is monitoring, to assist the operator in determining the cause of an alarm, and to suggest actions to take in response to an alarm. The means used to communicate this information consist of a number of displays, each designed to handle a specific task. Secondarily, SHARP also stores the results of some of its processing in log files and databases. SHARP is not an autonomous system in that it takes no control actions of its own. A human is "in the loop" at all times.

The displays of SHARP provide access to data resident in the system, provide an interface to allow the operator to change that data when appropriate, and dynamically indicate alarms and what systems are affected by them. The displays are summarized in Figure 2-3 and the text that follows. A more detailed description of the displays can be found in Section 5.

Display Name	Functionality
Alarm Warnings Attitude and Articulation Control Block Diagrams Channelized Data Plots Fast Fourier Transform Link Status Alarm Meters	Indicate System Status and Alarm Conditions
Alarm Limit Tables Integrated Sequence of Events	Display Data and Accept User Modifications
Channelized Data Monitoring Predicts Alarm History	Show Data or SHARP Status Information

Figure 2-3 SHARP Displays

Alarm History

This is a scrolling text display that shows all warnings given by SHARP during a session.

Alarm Limit Tables

This is a tabular display that presents the operator with the values of the alarm limits. The operator can modify alarm limits and other parameters using this interface.

Alarm Meters

This display is a collection of meters that shows which channels are currently in alarm, and gives the time of the last data value that caused the alarm.

Alarm Warnings

This is not actually a display, but a pop-up window that will appear whenever a warning message is given. This window will appear regardless of the primary SHARP display being viewed by the operator.

Attitude and Articulation Control

This display combines spacecraft motion parameters (pitch, yaw, and roll) and records spacecraft movement over time in an iconic display of spacecraft attitude.

Block Diagrams

This collection of displays based upon functional schematic block diagrams allows the user to view the current, instantaneous operational state of components of the communication path from the spacecraft through the DSSs.

Channelized Data Monitoring

This display is primarily for the SHARP system implementors to allow them to examine activity regarding the real-time data acquisition and database transactions.

Channelized Data Plots

This collection of displays gives graphical views of the collected real-time channelized data plotted in a variety of formats.

Fast Fourier Transform

This display shows the result of a Fast Fourier Transform (FFT) computation on a selected engineering data channel. An FFT is computed on the data values of a real-time data channel of the signal strength of the spacecraft transmissions received by a DSS. By being able to examine the relative magnitude of one component of this FFT, the SHARP system helps to determine when there is an antenna pointing problem at the DSS that is tracing the spacecraft.

Integrated Sequence of Events

This display allows the operators to search for and review summaries of selected events from the Integrated Sequence of Events (ISOE) that affect telecom operations. It also allows them to update the SHARP ISOE database to reflect the actual real-time commands sent to the spacecraft.

Link Status

This display shows DSS antenna assignments, uplink and downlink transmissions, and data rates for those transmissions. This display also helps the operators to predict when data quality may be degraded. In contrast to the block diagrams which show the instantaneous status, the Link Status display shows status over time.

Predicts

This display presents the predict information to the operator in a tabular form, and shows DSS view periods for the spacecraft in a graphical time-line format.

3. Inputs to SHARP

This section describes in more detail the types of data used by the SHARP system during the Voyager encounter, which would also be typical of data used for other subsystem and spacecraft applications. The previous section provided a summary of these data sources.

Predicts and Integrated Sequence of Events (ISOE) data were generated by other computer programs at JPL. Both the Predicts and the ISOE were available as ASCII format files. The Predicts covered a time period of six months. The ISOE data covered a period of one week. These were periodically retrieved when made available and processed for incorporation into SHARP databases.

The Alarm Limits and Diagnostic Rules were generated using information provided by the domain expert. The Channelized Data was retrieved and analyzed in real time.

3.1. Predicts

Predicts are numerical predictions of acceptable threshold values for particular spacecraft and DSS parameters that impact the performance of the telecommunications link, such as signal-to-noise ratio and antenna elevation angle. These predictions are generated for each spacecraft pass over each ground station, and can be divided into four categories: raw predicts, pass predicts, instantaneous predicts, and residual calculations.

In the present telecom monitoring environment, much of the predict calculation process is performed manually, and is both tedious and time consuming. SHARP stores the raw predicts in a database and generates the other predict information as needed in real time. Figure 3-1 gives an overview of the Predicts module within SHARP.

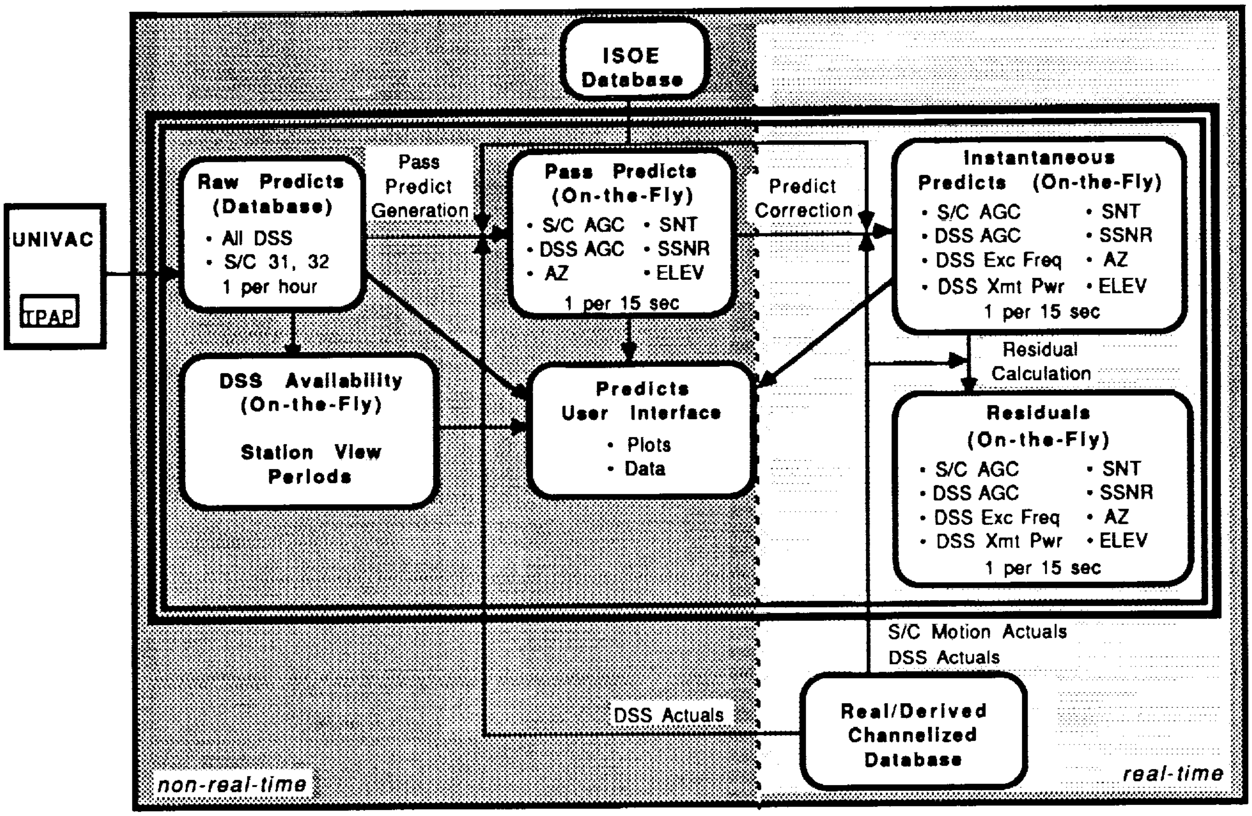


Figure 3-1 Predicts Overview

3.1.1. Raw Predicts

Raw predicts are static values which are calculated from spacecraft and DSS information. They are generated on an institutional Univac by a program called Telecommunications Performance Analysis Program (TPAP) using the following information:

- DSS center trajectory (STATRAJ) data, such as range and elevation angle. This information is derived by the Navigation team, stored on tape, and accessed for TPAP input by telecom personnel.
- Telecommunications link database information, i.e., static spacecraft and DSN parameters. This information evolves over time through such procedures as trend analysis.
- User input from telecom personnel, such as time, stations, and power levels. This information dictates the constraints for each TPAP run. For example, the user may request raw predict data for station 14 from Jan 1 - Mar 1 at high power.

Raw predicts files are generated for each antenna at a given power level for a specified number of days (typically 90, 120, 180, etc.). These files contain predict information *per hour* for each conceivable time period that the antenna can track Voyager 1 or Voyager 2 (per *half hour* during planetary encounters). The predict files are currently available to the telecommunications personnel in hardcopy form only, and are divided according to antenna size, type, and location as shown in Figure 3-2.

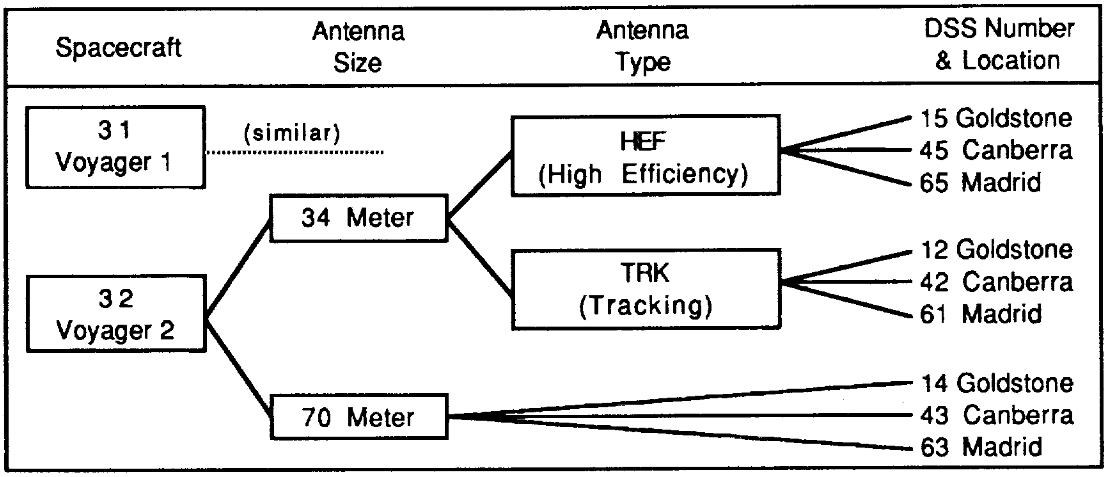


Figure 3-2 DSS Antennas

When the TPAP predict files are generated for the Voyager Spacecraft Team, the information is transferred via ethernet (using a TCP/IP based FTP utility) to the SHARP system. SHARP's raw predicts parser then processes these files and extracts the raw predicts for the predicts database.

Appendix A, Predicts, contains more information on the parsing of the raw predicts and how the other predict types are generated.

3.1.2. Pass Predicts

Pass predicts are raw prediction values tailored to a specific spacecraft pass, a viewing period of a spacecraft by a DSS. The current method of generating pass predicts involves searching the large hardcopy listings of raw predicts to find the correct spacecraft, station, time, and radio-frequency subsystem (RFS) mode. Predicts are then manually generated using a hand calculator to reflect the actual spacecraft state, and the results are manually recorded on a data sheet.

The tedious manual process for pass predict generation may take up to two hours each day, and limits calculations to one pass predict point per hour. Actual link parameters may be received every 15 seconds, leaving quite a disparity between the desired number of predictions and the incoming data.

Since the SHARP system maintains the raw predicts on-line, pass predicts can be automatically generated and interpolated to create values at 15-second intervals. All calculations are easily performed in real time and, therefore, are not stored in a database.

3.1.3. Instantaneous Predicts

Instantaneous predicts are pass predicts corrected in real time for actual spacecraft and DSS behavior, such as spacecraft pointing loss and DSS system noise temperature. The process of adjusting pass predicts is known as predict correction, or instantaneous predict generation. Some "instantaneous" predicts require no real-time adjustments, but are simply the pass predict values. Instantaneous predicts are the final predict values which are actually compared to the real data (in the form of residual calculations).

Instantaneous predicts are computed in real time within the SHARP system using the interpolated pass predict values. Again, these calculations are not stored in a database.

3.1.4. Residuals

Spacecraft and DSS residuals are difference measurements between actual telemetry values and their corresponding predicted values. Ideally, this calculation should always be 0, meaning that there is no difference between the values that were expected and the values that were actually observed.

Residual calculations are currently performed manually in real time at hourly intervals. These calculations are also performed by the Voyager Telecommunications On-line Processing System (VTOPS) program on a personal computer in the Voyager non-real-time telecom area. Residuals are examined by the computer and anomalies are flagged.

Residuals are automatically derived in real time by the SHARP system at 15-second intervals. Not only are the values alarmed, but they may be plotted as well for easy visual reference.

3.2. Integrated Sequence of Events

The Integrated Sequence of Events (ISOE) is a schedule of spacecraft activity integrated with corresponding tracking activity at the DSSs. ISOE data are used extensively throughout the spacecraft monitoring process: in prediction calculations, alarm determination, and anomaly diagnosis.

The SHARP system addresses the need to process ISOE data by acquiring ISOE information for on-line storage, processing, and viewing. Figure 3-3 shows the overall structure of SHARP's ISOE module.

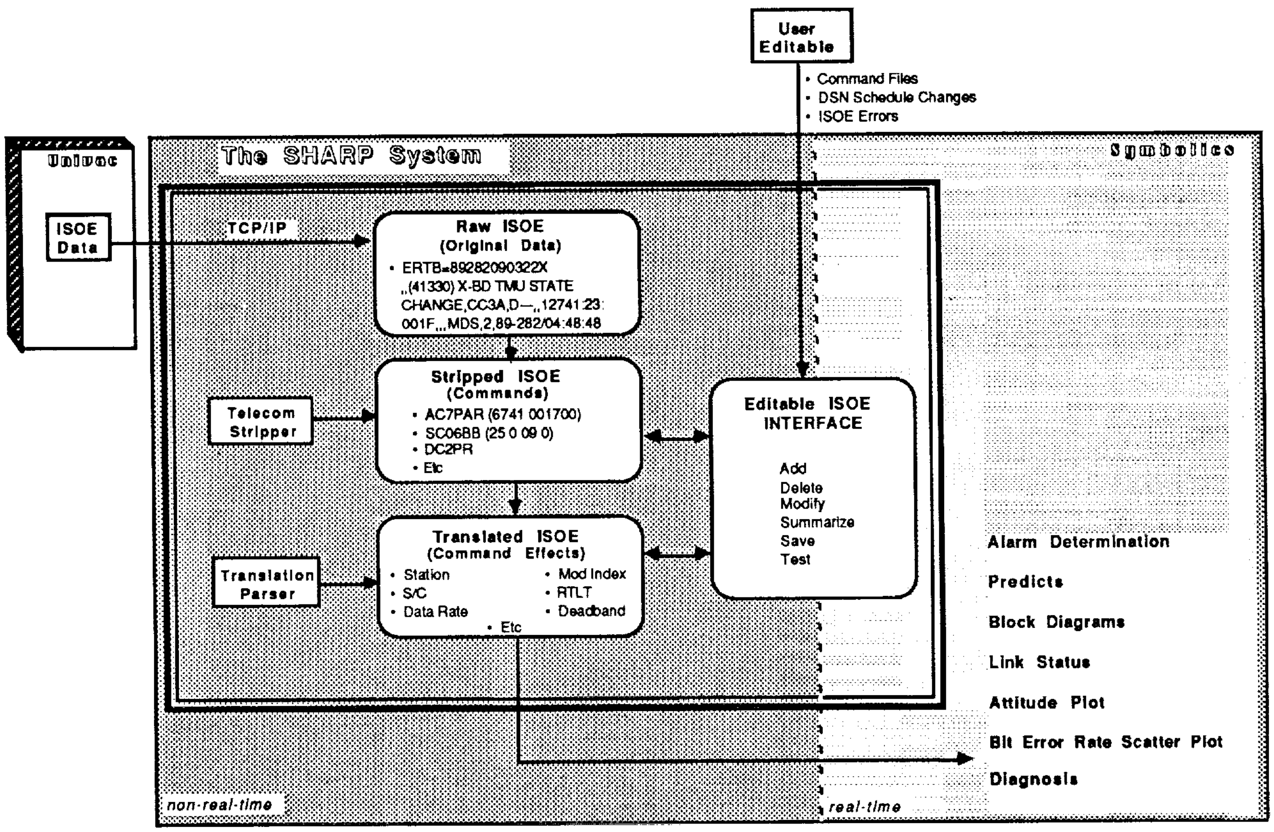


Figure 3-3 ISOE Module

In the current Voyager real-time operational environment, the Voyager ISOE, in hardcopy form (Figure 3-4), is visually scanned by the real-time operator, and telecom events manually highlighted so that pertinent telecom activity can be monitored. Modifications are made to an ISOE by issuing handwritten correction sheets to supplement the original listing (Figure 3-5).

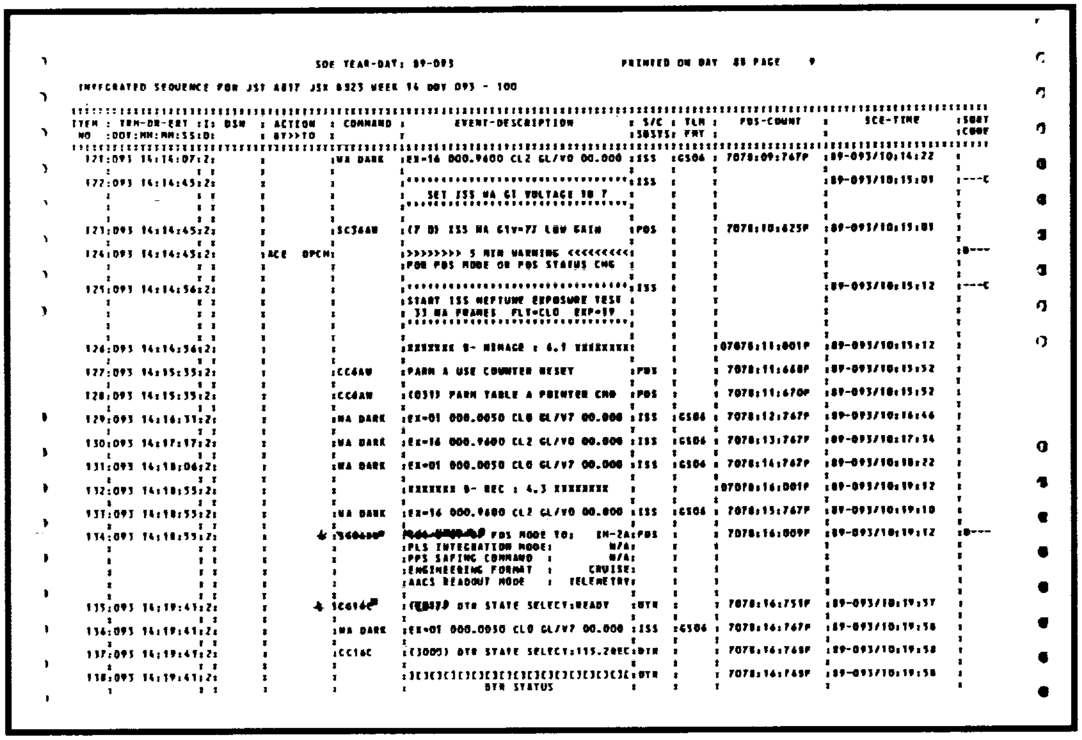


Figure 3-4 Sample Page of Voyager ISOE

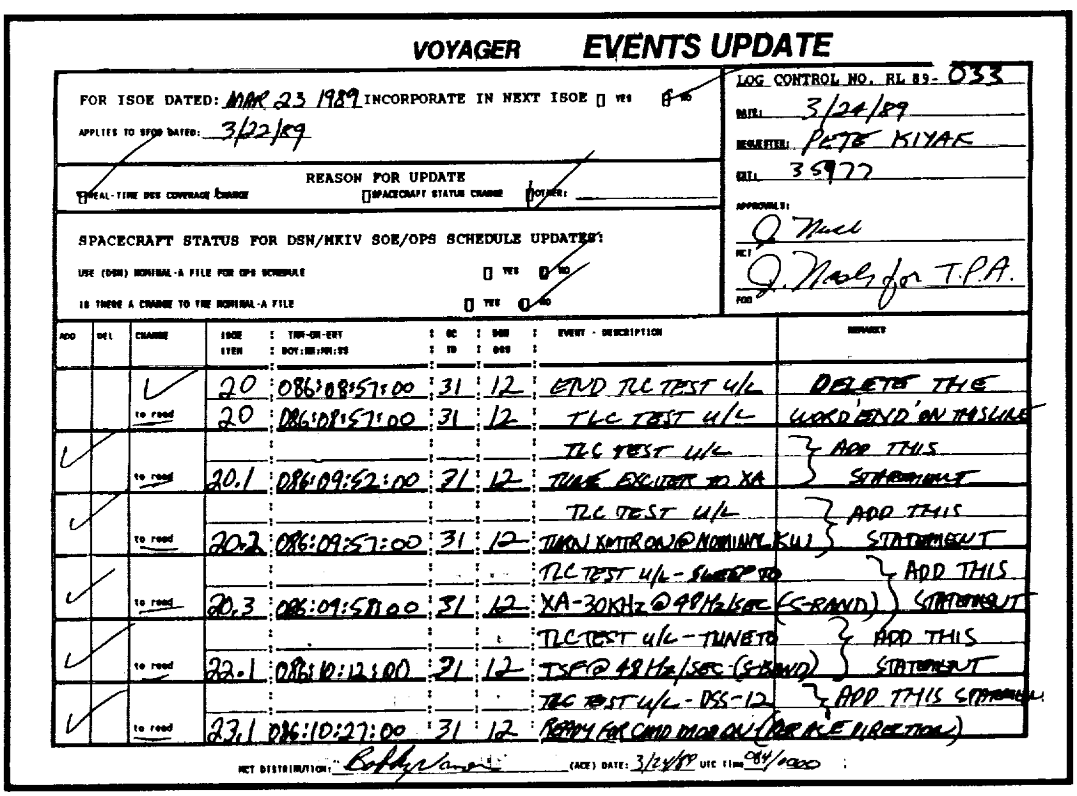


Figure 3-5 Sample Page of ISOE Corrections

ORIGINAL PAGE IS
OF POOR QUALITY

The current method of handling Voyager ISOEs prompts several complications. During periods of heightened activity, such as a planetary encounter, it is possible for a single telecom event to be embedded among several pages of another subsystem's events in the ISOE. It is easy to overlook events, and sometimes the ISOE is so extensive that operators do not even attempt to scan it. Rather, they rely on an unofficial graphical sequence hardcopied product, the Spacecraft Flight Operations Schedule (SFOS), to monitor critical events (Figure 3-6). Such use of the SFOS, which is manually highlighted with a marker to indicate changes, creates problems when users unknowingly do not reference the latest activity modifications.

