

1N-61-CR

217915

808

*Telescience Testbed Pilot Program
Fifth Quarterly Report*

*Maria L. Gallagher, ed.
Barry M. Leiner, ed.*

September 1, 1988

Research Institute for Advanced Computer Science
NASA Ames Research Center

RIACS M88.5



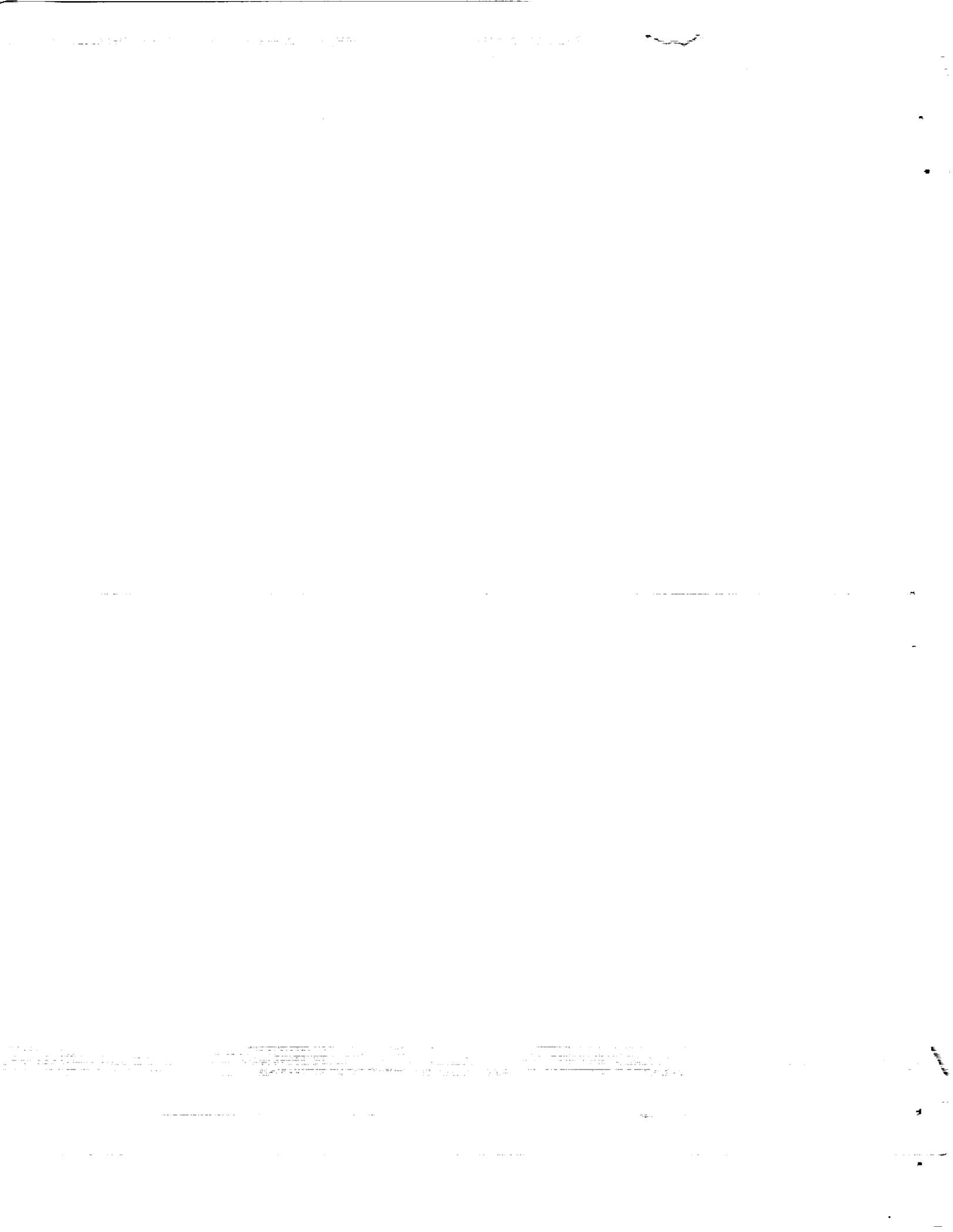
Research Institute for Advanced Computer Science

(NASA-CR-184594) TELESCEINCE TESTBED PILOT
PROGRAM Quarterly Report No. 5, 1 Jun. - 31
Aug. 1988 (Research Inst. for Advanced
Computer Science) 80 p CSCL 09B

N90-11455

Unclas
0217915

G3/61



Telescience Testbed Pilot Program Fifth Quarterly Report

*Maria L. Gallagher, ed.
Barry M. Leiner, ed.*

September 1, 1988

Research Institute for Advanced Computer Science
NASA Ames Research Center

RIACS M88.5

The Telescience Testbed Pilot Program is an innovative activity involving fifteen universities in a user-oriented rapid-prototyping testbed to develop the requirements and technologies appropriate to the information system of the Space Station era.

This is the fifth quarterly report, covering the period June 1 through August 31, 1988.

Work reported herein was supported by Contract NASW-4234 from the National Aeronautics and Space Administration (NASA) to the Universities Space Research Association (USRA).

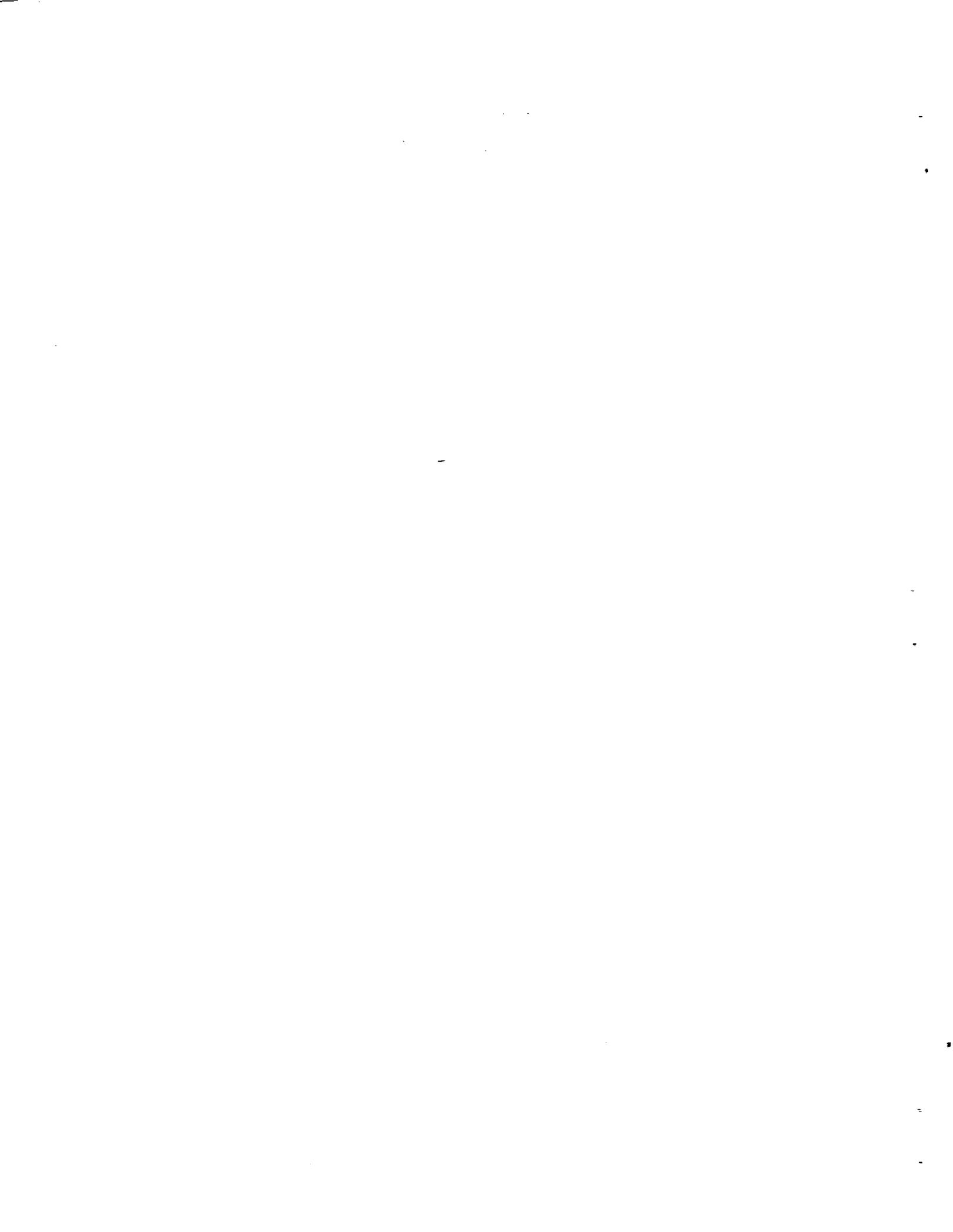


Table of Contents

1.0 INTRODUCTION	1
2.0 PROGRAMMATIC SUMMARY	1
3.0 TECHNICAL SUMMARY	1
4.0 SUBCONTRACTOR QUARTERLY REPORTS	1
4.1 UNIVERSITY OF ARIZONA, ASTRONOMY	2
4.1.1 Hardware Status	2
4.1.2 Communications Status	2
4.1.3 User Interface	3
4.1.4 Plans	4
4.2 UNIVERSITY OF ARIZONA, ENGINEERING & MINES	4
4.2.1 Introduction	4
4.2.2 Programmatic Summary	4
4.2.3 Astrometric Telescope (Astronomy and Astrophysics)	5
4.2.4 REMOTE FLUID HANDLING (Microgravity and Life Sciences)	10
4.2.5 TECHNOLOGY	15
4.2.6 REFERENCES	19
4.3 UNIVERSITY OF CALIFORNIA, BERKELEY	20
4.3.1 Teledesign	21
4.3.2 Extended Software Control System	21
4.3.3 Experiment Description and Summary	21
4.3.4 Issues Investigated	22
4.3.5 Method of Investigation	22
4.3.6 Experiment Results	23
4.3.7 Software Emulation of 8085 Microprocessors	23
4.3.8 Teleoperations	25
4.3.9 Remote Commanding and Telemetry Reception	25
4.3.10 Remote Data Analysis for EUVE	29
4.3.11 Real-time Rocket Data	32
4.4 UNIVERSITY OF CALIFORNIA, SANTA BARBARA	33
4.5 UNIVERSITY OF COLORADO, LASP	35
4.5.1 Remote Access to Data for Teleanalysis	35
4.5.2 Earth Science Telescience Field Campaign	35
4.5.3 Telescience Technologies	37
4.5.4 Astronomy Testbed	38
4.6 CORNELL UNIVERSITY	39
4.6.1 Current Status	39
4.6.2 Future Plans	40
4.7 UNIVERSITY OF MICHIGAN	40