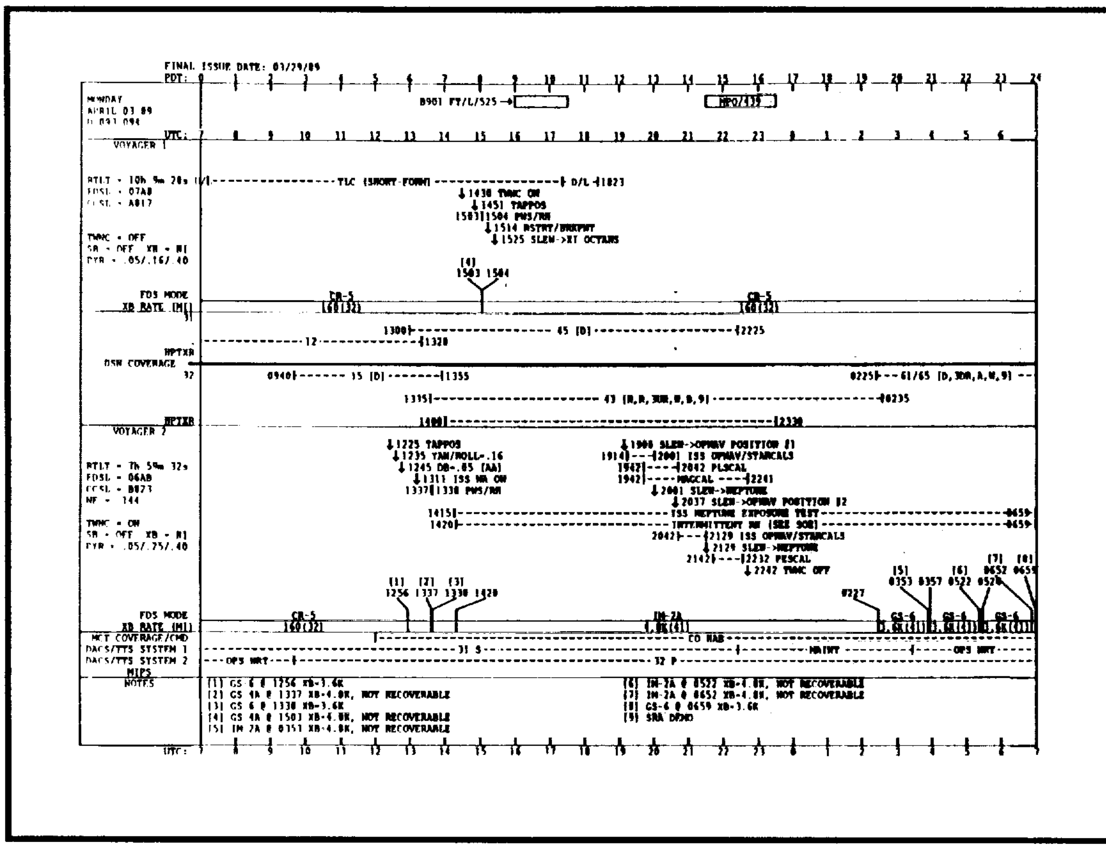


Figure 3-6 Sample Page of Spacecraft Flight Operations Schedule

The processing done by SHARP on the ISOE can be divided into five general areas for discussion:

- acquiring the unabridged ISOE,
- reducing its content with the telecom stripper,
- modifying the stripper output with the translation parser,
- storing and accessing information in the ISOE database, and
- operator interaction with the ISOE user interface.

The first four areas will be covered in this section. The user interface will be described in a later section.

3.2.1. Unabridged ISOE

The unabridged ISOE files are text files that contain the complete time-ordered sequence of spacecraft and DSS activities. An ISOE file is generated approximately weekly by a program that resides on an institutional Univac computer and is maintained by the Space Flight Operations Section of JPL. The files for January 1989 through August 1989 varied in length from 90 kilobytes to 2 megabytes. These ISOE files form the basis for the subsequent subsystem stripper program. Figure 3-7 illustrates an unabridged ISOE file.

The parsing of the unabridged ISOE files and the generation of the ISOE database is a two-step process. First an intermediate text file is produced containing the subsystem (e.g., telecom) events that have been extracted from each ISOE file. Each event has also had extraneous information eliminated from it. After this intermediate text file is generated, the second step converts the text file to a computer readable file which contains the events sorted by time and by event type. The intermediate text file, also referred to here as the stripped ISOE, is available for display and editing by the telecom operators.

```
 ERTB=89136193946X  ,,(7220) MISC LOGIC CNTL MEM DUMP,AC7MLD,,, 8374:59:
C                   602F,,,AACS,2,89-136/15:42:12
 ERTB=89136193956X  ,,(03 1 09 0) FDS MODE TO:   GS-4A,SC06BB,D---,, 837
C                   5:00:009F,,,FDS,2,89-136/15:42:24
N                   ,,PLS INTEGRATION MODE:        N/A
N                   ,,PPS SAFING COMMAND  :        N/A
N                   ,,ENGINEERING FORMAT  :  ENCOUNTER
N                   ,,AACS READOUT MODE   :  TELEMETRY
 ERTB=89136194045X  ACE  OPCH,,[[[[[[[[[[[[[[[[[[]]]]]]]]]]]]]]]]]],,D---,,
C                   8375:01:001F,GS-4A,,FDS,2,89-136/15:43:12,FDS,
N                   ,,            FDS STATUS
N                   ,,GS-4A/X4.8K/S40
N                   ,,MEM:PB/FDS09 SA/FDS18
N                   ,,[[[[[[[[[[[[[[[[[[]]]]]]]]]]]]]]]]]]
 ERTB=89136194145X  ,,--------------------------------,,---C,,,,,AACS,2,
C                   89-136/15:44:11
N                   ,,SLEW:----------------> MARK 509
N                   ,,--------------------------------
 ERTB=89136194145X  ,,XXXXXXX B- SLEW : 5.1 XXXXXXXX,,,,08375:02:218F,,
C                   ,2,89-136/15:44:13
 ERTB=89136194145X  ,,(1 12 2 083 077) EL SLEW R=LOW,AC7SPK,,, 8375:02:2
C                   19F,,,AACS,2,89-136/15:44:13
 ERTB=89136194145X  ,,(2 12 2 173 127) AZ SLEW R=LOW,AC7SPK,,, 8375:02:2
C                   20F,,,AACS,2,89-136/15:44:13
 ERTB=89136194558X  ,,EL=155.131 AZ=231.113 TS=30SEC,SLEWEND,,,08375:07:
C                   424F,,,,2,89-136/15:48:25
 SCEA=89136200518T  ,,VGR-31 OWLT IS 5H  6M  0S,,D---,,,,,,,1
 TRMB=89136214500X  ACE  OPCH,,<><><><><><><><><><><><><><><><>,,D---,,,
C                   ,,,2,,ARRE,43,49,,,
N                   ,,    END DSS-43/49 ARRAY PASS
N                   ,,<><><><><><><><><><><><><><><><>
 TRMB=89136214500X  OPCH  ACE,, LOS DSS-49 PASS-4299              ,,D---,49
C                    LOS,,,,,2,,LOS,
 TRMB=89136214935X  ,, SET DSS-49,,,49 SET,,,,,2,
 SCEB=89136221250X  ,,VGR-32 OWLT IS 3H 57M 34S,,D---,,,,,,,2,
 TRMB=89136232841X  ,, STATION 10 DEG ELEVATION POINT ,,D---,43,,,,,2
 TRMB=89136234500X  OPCH  ACE,, LOS DSS-43 PASS-4299              ,,D---,43
C                    LOS,,,,,2,,LOS,
 TRMB=89136234804X  ,, SET DSS-43,,,43 SET,,,,,2,
 TRMA=89137011500T  OPCH  ACE,, AOS DSS-61 PASS-4283 (D/L ONLY),,D---,61
C                    AOS,,,,,1,,AOS,18,
 SCEA=89137014236T  ,,VGR-31 OWLT IS 5H  6M  1S,,D---,,,,,,,1
 TRMA=89137022941T  ,, RISE DSS-15,,,15RISE,,,,,1
 ERTB=89137035734X  OPCH  ACE,,++++++++++++++++++++++++++++++++,,D---,,0
C                   8385:22:005F,GS4A,,RFS,2,89-137/00:00:00
N                   ,,            RFS STATUS
N                   ,, TWNC = OFF <--> FDS MODE = GS4A
N                   ,,S-BD PWR = ON     X-BD PWR = HI
N                   ,,     RNG = OFF         RNG = ON
N                   ,,     M.I.= 28          M.I.= 41
N                   ,,++++++++++++++++++++++++++++++++
```

Figure 3-7 Section of an Unabridged ISOE File

3.2.2. Telecommunication Events Stripper

A generic capability to extract subsystem-specific information was developed so that telecom-specific events, or any other subsystem events, could be stripped from the ISOE. This information could later be displayed to enable rapid identification of significant activities to be monitored during any particular pass. The telecom event stripper gathers from the unabridged ISOE file only that information which is critical to telecommunications analysis. This automated process replaces the task of manually highlighting the embedded telecom events.

Initially the stripper program was developed in BASIC on the Univac where the ISOE resides. It would strip out telecom data based on keyword identification. The result, a stripped ISOE file, was then transferred to the Symbolics development machine via a modem connection. Eventually, a method was developed to transfer the entire ISOE to the Symbolics machine via Ethernet (using TCP/IP FTP), and the Univac BASIC stripper program was rewritten in LISP on the Symbolics. Obtaining ISOE data directly made the ISOE module easier to maintain and manipulate. Figure 3-8 gives an example of the output of the stripper program. This is the output that would have been generated if the information of Figure 3-7 were used as the input. This example shows the amount of the ISOE data that is eliminated that is not needed for telecommunications analysis.

```
0277  136 19 39 56    SC06BB (03 1 09 0)
0278  136 19 40 45                        FDS GS4A 4.8K S40
0286  136 21 45 00 LOS 49
0290  136 23 45 00 LOS 43
0295  137 03 57 34              RFS OFF GS4A ON OFF 28 HIGH ON 41 3 57 34
```

Figure 3-8 Section of a Stripped ISOE File

3.2.3. Translation Parser

As illustrated in Figures 3-7 and 3-8, the ISOE data appears in a somewhat obscure form. However, each ISOE command represents a specific DSS or spacecraft activity. The Voyager commands list contains all spacecraft commands and their corresponding translation. SHARP's translation parser uses this commands list to translate the spacecraft commands from their original form into annotated summaries of spacecraft activity. In addition to direct translation, the parser also includes information about DSS events and information about the impact of each command, e.g., when telemetry data will be lost or degraded.

The output of the translator are LISP files that form the basis for SHARP's ISOE database. A sample of the output of the translator is shown in Figure 3-9.

```
;Parsed SOE.
(...
 (277 #.(CreateTime 136 19 39 56) SC06BB (3 1 9 0))
 (278 #.(CreateTime 136 19 40 45) FDS (GS4A 4.8K S40))
 (286 #.(CreateTime 136 21 45 0) LOS 49)
 (290 #.(CreateTime 136 23 45 0) LOS 43)
 (295 #.(CreateTime 137 3 57 34)
          RFS (OFF GS4A ON OFF 28 HIGH ON 41 #.(CreateTime 0 3 57 34)))
 ...)

;Translated SOE.
(...
 (43 (:LOS (#.(CreateTime 135 23 50 0) . 43)
           (#.(CreateTime 136 23 45 0) . 43) ...)
     (:AOS (#.(CreateTime 135 12 50 0) . 43) ...))
 ...)
 (:SC (:DTRMODE (#.(CreateTime 138 11 38 44) . :READY) ...)
      (:XBDDRIVER2 (#.(CreateTime 138 7 31 7) . :ON) ...)
      (:XBDDRIVER1 (#.(CreateTime 138 7 31 7) . :ON) ...)
      (:SUBCARFREQ&DATARATE  (#.(CreateTime 138 7 31 7)  . :NORMAL) ...)
      (:XBDDATALINE  (#.(CreateTime 138 7 31 7) . :HIGHRATE) ...)
      (:XBDSUBCARRIERFREQUENCY  (#.(CreateTime 138 7 31 7) . :HIGH) ...)
      .
      .
      (:XBDMODINDEX ... (#.(CreateTime 137 3 57 34) . 41) ...)
      (:XBDRNG ... (#.(CreateTime 137 3 57 34) . :ON) ...)
      (:XBDXMTLEVEL ... (#.(CreateTime 137 3 57 34) . :HIGH) ...)
      (:SBDMODINDEX ... (#.(CreateTime 137 3 57 34) . 28) ...)
      (:SBDRNG ... (#.(CreateTime 137 3 57 34) . :OFF) ...)
      (:SBDXMTPWR ... (#.(CreateTime 137 3 57 34) . :ON) ...)
      (:DATAMODE ...
            (#.(CreateTime 136 19 40 45) . :GS4A)
            (#.(CreateTime 137 3 57 34) . :GS4A) ...)
      (:TWNC ... (#.(CreateTime 137 3 57 34) . :OFF) ...))))
```

Figure 3-9 Sample Output of ISOE Translator Program

3.2.4. ISOE Database

The SHARP ISOE database contains the translated ISOE information, time-tagged and indexed as DSS or spacecraft events. Each event is further indexed and contains a list of associated time-value pairs.

This allows the user to index and retrieve the various individual states of the spacecraft based upon state type, time, spacecraft ID, and receiving station. This translated form of the data is used by all other modules in the SHARP system.

At any given moment in time, SHARP usually has thousands of lines of translated ISOE data loaded. This data must be searched at least seven times for every telemetry value received and every point plotted. The diagnostician may require this data to be searched hundreds of times for a typical diagnosis.

The state of the spacecraft changes very slowly over time. Some states have not changed for years. Searching years worth of ISOEs would be impossible to do and also maintain real-time processing of all data. This required a cache to be added to the facilities of the ISOE database.

The ISOE cache allows the incremental adding and deleting of facts to the ISOE database without reparsing any more than the minimal set of ISOE database lines in order to determine the new state of the spacecraft.

The ISOE cache has changed the average accessing time from an uncached ISOE request from four seconds to 600 microseconds.

3.3. Channelized Data

The third data input to SHARP, channelized data, includes real-time telemetry data from the spacecraft, tracking stations, and other relevant systems. It is collected and distributed by JPL computers, including elements of the DSN, the Ground Communication Facility (GCF), the Mission Control and Command Center (MCCC), and the Voyager Test and Telemetry System (TTS). The telemetry data is separated into channels, each containing information regarding a single system, subsystem, or component. The channelized data gives the values of hundreds of spacecraft engineering status parameters and station performance parameters. The data channels giving the spacecraft status are called engineering channels; the channels giving the DSN parameters are called the monitor channels. For the Voyager telecom application, SHARP processed just over 100 engineering and monitor channels.

Within SHARP, the channelized data is stored in a database as it is received. Also functions are automatically executed upon the receipt of the data so that an analysis of the newly acquired data can occur. Several of the functional modules of SHARP use this data and access it through the database (Figure 3-10). In the current telecom environment, the channels are plotted on black-and-white video terminals and are visually monitored to ensure that they remain within their prespecified limits.

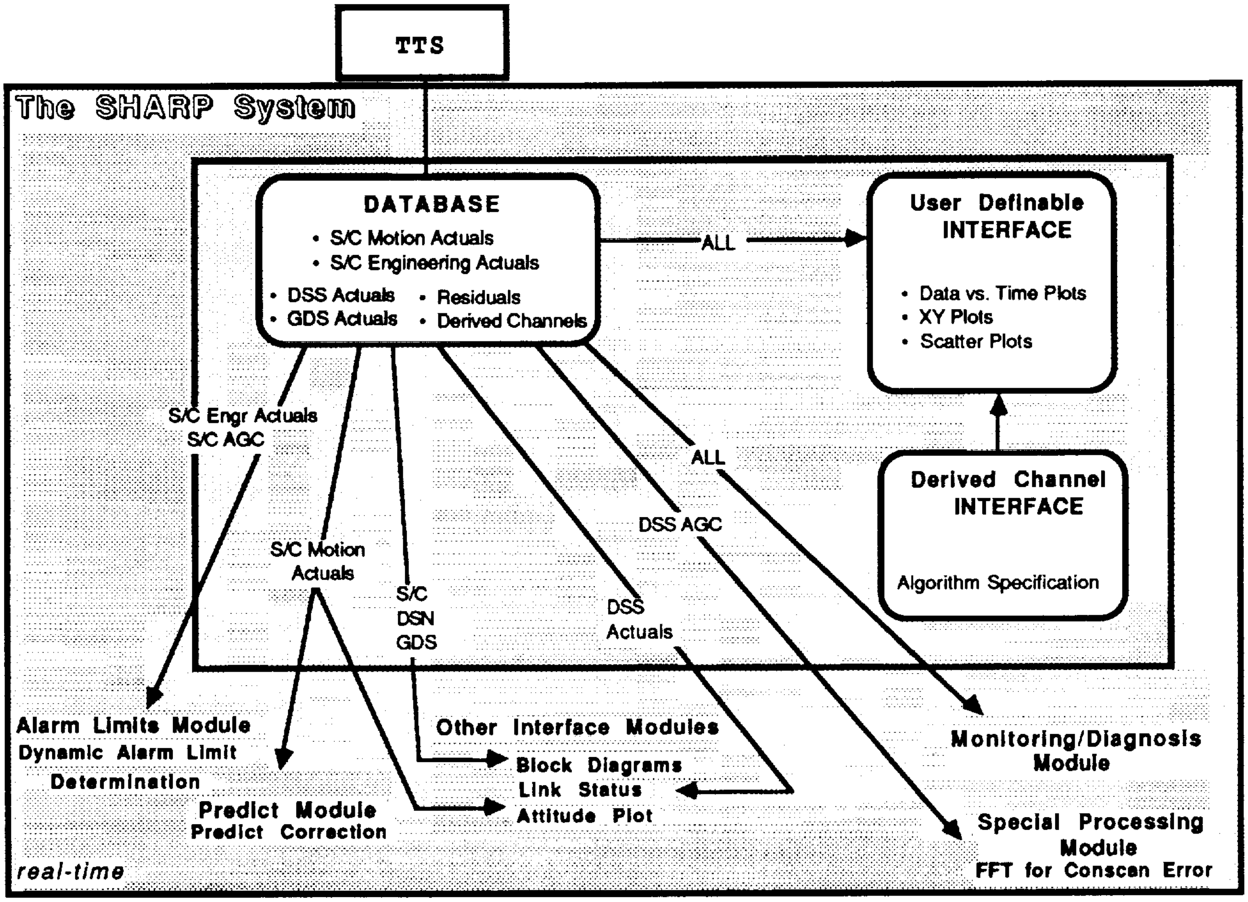


Figure 3-10 Channelized Database and Clients

3.3.1. Data Source

During the Voyager encounter, SHARP's real-time data source for telemetry data was an RS-232 data connection from the Voyager TTS computer. This was the same data that was sent to a line printer located in the Voyager real-time monitoring area. The data was received as ASCII text.

Figure 3-11 gives an example of the format of the received channelized data. The first two lines are a header line which was received each time a new page of data was output to the line printer. The only datum that was extracted from this line was the number of the day. Each of the remaining lines gave the channel number, the time (hours, minutes, seconds, and milliseconds), an abbreviated name for the data channel, and three columns of data. The first data column displays the data in engineering units. This is derived from the second column which is formatted in data units. The final column shows the amount of change in the data units from the previous receipt of data on that channel. SHARP uses the values contained in the second data column.

```
SC32 52801.41  H-63 ITR=40     F CE-40 UNDEF   W-    ITR=         F NONE
           PB=        DAY 160   P E15 PAGE 604

E1-025 00.33.17.085     RCV AGC      -9916-02        0       0
E1-146 00.33.17.085     FDS M WD        0000E   000016  000321
E1-168 00.33.29.085     FDS INDEX                   45       1
 M-776 00.33.59.798     XMTPWR                      50       2
```

Figure 3-11 Sample of Channelized Data

3.3.2. Acquiring Channelized Data

To acquire the channelized data, several tasks must be performed: establish a connection to the real-time data source, monitor the status of the data source for communication failures, automatically reconnect back to the data source in the event of a communication failure, and parse, extract, and store the ASCII data as channelized time/value pairs. These tasks are all managed automatically by the SHARP system.

3.3.3. Database Indexer and Retriever

This module includes one of the central databases for SHARP. It has the following capabilities:

- to dynamically create new areas in which to store data for a channel,
- to expunge all data associated with a channel,
- to add new data for a channel indexed by spacecraft time,
- to retrieve data indexed by spacecraft time,
- to delete individual values from a channel's database,
- to search for data based upon a channel, and
- to enumerate the data of a channel over some time interval.

Database daemons have been constructed to implement spontaneous computations. The daemons analyze the incoming data and activate function calls, such as requests to plot the data or to determine if the data indicates an alarm condition. Requests can be made to the database to trigger arbitrary activities when a complex combination of past, present, and future events occur. A wide selection of retrieval methods by time or value highlight the flexibility inherent in the database. Requests to the database can be handled serially or in parallel.

3.3.4. Database Partitioning

The central database breaks data for each channel into files that partition the data into four-hour chunks. Partitioning is required to keep empty areas in the database, time intervals where no data is stored, small. Since data is stored with a resolution of one second, for any one year there could be as many as 31,622,400 separate data values for each of the approximately 100 telemetry channels that SHARP monitors.

3.3.5. Database Open Status Reducer

While SHARP is running, it opens many partitions of the central database to access the data. There may be multiple partitions open at any one time as the different modules within SHARP may each be needing data. These partitions are left open after they are used because they may be accessed soon again and it is a costly operation to open a partition of the database. As a result, at any given moment there may be many partitions opened that have not been accessed for a long time.

The open status reducer will close off those partitions not accessed and free up any internal storage that has been allocated. Partitions that have not been accessed for 30 minutes are closed automatically.

In addition to closing partitions that have not been recently accessed, SHARP will only open a maximum of 20 partitions at any given time. Here again the least recently accessed and oldest partition will be closed to make room for a new partition that needs to be opened.

3.3.6. Automatic Database Archival and Size Reduction Module

This module scans all the telemetry channels of the database and deletes and expunges all data that are older than a certain number of days, where the number is easily set by an applications programmer. Currently data older than 14 days are eliminated. This is done to regain free space on the hard disk. It is assumed that data older than 14 days are not needed for real-time analysis, but could be retrieved from tape archives if necessary.

3.3.7. Daemon Pattern Matcher

This module allows one to associate one or more functions to be applied on the arrival of new real-time data or archived data for some time interval. When that time interval has expired, the daemon will automatically terminate.

This functionality is used:

- in the diagnostician to monitor the arrival of new data,
- in the channelized data plotter to plot new data as it arrives,
- in the link status module to perform its computations incrementally as the data arrives, and
- in the FFT module for the computation of scan errors based on certain channels.

3.4. Alarm Limits

The fourth data used within SHARP are the alarm limits.

Each time any data on a real-time channel are received by SHARP, the value of the data must be tested to determine if it may signal an alarm condition. These alarms suggest problems with the spacecraft and/or other parts of the telecommunication paths. Critical to the telecommunications link analysis are alarm limits, the threshold values for spacecraft and DSS performance measurements against which the channelized data are compared. These alarm limit values are selected according to the status of several parameters, such as the state of the spacecraft instruments and spacecraft events.

In the current telecommunications environment, however, the process to change these alarm limits is manual and must be performed in real time. The procedure is so burdensome, and occurs so often, that typically a wide threshold is selected that incorporates the entire range of parameter conditions that reflect the various spacecraft states. This creates a situation that increases the risk of undetected anomalies.

Broadened alarm limits present obvious complications. If, in fact, a component is in alarm within the broadened range, this condition may go undetected. Alternatively, alarm limits could be set to a narrower range that would be appropriate for the most common spacecraft state. However, when the spacecraft was placed into a different state, false alarms would be triggered by valid data that was outside of those limits.

One advantage of the SHARP system is its capability to use variable, dynamically selected alarm limits when examining the real-time channelized data. As the configuration of the spacecraft changes, the nominal limits on the real-time data received from the spacecraft are also changing. These alarm limits are stored into tables. Using both ISOE and channelized data, SHARP automatically computes which of the limits to use for real-time alarm determination. This is discussed more fully in Section 4.2.1.

3.4.1. Alarm Limit Tables

There are three types of alarm tables in SHARP: (1) tables for the Monitor channels which contain alarm limits for the ground station parameters, (2) tables for the Engineering channels which contain alarm limits for the spacecraft parameters, and (3) tables for the Predict Residual calculations. The values contained within these tables were specified by the domain expert, and reflect his judgement as well as the conventions accepted by the flight project.

These tables are accessible by the telecom operators in an editable format. This provides them a means to change the limits when warranted by changes in spacecraft or DSS operations or performance.

Other information is contained within the alarm tables in addition to the alarm limits. This includes a string for an abbreviated name for the channel, a list of the inclusive range of values within which the data are not in alarm, a list of the low and high limit values for hard alarms, a list of the default values for the low and high limits when plotting, and finally a string for arbitrary comments. There are exceptions to this format. An example of a portion of an alarm table is shown in Figure 3-12. The details of this table are also discussed in Section 4.2.1.

```
((:XBDXMTLEVEL :HIGH
  :SBDXMTPWR :ON
  :SBDXMTLEVEL :LOW
  :SBDXMTSELECT :SSA
  :TWNC :OFF
  :SCAGCLOCK :IN)
 (E-020 "RFS ST1" (146 146) E-020-Rfs-Status-1-Alarm-Test NIL "No Plot")
 (E-021 "RFS ST2" (207 207) E-021-RFS-Status-2-Alarm-Test NIL "No Plot")
 (E-022 "VCO C V" NIL NIL NIL "Dead; Not alarmed.")
 (E-023 "VCO F V" (130 131) (129 132) (125 135))
 ...
 (E-041 "X RFM T" (154 167) (152 169) (150 175))
 ...)
```

Figure 3-12 Portion of an Alarm Limit Table

3.5. Rules

The fifth data contained within SHARP are the rules. There are three places where rules are used.

3.5.1. Displaying System Status Rules

The first rules are found in the module that determines how to display the status of the spacecraft and ground station systems, subsystems, and components. Using information from the ISOE and the channelized data, SHARP computes the operating state of the components along the telecommunications path. A component could be turned on or off. Also, a component could be in alarm or not. This information is used to color the component or the system of which it is part in the appropriate block diagrams.

Information obtained from the domain expert was gathered into statements that described the parameters upon which the status of each component depended. For example, within the telecommunications block diagram description, the statement for the status of the Auxiliary Oscillator No. 1 appears in Figure 3-13.

```
10. AuxOsc1
     Turns GREEN when Uso = Off and SbdExcSelect = 1
   Turns WHITE when Uso = On or
               Uso = Off and SbdExcSelect = 2
   Turns RED/YELLOW when Sbd Exc 1 in alarm [E-034, E-050]
```

Figure 3-13 Statement of a Rule for Updating a Display Component

There are approximately 250 such statements that have been encoded into SHARP. They were encoded into two modules. The aspects of the rules that give information concerning the nominal operating state of the component (whether it is green for indicating it is on or white for off) were placed in the process that updates the block diagrams. The aspect that controls whether the display is to show an alarm state (red or yellow) was placed in the diagnostician module of SHARP.

3.5.2. Channelized Data Error Rules

The second type of rule determines whether or not the nominal operating state of a component corresponds to the actual state of the component. If not, then an error condition exists. Some of these determinations do not need rules but only simple numerical checks as described in the Section 4.2. However, for a number of components, more reasoning is required to understand their correct operating behavior. This knowledge is encoded within the special alarm functions that were discussed in the previous section. An example of one of these rules is shown in Figure 3-14.

For Bit 5 (Sbd SSA Pwr) of Channel E-020 (RFS Status 1):

If Bit 5 = 0 (Off) and
    Channel E-021 (RFS Status 2) Bit 3 (Sbd TWTA) = 0 (Off)
 Then SbdXmtPwr should be off.
        (Hard alarm if ISOE SbdXmtPwr differs)

If Bit 5 = 0 (Off) and
    Channel E-021 (RFS Status 2) Bit 3 (Sbd TWTA) = 1 (On)
Then SbdXmtPwr should be On and
        SbdXmtSelect should be TWTA.
        (Hard alarm if ISOE SbdXmtPwr or SbdXmtSelect differ)

If Bit 5 = 1 (On)
Then SbdXmtPwr should be On and
        SbdXmtSelect should be SSA.
        (Hard alarm if ISOE SbdXmtPwr or SbdXmtSelect differ)

Figure 3-14 Statement of a Rule for Determining Component State

This rule is contained within the processing that is done on data for channel E-020. There are approximately 62 such rules for all of the specialized alarm functions.

3.5.3. Diagnostic Rules

The final type of rules are used in diagnosing and hypothesizing the cause of error conditions. This aspect of SHARP is both the most difficult to implement and potentially the most rewarding as it requires the most detailed knowledge of the domain expert.

The process of acquiring the information from the domain expert required several iterations for many of the rules. The knowledge engineer of SHARP would interview the expert, and would then consolidate the notes from the interview into a flow chart or decision tree, a procedural representation easily understood by the domain expert. The expert would then review that and make deletions, modifications, or additions where appropriate. One important lesson learned from this exercise is that knowledge acquisition is a laborious task, even with the extremely cooperative domain expert who worked with the SHARP development team.

An example of one of the flowcharts is given in the Figure 3-15. There were approximately 20 rules that were encoded from similar flowcharts. These flowcharts were translated into a combination of rules for implementing the inferencing aspects and LISP code for implementing the algorithmic aspects. Each one of these flowcharts corresponds to a single main diagnosis.

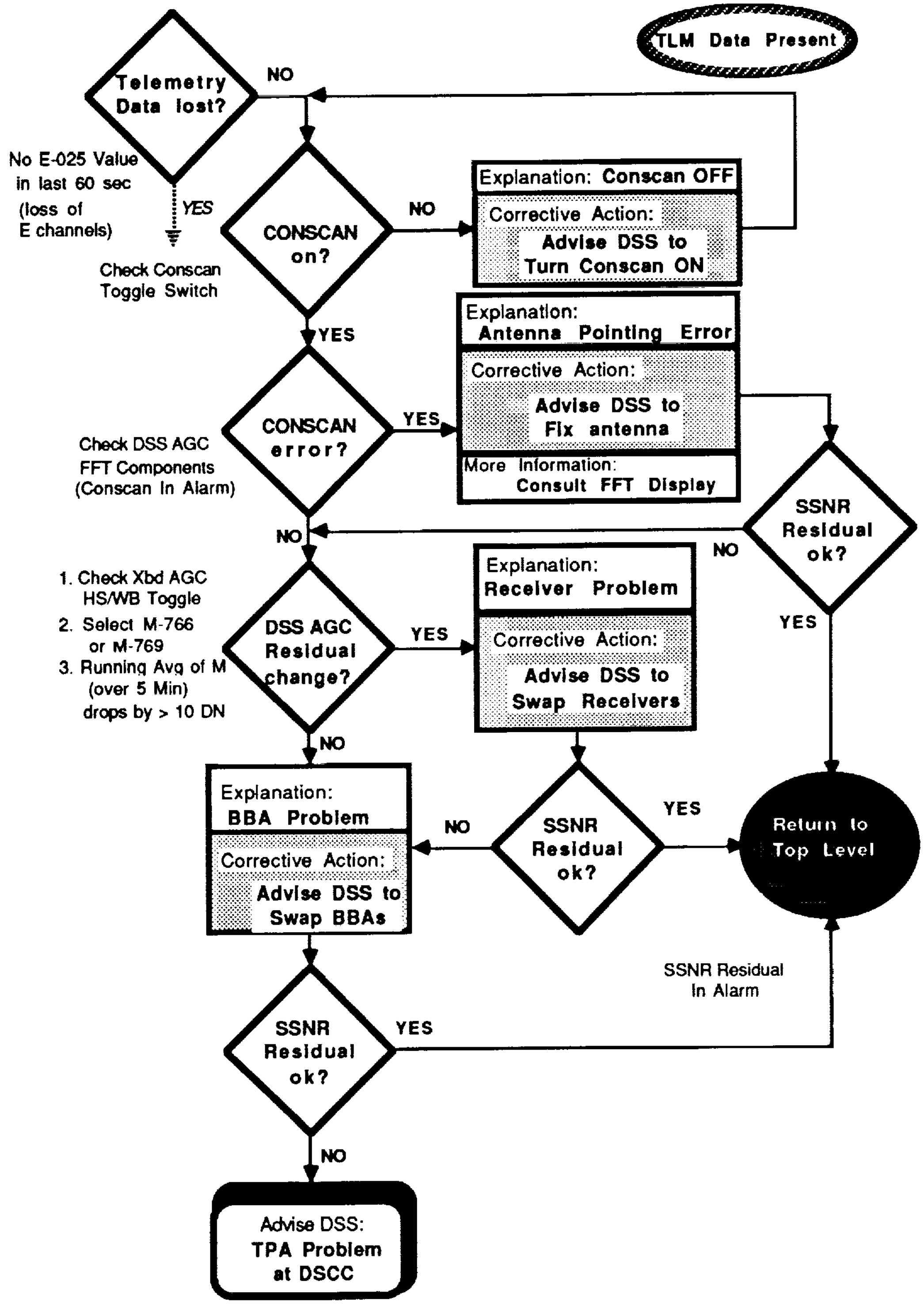


Figure 3-15 An Example of a Diagnostic Flowchart

4. Processing within SHARP

The processing within SHARP includes:

- data collection and display,
- automation of manual tasks done by telecommunication operators,
- alarm detection and diagnosis,
- specialized analysis of data, and
- user interface operations.

The primary focus of this section will be alarm determination and the way SHARP responds to alarms. Also, the Fast Fourier Transform (FFT) specialized analysis feature will be described. The means used to collect the data were covered in the previous section. The next section will discuss the displays and user interface.

4.1. Motivation for Telecommunication System Automation

When a spacecraft or DSS parameter goes into alarm, the cause must be determined. Within the current telecommunications monitoring system, the condition can often be a false alarm caused by inaccuracies precipitated by the limitations of the current system. In other instances, the alarm exists because of common problems that occur on a frequent basis, and therefore, these types of alarms are easily resolved. For actual spacecraft problems, such as the failed radio receiver, hundreds of people must be notified and put on alert to solve the emergency. Regardless of the cause or the severity of an alarm, a standard set of rules is routinely followed to determine the basis of the problems. Unfortunately, knowledge of these rules resides with a select few, and the first rule of the standard procedure is to consult the expert, even when the situation arises from a known false alarm.

The SHARP system automates much of the process of telecommunications diagnosis. By improving the monitoring process and correcting some of the inaccuracies of the current system, SHARP attempts to produce far fewer false alarms and reduce the mundane procedures required in handling the known common problems. When alarm conditions arise from any monitoring procedure within the SHARP system, such as channels in alarm, link status problems, antenna tracking errors, or attitude alarms, the information is automatically passed to the diagnostician to determine all possible causes for the anomaly.

Automating fault detection and diagnosis facilitates quicker response times to mission anomalies and more accurate conclusions. A second important benefit generated by the programming task to build this automation is the permanent record of expert knowledge in handling common as well as difficult and unusual problems.

4.2. Artificial Intelligence Diagnosis

The automation of fault detection and diagnosis is accomplished though the use of artificial intelligence (AI) programming techniques. The AI modules of SHARP are written in an expert system building language called STAR*TOOL, which is a programming language designed at JPL for use in the development of AI applications. Appendix C contains a description of the STAR*TOOL system.

AI techniques are distributed throughout all components of the SHARP system. Intelligent programming methodologies such as heuristic adaptive parsing, truth maintenance, and expert system technology enable more effective automation and thorough analysis for SHARP functions. Fault detection becomes almost immediate with a high degree of accuracy and precision, and the system quickly generates fault hypotheses. The structure of SHARP's AI module is illustrated in Figure 4-1. The following paragraphs highlight some of the principle AI concepts used in SHARP's diagnostic module. These concepts are described in detail in the general AI literature.

A blackboard architecture, provided by STAR*TOOL, serves as a uniform framework for communication within the heterogeneous multiprocess environment in which SHARP operates. Generally, when two or more processes are cooperating, they must interact in a manner more complicated than simply setting global variables. For example, they may need to add information to queues or share diagnostic context trees. The SHARP blackboard provides a standardized method of communication between multiple processes.

The diagnostic component of SHARP is composed of a hierarchical executive diagnostician coupled with cooperating and noncooperating mini-experts. Each mini-expert is responsible for the local diagnosis of a specific fault or class of faults, such as particular channels in alarm, antenna tracking conical scan errors, or loss of telemetry. A non-cooperating expert focuses only on its designated fault area, but a cooperating expert has the additional capability of searching beyond its local area to identify related faults that are likely to occur. Cooperating experts are used in situations where the identification of a particular fault cannot be made by examining a single fault class alone.

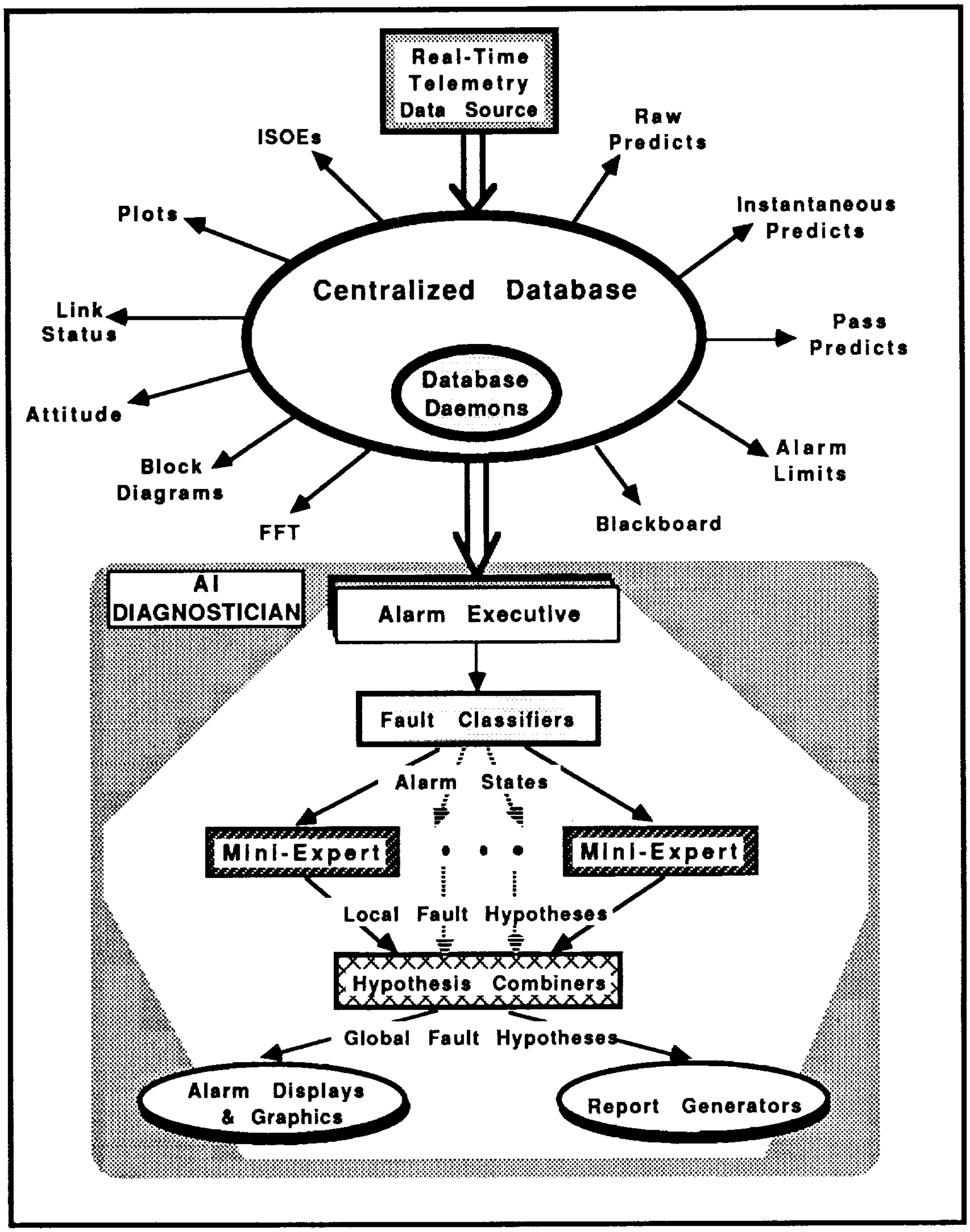


Figure 4-1 SHARP's AI Module

The executive diagnostician combines input propagated from each local diagnostician and reviews the overall situation to propose one or more fault hypotheses and recommended corrective actions. When multiple fault hypotheses are generated, the system lists all possible causes of the anomaly and ranks each according to plausibility.

If one or more of the cooperating experts fails, the executive diagnostician will continue to operate with only a reduction in the area of local diagnosis that would have been derived from the failed mini-experts. Similarly, if the executive diagnostician fails, the cooperating experts will locally diagnose the faults in isolation of multiple fault consideration.

The diagnostician is implemented in rules that execute in pseudo-parallel in pursuit of multiple hypotheses. Pseudo-parallelism is implemented in SHARP using facilities provided by STAR*TOOL, which includes parallelism as a fundamental control structure. The diagnostic rules operate in isolation of one another by executing in independent contexts provided by the STAR*TOOL memory model, and communicate through the Blackboard facility.

These contexts can be organized into a tree-like structure to represent contradictory information resulting from changes in facts or from the introduction of new or contradictory hypotheses. Facilities in the truth maintenance system handle data- and demand-driven diagnoses to ensure an appropriate balance between the persistence of hypotheses and sensitivity to new data.

The truth maintenance system constantly monitors for violations of logical consistency. For example, it performs conflict checking to maintain consistency among multiple rule firings, hypotheses, and the knowledge base, and allows the context-sensitive management of alarms through a complex response system to combinations of alarm conditions. Truth maintenance techniques also provide a variety of functions for temporal reasoning in multiple fault diagnosis.

4.2.1. Alarm Determination Module

Upon the arrival of each channelized data value of real-time telemetry, it is classified as to its state:

- nominal, where the channelized data has an acceptable value,
- soft alarm, in which case the operator should watch this channel more closely and inform others on the spacecraft team of potential problems, or
- hard alarm, in which case science data or the spacecraft are in jeopardy and the operator needs to inform others on the spacecraft team of serious problems.

To do this, SHARP invokes a function to check the datum against the channel's alarm limits. Channels that carry information about the condition of the spacecraft (engineering channels) may have alarm limits that depend upon the state of the spacecraft which is changing over time. Channels relaying information about the ground stations (monitor channels) have fixed alarm tables that do not change over time.

In order to determine the proper alarm limits to use for comparison against engineering channel data, SHARP examines the ISOE data and real-time data from a specific engineering channel (E-025, Receiver AGC) to determine the current state of the spacecraft. The conditions that define the relevant spacecraft configurations for Voyager telecommunications include:

Parameter	Values
X-Band Transmitter Power Level	High/Low
S-Band Transmitter Power Status	On/Off
S-Band Transmitter Power Level	High/Low
S-Band Transmitter Select	SSA/TWTA
Two-way Non-coherent (TWNC) status	On/Off
Automatic Gain Control (AGC) Lock status	In/Out

Given this information, a search is made through the alarm limit tables to find the proper limits to use.

The first part of this table contains the state information for the spacecraft. This is first checked to determine if the spacecraft state matches the given values.

In the majority of cases, a simple numerical check is used to determine if the data value for a channel is in alarm. In other cases, functions must be called that perform other types of analysis.

For an example of a numerical limit, refer to Figure 4-2. For the channel E-041, the data are expected to be 154 through 167. Data below 152 or above 169 are considered to be a hard alarm. Data that are 153 (i.e., between 152 and 154) or 168 are in soft alarm. If this channel is plotted, the default limit for the vertical scale is from 150 to 175.

```
((:XBDXMTLEVEL :HIGH
  :SBDXMTPWR :ON
  :SBDXMTLEVEL :LOW
  :SBDXMTSELECT :SSA
  :TWNC :OFF
  :SCAGCLOCK :IN)
 (E-020 "RFS ST1" (146 146) E-020-Rfs-Status-1-Alarm-Test NIL "No Plot")
 (E-021 "RFS ST2" (207 207) E-021-RFS-Status-2-Alarm-Test NIL "No Plot")
 (E-022 "VCO C V" NIL NIL NIL "Dead; Not alarmed.")
 (E-023 "VCO F V" (130 131) (129 132) (125 135))
 ...
 (E-041 "X RFM T" (154 167) (152 169) (150 175))
 ...)
```

Figure 4-2 Portion of an Alarm Limit Table

An example of a functional alarm test is shown for the channel E-020. If the value of the received data is not 146, the function E-020-Rfs-Status-1-Alarm-Test is called with the received value and the time it was received. The data for this channel contain information about the configuration of the spacecraft on a bit-by-bit basis. When a data value is received, it must be compared against the ISOE data to determine if there is an alarm condition. If there is an alarm condition, then this function returns both the type of the alarm (soft or hard) and a description of the alarm.

The channel E-022 is never considered to be in alarm because is has a NIL (a LISP data value) where the alarm limits should be.

A channel may have its alarm limits designated by (0 NIL). In this case, there is no upper limit for a hard alarm.

The data for the channel E-025 must also be analyzed specially. Both the state of the spacecraft and the range within which the data value lies determine the results of its processing.

4.2.2. Alarm Executive

Once it has been determined that an alarm has occurred, the alarm executive assumes responsibility. This module performs several functions:

1. If a channel is just going into alarm, notify the fault classifier of the channel's name, time, value at the time of alarm, and any initial hypothesis.

2. If the Fault Classifier cannot locate any mini-expert rules that can classify the fault, display an alarm condition message.

3. If the channel is already in alarm and the diagnosis for the current alarmed value is the same as the last diagnosis for the channel, then the new alarm has no additional effect.

4. If the severity of the alarm changes, i.e. from soft to hard or from hard to soft, then notify the user of the change in status.

5. If the new data value is not in alarm and the channel's last value was in alarm, then notify the user that the alarm condition for that channel no longer exists. Also, update all the various SHARP modules and displays to indicate that this channel is no longer in alarm.

6. Update the alarm status display to show the total number of soft and hard alarms. That display will also indicate when no alarms are currently present.

7. Update the appropriate block diagrams if the channel is related to one of the block diagram components. An example of a rule that does this is shown in Figure 4-3.

8. Create a meter, displayed on the alarm meter window, that will track and display all of the consecutively alarmed values. This meter will automatically go away when the channel is no longer in alarm.

9. Post the channel's state to the blackboard for use by other SHARP modules.

```
(DefBlockRule SC-SbdExc.Driver (Channel Time Value)
  :Channels (E-034 E-050)

  :Alarm (COND
          ((AND (EQ Channel 'E-034)
                (EQ (GetValueFromSoeOrDefault :SC :SbdExcPwr Time)
                    :On))
           (UpdateBlockDiagramIcon))

          ((EQ (GetValueFromSoeOrDefault :SC :SbdExcPwr Time)
               :Off)
           (UpdateBlockDiagramIcon))))
```

Figure 4-3 Sample Rule for Updating a Block Diagram

4.2.3. Fault Classifier

This module takes the spacecraft's current status as determined from the integrated sequence of events and any initial hypothesis generated from the alarm determination module, and asserts this information into the AI database to start its rules. The rules of the fault classifier take this information and determine which mini-expert rules can diagnose some aspect of this alarm condition.

After all applicable mini-expert rules are determined, the mini-expert Rule Interpreter is notified of the enabling set of mini-expert rules that should be tried.

This module allows the mini-expert rules to be written in a far more general manner. Each mini-expert rule does not have to specify all of the various preconditions that might pertain. Instead the mini-expert rule can contain only the information necessary to diagnose this type of problem rather than needing, in addition, a mechanism for determining all possible conditions that can cause this fault to occur.

4.2.4. Mini-Expert Rules

Rule definitions take the form of high-level descriptions of preconditions, activation and execution contexts, spacecraft state descriptions, real-time data, initial hypothesis, and diagnostic actions to be performed. Two examples of such rules can be found in Figures 4-4 and 4-5.