- 4.7.1 Offline Guidance Requirements 40
- 4.7.2 Online Guidance 42
- 4.7.3 Portable Workstations 44
- 4.7.4 Communications Architecture 45
- 4.8 MIT, ASTRONOMY 47
 - 4.8.1 Status and Accomplishments of the Telescience Effort 47
 - 4.8.2 Experiments Under the MIT Astronomy Telescience Testbed 48
- 4.9 MIT, LIFE SCIENCES 53
- 4.10 PURDUE UNIVERSITY 54
 - 4.10.1 Equipment and Software 54
 - 4.10.2 Spectrometer/Multiband Radiometer Database 54
 - 4.10.3 Field Research Database on Optical Disk 55
 - 4.10.4 University of Wisconsin's McIDAS System 56
 - 4.10.5 University of Colorado's OASIS 56
 - 4.10.6 Recent Experience with the Internet 56
 - 4.10.7 Other Notes 56
- 4.11 RENNELAER POLYTECHNIC INSTITUTE 57
 - 4.11.1 Experiment Title 57
 - 4.11.2 Issue Investigated 57
 - 4.11.3 Experiment Hypothesis 57
 - 4.11.4 Method of Investigation as Planned 58
 - 4.11.5 Method of Investigation 59
 - 4.11.6 Experimental Results 60
 - 4.11.7 Microgravity Materials Science Summary 61
- 4.12 SMITHSONIAN ASTROPHYSICAL OBSERVATORY 62
- 4.13 UNIVERSITY OF WISCONSIN 63
- Appendix 1 LOA 65
- Appendix 2 TTPP BIBLIOGRAPHY 70

1.0 INTRODUCTION

The Telescience Testbed Pilot Program (TTPP) is developing initial recommendations for requirements and design approaches for the information systems of the Space Station era. Multiple scientific experiments are being carried out, each exploiting advanced computer and communication technologies and approaches, and each emulating some aspect of Space Station era science. The aggregate results of the program will serve to guide the development of future NASA information systems.

This quarterly report covers the period 1 June 1988 to 1 September 1988.

2.0 PROGRAMMATIC SUMMARY

During the past quarter, the various University subcontractors continued their efforts in testbedding. All activities are anticipated to complete this first phase by 30 November 1988, in time to have the final report completed in draft form by the end of the contract (31 December 1988).

Planning has begun for a meeting of all University and NASA participants to integrate the results of the program into a final report, documenting the results for each participating scientific discipline. This meeting is to be held in October 1988 at RIACS/Ames.

Support has continued to be provided for the NASA planning of the future information systems for the scientific community.

RIACS, as well as several of the University subcontractors, presented papers on telescience at the International Geographical Congress, held 22-26 August 1988 in Sydney, Australia.

3.0 TECHNICAL SUMMARY

During this past quarter, the focus of activity has been on completing the first phase of technical testbedding activities, represented by this Pilot Program. Drafting of the final reports of the various participants has been initiated, with several drafts being included in this Quarterly report as the University technical reports. Because the various participants have been focussed on technical work and not coordination and reporting during this past quarter, we have chosen to omit discipline area summaries for this Quarterly report. The next section contains quarterly reports from a subset of the University contractors.

No significant problems are anticipated in completing this effort.

4.0 SUBCONTRACTOR QUARTERLY REPORTS

4.1 UNIVERSITY OF ARIZONA, ASTRONOMY

Institution: University of Arizona (Steward Observatory)

Principal Investigator: Erick Young

Objectives:

1. acquisition of astronomical images from a remote infrared array camera;
2. investigation of network performance for the remote access of a large astronomical database at the Infrared Analysis and Processing Center; and
3. study of techniques for sharing software and infrared data among SIRTf instrument teams.

Network Connections: Internet, BITNET, SPAN

Mail Address: eyoung@as.arizona.edu

*******Note:** This is new since we have changed our gateway machine*****

Computer: Sun 3/160C workstation networked with MicroVAX II and Sun 3/150C within infrared group and to departmental computers.

Operating System: Sun UNIX with NFS

Software Packages: IRAF, AIPS, Mongo (scientific plotting package)

4.1.1 Hardware Status

It was recognized very early in this project that the 141 MByte disk standard with our Sun workstation configuration would be inadequate for the kinds of activities we were investigating. In particular, the large IRAF (Image Reduction and Analysis Facility) package combined with substantial image data would quickly fill up the existing disk space. Moreover, additional packages such as OASIS and TAE would be difficult to install with the existing hardware limitations. Using other funds, we have replaced the 141 MByte disk with a pair of Control Data Wren-IV drives. These drives have a combined formatted storage capacity of over 490 MBytes and will, for the time being, relieve the space pressure. We found the installation of the units to be extremely easy since the drives have a built-in SCSI (Small Computer System Interface) and the cost was significantly less than the comparable Sun enhancement.

4.1.2 Communications Status

A major disappointment in the networking experiments has been the very slow time frame for connecting the University of Arizona campus to the NASA Science Internet. Although

the hardware for the gateway has been on campus since January 1988, it now appears that the physical connection to the network will not take place until October or November 1988, at the earliest. The poor response time of the existing Internet has precluded useful wide-area telescience experiments among the SIRTf teams.

The bulk of the communications experimentation has been done on the local area network to study the remote operation of the infrared array camera. A variety of links between the Sun 3/160C and the instrument control Compaq 386-20 including hardwire serial, ethernet, and dialin serial were tested. We are using Sun's Network File Server (NFS) software to mount the image data files on the Compaq, in order to make them available to the analysis software and the Sun user. A number of problems have been identified with the dial-in operation of NFS and therefore, we have been concentrating on the ethernet and hard-wire serial modes of operation.

4.1.3 User Interface

During the past quarter, we investigated a number on software environments for the remote operation experiments. In particular, we installed a beta-test version of TAE-Plus (Transportable Application Executive), in order to study the applicability of the windowing tools and user interface to our experiment. Although the TAE system proved to be potentially very useful in the design of the user interface, a major complication encountered concerns the processing and display of the quick look image data. The data from an infrared array camera normally requires a significant amount of processing before they are intelligible. Specifically, bad pixels must be identified and removed. Background variations and pixel-to-pixel gain variations must be corrected, and an image must be displayed. Since all of the necessary tools are available in the IRAF package, we have instead decided to use IRAF as the basis for the system and will write the remote observing interface as an IRAF subtask. At present IRAF and TAE use different windowing standards (SunView and XWindows respectively), thus, making the use of both capabilities complicated. IRAF has stated the intention to migrate to the XWindows standard, and we will reevaluate the use of TAE when appropriate.

An initial version of the observing interface is currently running on the Sun. Since IRAF is primarily a command line driven interface, not well suited to real-time applications, we are running the observing module as a fairly independent subtask with a more useful user interface. The camera control will include default parameter files, a status window, a quick look image display, and a command line window. Most commands are available via mouse-click buttons and pull-down menus.

We are modifying the controller program for the 128×128 pixel infrared array camera, for use in a simulation mode. The hardware links to the camera will be stubbed but, otherwise, the program will be fully functional. The program will then be used on one of the laboratory computers which will act as surrogate for the camera. Real image data will be used to test processing speeds. In this way, we will be able to experiment with a wide variety of user interfaces without requiring the actual camera to be functioning. When useful interfaces are developed, they will be easily dropped into the actual observing program.

4.1.4 Plans

In the remaining two months of the TTPP activity, we will complete the remote observing user interface running under the IRAF environment. We will also provide support for the last TTPP workshop and the final report.

4.2 UNIVERSITY OF ARIZONA, ENGINEERING & MINES

4.2.1 Introduction

This is the Summer 1988 quarterly report for the astrometric telescope, remote fluid handling, and technology assessment projects at the University of Arizona. It does not include the University of Arizona involvement in the SIRTf project.

Telescience research can be split into two main areas:

1. remote control, sensing, manipulation, etc.; and
2. remote data storage, searching, retrieval, etc.

The goals of our two telescience testbeds are related to the first main area. The first testbed involves teleoperation of a forerunner of the Astrometric Telescope Facility (ATF), which is to be attached to Space Station. The second involves the development of systems and software for Remote Fluid Handling (RFH) in support of the microgravity and life sciences. These seemingly quite different testbed demonstrations were selected in order to pursue the following goals in our research:

1. to design a set of tools to allow teleoperation of scientific experiments. These tools include the man/machine interface at the remote commanding computer, the machine/machine interface between the remote commanding computer and the local controlling computer, and the machine/instrument interface between the local controlling computer and the equipment (robots, telescopes, measurement instruments, analytical instruments, etc.) which may comprise any specific experiment;
2. to ensure that these tools are *GENERIC* and *MODULAR*, so that they can easily be applied to a wide variety of scientific applications on Space Station (or other platforms) and so that the individual modules can be revised without significant impact on the remaining parts of the software; and
3. to evaluate the technologies underlying the above developments and develop recommendations (specifications) for future development.