A rule compiler takes the high-level rule descriptions and generates Common LISP code from their descriptions. In addition, the rule compiler generates the necessary bookkeeping code to link the rule into SHARP's run-time rule environment. The run-time environment provides various debugging facilities, the blackboard for global communication among mini-experts, and other features.

Each rule is not required to diagnose a single fault. Many rules may operate in cooperation by sharing information on the blackboard. The rules locally diagnose their faults by posting their fault hypothesis on the blackboard and having the various fault combiners group this information into one or more conclusions.

```
(DefMiniExpert E-025-EU (Channel Time Value)
  :LocalVars ((t0
               (Channel-Alarm-Limits Channel Time)))

  (DiagnosticMessage "~A Alarm: Channel ~A is in alarm at time ~A.~2%~
                      Source: S/C engineering data.~2%~
                      Value: ~A~2%~
                      Explanation: There is a ~A problem.~2%~
                      Corrective Action: Advise ~A:~%~
                      ~A"
       (GetAlarmTypeString DiagnosticianAlarmType T)
       (GetFancyChannelName Channel)
       Time
       (AlarmPercentageText Value
                            t0@1
                            t0@2
                            (GetChannelDescription Channel))
       (if (>= value -100)
           "spacecraft"
           "telecommunications")
       (If (>= value -100)
           "Spacecraft Team, Mission Control,
                   and Deep Space Station (PANIC!!!)"
           "Mission Control and Deep Space Station")
       DiagnosticianAlarmArg))
```

Figure 4-4 Sample of a Simple Mini-Expert Rule

```
(DefMiniExpert DetectSSNR-ResidualInAlarm (Alarm Time Value)
  :Channel E-025
  :LocalVars (t0)

  (COND
    ((EngrDataPresentP 60.)
     (Hypothesize 'TelmentryDataPresentForSSNR-ResidualInAlarm))
    ((Not (MonitorDataPresentP 30.))
     (Hypothesize 'MonitorDataNotPresentForSSNR-ResidualInAlarm))
    ((OR (<= (SETQ t0 (FindMinusTimeValue (If X-BandAGC-SourceHighSpeedFlag
                                              'M-766
                                              'M-769)
                                          Time
                                          T))
          -1700.)
         (>= t0 -1000.))
     (Hypothesize 'DSS-RcvrOutOfLockForSSNR-ResidualInAlarm))))
```

Figure 4-5 Sample of an Inferencing Mini-Expert Rule

4.2.5. Hypothesis Combiner

This module is implemented as rules and LISP code that trigger on hypotheses that have been posted to the blackboard. It then combines related hypotheses into a single more encompassing one, eliminates redundant hypotheses, and generates multiple conclusions and ranks them from multiple unrelated hypotheses.

4.2.6. Alarm Message Manager

When the diagnosis part of the processing has been completed, the user must be informed about the results of that processing. The alarm message manager module informs the user of an alarm by updating an alarm status window and by displaying a message window giving the details of the alarm.

In addition it provides two logs of all diagnosis messages, alarm messages, error messages, status messages, or general notifications. One log is to a scrollable color-coded history window and the other is a text file that can be printed out on conventional printers. The text log file is maintained even in the event of a software or hardware failure.

4.2.7. Initial Alarm Status Calculator

Whenever SHARP is started from a cold state, such as from system booting, the current health of the spacecraft has to be determined based upon what is last known and is currently being seen. To wait for a complete set of real-time data to arrive in order to properly initialize SHARP's current state could take hours. A special capability had to be implemented to shorten this process.

This module analyzes the various SHARP databases, e.g. archived channelized data, integrated sequence of events, raw predicts, and instantaneous predicts, and initializes the diagnostician and all of SHARP's blackboards to the current state of the spacecraft. This analysis is performed in parallel with the receiving of real-time data and as real-time data replaces various assumed states based upon the last recorded archived data, SHARP updates its states accordingly.

4.3. Fast Fourier Transform

Special processing modules to perform subsystem-specific analysis are easily integrated with SHARP monitoring and diagnosis functions. For the telecommunications subsystem, a Fast Fourier Transform (FFT) of the DSS conical scanning component is performed to indicate when the antenna is going off point.

Conical scanning (conscan) by the ground stations is a technique used to maintain antenna tracking of the spacecraft. Instead of pointing the antenna directly at the spacecraft, the antenna is moved slowly in a small circle around the location. Ideally, the center of this antenna pointing direction is on the spacecraft at all times. Periodically, however, the antenna will begin drifting off target. This is a relatively common event, which currently may take a significant amount of time to detect and correct. Both spacecraft and scientific information can be permanently lost when this situation occurs.

SHARP calculates an FFT on data from the channel currently designated as the relevant AGC channel. A DSS X-Band AGC data signal can arrive on the high speed data channel (M-766) or the wide band data channel (M-769). Upon initialization of SHARP and acquisition of a connection to the channelized data source, SHARP begins to accumulate data from this channel. Data from this channel would ideally arrive approximately every 15 seconds; actually there are frequent gaps in the data. When a new datum arrives at a time which is within a certain tolerance of an integer multiple of 15 seconds after the time of the previous datum, SHARP fills in the gap, if any, by linear interpolation. When SHARP has accumulated 64 (actual and/or interpolated) data points, it calculates the FFT to resolve the AGC signal into frequency components. It compares the magnitude of the component at the conscan frequency with the average of the magnitudes of selected other components. If the conscan component's magnitude is more than twice the average of the other components' magnitudes, SHARP signals an alarm indicating that the antenna may be about to go off point.

Once it has computed an FFT on 64 data points, SHARP will compute a new FFT each time it receives an AGC datum at a time that is within the tolerance of an integer multiple of 15 seconds after the previous datum. Thus, if data were to consistently arrive every 15 seconds, SHARP would compute the first FFT approximately 16 minutes after starting, and would compute another one every 15 seconds thereafter.

5. SHARP Outputs

The SHARP system provides numerous sophisticated graphical displays for spacecraft and ground station monitoring. A comprehensive user interface has been developed to facilitate easy access to all pertinent data and analysis. An interface exists for each major module of the SHARP system. Each interface provides customized functions that allow data specific to that module to be easily accessed, viewed, and manipulated. Each SHARP module can be accessed from any other module at any time, and all displays are in color with mouse sensitivity and menu-driven commands.

As one of the primary purposes of SHARP is to provide warnings about alarm conditions within the telecommunications path, the displays have been designed to provide this information. Several graphical displays in SHARP automatically highlight alarmed events as they occur. These displays offer information ranging from the location of a problem to the probable cause of the alarm.

The operators of a SHARP system need to be able to review and, on occasion, change the data stored within the SHARP system. For example, if late changes are made to the Integrated Sequence of Events or if the alarm limits need to be changed on a particular real-time data channel, these need to be incorporated to prevent erroneous behavior within the SHARP diagnostic module. Therefore, some of the SHARP displays provide the operators the ability to display and edit portions of the data stored within the system.

Displays exist that allow the operator to review and possibly change the data within SHARP. Other displays are driven by the real-time data. These are given in the Figure 5-1.

Displays to review and change data
Alarm Limit Tables Integrated Sequence of Events Predicts
Displays driven by real-time data
Attitude and Articulation Control Alarm Meters Alarm History Alarm Warnings Block Diagrams Channelized Data Monitoring Channelized Data Plots Fast Fourier Transform Link Status

Figure 5-1 SHARP Displays

Each of these displays, except for the Channelized Data Monitoring display, has a mechanism to give alarm warnings in real time. For example, the Channelized Data Plots change the color of the graph for a channel when its data values go past the alarm limits. The mechanism used within each display will be described in more detail within the discussion of each.

There are certain features that are common to all of the configurations of the SHARP displays. Refer to Figure 5-2. (The SHARP displays use color extensively. The black and white images in this document use shading to try to give an indication of the use of color.) The top line of the display contains the JPL logo, a title bar indicating the current display, and a clock giving the Greenwich Mean Time. The next section down contains two or more menus with an alarm status window on the right. The alarm status window gives an indication of the number of soft and hard alarms currently active. The menu immediately to the left of the alarm status window, titled "Program Selection," allows the user to change the configuration of the display from one module of SHARP to another. Any other menus allow the user to modify the contents of the selected displayed configuration.

The large central portion of the display changes to display data, graphics, or plots, depending upon what configuration the user has selected. The area will contain one or more windows.

The bottom of the display contains a window that allows the developers of the SHARP system and the more experienced users to check the status of SHARP and recover from various errors. It provides an interface into the LISP environment.

Each display will be described in the remainder of this section.

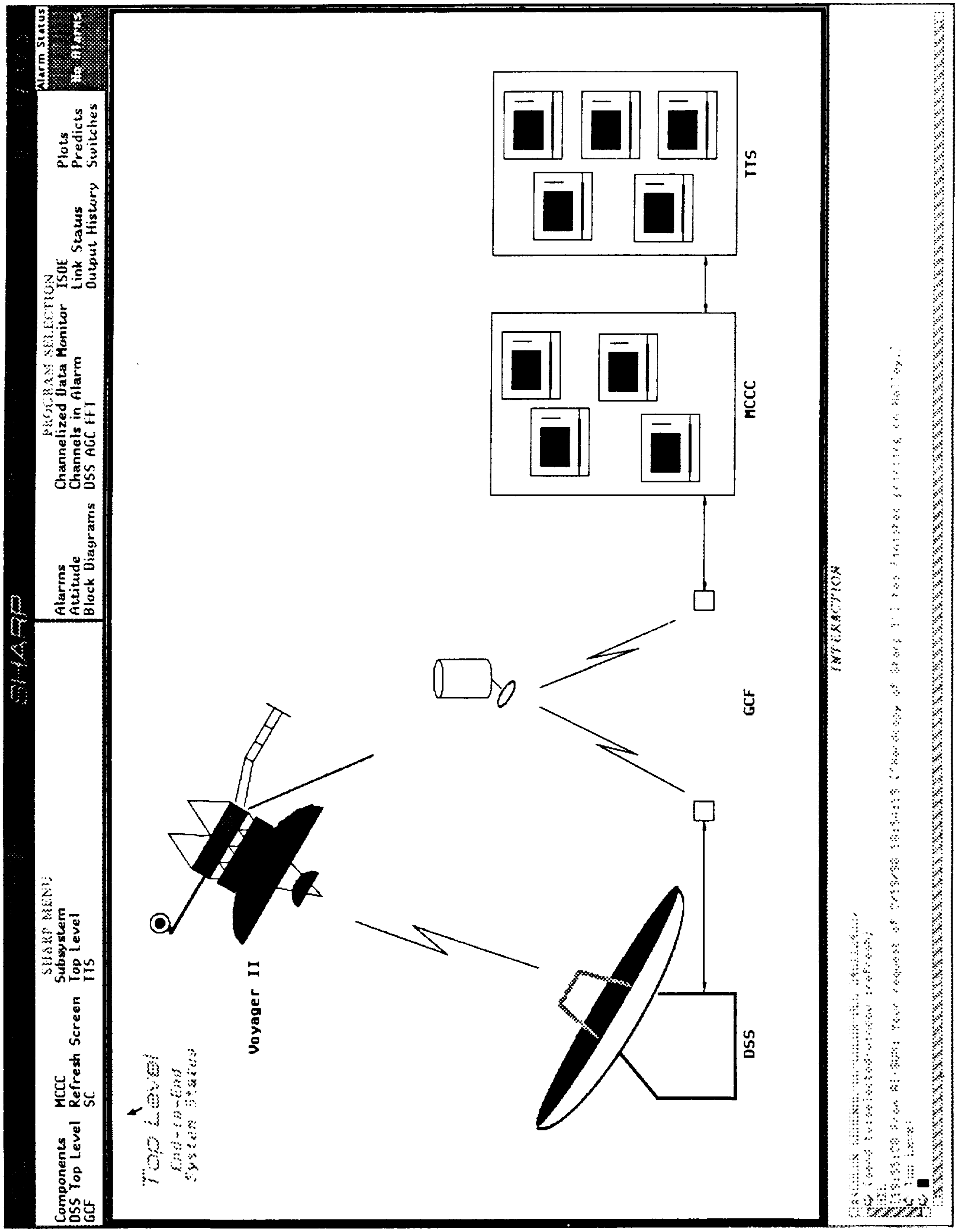


Figure 5-2 SHARP Top Level

ORIGINAL PAGE IS
OF POOR QUALITY

5.1. Alarm Limit Tables Display

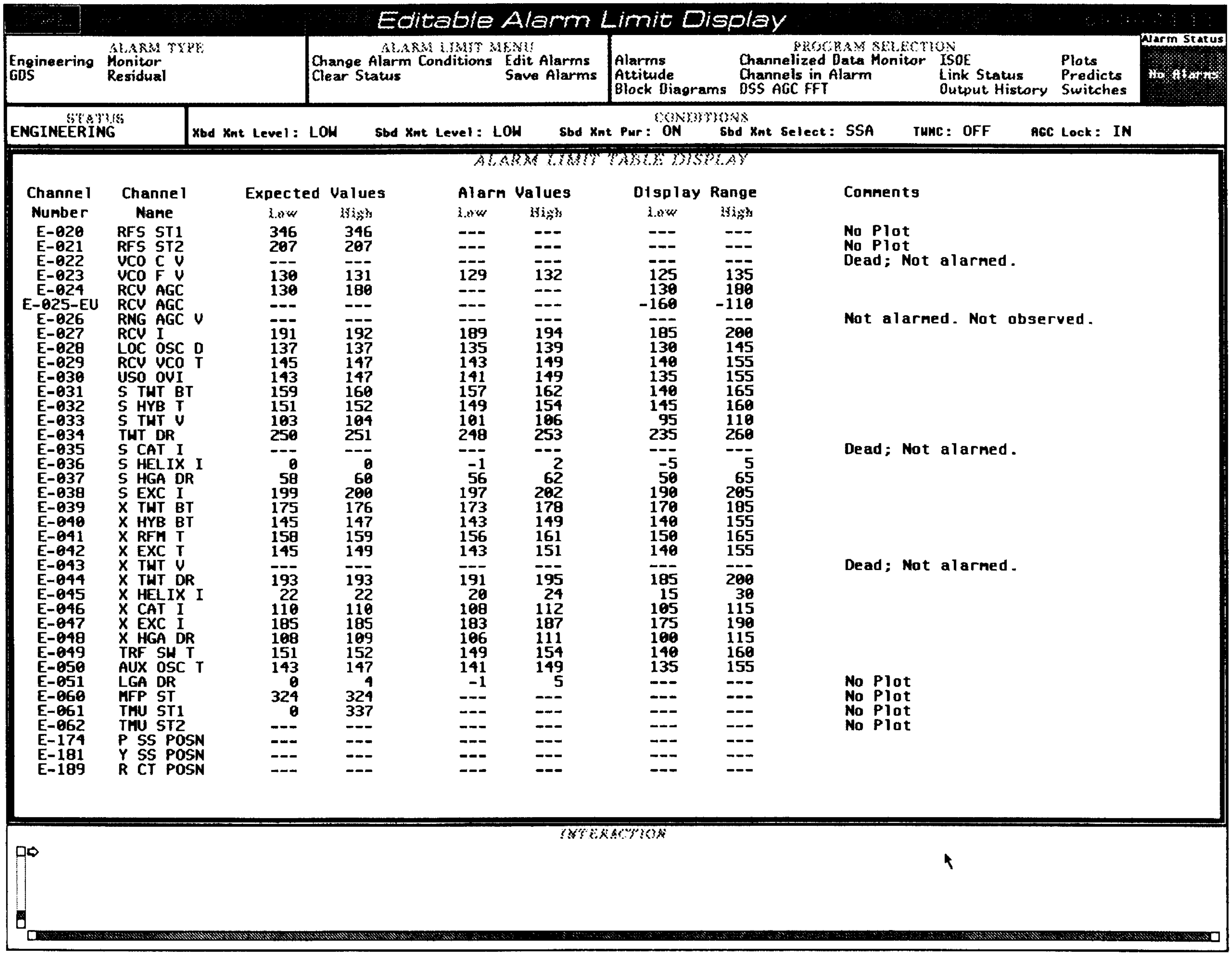
The SHARP system provides an alarm limit interface which allows on-line viewing and editing of established spacecraft engineering alarm limits, DSS performance limits, ground data system limits, and residual thresholds. See Figure 5-3. The user specifies which type of alarm limit to view, and for each channel of that type, the following information is displayed:

Channel Number
Channel Name
Lowest Nominal Value
Highest Nominal Value
Low Alarm Limit
High Alarm Limit
Low Display Range Value
High Display Range Value
Comments

If the specified alarm limits depend on a particular spacecraft configuration, the user is prompted to identify the configuration (e.g., whether the the X-Band Transmitter Power Level is high or low, etc. Refer to Section 4.2.1.). The limit values are then displayed, along with the conditions of the specified configuration, as demonstrated in the upper portion of Figure 5-3. The operator may subsequently change any configuration condition to display the appropriate table of limits.

Users are able to make permanent or temporary changes to these tables. To temporarily alter any of these limits, the user makes the appropriate changes to the desired tables. Changes are made by selecting a channel in the table to modify and then filling in the ensuing menu with the desired modifications. This capability, which provides a manual override option, enables the operators to suppress alarms or to set tighter alarm limits for closer scrutiny of a particular event, with no intervening procedures that would otherwise be required in the current operations environment.

To permanently change any alarm tables, the user follows the same procedure as above and then saves the changes. To rewrite the data files that contain the alarm limit tables so that they will be used the next time that SHARP is started, the user selects *Save Alarms* in the Alarm Limit menu. For the SHARP prototype no security feature was included that would prevent unauthorized users from making changes to the alarm limits. This feature can be included so that only authorized users could make such changes.



Channel Number	Channel Name	Expected Values Low	Expected Values High	Alarm Values Low	Alarm Values High	Display Range Low	Display Range High	Comments
E-020	RFS ST1	346	346	---	---	---	---	No Plot
E-021	RFS ST2	207	207	---	---	---	---	No Plot
E-022	VCO C V	---	---	---	---	---	---	Dead; Not alarmed.
E-023	VCO F V	130	131	129	132	125	135	
E-024	RCV AGC	130	180	---	---	130	180	
E-025-EU	RCV AGC	---	---	---	---	-160	-110	
E-026	RNG AGC V	---	---	---	---	---	---	Not alarmed. Not observed.
E-027	RCV I	191	192	189	194	185	200	
E-028	LOC OSC D	137	137	135	139	130	145	
E-029	RCV VCO T	145	147	143	149	140	155	
E-030	USO OVI	143	147	141	149	135	155	
E-031	S TWT BT	159	160	157	162	140	165	
E-032	S HYB T	151	152	149	154	145	160	
E-033	S TWT V	103	104	101	106	95	110	
E-034	TWT DR	250	251	248	253	235	260	
E-035	S CAT I	---	---	---	---	---	---	Dead; Not alarmed.
E-036	S HELIX I	0	0	-1	2	-5	5	
E-037	S HGA DR	58	60	56	62	50	65	
E-038	S EXC I	199	200	197	202	190	205	
E-039	X TWT BT	175	176	173	178	170	185	
E-040	X HYB BT	145	147	143	149	140	155	
E-041	X RFM T	158	159	156	161	150	165	
E-042	X EXC T	145	149	143	151	140	155	
E-043	X TWT V	---	---	---	---	---	---	Dead; Not alarmed.
E-044	X TWT DR	193	193	191	195	185	200	
E-045	X HELIX I	22	22	20	24	15	30	
E-046	X CAT I	110	110	108	112	105	115	
E-047	X EXC I	185	185	183	187	175	190	
E-048	X HGA DR	108	109	106	111	100	115	
E-049	TRF SW T	151	152	149	154	140	160	
E-050	AUX OSC T	143	147	141	149	135	155	
E-051	LGA DR	0	4	-1	5	---	---	No Plot
E-060	MFP ST	324	324	---	---	---	---	No Plot
E-061	TMU ST1	0	337	---	---	---	---	No Plot
E-062	TMU ST2	---	---	---	---	---	---	No Plot
E-174	P SS POSN	---	---	---	---	---	---	
E-181	Y SS POSN	---	---	---	---	---	---	
E-189	R CT POSN	---	---	---	---	---	---	

Figure 5-3 Alarm Limit Display

5.2. Integrated Sequence of Events

The SHARP system provides a user interface for the Integrated Sequence of Events which offers numerous capabilities to the operator. Figure 5-4 shows a sample ISOE display. Viewing of any ISOE is available by specifying the day or week of interest. The stripped telecommunications data will appear in a scrolling window near the top of the display, and on-line additions, deletions, and other changes can then be performed on the selected ISOE via menu-driven commands. Edits can be performed with ease, as the self-guiding menus contain explanations of the complex ISOE data.

SHARP's on-line editing capabilities allow the operator to enter into the SHARP system the latest modifications as specified by the ISOE correction sheets and reduce the likelihood of an operator referencing outdated material.

Translation of spacecraft commands from their raw form into more understandable summaries of spacecraft activity may be performed. The user can request status summaries of any activity. For example, a request can be made to view all values of the variable "Data Mode" during a user-specified period of time.

A history of user actions is maintained so that as the ISOE is updated, the operator can verify all modifications that are made. The operator then has the option to save these changes or revert back to the original ISOE.

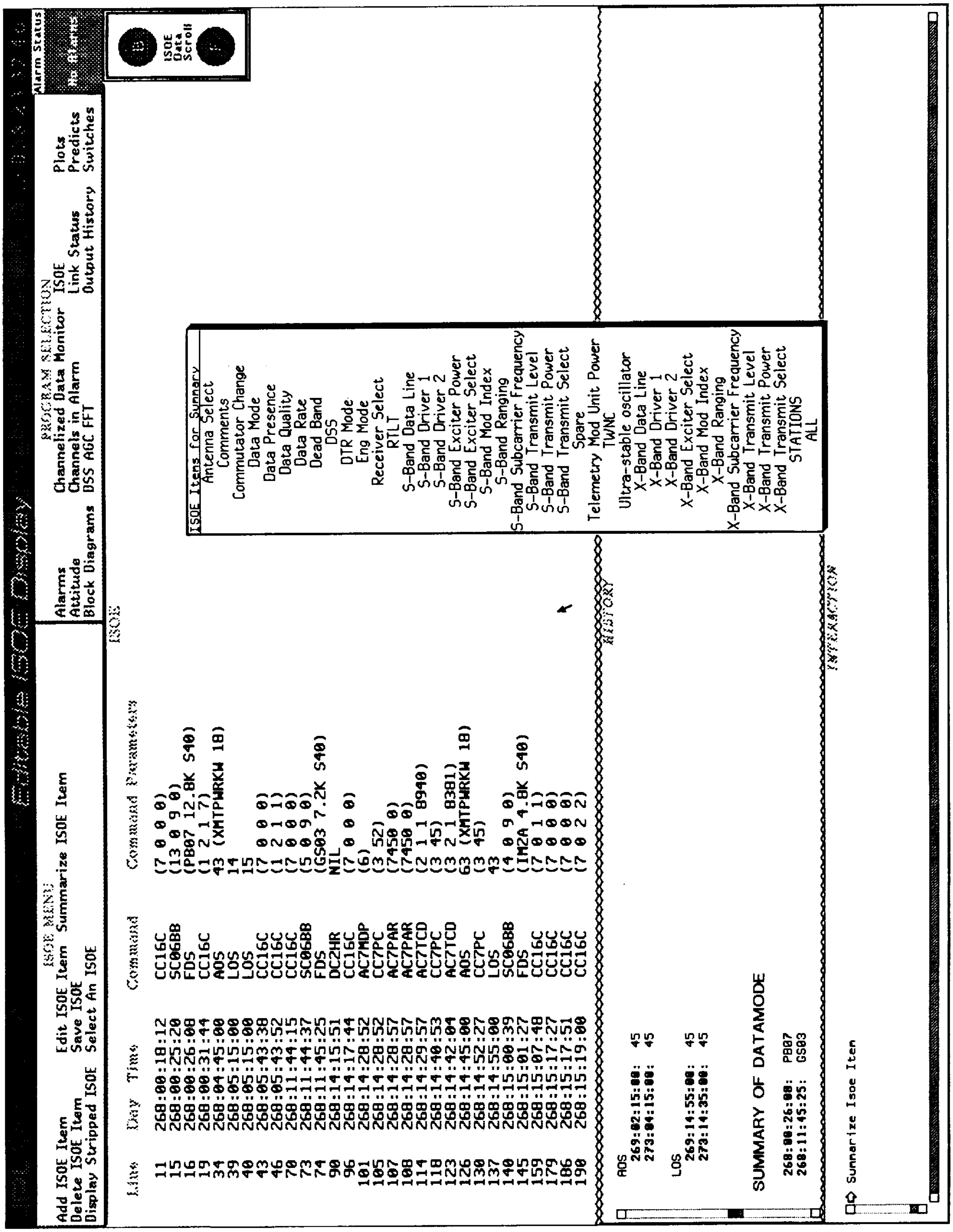


Figure 5-4 Integrated Sequence of Events

ORIGINAL PAGE IS
OF POOR QUALITY

5.3. Predicts

The Predicts interface in SHARP allows both tabular and graphical displays of prediction data. An example of this display is in Figure 5-5. The operator can display in tabular form raw predicts, pass predicts, instantaneous predicts, and residuals for any specified time range. Two display windows exist in the predicts interface so that the user may simultaneously view alternate forms of data. The time period, station, and power mode are displayed above each window to identify the predict data context.