4.2.2 Programmatic Summary

Larry Schooley and Francois Cellier attended the Space Station User Support Environment (USE) Working Group Meeting in Boulder, Colorado, on June 21-23, 1988. This symposium, chaired by Karen Moe of Goddard Space Flight Center, centered on discussions of the Common User Interface (CUI) being proposed for use on Space Station and related activities. Topics of interest included the user interface language (UIL), the direct manipulation interface language (DMIL), the intermediate language (IL), the human-computer interface guide (HCIG), the software support environment (SSE), and interactions with other projects.

This was a very useful experience. We were happy to discover that the basic architectures and software developments which we have proposed for telescience are quite compatible with current plans for the USE. However, telescience applications call for several extensions to the USE, which we mentioned to the working group. These are discussed in the technology section of this report. Francois Cellier is a member of the USE working group, and of the UIL specification task force within the USE working group, and will make sure that the interests of telescience are properly represented.

Larry Schooley and Francois Cellier visited Ames Research Center (ARC) on July 27-29. The major reasons for this visit were as follows:

1. Coordination of research with the Ames Life Science group (formerly under Daryl Rasmussen, now under Frank King). The University of Arizona and Ames have acquired the same hardware (workstation, PC, Ada compilers, robot, multifunction control board, syringes, and syringe adapters), in order to coordinate their testbed activity. One of the aims of this visit was to determine near-term objectives and schedules, and to avoid duplication of effort.

2. Since Phase I of the TTPP program will soon come to an end, and since the mechanisms for a potential Phase II of TTPP have not yet been established, it was necessary to discuss the extensions of research at the University of Arizona Telescience Laboratory which might be proposed in the interim. These discussions included Daryl Rasmussen (Telescience), Frank King (Life Science), Andy Goforth (Robotics and AI), and Ken Nishioka (Astrometric Telescope Facility), as well as Jack Faber (University of Colorado, LASP).

3. The University of Arizona team presented the current results of Phase I of our TTPP projects to RIACS (Barry Leiner, Richard Haines, Vicki Johnson), the above ARC personnel, and several additional members of the ARC Life Science group. Our new video tape on remote fluid handling was shown to this group, and the results of our efforts to date were discussed.

4.2.3 Astrometric Telescope (Astronomy and Astrophysics)

Status

The current demonstration involves teleoperation of a simulation of the Thaw Astrometric

Telescope. A remote commanding computer (RCC), which is a MicroVAX workstation running OASIS, communicates at 9600 baud over dialup phone lines with a second MicroVAX workstation. Intermediate command language statements, simulated telemetry, and simulated scientific data are exchanged using DECnet and CCSDS packets as the communications protocols. The second workstation includes the functions of a local controlling computer (LCC), as well as the telescope simulation.

The telescope simulation includes a failure mode simulator. Failure simulation is used for training purposes and also to aid in the design of more robust remote control software. The failure simulation is accomplished by drawing a random number (uniformly distributed between 0 and 1) prior to command execution. This random number is then compared to a preset threshold which is user specified when the simulator is started. Whenever the random number is above the threshold, a failure in the command execution is to occur, and the operator must then deal with this situation. The telescope simulator has been documented in Alfie Lew's MS Thesis entitled: "Astrometric Telescope Simulator for the Design and Development of Telescope Teleoperation." This document is available upon request as Telescience Laboratory Report TSL-016/88.

The communications functional design contains provisions for the future addition of multiple remote experimenters (multiple RCC's) running OASIS and one local controlling computer connected through a communication network. The software design will allow one OASIS user, holding an allocatable privilege key, to send intrusive telecommands, such as slew the telescope, or nonintrusive telecommands, such as: show telescope position to the simulator. Other (non-privileged) OASIS users may either issue nonintrusive commands to the local controlling computer, in order to observe the ongoing experiment, or may request a shadow image of the privileged user's session to be displayed. Non-key holders may also exchange messages with the privileged user and with each other (e.g., to request a transfer of the privilege key), and they can schedule a privileged session for themselves at some time in the future.

For the Phase I demonstration, however, the implemented software allows only one OASIS user. The Ada code for the local controlling computer's communication protocol software is complete. This software includes the implementation of Ada tasks that remove CCSDS telecommand headers from telecommand packets, thus leaving only the commands to the local controller, and concatenating CCSDS telemetry headers to telemetry data being sent back to the remote commanding computer running OASIS and Ada/DECnet interfaces. A significant amount of effort has been made to make this software as application independent as possible. As an example of this, it was decided to implement the CCSDS telemetry header concatenation and telecommand header removal as two tasks each. One task for each process performs CCSDS primary header operations, and one task for each process performs CCSDS secondary header operations. The CCSDS primary headers are application independent but the secondary headers are application dependent, and thus software changes for new applications will only be needed for the two tasks which perform the CCSDS secondary header operations.

Lower layer communication between the RCCs and the LCC will use the DECnet protocols.

The higher layer protocol functions between the various participating computers will be handled in accordance with the CCSDS Telecommand Data Management Service and Packet Telemetry recommendations. The current communication software design assumes that telecommands will only flow from the remote commanding computer to the local controlling computer, and telemetry will only flow in the opposite direction. Future software changes will allow for both telecommands and telemetry data to flow in either direction, which is in better accordance with NASA's concept of a "virtual terminal" for Space Station.

When telecommands arrive at the local controlling computer, the DECnet software removes the DECnet headers and trailers and sends the CCSDS telecommand packets to a DECnet/Ada interface procedure. This interface procedure then routes the telecommand packets to a CCSDS depacketizer procedure which performs various checks on the CCSDS primary and secondary headers, removes the headers and sends the telecommands, packet content and packet originator information to a queuer procedure. The CCSDS secondary header (optional according to CCSDS recommendations) contains 96 bits and includes the packet content type, a time-stamp indicating the time of the packet transmission, the current OASIS version number, and the packet originator information. The OASIS version number is important to ensure full compatibility since it cannot be guaranteed that all RCC's and the LCC will, at all times, have been updated to the same software level.

Error messages, status data and scientific data to be transmitted back to OASIS users are sent to a CCSDS packetizer task. This packetizer task places the messages and data in a CCSDS telemetry packet and then sends each telemetry packet to an Ada/DECnet interface procedure. Each CCSDS telemetry packet contains a secondary header. The secondary header is discussed in the CCSDS telemetry packet recommendation, and will be identical to the secondary header used in the CCSDS telecommand packet. The Ada/DECnet interface procedure sends each CCSDS telemetry packet to the DECnet software to transmit it back to the OASIS users.

The application data formats for the CCSDS telemetry packets have been defined for each error message, status data, and scientific data packet. These packet type definitions were used to build the OASIS communication bit stream decomposition table. So far, 20 different telemetry packet types have been defined.

It is expected that this design will be sufficiently generic that adaptation to the remote fluid handling demonstration will require little or no modification, aside from recompiling the Ada routines for the PC (the LCC in the case of the remote fluid handling demonstration), using the Meridian compiler, and interfacing ADA with the DECnet/DOS software, which is written in C.

Development of this demonstration is complete. Visitors are welcome at any time. A videotape of the demonstration is being prepared for the October meeting at Ames.

Preliminary Results

1. Observations

- a. Remote operation of ground based telescopes is not commonly done, not because the control task is particularly difficult to solve (it is not), but because most telescopes are not properly equipped with the hardware required for remote operation.
- b. The most serious obstacle to remote operation of a telescope seems to be safety considerations, and while in space there will not be a cleaning lady around into whom you can slew the instrument, it might still be considered bad taste if the ATF were slewed mistakenly into an approaching Space Shuttle. Such problems can be avoided with properly designed hardware and software.
- c. There is currently no ground based astrometric telescope which is remotely controlled. Most instruments with some sort of remote operation facility are infrared telescopes, and the rest are millimeter wave telescopes. In both of these cases, the problems are quite different from those which can be expected when operating the ATF.
- d. Astronomers are not usually interested in data reduction techniques. For understandable reasons, they prefer to store every bit of information available to them. Consequently, little research has been done on data reduction for transmission or storage of star field images. However, the automatic control circuitry needs the star field information also (for the automated finder scope and guider scope control). The control circuits very definitely do not need the total star field information. On the contrary, the task of these circuits can be greatly simplified if the information contained in the star field is reduced to only those pieces that are needed for control purposes.

2. *Achievements*

- a. Our development of the Thaw simulator has resulted in the definition of a set of sensors, controls, and safeguards needed for teleoperation of the Thaw telescope. Whether or not this is ever actually implemented, it has shown that the control, sensory, and safeguard systems of the ATF should be developed with the end user in mind. Successful telescience applications of this instrument call for a set of controls, sensors, and safeguards that should be defined early on, before the instrument has been totally developed. It is much simpler and less costly to design any telescope from the beginning with the telescience user in mind, rather than to be forced to retrofit the instrument at a later date.
- b. The goal of developing generic, modular software has been attained. Further discussion of this is contained in the RFH and TECHNOLOGY sections of this report. This result could not have been achieved without the previous OASIS development by the University of Colorado.
- c. The work on video data compression for transmission of telescope pointing data

has been completed. A form of run length coding, followed by variable word length (Huffman) coding was found to be the best for this application. This compression algorithm provides reduction from 8 bits per pixel to 0.015 bits per pixel for telescope pointing data. It appears that use of this technique will also allow compression of typical scientific observation data (of a general starfield) for storage or transmission by at least a factor of 10, and perhaps as much as 100. This research has been documented in Wendy Tolle Walker's MS Thesis entitled, "Video Data Compression for Telescience," and is available upon request as Telescience Laboratory Report TSL-017/88.