Along with the tabular data, a color-coded Deep Space Station availability graph has been designed which enables rapid identification of stations within view of the spacecraft for any specified time period. Situations that mandate that another Deep Space Station be acquired can be addressed immediately as opposed to the more arduous current method which requires the manual look-up of each station at the specified time period.

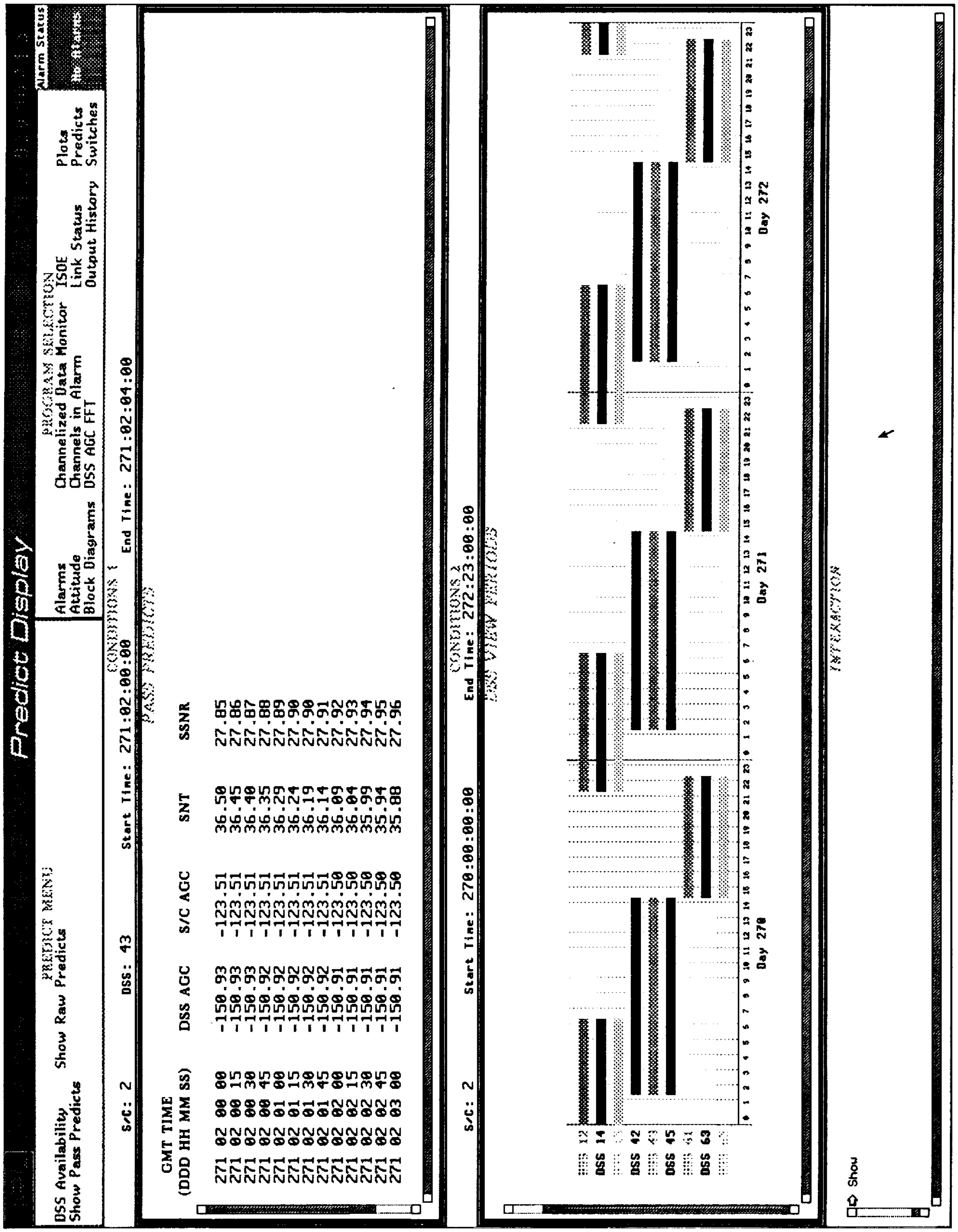


Figure 5-5 Predicts Display

5.4. Channelized Data Plotting

Telemetry data from the spacecraft, tracking stations, and other relevant systems are collected in the JPL computers and separated into channels that are distributed for processing. For the Voyager project, the data were distributed from the Test and Telemetry Subsystem (TTS) computers. These channels contain the values of hundreds of spacecraft engineering parameters and station performance parameters. The channels are plotted in the Voyager real-time area and are visually monitored to ensure that they remain within their pre-specified limits.

The current Voyager data display system consists of plots on black and white computer screens. The constraints of this system allow the construction of only five plot display pages for the entire spacecraft team, of which the telecommunications subsystem has control of just a single page. One display page is capable of showing up to three plotted channels. In order to change the plot parameters to select different channels to display, the operator must punch a card and feed it into the system's card reader. To obtain an additional plot, special permission must be secured from personnel of another subsystem who are willing to temporarily give up one of their own plots.

SHARP's display that plots the channelized data, illustrated in Figure 5-6, is a significant improvement over existing capabilities. The user can dynamically customize the display at any time by selecting which and how many channels to view, the time scale, and the data range for each plot. Each plot can be color-coded by the user for easy visual distinction between displayed channels.

In the example in Figure 5-6, four channels are plotted. The data in view cover a period of 1.5 hours. The data range for Channel E-025 data, for example, is from -95 to -155. The colors used for each channel were different shades of blue or green.

When a channel is in alarm, its corresponding data points and connecting lines are plotted in red (hard alarm) or yellow (soft alarm), allowing the operator to quickly notice an alarm condition. The channel's associated alarm limits may be optionally overlaid onto the channel's plot for further information. For example if Channel M-764, a monitor channel with fixed alarm limits, had a soft alarm limit of 1000 and a hard alarm limit of 1025, a horizontal yellow line would be drawn at the value of 1000, and a red line at 1025. These lines would allow the operator to see how close the data were to each alarm limit. For engineering channels where the

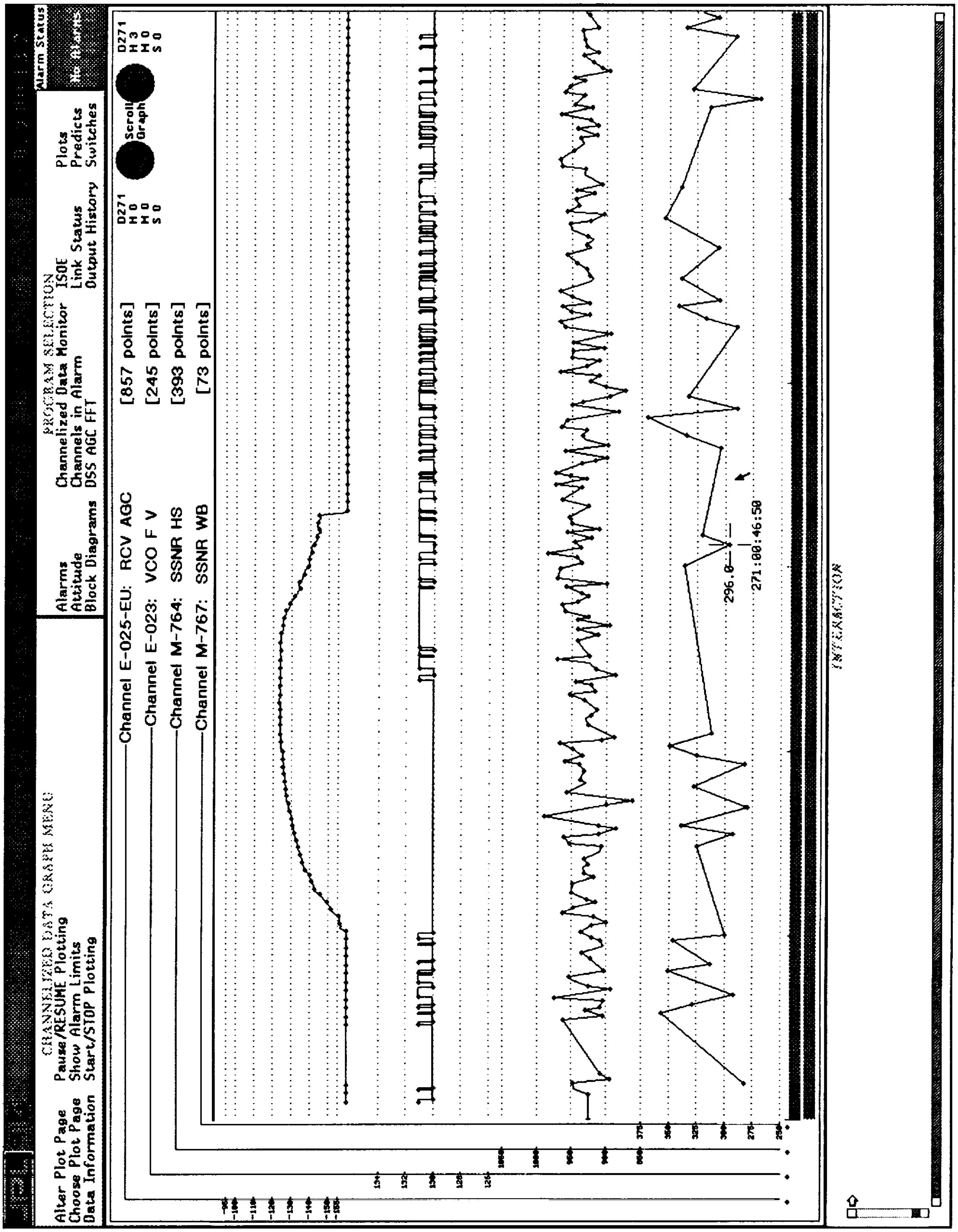


Figure 5-6 Channelized Data Plot

alarm limits may be changing over time, the red and yellow alarm limit lines would not be just constant horizontal lines, but would show steps to indicate when the configuration changes of the spacecraft would cause changes to the alarm limits.

Each data point is mouse sensitive to provide time and numerical value indicators. Clicking the mouse on a data point will display (or erase) cross hairs with the time and value. At the bottom of Figure 5-6, one can see the data point on Channel M-767 has a value of 296 at a time of 00:46:50 on day 271.

An automatic counter continually indicates the number of data points per plot. This information is located at the top of the plotting window after each label for the channels. 857 points have been plotted on Channel E-025.

Real-time data can be plotted as it is being received. Also, if the operator chooses to review past data, historical data that are stored in the channelized data database can be plotted. The displays are can be scrolled along the time axis to allow the operator to move through more data than that which are currently being presented.

These plots can represent information as graphs of actual or derived data versus time, x/y plots, or scatter plots. Each type of plot can use linear or logarithmic scales. One example of a plot of derived data is a residual plot which shows the variance between the real-time data of a channel and its expected value.

For the Voyager encounter, two types of plots were used. The majority of the plots were channelized data versus time. A second type of plot used was the scatter plot of one data channel graphed against another data channel. Telecommunications personnel specifically requested that a scatter plot of Bit Error Rate (BER) versus DSS Symbol Signal-to-Noise ratio (DSS SSNR) be constructed. This provided information on whether the actual Bit Error Rate was consistent with the received SSNR. A sample BER scatter plot is shown in Figure 5-7.

The DSS SSNR can be received on either of channels M-764 or M-767 depending upon the state of the telecommunications link. The BER is determined from a combination of data from channel S-752 and the ISOE.

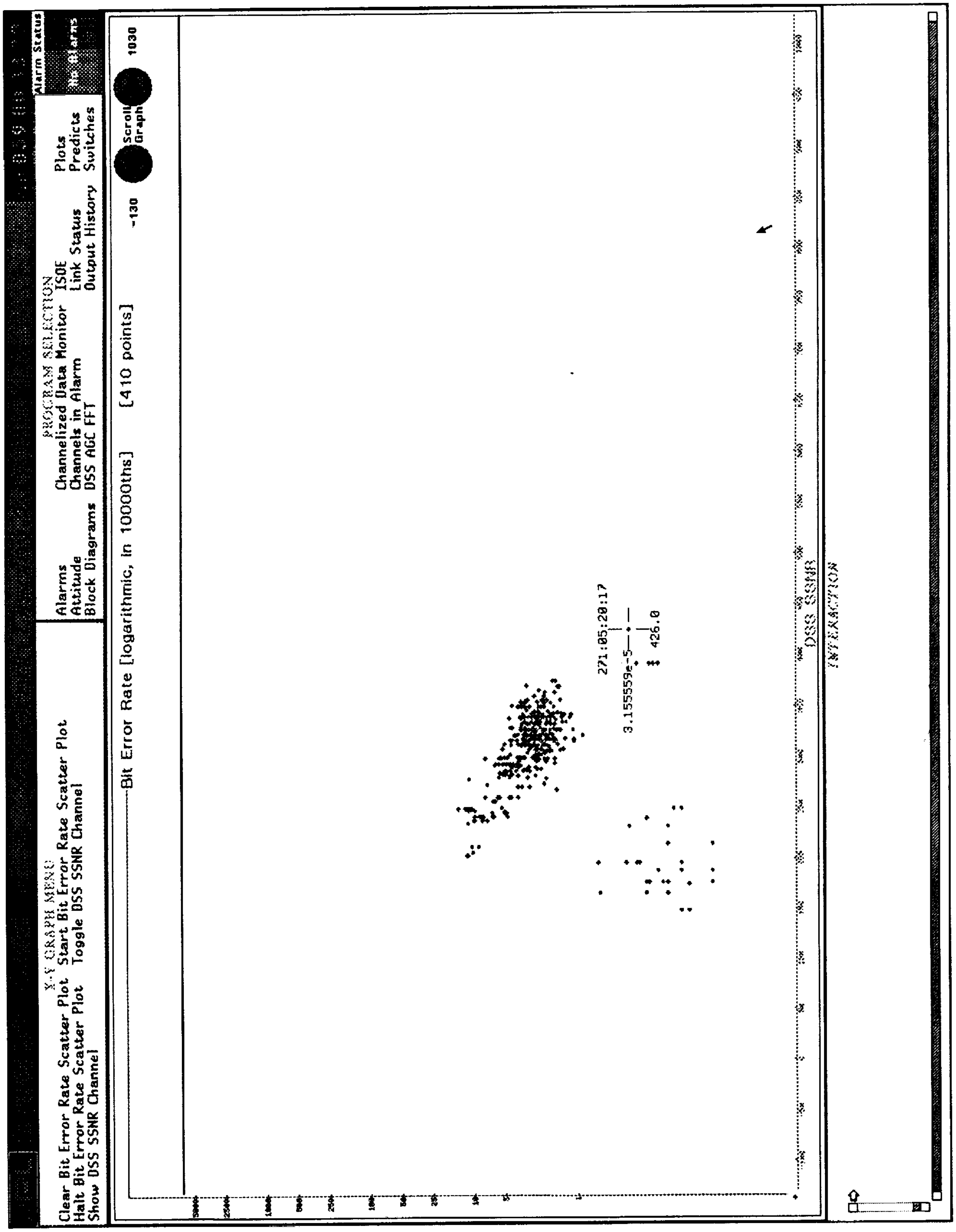


Figure 5-7 Bit Error Rate (BER) vs. Symbol Signal-to-Noise Ratio (SSNR)

ORIGINAL PAGE IS OF POOR QUALITY

5.5. Block Diagrams

The SHARP system provides a graphical capability in the form of on-line functional block diagram schematics of the end-to-end communications path from the spacecraft through a Deep Space Station (DSS) and Ground Communications Facility (GCF) to the Mission Control and Computing Center (MCCC) at JPL and final destination of the Test and Telemetry System (TTS) computers (see Figure 5-2). This facility allows the operators to view various components or subsystems of the telecommunications path in a schematic form.

The top-level diagram could be considered to be the base node of a tree structure of diagrams. (See Figure 5-8.) Each block diagram may contain links to one or more other block diagrams. By moving through the block diagrams, the user can see successive levels of detail of the telecommunications system. The SHARP system focuses on two areas: the spacecraft and the DSS. The telecommunications subsystem is very comprehensive, as spacecraft schematics have been developed for all of its individual components. Diagrams of two DSS areas have also been developed. Figure 5-8 illustrates the Voyager telecommunications subsystem.

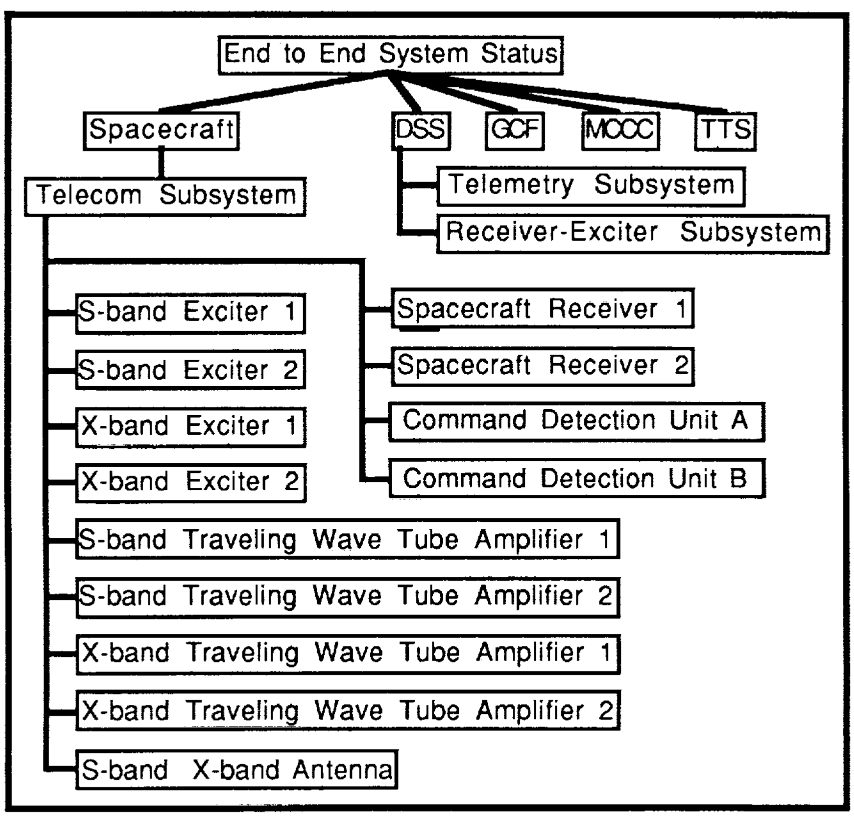


Figure 5-8 Block Diagrams

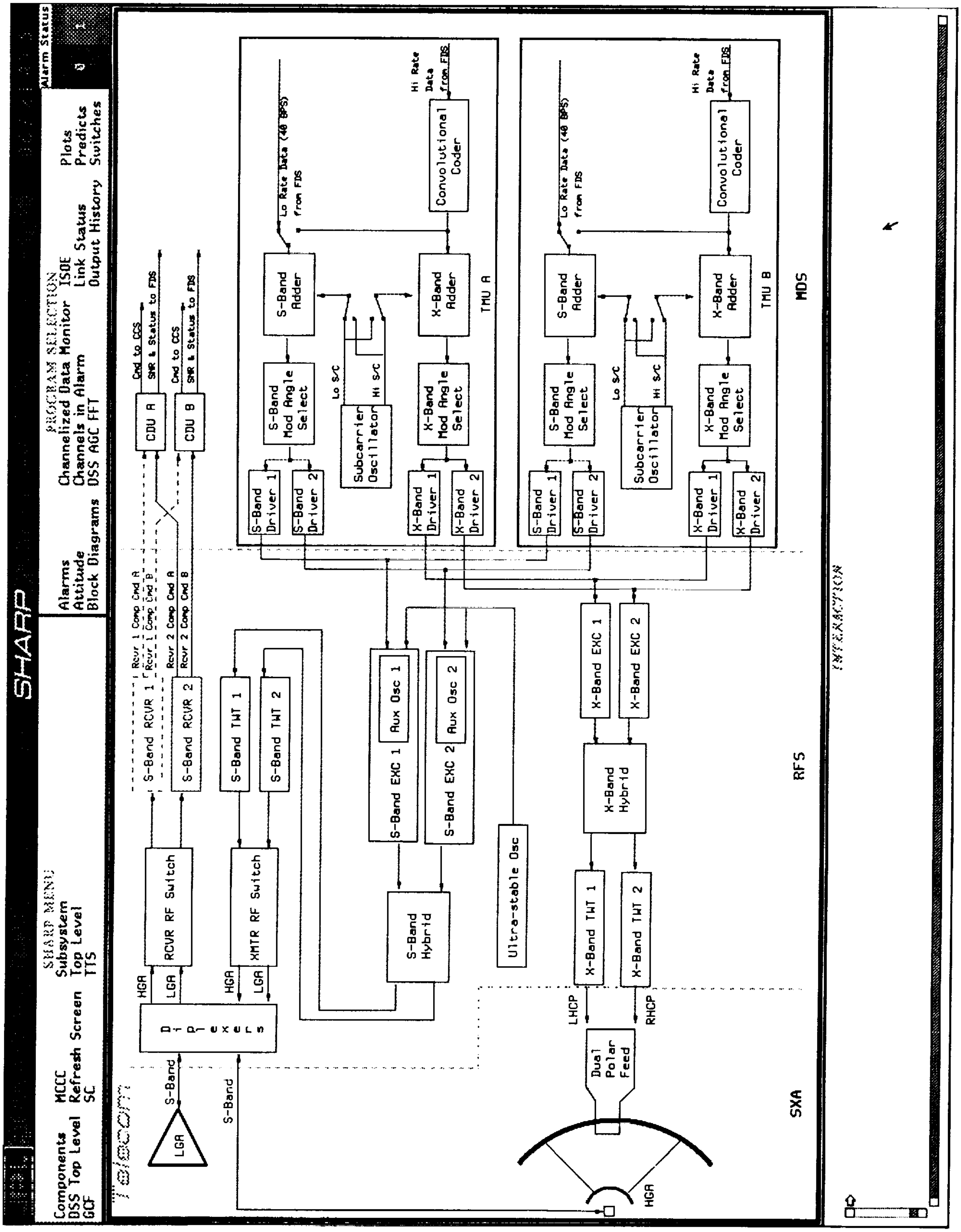


Figure 5-9 Voyager Telecommunications Subsystems Block Diagram

ORIGINAL PAGE IS
OF POOR QUALITY

The icons of each diagram represent a component or subsystem. These are mouse sensitive. When an icon is selected, the block diagram it represents is displayed.