Future Work

The original plan was to move the simulation to Allegheny Observatory, and to operate it from Tucson during Phase 1. Then, during Phase 2, teleoperation of the Thaw telescope itself was to be accomplished. This is not currently feasible. The design and installation of the physical interface between the local controlling computer and the telescope is in progress, but proceeding very slowly due to lack of funding. Thus, remote operation of the actual telescope and scientific instrument will not be possible for at least another year, and may or may not be desirable at that time. This is because the development of the multidetector medusa instrument head is unfunded.

Because of the above, and because availability of the NASA Science Network (NSN) has slipped until at least January 1989, it will not be useful now to physically move the telescope simulator to Allegheny and to operate it from Tucson. All the objectives of this portion of the demonstration can be accomplished in a more timely and cost effective manner by sending the simulator software electronically to the University of Colorado and demonstrating teleoperation of the simulation from Tucson. This could also be done in the reverse direction, i.e., operate the simulation in Tucson from a workstation running the OASIS application in Boulder.

In view of the above developments, it seems most logical to proceed with the design, implementation, and teleoperation of a realistic simulation of a Space Station Astrometric Telescope Facility for the next phase of the Telescience Program. If other funding permits, this would also include a half-scale brassboard simulator of the telescope itself. The effort would begin with development of a concise statement of the objectives of the testbed, based on requirements for autonomous teleoperation of the telescope facility, as well as provision for maintenance, calibration, and response to anomalous events and conditions.

A preliminary list of these objectives includes:

1. identification of a complete set of controls, sensors, and safeguards needed for remote operation of the ATF;
2. identification of a complete set of scenarios for normal operation of the ATF and development of a set of software modules implementing these scenarios; and

3. identification of a generic set of rules which detect anomalies in time for corrective actions to be taken; determination of a complete set of exception procedures to identify anomalies once they are detected (shut-down procedures); and finally, development of a complete set of recovery procedures to return to normal operation after the instigating anomaly has been removed (start-up procedures); and

4. design of an appropriate user interface to remotely operate the ATF in a convenient fashion for calibration, maintenance, and other similar functions.

The refinement of this set of objectives will be based upon a detailed list of requirements derived from discussions with experienced researchers at JPL and Allegheny Observatory to identify probable scenarios. From this list of requirements will be developed a comprehensive list of the activities to be simulated. This process will also involve appropriate NASA centers (probably Ames and JPL). It will then be a relatively straightforward task to extend the existing Thaw simulator into a realistic ATF simulator, and to demonstrate teleoperation of the simulation. Depending on ultimate availability of an automated multihead detector system, it may then be useful to proceed with the teleoperation of the Thaw telescope at some later date.

4.2.4 REMOTE FLUID HANDLING (Microgravity and Life Sciences)

Status

The initial demonstration involves teleoperation of a laboratory which provides automated handling and analysis of fluids, such as those which might be extracted from laboratory animals or human subjects as part of the life sciences program onboard Space Station. The experiments selected are determination of the pH of a solution, and the separation of a solution into its charged components using electrophoresis. The laboratory equipment to be controlled includes a SCORBOT robot, special fluid handling apparatus, and the instruments and electronics required for the two experiments. The architecture of the teleoperations system is identical to that used for the ATF demonstration. The remote commanding computer is the same MicroVax workstation which also runs an OASIS application for the man/machine interface. The local controlling computer is an IBM PC equivalent with 640K memory, an 8087 coprocessor, a 20 Mbyte hard disc, and a LabTender multifunction board. The machine/machine interface is also the same as for the other demonstration.

Efraim Raize, our Visiting Scholar from Israel, completed his Sabbatical leave and returned to his home country. He was responsible for the syringe driver assembly, and a special computer interface for the isotachopheresis instrument. The results of his efforts are documented in our Telescience Laboratory Reports TSL-011 and TSL-012, which are available upon request.

The design of the user interface (OASIS application) has been completed, and has been documented in Byron Hack's MS Thesis entitled, "Man/Machine, Machine/Machine, and Machine/Instrument Interfaces for Teleoperation of a Fluid Handling Laboratory Aboard

Space Station." This document is available upon request as Telescience Report TSL-013. It supersedes and extends the previous documents TSL-006, TSL-009, and TSL-010.

A video tape has been completed which demonstrates a robot controlled fluid handling laboratory. The Scorbot robot is able to manipulate fluids by operating the specially built syringes and syringe adapters. The robot and instruments are operated by the local controlling computer. The remote control software has not yet been completed, and will be demonstrated with a subsequent video tape (completion date: October 15, 1988 -- see schedule below). Copies of our current video tape are available upon request as Telescience Laboratory Report TSL-015/88.

The implementation of the man/machine interface (the development of an appropriate OASIS application database), the machine/machine interface (the intermediate language), and the machine/instrument interface are in progress and will be completed shortly. The software interface between Meridian Ada and BASIC (the language in which the robot control macros are written), and between Meridian Ada and C (the language in which the DECnet/DOS software is coded) has begun.

The detailed schedule is given in the table below. For comparison, the corresponding tasks are shown for the astrometric telescope demonstration. This emphasizes our success in developing generic systems/software for two quite different experiments. Only the machine/instrument interface is different (of course, the databases which drive the man/machine interface (OASIS) and the machine/machine interface differ also).

PHASE I DEMONSTRATION SCHEDULE

TASKS	SCHEDULED COMPLETION DATE	
	ATF	RFH
REMOTE COMMANDING COMPUTER		
OASIS Database	done	9/16/88
OASIS CCSDS Protocol Changes	done	9/16/88
LOCAL CONTROLLING COMPUTER COMMUNICATIONS		
DECnet/Ada interface	done	9/23/88
CCSDS Depacketizer	done	9/16/88
Telemetry Packet Storage	N/I	N/I
CCSDS Packetizer	done	9/16/88
Ada/DECnet Interface	done	9/23/88
TELECOMMAND STORAGE & RETRIEVAL		
Queue Manager	done	done
Mailboxes	done	done
Retriever	done	done
SCHEDULER	N/I	N/I
COMMAND PROCESSING		
Scanner	done	9/16/88
Parser	done	9/16/88
Interpreter	done	9/16/88
LOCAL CONTROLLING COMPUTER INTERFACES		
Local Keyboard Interface	done	done
Local CRT Interface	done	9/16/88
TELEMETRY HANDLERS		
Status Data Handler	done	9/16/88
Scientific Data Handler	done	9/16/88
INSTRUMENTATION		
Telescope Simulator	done	N/A
Telescope Simulator Database	done	N/A
Failure Simulator	done	N/A
Ada/BASIC Interfaces to Labtender & Robot Software	N/A	9/23/88
Application Programs in BASIC	N/A	9/16/88
Rack	N/A	done
Syringe Driver & Assembly Control	N/A	done
Pipette Modification	N/A	9/16/88
System Integration & Test	9/1/88	10/1/88

N/I - not implemented in phase I		
N/A - not applicable		

Other efforts of the remote fluid handling group have focused on modification plans for the recently acquired motorized pipette. It is hoped that the use of this device will eliminate the need for the syringe holder and fixture; will be more in line with the current industry trend towards fixtureless assembly; and will reduce the overall weight of the components used in the system. Since this pipette has been designed for use in ground-based laboratory work, and picks up fluids by creating a vacuum in a chamber above the disposable tip, changes will be necessary to prevent the fluid from floating up this chamber in reduced gravity. The current thinking calls for the use of syringes with needles, rather than the provided plastic tips, where each syringe will be fitted with a plastic membrane attached to a spring, such that fluid pressure will push the membrane/spring pair up in the syringe barrel.

In the meantime, the syringe driver assembly, which also uses a stepper motor, has been interfaced to the multifunction board. Software development for PC control of the syringe driver has been completed. This software can be used with very little modification to operate the pipette later on. The work on design and construction of a control board to operate the pipette motor is near completion. This board will be mounted in place of the existing control board and will provide the necessary interface to the PC.

Present capabilities of the pipette permit four modes of operation at two speeds, namely pickup, dispense, multi-pickup, and multi-dispense at regular and fast speeds. The on-board software that performs these operations will then be replicated in the PC in Basic. The present manual triggering, which starts each mode of operation, will be replaced by an appropriate command from the PC.

Preliminary Results

1. Observations

- a. Analytical work in the life sciences/microgravity sciences cannot proceed in an economically feasible fashion without facilities for automated remote fluid handling. The work is tedious and repetitive, and crew time is exceedingly expensive. Thus, it is essential that the telepresence principles developed in this testbed be extended and integrated with other activities, such as the Space Station Life Sciences Glovebox. In particular, an instrument rack (automated laboratory) should be added immediately to the design, adjacent to the glovebox, with provision for passing samples back and forth between the two areas.
- b. The robot configuration must be geometrically compatible with the workspace. For example, in front of a flat rack, it is inappropriate to use a robot that is unable to move horizontally and vertically along the rack. It is insufficient that we are somehow able to reach each point in space. Unless the robot configuration matches the workspace topology, the necessitated indirect approximation of the desired robot motion is paid for by an unavoidable loss in positioning accuracy and by unnecessary complexity of the command sequences.

- c. The handling of small amounts of fluid in space calls for special equipment (syringes or pipettes). The laboratory robot must be able to handle these devices. Today's commercial laboratory robots are not equipped with an appropriate end effector for that purpose.
- d. The major problems which must be solved relate to contamination control and waste disposal, coupled with the fact that there can be no air/liquid interfaces which are not completely controlled with surface tension.

2. *Achievements*

- a. Our testbed has produced the design of an appropriate end effector/syringe coupling. Several alternatives were investigated, and several alternatives were actually built. This approach works but is somewhat cumbersome in terms of detailed steps required in the robot programming.
- b. We are currently experimenting with a setup using pipettes. In this approach, only the cheap, light, and small plastic pipette tip needs to be discarded. All other parts are reusable. This configuration seems also in line with industrial developments on the ground. In 1984 alone, industry consumed 5 billion of these pipette tips. It will be necessary to modify the tip to provide containment/insertion for contamination control, and to avoid air/liquid interfaces. Our testbed will help to decide whether or not this technology will work satisfactorily for telepresence operations in microgravity environments.
- c. Successful teleoperation of a representative (albeit small scale) fluid handling laboratory has been demonstrated. We have been successful in producing generic man/machine and machine/machine interfaces which work equally well for teleoperation of two quite different demonstrations. This could not have been accomplished without the prior development of the OASIS software by the University of Colorado.
- d. We have determined that the recommended video data compression technique for the robot/laboratory observation data is one that currently has wide-spread use in the area of video conferencing. Since color images are needed for this application, data rates of 80 to 400 kbits/sec (depending on rates of motion) are required for adequate image quality.

Future Work

While it is important to design the robot, as outlined above, in an intelligent and flexible manner, it is generally cheaper and better to adapt exotic instruments to the capabilities of the once designed robot, than to retrofit new capabilities into the robot. We therefore suggest that an appropriate laboratory robot be designed first. Thereafter, all instruments which are to be placed in the instrument rack should be designed such that they can be easily manipulated by the laboratory robot. This will necessitate a major redesign of essentially

every single laboratory instrument to be placed in the rack, but they must be redesigned and qualified for space use anyway. While our current testbed will help to develop a set of design parameters for the laboratory robot, we currently do not have the financial means to actually build a prototype of such a robot. However, in a later stage of the telescience program, this is a task we would be well qualified to pursue.

More research needs to be done with respect to the type and amount of video-feedback necessary for remote fluid handling aboard Space Station. While it has already been established that video-feedback is an indispensable component of this telescience application, we do not know yet exactly where to place the cameras, and how many of those will be needed. One of the things we pointed out at the USE conference was the need to provide for video feedback to the scientists, with remote control of the video equipment. No video feedback had been included in the USE presentation output layer.

Scenarios involving multiple remote experimenters and/or observers call for specific decisions relating to the design of the user interface language. These should be taken into consideration. As an example, it should be possible to request a copy of the display presented to the experimenter to be broadcast to one or several other sites that are communicating with the experimenter through an audio link. This demand requires all DMIL commands even housekeeping commands such as changing the background color of a screen window) to have their UIL equivalent. Provision for multiple users were not included in the USE requirements. Similar considerations are involved in provision for simultaneous multiple experiments.

4.2.5 TECHNOLOGY

As detailed implementation of the two testbeds proceeds, much is being learned about the underlying technologies. The following represents a preliminary statement of our conclusions and recommendations:

Scientists Will Require Remote Multiple Access for Remote Teleoperation

While a number of TTPP participants have looked into the problem of remote multiple access to distributed databases, little has been done with respect to multiple control of experiments. Outside of the telescience community, there exists even less awareness of the need for multiple simultaneous users of the same equipment. We have performed a feasibility study which indicates that multiple simultaneous control and/or observation of an ongoing experiment can be attained.

The Need for Multiple Access Must Be Addressed Now

It is necessary to foresee multiple simultaneous controllers/observers right from the beginning, since this mode of operation calls for additional communication protocols and equipment. For example, if we want to allow an observer to obtain on his screen a shadow image of everything that is displayed on the main experimenter's console, all screen commands, including the housekeeping commands, such as changing the background color of

a screen window, must be communicable rather than being treated as strictly local activities. A more obvious example is the use of token-passing (key ownership) to ensure that only one location is in active control at any particular time. It is therefore not feasible that the system be developed for single users only, while the design of a multiple user capability, for financial and scheduling reasons, is postponed until later. It is important that this capability is foreseen right away. While our current financial means are insufficient to actually testbed a multiple controller environment, this is one of the activities that we would suggest be pursued in a future extension of the telescience program.

Multiple Experiments Must Also Be Considered

Due to a lack of appropriate coordinating information, all activities are currently treated as independent units. The general scenario is that of: an experiment on one end remotely controlled by an experimenter on the other end. In reality though, all communication to and from Space Station will be routed through a central (though distributed) operating environment (the Operation Management System (OMS)). The integration of the telescience activities into the OMS will create additional problems (such as, additional time delay), and will call for additional communication protocols. It is important that the implications of this integration into the OMS be considered early on. It might be useful to create an OMS simulator, and make this simulator available to the telescience community. The TTPP participants will then be able to route their commands through the OMS simulator to more accurately determine what will happen to their experiments onboard Space Station.

Overall System Response Times Must Be Specified

We have studied the effect of communication delays on stability of remote closed loop control. It appears that a total delay of 0.8 sec can be made acceptable by use of special techniques, but much more than that will result in unstable behavior of the control circuit. For reference, this is about equal to the worst case communication delay (two TDRSS hops, but with no processing or ground network delay). Since we will not be able to guarantee 0.8 sec delay for normal operation (even with one TDRSS hop), it will not be possible to routinely teleoperate any robots with remote man-in-the-loop control configuration. Instead, the local robot control must be automated. The remote operator cannot control the robot directly. This mode of interaction must be limited to specifying set points and preprogrammed procedures. If something goes wrong, however, it must be possible to remotely fix the problem. For that purpose, it should be possible to temporarily request a point to point connection (special communication link) which guarantees a delay time of less than 0.8 seconds round trip (0.4 seconds per link). For normal operation (specification of set points or procedures), our testbed has shown that a response time of 5 seconds will be sufficient, or rather, that the control circuits can be designed to operate under such conditions.

Delay Times Must Be Verified By Simulation

It is important to study the effect of these multiple software and communication layers on overall system response time. Since the development of these software tools has been

distributed among several contractors, nobody seems to have a clear picture of the effect of these multiple software systems on overall system response times. It is suggested that an integrated end-to-end real-time performance simulator be created to study this problem. We would be very interested in pursuing this effort in cooperation with the appropriate groups.

Public Data Switching Networks Cannot Be Used for Teleoperations

The performance available through unlimited access data switching networks (such as, NSN/NSF) will not suffice under any conditions. The large, unpredictable, and variable delay times imposed by these networks will jeopardize any successful teleoperation of equipment aboard Space Station (the problem is with the delay, not with the bandwidth - NSN will probably work fine for remote access to distributed data bases). There are those who claim that this problem will be eliminated by installation of higher capacity networks, but this is not true unless access is limited. As an example of this, the NSFnet backbone was upgraded from 56 kbps to 448 kbps last July, and it is planned to increase capacity to 1.54 mbps in 1989 and 45 mbps in the early 1990's. The problem is that usage is also increasing (100% per year) so that the current grade of service is not predicted to improve much. This situation could be altered with provision of a virtual circuit capability, but this seems far in the future, if it occurs at all.

Video Feedback Has Separate Requirements

Bandwidth and delay problems occur with the incorporation of video feedback. It is suggested that this problem be considered outside the OMS. Special point to point connections with a guaranteed delay time of below 2.5 sec must be established which bypass the OMS and can deliver video feedback directly to the teleoperator. More research is needed to establish appropriate data links, switching control, and data reduction schemes for different applications. This is yet another activity in which we could participate in a future extension of the Telescience program.

Ada Is An Excellent Language Choice for Telescience

Ada has proven to be an excellent choice as the standard programming language! In particular, it is a very convenient language for telescience applications. Especially useful are the *exception handling* capabilities, which allow separation of the error handling from the processing of correct commands (increased readability and modularity); the *information hiding* features (private types), which allow for a previously never seen ability to modularize software, and to distribute the coding of subsystems among several team members without fear of side-effects; and finally the *multi-tasking capability* which allows capture of parallel processes in parallel software modules in a very convenient manner.

OASIS Should Be Modified and Extended

We have extensively tested OASIS for the purpose of teleoperation of experiments aboard Space Station. We found that OASIS presents us with a flexible and convenient state-of-

the-art user interface to command experiments. However, OASIS is not yet satisfactory with respect to its flexibility and performance for communicating with the equipment located at the other end of the communication link, and for interactions of multiple experimenters with multiple experiments.

It is clear that OASIS forms a very good starting point for a generic teleoperations software, but that it must be significantly modularized and expanded for future applications. A number of specific suggestions are presented below. These improvements are all feasible, and the cost to update OASIS will be significantly less than following the route of redesign/reimplementation.

1. A future successor of OASIS must be modularized. In this software, a clear distinction must be made between:
 - a. the scanner/parser that handles user input and translates it into an intermediate language;
 - b. the command interpreter that interprets the intermediate language commands;
 - c. the communication software that communicates with other workstations or computers; and
 - d. the instrument interface that communicates with the experiment.
2. Provisions must be made that allow for multiple remote operators operating through separate workstations that may be geographically distributed. These workstations must be able to communicate with each other, as well as with the local controlling computer, which should simply be yet another "user" on this symmetric communication network.
3. This workstation software should allow for convenient access to a large variety of different experiments to be performed. It is suggested that a "room concept" be used for that purpose, which provides for a pictorial representation of the SSIS as a whole, and allows the user to select (by entering the appropriate "room" of Space Station) the activity to be performed. In terms of the OASIS successor, this requires the capability of hierarchically chaining separate databases.
4. The OASIS successor must provide for a second entry at the intermediate language level for commands that have been issued by other workstations and which have already been scanned/parsed. This will be the "normal" mode of operation for the local controlling computer, except for local operator overrides.
5. The workstation software must provide for better handshaking mechanisms than OASIS currently does. Currently, there are no provisions made for acknowledgment of CCSDS packages, for resubmission of packages in case of timeout, for automated grouping of incoming packages for the purpose of more efficient handling of the requests, etc.
6. The software must allow for selective transmission of packages. In a complex experiment, it is unacceptably wasteful to request all telemetry data to be transmitted at all times. It should be possible to specify much more flexibly which data are to be transmitted,

and where they are to be broadcast to (in case of multiple participants). The secondary CCSDS headers can be used for that purpose, but the current OASIS software gives little or no support for using this potential communication capability.