The block diagrams were constructed by using an object-oriented drawing program developed specifically for the SHARP effort. (Refer to Appendix D for a discussion of GEd, a graphics editor.) Since each component's icon is implemented as an object, it is relatively easy to change its color. This feature is used so that components that are functioning normally are colored green; components that are off are white; and components that SHARP has diagnosed to be in an alarm condition are red.

The status of the various spacecraft and DSS systems are continually updated on the diagrams, allowing the operator to quickly identify problem areas as anomalies arise.

There are two modes in which the block diagrams can be activated. They can either be passive, in which case the user must act to cause a diagram to be displayed. Alternatively, the block diagrams can be responsive to the changing status of a component. In this case, when a block diagram changes, it automatically comes to view. The former behavior is the default by choice of the designers of SHARP. It was felt the users would not want the intrusive behavior of the latter mode.

Whether a component is colored green (on) or white (off) is determined from its operating status. When SHARP is first started, the ISOE data is searched to determine the status of each component at that point in time. The components of each diagram are then colored as appropriate.

SHARP must guarantee that the block diagrams follow changes in the functioning of the components as specified in the ISOE. To do this a separate process runs that updates the block diagram component colors as changes occur in their states. This updating process searches forward in the ISOE from the time of the most recent change. This search is looking for the next time when an event occurs that will cause a change in the block diagrams. When this time is determined, the updating process becomes inactive. At the arrival of that time the process becomes active, changes the coloring of the proper block diagram component, and then searches again for when the next change will occur.

The determination of whether a component is turned red (to indicate an alarm state) occurs when real-time data is being received. The AI module contains the necessary information needed to be able to change one or more diagrams when an alarm is indicated. As the AI module detects a channel going into or out of alarm, it does the work necessary to make these changes appear in the appropriate block diagrams.

5.6. Alarm Warnings

Alarm warnings do not actually have their own display configuration. Instead, this is a part of the interface that allows the user to be notified whenever an alarm occurs regardless of the context the user is in. Whenever an alarm occurs, a window appears on the screen and waits for the user to acknowledge the message. A sample of this is shown in Figure 5-10. It is possible to change the behavior of SHARP so that alarm messages are not displayed, but are only recorded to the Alarm History. The user has control over what classes of alarms should or should not be presented to him. It is also possible to set a time-out period so that the alarm warning windows automatically close if that period has elapsed.

```
Message from the diagnostician at GMT 163 20:22:54;

Hard Alarm: E-025 S/C AGC (Automatic Gain Control) in alarm.

Source: S/C Engineering Data.

Explanation 1: The DSN Exciter Frequency is in alarm.  The wrong ramp
was entered into the Digitally Controlled Oscillator (DCO).

Corrective Action: Advise DSN to restart the DCO with correct frequency offset
and ramp rate.

Explanation 2: The AGC detector has failed.

Corrective Action: Notify SCT personnel.

Explanation 3: S/C Antenna is off point.

Corrective Action: Check Attitude Control data.

More Information: Consult data in the Channelized Data Display
for channels E-025 (S/C AGC), M-777 (Exc Freq), E-074 (Attitude Pitch),
E-181 (Attitude Yaw), and E-189 (Attitude Roll).

Please click any mouse button to acknowledge.
```

Figure 5-10 Sample Alarm Warning

5.7. Alarm History

The Alarm History provides a visual display of all alarm messages that are generated by SHARP. It allows the user to scroll the display in order to access messages that are off the screen. The display includes a status window of how many of the various types of alert messages have been generated so far, e.g., alarm messages, diagnostic messages, etc.

5.8. Alarm Meters

This display of SHARP, shown in Figure 5-11, provides to the operator a simultaneous view of the values of all channels in alarm, and (for most channels) where these values stand with respect to the respective channels' alarm limits. When a channel goes into alarm, a new meter is created for it. When a channel that had been in alarm goes out of alarm, its meter is deleted. The meter pointers showing channel values are updated whenever new data values are received.

The large central area of the display configuration can contain one or more meters, each corresponding to a different channel. Each meter consists of a horizontal line segment, ticks, labels of the numbers corresponding to the left and right ends of the segment, a label indicating the channel name, and a pointer. For simple numerical alarm limits, the horizontal line has regions colored red, yellow, and green to indicate ranges of hard alarms, soft alarms and nominal behavior for the channel. For other more complicated alarm determinations, the line is colored blue.

The pointer points to the coordinate on the line segment corresponding to the channel value; and a displayed string, such as "146.0 at **277:02:46:37**", gives the value and time of the last datum on the channel. After the value and time, the string may also contain such a substring as "(OK at **277:02:57:22**)", indicating that the alarm limits on the channel have changed since the latest datum was reported, and that if the channel still has that latest value, it is not in alarm any more. If the meter's horizontal line segment is drawn in green, yellow, and red, the pointer triangle and string will be drawn in the color indicating the appropriate level of alarm (as of the time of the datum). If the line segment is blue, the pointer triangle and string will also be blue.

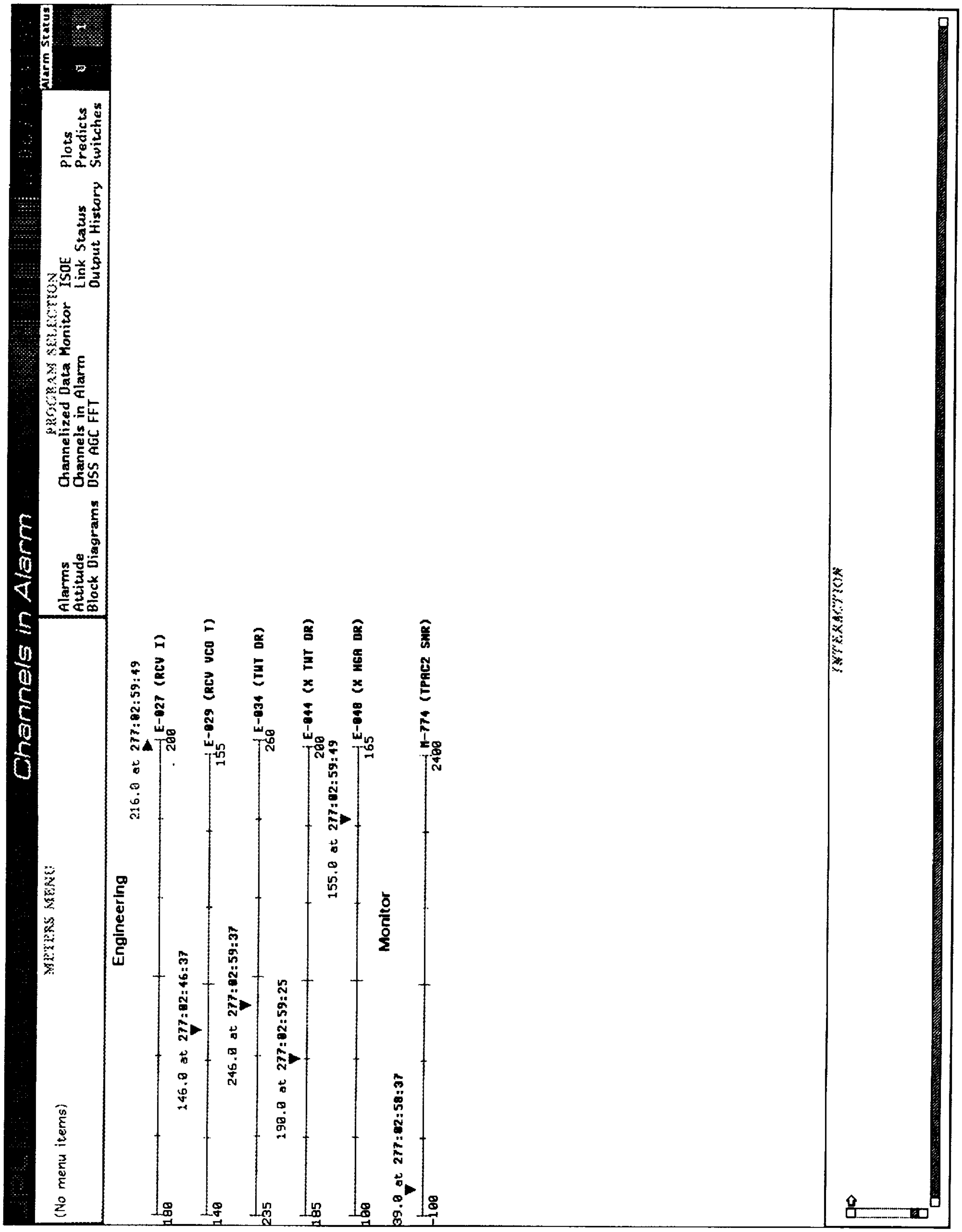


Figure 5-11 Alarm Meters

5.9. Attitude and Articulation Control Subsystem Display

Currently telecommunications (and Attitude and Articulation Control) personnel examine a black and white plot page which contains three individual plots depicting spacecraft pitch, yaw, and roll movement. These displays are necessary to monitor the attitude behavior of the spacecraft, yet do not allow easy recognition of alarm conditions.

SHARP's Attitude and Articulation Control Subsystem display combines various sources of information to produce an integrated graphical representation of spacecraft attitude and articulation. As shown in Figure 5-12, this display combines spacecraft motion parameters (pitch, yaw, and roll) and records spacecraft movement over time. In addition to the graphical layout, actual motion parameter values and time are also shown to augment the display. As new attitude data become available, both the graphical and textual components of the display are updated to reflect current spacecraft positional information.

The spacecraft is represented as a large cross hair in the attitude display. This icon moves around the display, and turns red when any of the attitude parameters are in alarm.

A limit cycle box which represents defined spacecraft deadband limits (limits for pitch, yaw, and roll) encloses the spacecraft icon. The deadband limits are obtained from the ISOE and the limit cycle box changes size and shape according to pitch and yaw deadband changes. The roll deadband box is represented as a Maltese cross around the spacecraft icon. The spacecraft icon rotates within the roll limit cycle box to indicate roll alterations.

As the yaw and pitch values change, a line is drawn from the old value to the new value. These trailing vectors from the spacecraft icon establish the path of the spacecraft and enable the visualization of spacecraft movement over time.

Alarm conditions are easily detected as the spacecraft icon drifts outside of the designated deadband box. The spacecraft turns red to denote the alarm condition, and the corresponding pitch, yaw, or roll information is highlighted red as well. This information is passed to the diagnostician which will subsequently notify the operator of the attitude alarm state.

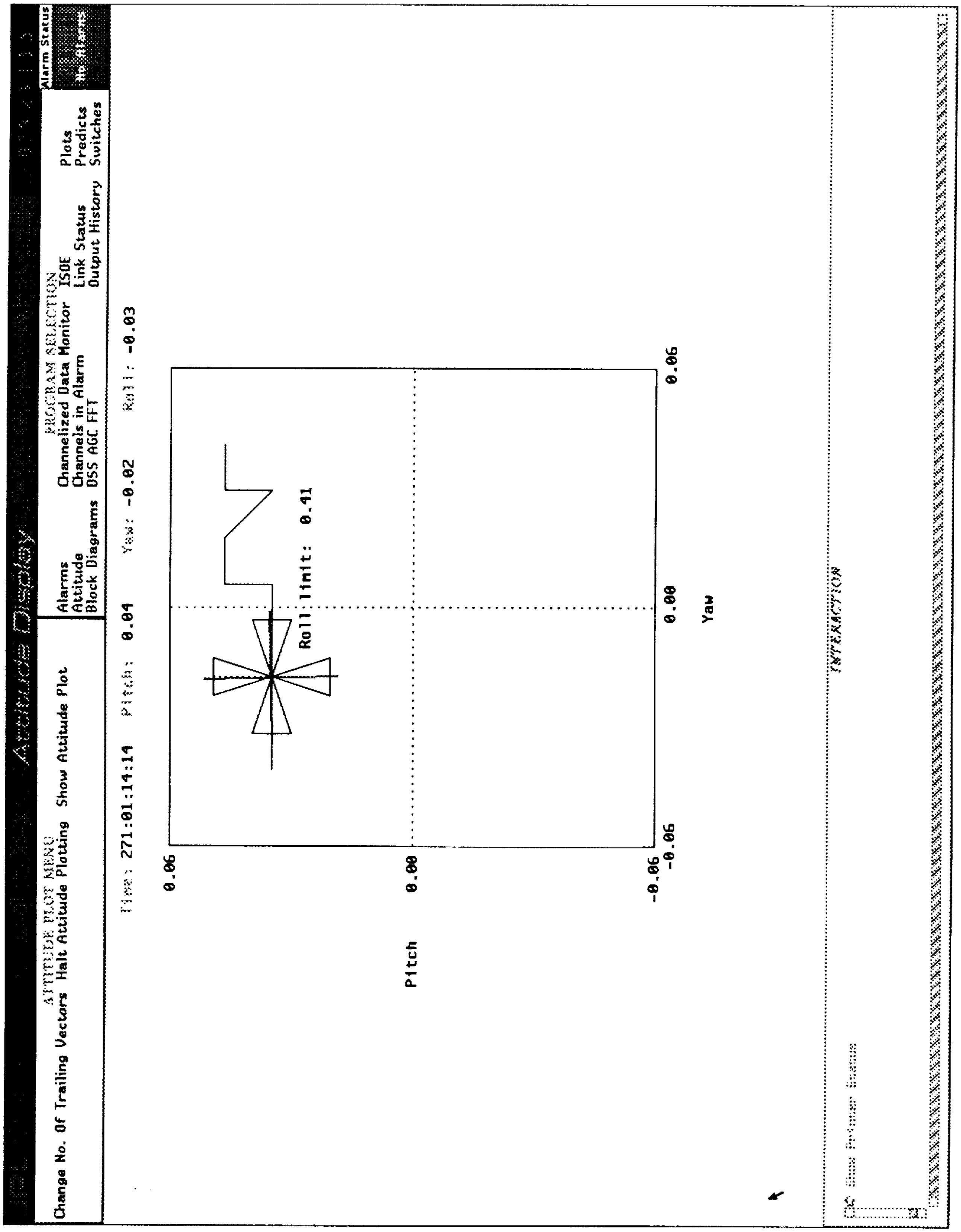


Figure 5-12 Attitude and Articulation Control Subsystem Display

5.10. Fast Fourier Transform

The "Fast Fourier Transform of DSS AGC", or "DSS AGC FFT", display shows the operator a frequently-updated bar chart of the magnitudes of the components of the FFT of the automatic gain control (AGC) channel, with the conscan component distinguished by its color. The display also shows a list of the times at which actual channel data were received for this FFT and the gaps during which data were estimated by interpolation. An example of this display is given in Figure 5-13.

When this configuration is selected, the bar chart display is updated every time a new FFT is computed. The heights of the bars are proportional to the magnitudes of the components, with the scale indicated by the vertical axis. Most of the components are shown as blue bars.

The conscan component is shown as a green bar if it is not in alarm. If it is in alarm, the part of the bar above the momentary alarm limit (twice the average of the magnitudes of the second through thirty-second components of the FFT except the conscan component itself) is red, and the bottom is green.

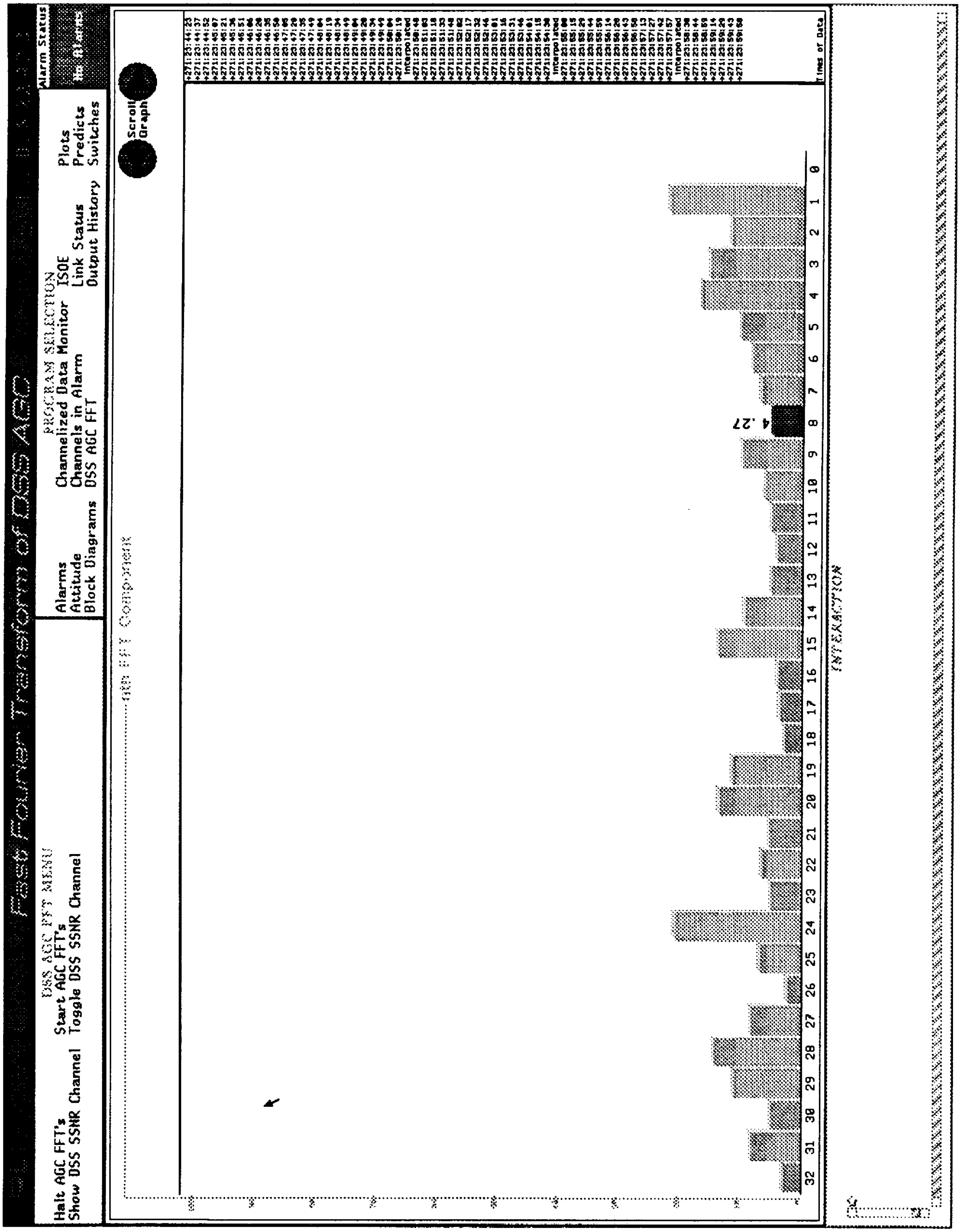


Figure 5-13 FFT Display of DSS AGC

5.11. Link Status

Among the new graphical analysis capabilities provided by the SHARP system is the telecommunications link status display, as shown in Figure 5-14. This facility was developed to combine multiple sources of data into one graphical presentation.

The SHARP link status display is a significant improvement over current telecommunications capabilities, where station coverage, spacecraft transmitter power status, and data rate are listed in an unofficial graphical sequence hardcopy product (SFOS), or are determined by manually searching through the hardcopied ISOE. Station uplink must be viewed on a black and white DTV terminal. Projected downlink, data outages, spacecraft-DSS lock status, and spacecraft data quality are manually determined in real time, and manual analysis is performed when the alarms indicate a possible anomaly.

SHARP's link status display capability provides an abundance of information to the operator at one glance. The analysis provides the user with such valuable information as time ranges and explanations of data outages. It can warn the operator when to expect noisy or corrupted data and why.

Various types of data are graphed over time, which appear on the horizontal axis as hourly (the default) increments. The upper portion of the graph represents station coverage of the specified spacecraft. The stations are color coded according to their size and type: 70-meter (DSS 14, 43, 63), 34-meter high efficiency (12, 42, 61), 34-meter tracking (15, 45, 65), and non-DSN (Parkes, Australia, the Very Large Array in New Mexico, and Japan). SHARP examines the ISOE data to determine station coverage during the specified time period, and then draws horizontal bars to represent the stations. Each bar is labeled with its corresponding DSS identification number.

Also drawn in non-real time is the spacecraft data transmission rate. Time bars are drawn in green and labeled with the corresponding data rate during that time interval. This information, along with data outages, is taken from the ISOE database. When the ISOE is translated, those events which are known to cause telemetry outages or degradations (e.g., no station coverage or ongoing spacecraft maneuver) are noted for uses such as this one. The outage information is overlaid in black on the data rate line.

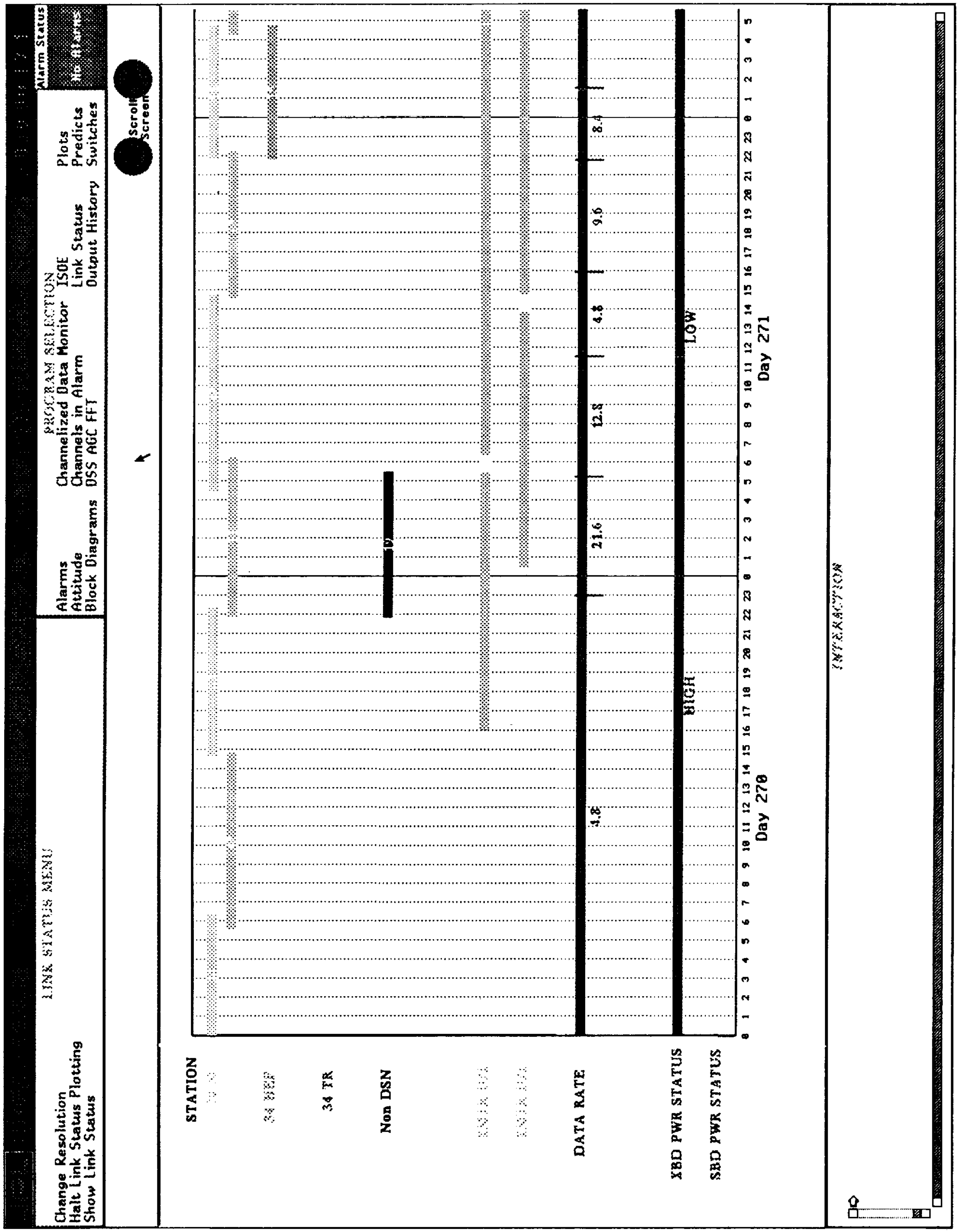


Figure 5-14 Link Status

ORIGINAL PAGE IS
OF POOR QUALITY

At the bottom portion of the graph, s-band and x-band transmitter power status is drawn. The bars are color coded to represent high-power level (green), low-power level (yellow), or off status (black).