7. The software must handle textual data for communication between different remote users. Currently, OASIS allows for a brute force encoding of textual data, but the decoding does not properly work, since the data fields are of a predefined fixed size. The software ought to provide for *mail* and *phone* facilities of its own, rather than relying on operating system protocols (unless the OMS assumes this responsibility in a standardized manner).

4.2.6 REFERENCES

1. R. A. Bienz and L. C. Schooley, A Survey of Computer Networks, Technical Report TSL-001/87, Electrical and Computer Engineering Department, University of Arizona, Tucson AZ, February 1987.
2. Y. Pan, Teleoperation of Mechanical Manipulators Aboard the U.S. Space Station, Technical Report TSL-002/87, Electrical and Computer Engineering Department, University of Arizona, Tucson AZ, December 1987.
3. L. C. Schooley and F. E. Cellier, Telescience Testbed Pilot Program Quarterly Report, Technical Report TSL-003/87, Electrical and Computer Engineering Department, University of Arizona, Tucson AZ, December 1987.
4. F. E. Cellier, Teleoperation of the Thaw Telescope at the Allegheny Observatory: A Case Study, Technical Report TSL-004/87, Electrical and Computer Engineering Department, University of Arizona, Tucson AZ, December 1987.
5. L. C. Schooley, R. A. Bienz and F. E. Cellier, Basic Research in Telescience Testbed Program, Final Report: NASA Grant NAGW-1073, Technical Report TSL-005/88, Electrical and Computer Engineering Department, University of Arizona, Tucson AZ, January 1988.
6. Superseded by TSL-014/88.
7. L. C. Schooley, D. G. Schultz and F. E. Cellier, University of Arizona Presentation, Second Telescience Testbed Pilot Program Meeting, Boulder Colorado, March 7-9, 1988, Technical Report TSL-007/88, Electrical and Computer Engineering Department, University of Arizona, Tucson AZ, March 1988.
8. L. C. Schooley and F. E. Cellier, Telescience Testbed Pilot Program Quarterly Report for Winter 1987-1988, Technical Report TSL-008/88, Electrical and Computer Engineering Department, University of Arizona, Tucson AZ, March 1988.
9. Superseded by TSL-014/88.
10. Superseded by TSL-014/88.

11. E. Raize, Computer Interface for Electrophoresis Apparatus, Technical Report, TSL-011/88, Electrical and Computer Engineering Department, University of Arizona, Tucson AZ, May 1988.
12. E. Raize, Syringe Driver Assembly for Automated Fluid Handling Laboratory, Technical Report, TSL-012/88, Electrical and Computer Engineering Department, University of Arizona, Tucson AZ, May 1988.
13. L. C. Schooley and F. E. Cellier, Telescience Testbed Pilot Program Quarterly Report for Spring 1988, Technical Report TSL-013/88, Electrical and Computer Engineering Department, University of Arizona, Tucson AZ, June 1988.
14. B. W. J. Hack, Man to Machine, Machine to Machine and Machine to Instrument Interfaces for Teleoperation of a Fluid Handling Laboratory, Technical Report TSL-014/88, Electrical and Computer Engineering Department, University of Arizona, Tucson AZ, June 1988.
15. D. G. Schultz and R. Fardid, An Automated Remote Fluid Handling System, Technical Report TSL-015/88 (videotape), Systems and Industrial Engineering Department, University of Arizona, Tucson AZ, July 1988.
16. A. K. Lew, Astrometric Telescope Simulator for the Design and Development of Telescope Teleoperation, Technical Report TSL-016/88, Electrical and Computer Engineering Department, University of Arizona, Tucson AZ, September 1988.
17. W. T. Walker, Video Data Compression for Telescience, Technical Report TSL-017/88, Electrical and Computer Engineering Department, University of Arizona, Tucson AZ, September 1988.
18. L. C. Schooley and F. E. Cellier, Telescience Testbed Pilot Program Quarterly Report for Summer 1988, Technical Report TSL-018/88, Electrical and Computer Engineering Department, University of Arizona, Tucson AZ, September 1988.

4.3 UNIVERSITY OF CALIFORNIA, BERKELEY

The U. C. Berkeley Telescience effort focuses on the Extreme Ultraviolet Explorer (EUVE) project. From its inception, the design philosophy of the EUVE project has incorporated many telescience concepts. Three experiments were initially proposed for TTPP, working on both teledesign and teleoperations. In the area of teledesign, we worked on an extended Software Control System to allow widely separated research groups to collaborate on hardware and software development. A software microprocessor emulator would allow instrument software to be developed and tested before the flight hardware is available. Our teleoperation experiment involved remotely operating the prototype EUVE electronics.

Two additional experiments were added during the course of the TTPP. During the teleoperation experiments, it became apparent that visual feedback would enhance the operator interaction with the instruments. A method of transferring low-resolution, low-rate video images over the same network being used for the remote operations was developed, using inexpensive off-the-shelf hardware. A rocket experiment provided an opportunity to demonstrate real-time data collection/analysis from a short (20 minute) mission. We also investigated the use of the electronic browsing for EUVE guest observer data analysis.

4.3.1 Teledesign

We have conducted two teledesign experiments under the TTPP program. The first experiment was to develop methodologies for distributed software development. The second experiment focused on the simulation of the operations of eight Intel 8085 microprocessors to be flown aboard the EUVE satellite. We describe the experiments and the results in the following sections.

4.3.2 Extended Software Control System

In the Space Station era, it is expected that experiments will be built by a consortium of experimenters located in various corners of the world and interacting over computer networks. To facilitate the design, fabrication, and flight operation processes, it will be necessary to share software and data files among various sites. In order to prepare for such a scenario, we have extended a software control system (SCS), which we developed for flight, and data analysis software for the EUVE satellite. Software for the EUVE project is being developed by a team of programmers, scientists and technicians over a local area network connecting a number of workstations. The mechanism has been in use for the past five years and proved extremely beneficial in the hardware/software development for the EUVE project.

The TTPP experiment we undertook was to extend the SCS to include a wide area network (WAN) connecting the Space Sciences Laboratory (SSL) of UCB to a group at the Massachusetts Institute of Technology (MIT). It is aimed for use by the X-ray Timing Explorer (XTE) project, where a portion of the flight software is being developed jointly by the two groups.

4.3.3 Experiment Description and Summary

The EUVE SCS provides an environment in which the software developers and users can coexist without impeding the progress of either. It compartmentalizes software into three categories:

1. *New*. The software in this category is under active development and the only allowed users of this partition are the software developers themselves. Once the software is fully developed, it is promoted to an area called "test."

2. **Test.** Software in this category is used by the software developers and users for testing and validation purpose. The majority of the interaction between the software engineers, programmers, and the scientists take place here. After a reasonable amount of testing, the software is promoted to the "operational" area.

3. **Operational.** After the software has undergone a thorough testing it is promoted classified as operational. At this point, the software is considered rugged and the users are encouraged to use the codes residing in this area.

Furthermore, it allows critical software and data files to be copied to multiple selected computers on the network, providing backup functions. This allows the detector development group, for example, to continue to gather data and analyze them even if the computer which has the master copy of the software unavailable.

This system has proved to be extremely useful for the EUVE project. Under the TTPP program we decided to test its performance over a wide area network connecting the MIT and Berkeley. We used the internet for this purpose. Appropriate servers were installed on two machines, one at each location. Various issues relating to file updates and network performances were investigated.

4.3.4 Issues Investigated

The aim of this experiment was to install a software system so that programmers at MIT and Berkeley could jointly develop software for use on the XTE project. Since the basic system was already operational at UCB, the key issues investigated were the bandwidth, delay, and reliability of the Internet to support the SCS.

5.1.1.3 Experiment Hypothesis

The two hypotheses for this teledesign experiment are as follows:

1. The technology is mature enough so that software development from widely distributed geographic locations using the existing network is feasible.
2. The Internet bandwidth and performance are sufficient to support the required operations

4.3.5 Method of Investigation

We had two meetings with Dr. Hale Bradt and colleagues at MIT to discuss the implementation plans. Afterwards, we installed appropriate software at the XTE1 machine, at the MIT Center for Space Research. Several files were created at both locations. We wanted to see how changes at one location get updated at the other. Depending upon the originating location, the file in one location was designated as the "master" with a "slave" copy at the other location. Every night, the SCS updated the changes in the files at the slave locations using the master copy. The first attempt included updating the complete file

whenever it was changed. This is easiest to implement but required a large data volume to be transferred over the Internet. In the final phase, we would include the updates of only those lines of the files containing changes. Due to various problems, we were unable to complete this version.

4.3.6 Experiment Results

We have successfully implemented the first version of the wide area SCS. We discovered a serious problem with the Internet throughput while whole files were being copied. To minimize the amount of data volume over the network, we attempted to implement the second version. Limitations in the internet routing software prevented us from taking advantage of high speed NASA Science Internnet (NSI) links. During the day, a simple login to MIT from Berkeley was virtually impossible. The situation has improved somewhat since the NSI backbone was upgraded to T1. Late at night the situation was somewhat improved. We concluded that unless the throughput of the Internet is increased, this scheme, although very useful, may not be implemented to the satisfaction of the scientific users.

4.3.7 Software Emulation of 8085 Microprocessors

The second teledesign experiment simulated the operations of multiple Intel 8085 microprocessors to be flown in the EUVE satellite. The advantages of a microprocessor level simulation of the EUVE instruments are twofold. First, it allows us to test the ISW in ways not otherwise possible before the instrument is complete. Second, provides an interactive debugging environment, not available with the flight development hardware itself.

4.3.7.1 Experiment Description and Summary

The EUVE uses an integrated hardware/software development strategy. Its implementation uses an End-to-End system which simulates all major subsystems [Marchant et al., 1987; Chakrabarti et al., 1988a]. This simulation attempts to provide the user with one interface to the instrument and software, during all phases of the instrument development, calibration, and flight operations. Our simulation experiment was to enhance the EES to include the interactions of the microprocessors at a low level. This allows a thorough testing of the software written for use by the flight microprocessors.

4.3.7.2 Issues Investigated

The EUVE instrument presents a complex microprocessor environment, with eight 8085 CPUs controlling the different instruments and interacting closely with each other to collect data and generate telemetry. As stated above, a software simulation of the microprocessor environment would allow the instrument software to be developed and tested in advance of the availability of the actual instrument hardware. Such a simulation must duplicate the close interaction of the microprocessors, while providing an interactive debugging environment for the programmer. At the same time, the simulation should be fast enough to be able to run at a significant fraction of the actual hardware speed.

4.3.7.3 Experiment Hypothesis

Multitasking techniques are well established, and indeed, UNIX provides facilities for writing multitasking applications. Such general purpose facilities come with considerable overhead, however. For this experiment, we hypothesized that a cooperative multitasking scheme, tailored to the needs of the simulator, would minimize the overhead associated with task switching. This is essential for a simulator that must be able to switch between simulated processors at the level of the processor clock speed.

Using a time slice of a single (simulated) processor cycle would still bog down the simulator with task switching overhead. It turns out, however, that the events that generate interactions between the processors (detected photons, telemetry interrupts) occur at a much slower rate than the processor clock (a few Khz compared to 3 Mhz). Therefore, we also hypothesized that a variable time slice would allow us to approximate the behavior of the actual instrument processors while enabling the simulator to run at a reasonable speed. The time slice would be set to a few hundred processor cycles until an interrupting event occurred, when it would be reduced to a single cycle until the event had been processed.

4.3.7.4 Method of Investigation

The first phase of the simulation used an interpretive programming language called Magic/L. Magic/L is an interactive environment. It was chosen for the 8085 emulator because it provides a ready command interface for the required debugging support. We have verified the operation of the emulator in a multiprocessing mode through the use of two test programs operating in parallel and sharing data through simulated I/O ports.

In the second phase, the core of the emulator was rewritten in Motorola 68000 assembly language for use on our telescience Sun Microsystems workstations. (The original version of the assembly language emulator was written by Mr. Steve Rosenthal of MIT). The user interface was maintained in Magic/L.

4.3.7.5 Experiment Results

The two programs were run successfully on both versions of the emulation. However, the interpretive nature of the Magic/L language imposes severe speed limitations. We found that a single processor simulation runs at a speed that is approximately six hundred times slower than the actual hardware.

The assembly language version was a significant improvement over the Magic/L version. The single processor speed was found to be 33% of the real time speed. With the additional overhead of the hardware and multiple-processor simulations, the actual speed is approximately an order of magnitude slower than the hardware. A multiprocessor simulation running at, say, 1% of real time speed, would simulate several minutes of actual operation in an overnight run.

4.3.8 Teleoperations

One of the important goals of the telescience testbed program is to develop methodologies to allow scientists to interact with space based instruments as they would in a laboratory environment. To this end, we have undertaken a teleoperation project to determine if currently available technologies are mature enough to support real time operation with instruments from a remote site.

4.3.9 Remote Commanding and Telemetry Reception

Our teleoperation experiments used the EUVE payload, which is under development for a 1991 launch. They demonstrate transparent command and telemetry handling over various computer networks. Details of the experiments and results can be found in [Chakrabarti et al., 1988b].

4.3.9.1 Experiment Description and Summary

A hardware/software simulator for the EUVE, called Kiwi (for the bird that does not fly), was developed for EUVE. The Kiwi simulates the science payload, as well as the spacecraft, and was used in our experiments. The spacecraft was simulated on an IBM PC, equipped with special hardware and software. A simulation of a Science Operations Center (SOC) was created on a Sun Microsystems workstation which was connected on the SSL LAN. In this way, the instrument on a spacecraft is treated like a node in a computer network.

The Kiwi was always located at the SSL for all experiments. The user interacted with the Kiwi from:

1. another workstation on the SSL LAN;
2. a MicroVAX workstation running the Ultrix operating system located at the Stanford University; and
3. a Sun Microsystems workstation located at the University of Colorado at Boulder.

The user sent commands to the EUVE Kiwi from the workstations described above and received telemetry from it. In the last experiment, a video camera was used at the Kiwi to provide a visual confirmation of the execution of commands to Kiwi.

4.3.9.2 Issues Investigated

The first issues from these experiments indicate that current technology is adequate to support transparent teleoperations. We have run the experiment using workstations from different manufacturers. Our experience indicates that the use of the UNIX operating system provides some degree of vendor independence. However, various versions of UNIX are different enough that it was not possible to run the experiments without modifications to the software or the operating environment.

These experiments have highlighted a number of issues related to the applications of computer networks in space based research. Continuous access to network resources is mandatory for realistic teleoperations. Our experiments using the Internet were severely hampered by the poor reliability and low effective bandwidth of the channels. In running our demonstration between Boulder and Berkeley, for example, it often took several attempts to establish the connection to UCB. Once connected, we would experience generally good throughput for 5-20 minutes or so, but then the connection would be broken. A conclusion may be drawn that existing networks are adequate for electronic mail, block file transfer, and terminals but are inadequate to support the continuous bidirectional data flow necessary to support telescience. Current network technology is adequate, if specific links are dedicated to operation of instruments.

Software and, to a lesser degree, hardware compatibility still remain major problems. We attempted to run the teleoperation experiment from a DEC VAX 11/780 computer, running the Eunice operating system. Eunice provides one version of the UNIX operating system environment which runs on top of the VMS operating system. Although Sun's UNIX and Eunice appear to users to be very similar, we were unable to run the testbed with the clients running Eunice. We investigated the problem and determined that it was due to the security system's refusal to allow us to change the UNIX socket configuration. With proper access, one should be able to overcome this hurdle. However, we have successfully conducted the experiment with the client environment running Ultrix on a MicroVAX.

Using the UNIX operating system for both clients and servers has helped ameliorate some of these problems but an enhanced effort is required to develop networking tools that are independent of language and operating system. These include programming language bindings and basic packaged tools.

Command encryption and validation will be required during flight operations. We have made a small attempt to address the data link security control needs. For the time being, we are using the UNIX "crypt" program for testing purposes. Although "crypt" is not as secure as the Data Encryption Standard (DES), it provides a test of the software overhead involved in command encryption. This overhead can be substantial: during the Boulder demonstration it was necessary to disable the UNIX encryption tool crypt in order to make sure all the commands were correctly received and executed. Software implementations of DES, for instance, did not appear to have the speed to support telescience data rates. Sun Microsystems has provisions for hardware support of DES, but we need to evaluate capabilities offered by equipment from other vendors. We believe that further work needs to be done to standardize the encryption protocols to be used over telescience data links. We also note that the U.S. Government restricts the distribution of DES hardware and software outside the United States.

4.3.9.3 Experiment Hypothesis

The hypotheses for the teleoperation experiments were:

1. Technology is mature enough to support routine interactions with instruments in a

laboratory-like manner.

2. The computer networks are reliable enough to run the tests. Furthermore, the network bandwidth is sufficient to run the tests, provided data compression is applied to the telemetered data and video images. We also assumed that the delay of the Internet is representative of those encountered during interactions with a space-based instrument.

4.3.9.4 Method of Investigation

Our teleoperation testbed concentrates on the operation of the EUVE instruments over the NSI network. The Kiwi can be reached through a Sun Microsystems workstation that is connected to the SSL's LAN, so instrument commands can be sent to the CDP from any workstation connected to this network. This is not done with a simple remote login: servers on each workstation exchange data packets directly using TCP/IP.

The Kiwi is connected to the outside world through command and telemetry server workstation. A single-board 68000 computer acts as the hardware interface to the Kiwi. It buffers all telemetry and command requests. The server machine contains the telemetry and command servers, which are awakened by appropriate service requests from the clients (users). The clients can be conceived as operations requested from a SOC. The system responds in the same manner to a request from a machine on the SSL LAN, as to a request from a machine anywhere on the Internet; all that is required is that the machines and users have appropriate access permissions.

Each telemetry request is assigned a telemetry slave, but only one command slave exists at a given time. Subsequent command requests are queued. When control of the instrument is relinquished by the current commander, the next command request is served.

The purpose of the client-slave process pair is to mask, from the user, the physical implementation of accessing the payload. A client allows the user to treat a payload as a local, dedicated resource. This frees the user and tool developer from the specific development state of a payload. When tools are independent of the communications medium, they will function during development (when the breadboard payload is on a bench next to the user), or when the payload is on orbit.

A client provides only basic access to a payload at the lowest layer. Tools must be built to control the masses of data flowing to and from a client. Some of those can be utilized by many users; some of them will be unique to an individual user.

The UCB campus is connected to several of the TCP/IP networks that make up the Internet: NSFnet (National Science Foundation Network), ARPAnet (Advanced Research Projects Agency Network), and BARRnet (Bay Area Regional Research Network). Gateways to non-TCP/IP networks, such as Bitnet, SPAN (Space Physics Analysis Network), and Usenet, are also available. The SSL LAN is connected to the campus network through a 56K bps link.