Two lines representing station transmitter uplink (signal transmission) and projected downlink are drawn in real time. When an uplink begins, a bar starts to be drawn and advances in time as the uplink continues. As that is plotting, the expected resulting downlink is drawn at the time it should occur, which is a round-trip light time (RTLT) into the future. In Figure 5-14, an uplink begins on day 270 at 16:00. The resulting downlink begins on day 271 at 00:30.

If the two-way non-coherent (TWNC) mode is scheduled to be off during the downlink, the data rate line at RTLT after uplink is painted yellow to indicate two-way data mode. When an uplink is low and the TWNC is off, there may be degraded data. To determine if there is degraded data, the Spacecraft Receiver AGC (E-025) is examined at the RTLT after an uplink.

To warn an operator the data rate line will turn from yellow to orange when the uplink is low, indicating that there is noisy data due to excessive two-way phase noise. When the uplink is extremely low (<-140 dB), the data rate line turns red to indicate that the uplink is in the "Quasi region" and data have been destroyed because of extreme amounts of excessive phase noise due to a best lock frequency (BLF) estimate error.

Spacecraft-DSS lock status information is overlaid on the downlink line, which turns from yellow to orange to indicate Quasi region data, and red to indicate that either the DSS missed the spacecraft because it was tuned to the wrong frequency, the DSS transmitter was not radiating, or there was a spacecraft failure.

Each time bar is mouse sensitive, and will display the starting and ending time of the bar, or event. If there is an explanation associated with the event, such as a spacecraft maneuver telemetry outage or noisy data, then this information will be presented to the user.

5.12. Channelized Data Monitoring

This module allows a SHARP application developer or maintainer to watch and debug the overall operation of SHARP with respect to all of its data sources. It maintains and displays numerous status icons in a variety of formats such as raw numbers, tables, graphs, meters, and textual displays. This display is primarily intended for the developers of SHARP but may be useful for the SHARP user when telemetry status is needed at a glance.

6. Voyager Demonstration

6.1. Installation in Telecommunications Real-Time Area

The development of the SHARP system took place within the facilities of the Artificial Intelligence group. It was possible from this location to acquire all of the necessary ISOE, Predict, and real-time channelized data required for the operation of SHARP. This allowed the developers to work on the system in an environment similar to that of the real-time operations area without disrupting the telecommunications personnel.

Approximately one month before the Voyager encounter with Neptune, a Symbolics workstation with SHARP loaded on it was moved into the real-time telecommunications work area. The installation of the SHARP system in the real-time area was very simple. There were two differences between the development environment and the real-time environment.

The first difference between the two areas was in the way the channelized data was received by SHARP. In the development environment, the channelized data was received via a modem connection from the real-time area. In the real-time area the data was received by SHARP via a direct connection. The modem connection was sometimes unreliable. On occasion, the developers of SHARP would have to call the operators in the real-time area and ask them to reset the modem transmitting the data when it had become locked up. In contrast to the sporadic unreliability of the modem connection, the reliability of the channelized data connection in the real-time area was flawless.

The second difference between the two areas was the availability of network connections. The machines in the development area were all connected together on Ethernet with several machines acting as file servers. The processor in the real-time area was not networked. Unabridged ISOE files and raw Predict files could be acquired in the development area over the Ethernet connection. Therefore all ISOE and Predict database generation was done in the development area. In order to take database updates and software changes to the real-time area, tapes were carried between the two sites. A permanent application of SHARP would ideally have network connections to all of its data sources.

6.2. Events

During the demonstration period, SHARP helped find the cause of a Voyager science data error anomaly which appeared in the telemetry from the spacecraft as an excess error count. After SHARP detected the problem, its graphical displays were used by telecommunications personnel to identify the problem and to characterize its magnitude. The problem was in the Voyager ground data system and was corrected by the replacement of a wide-band interface unit in the Voyager Data Acquisition and Capture System (DACS). SHARP helped verify that the replacement of the unit actually fixed the problem. In a matter of hours, SHARP was able to assist operators in solving an anomalous condition which could have easily escalated to a more serious problem during the encounter itself, and could have taken human operators days or weeks to isolate without SHARP.

Also during the demonstration period, the knowledge engineer of SHARP and the domain expert would review alarms that SHARP had given. Generally, these alarms were correct. In one alarm situation, SHARP was giving warnings about the loss of the telecommunications signal. This ultimately turned out to be a false alarm as the spacecraft was undertaking a particular maneuver that the SHARP knowledge base did not contain, thereby leading the diagnostic system into an erroneous conclusion about antenna pointing. In other cases, SHARP was able to detect conditions where the Deep Space Station antenna tracking the spacecraft was drifting off point. SHARP detected these problems in a matter of seconds, and reported the condition to the telecommunications operators. Unfortunately, due to their previous lack of ability to detect and diagnose antenna pointing problems, the real-time telecommunications operators at JPL did not have procedures for alerting the Deep Space Station operators (possibly on the other side of the world) to antenna drift situations detected by SHARP. When the antenna drift reached a sufficient magnitude and urgency for the station operators to notice and correct, SHARP was able to detect the resolution of the problem and cancel the alarm situation. SHARP detected and correctly diagnosed other non-critical problems with the receiver automatic gain control and the S-band traveling wave tube temperature on board the spacecraft.

On the whole, the encounter with Neptune went extremely smoothly for the Voyager spacecraft. SHARP did not get a chance to make any really dramatic diagnoses, and the diagnostic system described in this publication did not get a strenuous operational test. This underscores the difficulty in testing the diagnostic ability of real-time monitoring and control expert systems in operation settings: you may not get any problems! Using simulated data (based on historical problems with the spacecraft and based on synthetic situations) we were able to test SHARP much more thoroughly in the laboratory. SHARP is able to analyze 39 classes of telecommunications problems, and make about 60 unique diagnoses which require some problem solving by the mini-experts to determine. Another 20 telecommunications problems are detectable by SHARP, but can be reported directly to the operator. Our domain expert estimates that SHARP covers approximately 80% of the known types of faults experienced in spacecraft telecommunications for Voyager. The remaining 20% include diagnoses which could be made if SHARP had the appropriate real-time data and additional knowledge engineering. As with most complex systems, there is always the possibility of novel faults. SHARP does not have the ability to successfully diagnose and explain a novel type of fault (nor was it intended to), but we are confident in the system's ability to detect departures from expected, nominal behavior.

6.3. Evaluation

The evaluation of SHARP by telecommunications personnel was not as thorough as the developers would have liked. By the time SHARP was installed in the real-time area, the quantity of work required by the operators had increased in anticipation of the Neptune encounter. As a result, it was not possible for the developers and the operators to work together for extended periods in order to thoroughly walk through the capabilities within SHARP.

The response by the telecommunications operators to SHARP left two major impressions with the developers. The first was that the operators were enthusiastic about the potential of a fully operational SHARP system. The second was that they would have liked a more responsive display interface.

SHARP functioned well during the installation in the real-time area. A number of problems in the software were discovered, but this was not surprising. This version of SHARP was not built to be an operation system with minimal bugs; it was built as rapidly as possible to get a prototype running to determine if the ideas motivating the SHARP architecture were correct. The developers feel that the SHARP demonstration proved that the ideas were correct, but that work needs to be done to improve their implementation. This work is under way.

7. Conclusions and Continuing Work

Spacecraft and ground data systems operations present a rigorous environment in the area of monitoring and anomaly detection and diagnosis. With a number of planetary missions scheduled for the near future, the effort to staff and support these operations will present significant challenges.

The SHARP system was developed to address the challenges of automation in a multi-mission operations environment by augmenting conventional automation technologies with artificial intelligence. Its successful development and demonstration have led to a number of important conclusions. First and foremost, artificial intelligence technology is ready for application to spaceflight operations. The techniques can be used alongside conventional computer science techniques, and diagnostic knowledge-based systems can be embedded in the resulting application system. Acceptable real-time performance can be achieved. SHARP was never pushed to the limit of its speed or memory resources; in fact, most of its time was spent idle, waiting for new engineering data to process. This gives us confidence for broadening the approach in SHARP to multiple spacecraft subsystems.

The evaluation by Voyager personnel also taught us that the types of automation provided by SHARP are highly desired by operations personnel, and are not viewed as job-threatening (although they may be in some cases). Operators were able to readily use the system with minimal training, and were enthusiastic about using the wide variety of graphical displays and options.

7.1. Benefits of SHARP to Mission Operations

There are four principal areas where the JPL telecommunications users of SHARP expect to see benefits from application of the system and its descendents, which we are now developing. These areas are safety, workforce savings, reliability, and productivity.

Through its accurate detection, analysis, and tracking of the antenna drift and pointing conditions during the encounter, SHARP showed that it can detect and analyze important problems in a matter of seconds which currently take human operators minutes or hours. This provides an extra margin for ensuring the safety of the spacecraft, and thereby supports the success of the mission as a whole. The SHARP Voyager telecommunications domain expert, a man with over 20 years of experience who has cognizance not only for Voyager telecommunications operations but for other spacecraft as well, has stated publicly that the Soviets would not have lost the first Phobos spacecraft if they had applied SHARP to their telecommunications operations. One of the stated causes of the loss of the Phobos spacecraft was that the spacecraft antenna drifted until the telecommunications link was lost due to a faulty attitude control command.

A second major benefit from application of SHARP will be in the area of workforce savings. Through its automation of many manual functions, SHARP promises to reduce the real-time link analysis operations staff by a factor of five, and there is reason to believe that similar savings may be possible in other operations areas. This is precisely the type of benefit from automation which is necessary to support the single multi-mission flight team in the new JPL Space Flight Operations Center.

The system-wide status monitoring afforded by SHARP helps operators assure correct telecommunications system configuration. This is expected to reduce the number of commanding errors to the spacecraft and ground systems, and thereby reduce the loss or corruption of data due to configuration problems.

Finally, the SHARP system is expected to enhance the productivity of operations personnel by freeing them from the tedium of watching raw data and interpreting it for themselves. SHARP shifts the burden of routine monitoring operations, and most of the boring, manual computations which are involved, away from the operator to itself. This will enable operations personnel to perform required analyses more efficiently, and to exert a higher level of "supervisory monitoring" over multiple spacecraft subsystems on multiple spacecraft.

7.2. Current SHARP Development Effort

SHARP is now being extended and developed to a higher level of readiness so that flight projects such as Voyager, Magellan, Galileo, and others can use it directly. The system will be completed in 1990 and delivered to the Space Flight Operations Center for further evaluation and application to Magellan telecommunications. Separately, SHARP is also being applied to the Deep Space Network, Network Operations Control Center at JPL, with an operational system planned for 1991. Applications for remote monitoring and control of spaceborne instruments and experiments are also under consideration.

A major aspect of the current activities is to move SHARP from the Symbolics platform to a Unix platform. This will increase its portability, and allow a more ready access to data within the different operation centers at JPL.

8. References

Atkinson, David J., James, Mark L. and Lawson, Denise L. "System Autonomy Demonstration: Subsystem Selection", Technical Memorandum 366.3/87-234, Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California, November 10, 1987.

Doyle, Jon, "A Truth Maintenance System," *Artificial Intelligence*, Vol. 12, No. 3, 1979.

James, Mark, and Atkinson, David, "STAR*TOOL — An Environment and Language for Expert System Implementation," *Jet Propulsion Laboratory Report NTR C-17536*, Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California, August 19, 1988.

McDermott, D. V., "Very Large PLANNER-type Databases," Memo AIM-339, AI Laboratory, Massachusetts Institute of Technology, Cambridge, Massachusetts, 1975.

Reiger, Chuck, "Spontaneous Computation and Its Role in AI Modeling," in *Pattern-Directed Inference Systems* (Waterman, D.A., and Hayes-Roth, Frederick, eds.), Academic Press, Inc., Orlando, Florida, 1978.

Woods, W.A., "Transition Network Grammars for Natural Language Analysis," *Communications of the ACM*, Vol. 13, No. 10, pp. 591-606, October 6, 1970.

A. Appendix A - Predicts

A.1. Raw Predicts: Content and Parsing

Each raw predict file contains information which is divided into four sections:

Uplink Carrier Design Control Table
Downlink Carrier Design Control Table
Telemetry Channel Design Control Table
Telemetry Performance Tabulations

Figures A-1 to A-4 (at the end of this appendix) show sample pages of each type of information. The SHARP predicts module is concerned only with particular predict values from the telemetry performance tabulations section. As seen in Figure A-4, this section contains header information identifying the spacecraft, the x-band transmitter power level, and the Deep Space Station (DSS), followed by actual predict values for elevation, uplink carrier, total power (PT), downlink carrier, system noise temperature (SNT), data power with respect to noise (PD/N_0), telemetry margin, tolerance, and Bit SNR (ST/N_0).

Heuristic adaptive parsing is implemented for SHARP's raw predicts database. Periodically the format of this data source changes without mission operations being notified. Generally this would require the raw predicts parser to be rewritten to incorporate the new format. However, SHARP utilizes Augmented Transition Network (ATN) techniques to accomplish adaptive parsing. The advantage of such an ATN lies in its ability to parse the database according to semantic content rather than syntactic structure. The raw predicts database can therefore be modified and yet remain successfully parsable. This heuristically controlled, format-insensitive parsing ensures continuity despite format modifications in predict generation.

The following sections give notes regarding the generation of the other predict information from the raw predicts.

A.2. Generation of Pass Predicts from Raw Predicts

I. Select the appropriate Pass parameters:
S/C: 32 (Voyager 2)
DSS: from ISOE (DSS)
X-band Transmitter Power: from ISOE (XbdXmtPwr)
Time: Time of Pass

II. Based on the Pass parameters, extract values from Predict File database:
Elev (deg), Az (deg), UL Carr (dbM), PT (dBm), SNT (k), PD/N_0 (dB)

III. Modify the raw predict values of PT and PD/N_0 for RFS mode as follows:

Predict File Power	ISOE XbdXmtLevel	Action
Hi	Hi	PT=PT and $PD/N_0 = PD/N_0$
Hi	Lo	PT=PT - 1.9 and $PD/N_0 = PD/N_0 - 1.9$
Lo	Hi	PT=PT + 1.9 and $PD/N_0 = PD/N_0 + 1.9$
Lo	Lo	PT=PT and $PD/N_0 = PD/N_0$

IV. Calculate pass predicts using the following algorithms:

1. SSNR Pass Predict (dB)
$= PD/N_0 - 10\log(\text{DataRate}) + 20\log(\sin\phi) - \text{SystemLoss} - 3.01$
All logarithms are base 10.
PD/N_0 value is in dB.
DataRate is obtained from ISOE. If DataRate in kbps, convert DataRate to bps.
$$\phi = 20 + \frac{\text{XbdModIndex}}{47} * 60$$
(i.e. ModIndex = 0 => $\phi = 20^{\circ}$)
ϕ value is in degrees, not radians
To determine the System Loss, use:

DataRate	SystemLoss
> .16 kbps	0.5 dB
.16 kbps	0.6 dB
.08 kbps	0.8 dB
.04 kbps	1.2 dB
< .04 kbps	2.0 dB

The term -3.01 is a result of converting from BSNR to SSNR.

2. DSS AGC Pass Predict (dBm):
$= PT + 20\log(\cos\phi)$ if ISOE XbdRng is off.
$= PT + 20\log(\cos\phi) - 0.2$ if ISOE XbdRng is on.
PT value is in dBm.
ϕ value is same as previous calculation.

3. S/C AGC Pass Predict (dBm):
$$= 10\log\frac{18}{\text{XmtPwrKw}} + \text{UL Carr Raw Predict} + \text{RSup}$$
XmtPwrKw = Transmitter Power in Kilowatts (from ISOE)
RSup = Ranging Suppression (DSS Actual, currently not channelized)

4. SNT Pass Predict (k):
= SNT Raw Predict

5. Elev Pass Predict (deg):
= Elev Raw Predict

6. Az Pass Predict (deg):
= Az Raw Predict

The hourly Pass Predicts are then interpolated to produce values per 15 seconds.

A.3. Generation of Instantaneous Predicts from Pass Predicts

I. Calculate Antenna Pointing Error, E_p, using S/C actuals, Pitch and Yaw:

$E_p = \sqrt{PE^2 + YE^2}$, where

PE = Pitch Axis Error = k(Pitch Telemetry - DeadBand midpoint)
YE = Yaw Axis Error = k(Yaw Telemetry - DeadBand midpoint)

Pitch Telemetry = E-174 (DN)
Yaw Telemetry = E-181 (DN)
DeadBand midpoint = 128
k = .00586

II. Calculate Antenna Pointing Loss, L_p:

L_p (dB) = 10 log $\cos^5$ ($100E_p$), or 50 log Cos ($100E_p$)

III. Calculate instantaneous predicts using the following algorithms:

1. Instantaneous SSNR Predict (dB):
   = SSNR Pass Predict (dB) - L_p (dB) + 10 log $\left(\frac{\text{SNT Pass Predict (K)}}{\text{Actual SNT (K)}}\right)$,

   Actual SNT = M-765 (HS) or M-768 (WB)

2. Instantaneous DSS AGC Predict (dB):
   = DSS AGC Pass Predict (dBm) - L_p (dB)

3. Instantaneous DSS SNT Predict (K):
   = SNT Pass Predict (no adjustments).

4. Instantaneous S/C AGC Predict (dB):
   = S/C AGC Pass Predict (dBm) + 10 log $\left(\frac{\text{Actual Xmt Pwr}}{\text{Pred Xmt Pwr}}\right)$ - CSup - RSup
   [RTLTadj],

   Actual Transmit Power = M-776
   Predicted Transmit Power = modifiable static value = 18.0 kw
   CSup = Command suppression (DSS Actual, currently not channelized, so use 0)
   RSup = Ranging suppression (DSS Actual, currently not channelized, so use 0)
   This predict is adjusted for RTLT. It is compared to the Actual S/C AGC one RTLT from now.

5. Instantaneous DSS ELEV Predict (deg):
   = ELEV Pass Predict (no adjustments).

6. Instantaneous DSS AZ Predict (deg):
   = AZ Pass Predict (no adjustments).

7. Instantaneous DSS EXC FREQ Predict (Hz):
   = Start Freq (Hz)+ (Current Time - Start Time)*Ramp Rate (Hz/sec)

8. Instantaneous DSS XMT PWR Predict (kw):
   = ISOE XmtPwrKw Value at time t (no adjustments)

A.4. Residual Calculations

I. Obtain actual values from the following telemetry channels:

Actual	Channel
DSS SSNR =	M-764 (HS) or M-767 (WB)
DSS AGC =	M-766 (HS) or M-769 (WB)
DSS SNT =	M-765 (HS) or M-768 (WB)
S/C AGC A=	E-025 (S/C 32)
DSS ELEV =	M-782
DSS AZ =	M-781
DSS EXC FREQ =	M-777
DSS XMT PWR =	M-776

II. Calculate residuals using the following algorithms:

1. DSS SSNR Residual (dB):
$$= \frac{\text{DSS SSNR Actual (DN)}}{100} - \text{DSS SSNR Inst Predict (dB)}$$

2. DSS AGC Residual (dB):
$$= \frac{\text{DSS AGC Actual (DN)}}{10} - \text{DSS AGC Inst Predict (dB)}$$

3. DSS SNT Residual (DN):
= DSS SNT Actual (DN_K)- DSS SNT Inst Predict (K)

4. S/C AGC Residual (dB):
= S/C AGC Actual (EU) - S/C AGC Inst Predict (dB)

5. DSS ELEV Residual (DN):
= ELEV Actual (DN_{deg}) - ELEV Inst Predict (deg)

6. DSS AZ Residual (DN):
= AZ Actual (DN_{deg}) - AZ Inst Predict (deg)

7. DSS EXC FREQ Residual (DN):
= DSS EXC FREQ Actual (DN_{Hz}) - DSS EXC FREQ Inst Predict (Hz)

8. DSS XMT PWR Residual (dB):
$$= 10 \log \frac{\text{DSS XMT PWR Actual } (DN_{kw})}{\text{DSS XMT PWR Inst Predict (kw)}}$$

VOYAGER UPLINK CARRIER DESIGN CONTROL TABLE

VOY 2 (JSX), 70M/18KW/30HZ, 0DB RNG, 0DB CMD, CLR WTHR
X-BAND TWT HP, HGA/NLC, 14.4 KBPS CODED, 2-WAY RADIO LOSSES

EPOCH 00/000/00/00 SPACECRAFT 2 STATION 14

TIME IN MISSION 89/119/08/00 TIME FROM EPOCH 32627 07:59

	DESIGN	FAV TOL	ADV TOL	MEAN	VARIANCE
TRANSMITTER PARAMETERS					
1) RF POWER, DBM	72.55	.50	-.50	72.6	.04
POWER OUTPUT = 18.0 KW					
TRANSMIT CIRCUIT LOSS, DB	.00	.00	.00	.0	.00
2) ANTENNA GAIN, DBI	62.10	.30	-.70	61.9	.08
ELEV ANGLE = 7.63 DEG					
3) POINTING LOSS, DB	-.03	.03	-.03		
PATH PARAMETERS					
4) SPACE LOSS, DB	-291.59			-291.6	.00
FREQ = 2113.31 MHZ					
RANGE = 4.285+09 KM					
= 28.64 AU					
5) ATMOSPHERIC ATTENUATION,DB	-.26	.00	.00	-.3	.00
RECEIVER PARAMETERS					
6) POLARIZATION LOSS, DB	-.12	.12	-.18		
7) ANTENNA GAIN, DBI	34.60	.39	-.39	34.5	.03
8) POINTING ERROR, DB	-.10	.10	-.10	-.1	.00
LIMIT CYCLE, DEG	.05	-.05	.00		
ANGULAR ERRORS, DEG	.00	.00	.00		
9) REC CIRCUIT LOSS, DB	.00	.00	.00	.0	.00
10) NOISE SPEC DENS, DBM/HZ	-166.71	-.10	.16	-166.7	.00
OPERATING TEMP, K	1545.00	-34.00	59.00		
HOT BODY NOISE, K	.00	.00	.00		
11) CARR THR NOISE BW, DB-HZ	12.72	-.24	.23	12.7	.01
POWER SUMMARY					
12) RCVD POWER,PT,DBM (1+2+3+4+5+6+7+8+9)				-123.1	.16
13) RCVD PT/NO, DB-HZ (12-10)				43.6	.16
14) RANGING SUPPRESSION, DB	.00	.00	.00	.0	.00
15) COMMAND SUPPRESSION, DB	.00	.00	.00	.0	.00
16) CARR PWR/TOT PWR,DB(14+15)				.0	.00
17) RCVD CARR PWR, DBM (12+16)				-123.1	.16
18) CARR SNR IN 2BLO,DB(17-10-11)				30.9	.17
				2.0S =	.8

Figure A-1 Uplink Carrier Design Control Table

VOYAGER DOWNLINK CARRIER DESIGN CONTROL TABLE

VOY 2 (JSX), 70M/18KW/30HZ, 0DB RNG, 0DB CMD, CLR WTHR
X-BAND TWT HP, HGA/NLC, 14.4 KBPS CODED, 2-WAY RADIO LOSSES

EPOCH 00/000/00/00 SPACECRAFT 2 STATION 14

TIME IN MISSION 89/119/08/00 TIME FROM EPOCH 32627 07:59

	DESIGN	FAV TOL	ADV TOL	MEAN	VARIANCE
TRANSMITTER PARAMETERS					
1) RF POWER TO ANTENNA, DBM				42.9	.04
TRANSMITTER POWER, DBM	42.87	.50	-.50	42.9	.04
TRANSMIT CIRCUIT LOSS, DB	.00	.00	.00	.0	.00
2) ANTENNA CIRCUIT LOSS, DB	.00	.00	.00	.0	.00
3) ANTENNA GAIN, DBI	48.20	.26	-.26	48.2	.01
4) POINTING ERROR, DB	-.10	.10	-.10	-.1	.00
LIMIT CYCLE, DEG	.05	-.05	.00		
ANGULAR ERRORS, DEG	.00	.00	.00		
PATH PARAMETERS					
5) SPACE LOSS, DB	-303.59			-303.6	.00
FREQ = 8415.00 MHZ					
RANGE = 4.285+09 KM					
= 28.64 AU					
6) ATMOSPHERIC ATTENUATION,DB	-.23	.00	.00	-.2	.00
RECEIVER PARAMETERS					
7) POLARIZATION LOSS, DB	-.08	.08	-.11		
8) ANTENNA GAIN, DBI	73.34	.60	-.60	73.1	.14
9) POINTING LOSS, DB	-.20	.20	-.20		
10) NOISE SPEC DENS, DBM/HZ	-182.71	-.50	.45	-182.7	.03
TOTAL SYSTEM NOISE TEMP, K	38.79	-4.24	4.24		
RECEIVER TEMPERATURE, K	13.20	-3.00	3.00		
GROUND CONTRIBUTION, K	9.59	-3.00	3.00		
GALACTIC CONTRIBUTION, K	2.56	.00	.00		
ATMOSPHERIC CONTRIB, K	13.44	.00	.00		
HOT BODY NOISE, K	.00	.00	.00		
ELEV ANGLE = 7.63 DEG					
11) CARR THR NOISE BW, DB-HZ	14.77	-.46	.41	14.8	.03
POWER SUMMARY					
12) RCVD POWER,PT,DBM (1+2+3+4+5+6+7+8+9)				-139.8	.19
13) RCVD PT/NO, DB-HZ, (12-10)				43.0	.22
14) RANGING SUPPRESSION, DB	-.22	.05	-.05	-.2	.00
15) TELEMETRY SUPPRESSION, DB	-12.33	.37	-.39	-12.3	.02
16) CARR PWR/TOT PWR,DB(14+15)				-12.6	.02
17) RCVD CARR PWR,DBM(12+16)				-152.3	.22
18) CARR SNR IN 2BLO,DB(17-10-11)				15.6	.27
				2.0S = 1.0	

Figure A-2 Downlink Carrier Design Control Table

VOYAGER TELEMETRY CHANNEL DESIGN CONTROL TABLE

	DESIGN	FAV TOL	ADV TOL	MEAN	VARIANCE
DATA CHANNEL PERFORMANCE					
19) DATA BIT RATE, DB	41.58	.00	.00	41.6	.00
BIT RATE = 14400.0 BPS					
20) DATA PWR/TOTAL PWR, DB	-.26	.02	-.02	-.3	.00
TLM MOD INDEX = 76.0 DEG					
21) DATA PWR TO RCVR,DBM(12+14+20)				-140.3	.19
22) ST/NO TO RCVR,DB(21-19-10)				.9	.22
23) SYSTEM LOSSES, DB	-.85	.07	-.20	-.9	.00
RADIO LOSS, DB	-.48	.05	-.05		
DEMOD, DETECT LOSS, DB	-.19	.02	-.19		
WAVEFORM DIST LOSS, DB	-.18	.04	-.03		
24) ST/NO OUTPUT, DB (22+23)				-.0	.22
25) THRESHOLD ST/NO, DB	2.31	.00	.00	2.3	.00
26) PERFORMANCE MARGIN,DB(24-25)				-2.3	.22
				2.0S =	.9

Figure A-3 Telemetry Channel Design Control Table

TELEMETRY PERFORMANCE TABULATIONS

VOY 2 (JSX), 70M/18KW/30HZ, 0DB RNG, 0DB CMD, CLR WTHR
X-BAND TWT HP, HGA/NLC, 14.4 KBPS CODED, 2-WAY RADIO LOSSES

S/C = 2 DSS = 14 MOD INDEX =76.0 TOL = 2.0 SIGMA

GMT TIME (DDD HH MM SS)	ELEV (DEG)	UL CARR (DBM)	PT (DBM)	DL CARR (DBM)	SNT (K)	PD/N0 (DB)	TLM MARG (DB)	TOL (DB)	ST/N0 (DB)
119 08 00 00	7.63	-123.06	-139.79	-152.34	38.6	42.73	-2.32	.94	-.01
119 08 30 00	12.55	-122.95	-139.54	-152.10	31.9	43.80	-1.14	.96	1.17
119 09 00 00	17.12	-122.91	-139.38	-151.94	28.7	44.42	-.46	.98	1.85
119 09 30 00	21.26	-122.89	-139.27	-151.82	26.9	44.82	-.03	.99	2.28
119 10 00 00	24.89	-122.88	-139.18	-151.73	25.6	45.11	.29	1.00	2.60
119 10 30 00	27.91	-122.87	-139.11	-151.67	24.8	45.32	.50	1.01	2.81
119 11 00 00	30.24	-122.86	-139.07	-151.62	24.3	45.46	.65	1.02	2.96
119 11 30 00	31.78	-122.86	-139.05	-151.60	23.9	45.54	.74	1.02	3.05
119 12 00 00	32.47	-122.86	-139.04	-151.59	23.8	45.58	.77	1.02	3.08
119 12 30 00	32.28	-122.86	-139.04	-151.59	23.8	45.57	.76	1.02	3.07
119 13 00 00	31.23	-122.86	-139.06	-151.61	24.1	45.51	.71	1.02	3.02
119 13 30 00	29.35	-122.86	-139.09	-151.64	24.5	45.41	.60	1.01	2.91
119 14 00 00	26.72	-122.87	-139.14	-151.69	25.1	45.24	.42	1.01	2.73
119 14 30 00	23.43	-122.88	-139.21	-151.76	26.1	45.00	.17	1.00	2.48
119 15 00 00	19.57	-122.90	-139.31	-151.86	27.5	44.67	-.19	.99	2.12
119 15 30 00	15.24	-122.93	-139.44	-151.99	29.7	44.21	-.69	.97	1.62
119 16 00 00	10.52	-122.98	-139.63	-152.18	34.0	43.44	-1.54	.95	.77
**									
120 08 00 00	8.29	-123.03	-139.74	-152.30	37.3	42.92	-2.10	.94	.21
120 08 30 00	13.17	-122.94	-139.52	-152.07	31.3	43.90	-1.03	.96	1.28
120 09 00 00	17.69	-122.91	-139.37	-151.92	28.4	44.48	-.40	.98	1.91

Figure A-4 Telemetry Performance Tabulations

B. Appendix B - Integrated Sequence of Events

There are three forms in which the ISOE may be present. The first form is the original, unabridged file that is retrieved from a Univac. This file is processed to generate the second form of the ISOE which is a text file that contains only the information from the original ISOE that is relevant to telecommunications operations. The final form for the ISOE is generated by a program that takes the second form of the ISOE and generates a lisp file suitable for SHARP's ISOE database.

One of the knowledge acquisition tasks of SHARP was to learn what events to extract from the ISOE, the interpretation of those events, and how the events impacted the building of the ISOE database.

B.1. Content of Original ISOE File

Two entries from the original ISOE appear as follows:

```
 ERTB=89016040208X  OPCH  ACE,,++++++++++++++++++++++++++++++++,,D---,,0
C                   4755:22:005F,CR05,,RFS,2,89-016/00:00:00
N                   ,,            RFS STATUS
N                   ,, TWNC = ON <--> FDS MODE = CR05
N                   ,,S-BD PWR = ON    X-BD PWR = LO
N                   ,,     RNG = ON         RNG = ON
N                   ,,     M.I.= 28          M.I.= 32
N                   ,,++++++++++++++++++++++++++++++++
 TRMB=89016105500X  OPCH  ACE,, AOS DSS-63 PASS-4178 (U/L PASS),,D---,63
C                    AOS,,,,,2,,AOS,
N                   ,,SET HI POWER TRANSMITTER TO 90KW
N                   ,,PER TELECOM TRACKING FORM
```

Each event contains 1 or more lines. The first line of an event begins with a space. The second line of an event begins with a "C". Any additional lines begin with an "N".

The first line of an event begins information denoting whether the information pertains to a satellite or a ground station. Following the equal sign is the time for the event. In the first example above, this is 89016040208. This gives the time in Greenwich Mean Time as YYDDDHHMMSS. For this example: the year is 1989. The day is 016. The hour is 4. The minute is 2. The second is 8.

The remainder of the first line and the second line (ignoring the "C" in the first column and the 19 leading spaces) consist of a number of fields separated by commas. The field before the first comma is considered to be field 0; the field after the first comma is field 1, etc.

The parsing process does not process all entries of the ISOE file, nor does it process all fields within an event.

B.2. Content of the Intermediate Text Files

These files are generated as a result of parsing each ISOE file. Each event in the ISOE is given a number during the parsing process. This number, the day and time of the event, and a concise description of the event is printed - one event per line.

A sample from one of the intermediate files is:

```
0010  016 02 45 00 LOS 45
0015  016 04 02 08                RFS ON CR05 ON ON 28 LOW ON 32 4 2 8
0026  016 10 55 00 AOS 63
0032  016 13 45 00 AOS 65
0034  016 14 05 00 LOS 63
0040  016 14 44 31                RFS ON CR05 ON ON 28 LOW ON 32 4 2 8
0056  016 14 44 34        CC16C  (7 0 1 1)
0067  016 15 05 00 LOS 65
0089  016 18 30 00 AOS 43
0093  016 18 57 07        AC7PAR (6741 001700)
0094  016 18 57 17        AC7PAR (6742 004540)
0102  016 19 04 30        CC16C  (7 0 0 0)
0105  016 19 08 08        DC2PR
0109  016 19 14 10        CC16C  (7 0 1 1)
```

B.3. Original ISOE File Event Parsing

The parser begins at the start of the original ISOE file and processes each event sequentially. The parser first examines a number of characters and fields to determine if the event is one to skip because it is not of interest to the telecom operators.

The following examples describe how events are processed. Four events are specified by field 12, one by field 9, the remainder by field 3.

Example 1. If the first 3 characters of field 12 are "AOS", this is an "Acquisition of Signal" event. The first 2 characters of field 5 give the ground station acquiring the signal. If the last 10 characters of field 2 are "(U/L PASS)", we know the transmitter is to be turned on at the ground station. Field 13 gives the power level in kilowatts for the transmitter. If there is no value for field 13, a default of 18 kilowatts is assumed. Written to the intermediate text file are the event number, the day and time, "AOS", the station number, and if the transmitter was on, "XmtPwrKw" with the power level value. For example:

```
0013 135 08 15 00   AOS 12 XmtPwrKw 18.
```

Example 2. If the first 3 characters of field 12 are "FDS", this is an FDS event. The first continuation line that contains the character "/" is parsed to determine the data mode, the data rate, and the s-rate. The characters describing the data mode are the 22nd character of the line to the first "/". The data rate is between the first "/" and the second "/" except for the first character. The s-rate is from the second "/" to either the end of the line or a comma. Written to the intermediate text file are the event number, the day and time, "FDS", the mode, the data rate, and the s-rate. For example:

```
0215 136 16 22 22                        FDS GS4A 4.8K S40
```

Example 3. If the first 5 characters of field 3 are "CC16C" and the first 2 characters of field 2 are "(1" or "(7", then take each of the four digits within the parentheses of field 2 and put spaces between for the output to the intermediate file. For example:

```
0420 138 13 30 14              CC16C (7 0 0 0)
```

There are approximately 50 different event types that are processed in a similar fashion.

B.4. Extracted ISOE Translation

Neither the original ISOE file nor the output of the program that extracts the ISOE events are comprehensible to persons unfamiliar with the notational conventions being used. The final processing of the ISOE data accomplishes two things: build a database that can be used by the rest of the SHARP system and translate the ISOE events into more understandable terminology.

The source for interpretation of events came from the domain expert and the Voyager commands list found in JPL Voyager internal document 618-804. The knowledge engineering task converted this information into a collection of about 15 tables that the programmers on the SHARP team could then encode into SHARP. An example of one of these tables is as follows:

CC7xx (Control)

SS (a) Sun Sensor Control			
a	Data Present	S/C Maneuver	Comments
1	nil, t in 105 sec	-	Search Enable
6	nil	-	Sun Point Enable
7	t	nil	Earth Point Enable
Other	IGNORE		

CC2x (Status Select)

x	STATUS	
BP	SbdExcSelect	1
BRP		2
CP	SbdXmtSelect	TWTA
CRP		SSA
FP	RcvrSelect	1
FRP		2
GP	XbdXmtPwr	On
GRP		Off
JP	XbdExcSelect	1
JRP		2
KP	SbdXmtPwr	On
KRP		Off
LP	XbdXmtSelect	1
LRP		2
MP	SbdExcPwr	On
MRP		Off

The construction of the database involved more than simply converting the extracted information into a Lisp format. Extra effort was required because events could have impact over a time period that was larger than the event itself. Also, there could be interactions between different event types.

Two examples from the notes generated during the knowledge acquisition period are:

Memory Readout
Keywords:
Memory readouts are indicated in the numerical field of SC06BB commands.
Info obtained:
All data should be ignored for the duration of this activity
Telemetry can be re-acquired within 5 minutes.

AACS Maneuver

Keywords:

Beginning of activity: AC7MDP (6) (All Axes Inertial)
or AC7MDP (5) (Roll Axis Inertial)

End of activity: CC7PC (307) (star acquired)

Info obtained:

Telemetry data will be lost or distorted during maneuver

Comments:

In the case of some maneuvers, a CHKPNT precedes the AC7MDP command. If the CHKPNT indicates a MINI-CRSMVR, then there will be data available during part of the maneuver. Anything other than mini-crsmvr (such as TCM) means that no data will be available until the maneuver is completed and a star has been acquired. Following the AC7MDP command there should be either an AC7VCD (gyro drift turn) or AC7TCD (turn command). Both indicate the start of a turn. The ensuing number in parentheses indicates the type of turn:

AC7VCD (1) = Pitch turn (lose telemetry signal immediately)
AC7VCD (2) = Yaw turn (lose telemetry signal immediately)
AC7VCD (3) = Roll turn (variations in telemetry signal)

For Roll Axis Inertial, there can only be AC7VCD (3).

The above information holds true for the AC7TCD commands as well.

B.5. ISOE Database Access

The ISOE database is a collection of files each containing the ISOE data for a period of one week. The database queries have three arguments:

- where: a query is directed at one of the DSS's or the spacecraft;
- what: the state, such as the X-band power level of the spacecraft's transmitter; and
- when: at this time, what was the value of the state.

Certain settings of components on the spacecraft change slowly if at all. Therefore, multiple files are often searched in order to determine the state of a component. When a query is made to the database, a search is made in the file for the week that contains the time within the query. If an event that specifies the requested state cannot be found at the designated time or earlier in that week, then the previous week's data is searched. This search continues until all files in the database have been examined. If no event is found, then a default value is returned. The database cache described in Section 3.2.4 prevents this exhaustive search from occurring every time a query is made.

C. Appendix C - STAR*TOOL

Knowledge-based systems for automated task planning, monitoring, diagnosis, and other applications require a variety of software modules based on artificial intelligence concepts and advanced programming techniques. The design and implementation of such modules requires considerable programming talent and time, and a background in theoretical artificial intelligence. Sophisticated software development tools that can speed the research and development of new artificial intelligence applications are therefore highly desirable. The STAR*TOOL system was developed specifically for this purpose. STAR*TOOL is currently available for license to industry and academia from the California Institute of Technology, Office of Patents and Technology Utilization.

The STAR*TOOL system is a set of high-level software tools that assist programmers in the creation of efficient artificial intelligence and knowledge-based (expert systems) software. Included in the system are facilities for developing reasoning processes, memory-data structures and knowledge bases, blackboard systems, and spontaneous computation daemons.

Computational efficiency and high performance are especially critical in artificial intelligence software. This consideration has been an important objective of STAR*TOOL, and has led to its design as a toolbox of AI facilities that may be used independently or collectively in the development of knowledge-based systems.

STAR*TOOL facilities are invoked directly by the programmer in the Common LISP language. For improved efficiency, an optional optimization compiler was developed to generate highly efficient Common LISP code.

When an application program is developed in STAR*TOOL, STAR*TOOL first translates the program to Common LISP code. It can then optionally pass the resulting Common LISP code through a source-to-source LISP code optimizer. STAR*TOOL generates code that is tailored for each application. There are no intermediate levels of interpretation for execution, unlike many other software systems. STAR*TOOL programs are executed directly by the LISP interpreter and compiled directly by the LISP compiler. This results in greater speed and better portability to other computers. STAR*TOOL augments the Common LISP programming language and environment so that programs written in STAR*TOOL have direct use of all of the features of the underlying LISP software system and computing environment.

Because a single line of STAR*TOOL code can translate into many lines of Common LISP code, all error messages generated from the STAR*TOOL compiler and run-time environment reference the original line of STAR*TOOL source code. In the case of run-time errors, the resulting Common LISP translation is also referenced.

STAR*TOOL provides the LISP programmer with the necessary software tools to build a wide variety of reasoning and inference engines such as planners, diagnosticians and simulators. STAR*TOOL's efficient implementation also enables the building of real-time monitors. When STAR*TOOL is run in an environment which supports multiple programming languages, STAR*TOOL's capabilities can be utilized via local and remote procedure calls and through shared data structures. This enables portions of the system to be developed in the most suitable programming language and allows such portions to be connected to the LISP application in a straightforward, natural way.

STAR*TOOL enables and encourages the development of embedded expert systems. Thus, STAR*TOOL could be a supervisor of many other systems written in either STAR*TOOL or conventional programming languages. Most of the software tools provided by STAR*TOOL, like the blackboard, memory model, process model and process scheduler can operate independently of one another. However, when operated in combination, they form a fully-integrated synergistic set of software tools. Since the user is able to choose only those portions of STAR*TOOL that are applicable to a given application, the problem of storing unnecessary excess software is eliminated. Thus, the resulting application program can run on a smaller computer than the one on which it was developed.

D. Appendix D - GEd (Graphics Editor) and GObS (Graphics Object System)

D.1. Introduction

This publication describes GEd and GObS, a graphics editor and object-oriented graphics substrate designed to assist in the design and implementation of user interfaces on Symbolics Lisp machines. GEd and GObS provide high and medium-level facilities for using the built-in graphics routines on the Symbolics.

D.2. GEd

GEd is a mouse-driven graphics editor which allows the user to draw and edit simple diagrams using the mouse and to use those diagrams as graphics objects in other programs. Shapes that can be added are lines, rectangles, ellipses, circles, polygons, and text.

Shapes can be combined together into groups so that they may be edited as a unit and share attributes.

The editing functionality within GEd allows the user to move, duplicate, delete, expose, and bury shapes or groups. The visual characteristics of a shape can be changed. Borders of shapes can have thickness, color, and patterns. Interiors of shapes can have colors and patterns and can be transparent or opaque. Lines can have arrowheads. Polygons can be drawn as smooth curves. Circles and ellipses can be arcs. Text can have color and different fonts.

Drawings that are made in GEd can be saved to files that can be loaded by other programs. The objects created within GEd can then be manipulated by these other programs. Within SHARP GEd was used in a number of areas. As an example, the block diagram drawings were built in GED. The structures provided by GEd made it very easy to write the code necessary to change the color of the various components as their status changed.

D.3. GObS

GObS is the substrate upon which GEd is built. It is a collection of flavors and methods for the construction of graphics objects. The following flavors are supported:

- SHAPE (abstract)
- SCALE (abstract)
- LINEAR-SCALE
- CIRCULAR-SCALE
- RECTANGLE
- CIRCLE
- REGULAR-POLYGON
- ELLIPSE
- LINE
- POLYGON
- TEXT
- GROUP

SHAPE is an abstract flavor; it is not intended to be instantiated. All the other flavors inherit from SHAPE. It carries information on a shape's location, size, interior and exterior color, current output stream, and certain other attributes common to all shapes.

The SHAPE flavor provides the following methods:

SHOW - makes a shape visible
HIDE - makes a shape invisible
REFRESH - redraws a shape if it is visible

In addition, it has methods that control various parameters of a shape's appearance such as color, location, size, etc.

TECHNICAL REPORT STANDARD TITLE PAGE

1. Report No. 90-21	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle A REPORT ON SHARP AND THE VOYAGER NEPTUNE ENCOUNTER		5. Report Date August 15, 1990
		6. Performing Organization Code
7. Author(s) Martin, R.G., Editor; D.J. Atkinson, M.L. James, D.L. Lawson, and H.J. Porta		8. Performing Organization Report No.
9. Performing Organization Name and Address JET PROPULSION LABORATORY California Institute of Technology 4800 Oak Grove Drive Pasadena, California 91109		10. Work Unit No.
		11. Contract or Grant No. NAS7-918
		13. Type of Report and Period Covered JPL External Publication
12. Sponsoring Agency Name and Address NATIONAL AERONAUTICS AND SPACE ADMINISTRATION Washington, D.C. 20546		14. Sponsoring Agency Code

15. Supplementary Notes

ORIGINAL PAGE IS
OF POOR QUALITY

16. Abstract

This publication provides a report on the development and application of the Spacecraft Health Automated Reasoning Prototype (SHARP) for the operations of the telecommunications systems and link analysis functions in Voyager mission operations. The report provides an overview of the design and functional description of the SHARP system as it was applied to Voyager. Some of the current problems and motivations for automation in real-time mission operations are discussed, as are the specific solutions that SHARP provides. The application of SHARP to Voyager telecommunications had the goal of being a proof-of-capability demonstration of artificial intelligence as applied to the problem of real-time monitoring functions in planetary mission operations. As part of achieving this central goal, the SHARP application effort was also required to address the issue of the design of an appropriate software system architecture for a ground-based, highly automated spacecraft monitoring system for mission operations, including methods for: (1) embedding a knowledge-based expert system for fault detection, isolation, and recovery within this architecture; (2) acquiring, managing, and fusing the multiple sources of information used by operations personnel; and (3) providing information-rich displays to human operators who need to exercise the capabilities of the automated system. In this regard, SHARP has provided the Jet Propulsion Laboratory with an excellent example of how advanced artificial intelligence techniques can be smoothly integrated with a variety of conventionally programmed software modules, as well as guidance and solutions for many questions about automation in mission operations.

17. Key Words (Selected by Author(s))	18. Distribution Statement
121: Ground Support Systems and Facilities (Space); 235: Computer Systems; and 383: Voyager Project -- Artificial Intelligence; Monitoring; Diagnosis; Spacecraft	Unclassified; unlimited

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	100	

JPL 0184 R 9/83