After performing the remote operation experiment from a foreign host on our LAN, we conducted the same experiment over a wide area network. We installed the telemetry and command clients on a Digital Equipment Corporation MicroVAX computer, running the Ultrix operating system. We then ran an experiment in which commands and telemetry were exchanged between the Stanford computer and the Kiwi at UCB. All of the commands were received and executed properly. This experiment demonstrated that, in addition to interacting with an instrument from a remote site, we were able to achieve some vendor-independence.

In a second remote operation experiment, we operated the instrument from a Sun workstation located at the University of Colorado in Boulder. The experiment was demonstrated at the TTPP II meeting held at the University of Colorado, Boulder. Command and telemetry clients were run on the Boulder machine. The instrument remained on the UCB LAN and was connected to the client machines through the Internet. We established a command stream to operate the instrument and monitored the telemetry to verify proper execution. All commands received were executed properly.

The telemetry rate of the EUVE experiment is 32 kilobits per second, but data compression reduced this considerably for the demonstration: only engineering data were sent, so the telemetry stream was relatively empty. It is difficult to gauge the actual network throughput attained, but we estimate that we averaged approximately 4 kilobits per second, with occasional bursts of up to 8 kilobits per second between UCB and Boulder.

Data compression was achieved with the UNIX compress command. It uses an adaptive Lempel-Ziv compression scheme that provides good error-free compression with reasonable computational overhead [Welch, 1984]. At the client end, we used the UNIX uncompress command to regain the original data. Standard UNIX tools allowed us to achieve useful data compression, although they may not have provided the highest possible compression. The same data compression scheme was used for both the telemetered digital data from the instrument, as well as the accompanying video signals (described later).

The primary purpose of this experiment was to operate a d.c. motor that is connected to a door covering a high-vacuum cavity. During flight, two micro-switch monitors indicate the position of the motor; for the teleoperation demonstration, we decided it would be useful to use a video monitor to confirm the effects of the command execution. The use of video signals over the Internet concurrently with the command and telemetry signal was our first step to test the present network infrastructure. We expect that similar video images will play an important role in space experiments in the space station era.

Time and funding constraints dictated that inexpensive, off-the-shelf tools be used for video support. We selected the Imagewise video digitizing system made by Micromint, Inc., of Vernon, Connecticut. It consists of a digitizer/transmitter that collects individual video frames and transmits them over an RS-232 line to a receiver/display unit that converts the data back to a standard NTSC video signal for display on a monitor. This system provided us with a low-frame-rate video image, which was then sent over the Internet for visual confirmation of the telemetered door position values. The digitizing system can be operated

at several different spatial resolutions and gray levels. The system uses a run-length encoding scheme for data compression. Operating at 19.2K bps with a resolution of 128 by 122 pixels, it can transmit a video frame in 10 seconds or less. This was sufficient for the experiment - viewers could see images of the mechanism before and after the commands were sent.

Some additional software was written to allow us to transmit the video images over the Internet. The Imagewise transmitter sent video frames to one of the workstations in the SSL LAN, where a server program performed some additional data compression and transmitted the compressed data to a client at the remote site in Boulder. The Imagewise receiver and monitor were not used in this experiment. Instead, the client software uncompressed the bit stream and displayed each video frame in a window on the console screen of the Sun color workstation. A display program called "imtool," developed at the National Optical Astronomy Observatories for their IRAF image analysis system, provided a ready-made display window for the video images [Tody, 1986].

4.3.9.5 Experiment Results

All commands and telemetry were properly received in all three experiments described above. We have, however, discovered some problems in running the tests as described in section 5.3.1.2. The video images were also received over the Internet during peak use hours as was demonstrated at the Boulder meeting. Although, the network reliability and delay caused some irritation, as described in section 5.3.1.2.

Teleanalysis

We have examined two aspects of teleanalysis. First, we have investigated remote data analysis issues for the EUVE guest investigator program. Second, we conducted a simple real-time teleanalysis experiment with a sounding rocket, launched from NASA Wallops Flight Facility on September 30, 1988.

4.3.10 Remote Data Analysis for EUVE

One of the important components of the EUVE project is the Guest Observer (GO) program for spectroscopic observations with the Deep Survey/Spectrometer instrument. The current plans for the GO data analysis include, a local database and distributed scientific users having various levels of access to the database. These components are all similar to those in the Browse system, developed at the University of California at Santa Barbara (UCSB). Furthermore, the EUVE data makes heavy use of images of astronomical objects, another similarity with the Browse system. Finally, the spectroscopic data are obtained by imaging spectrometers, and therefore, they can be treated like images. For all these reasons, we have explored the possibility of incorporating the electronic Browse system or some of its key components in the EUVE GO data analysis system. The details of the investigation is described in Chakrabarti et al., 1988c.

4.3.10.1 Experiment Description and Summary

For this study, we have examined the use of Browse for the EUVE GO data analysis program. Several features of the Browse system can be incorporated for the GO data analysis program as described below.

At the discretion of the guest observer, the offline mode of operations will require permanent files that are generated in the production processing. Binned photon and exposure maps are generated which may be deconvolved for some studies (e.g., extended sources and crowded fields), while detailed photon and exposure data are saved in "pigeon hole" files for studies of individual sources. It is expected that the latter data set will be more commonly used. The collection of detailed data into pigeon holes (as applied to spectra) is governed by the list of sources detected in the deep survey instrument which provides a direct image of the target region. These data will be similar in format to those in the Browse database. As with the survey data, sources may be found which were not expected or which may be brighter than expected and thereby contribute to noise in the spectrum. Thus, offline analyses are required to collect data for sources that were not in the original catalog. Data for newly discovered sources will be collected from prior observations for future analysis.

The main differences between the handling of survey and spectroscopic data are in the treatment after collection into a permanent database. Generally, all data are handled by the GO interface software which allows data and software transfers via phone lines, networks, or physical media (e.g., optical disks for large requests or printout for users without access to computer networks). Browse has schemes that account for the different levels of access to the database; even different types of user environments (e.g., dumb terminal vs. graphics workstation interfaces) are also included in their interface. We will examine these schemes before selecting the GO interface.

The GO Interface selects the applicable portion of the database to minimize file transfer time and should warn the remote user if the requested data are not in the current database. In such a case, a special request is generated that requires action by the database manager, which controls the archive data. Requests for unavailable data are handled by generating a set of new coordinates for guiding the selector program via the database manager. This process insures that the raw data are managed properly by an onsite operator of the database managing program. Unlike Browse, we plan to allow remote logins but access to the production network will be restricted for security reasons. Temporary files may be created by running remote processes and then transferred to the GO's home institution.

The GO database plan calls for a file named "History." It contains general orientation information, instrument housekeeping information and other parameters such as sun angle, positions of the moon and other planets, etc. There are also plans for storing intermediate results such as exposure information, photon maps, data quality indicators, etc. This database is therefore essentially the same as catalog in the Browse system, which contains information that describes the quality of the data and other attributes.

It is expected that a number of specialized I/O and data analysis packages will be written specifically for use in the IRAF data analysis environment, an image analysis package

developed by the National Optical Astronomy Observatories (NOAO). These packages will be available for remote execution, or transfer to the GO's home institution for installation in IRAF. This is similar in concept to the software Browse will provide to its remote users for manipulation of images. The packages supplied for EUVE data analysis will run on any hardware configuration to which IRAF has been ported. Using IRAF's potential networking functions, one may also invoke this package at the remote site without transferring the applicable files.

We also anticipate several simultaneous observations with EUVE by other instruments (ground- or space- based). These GO nodes will contain special data not in the EUVE database, as well as specific scientific interest and expertise. These characteristics are also incorporated in Browse and are included in the directory and user databases. We anticipate that the EUVE GO database management plan will include similar features.

4.3.10.2 Issues Investigated

We investigated the usefulness of electronic browse for the EUVE guest observer data analysis program. The Browse system developed at the UCSB was examined in detail to check that the major EUVE GO functions could be supported. The incorporation of the IRAF image analysis system in Browse was also explored.

4.3.10.3 Experiment Hypothesis

We hypothesized that the Browse system can be easily adopted for the EUVE GO program.

4.3.10.4 Method of Investigation

We visited the University of Santa Barbara and took part in a demonstration of the Browse system operating on a wide variety of remote sensing data. We also were briefed on its specific capabilities. At Berkeley, we had several meetings defining the EUVE GO data analysis requirements. A comparative study was undertaken to determine the adoptability of Browse for the EUVE system.

4.3.10.5 Experiment Results

We are very encouraged by the capabilities of the Browse system, which should be adaptable for the EUVE image and spectroscopic data. The image database and the astronomical catalogs to be used in the EUVE program is/are similar to the remote sensing data obtained from a variety of platforms. We feel that Browse is a potential candidate for the GO data analysis system.

Unfortunately Browse, as it is presently implemented, uses a VMS operating system. All of the EUVE software is built around a network of Sun workstations running UNIX. Our SCS scheme, described earlier makes heavy use of UNIX facilities. Therefore, we will not be able to incorporate Browse readily.

Due to financial and schedule constraints, we have not been able to use actual astronomical data and used the Browse system to analyze them from different remote locations. We also have not explored the availability of Browse for use at the SSL, UCB without regular supervision by the UCSB personnel.

4.3.11 Real-time Rocket Data

Our final experiment used real time telemetered data from a sounding rocket. The sounding rocket, called Berkeley EUV Airglow Rocket Spectrometer (BEARS), carried a 92 lbs. spectroscopic payload to 963 km altitude on September 30, 1988. The reduced telemetered data was sent to the SSL from the Wallops Island using a 9600 baud modem over a telephone line. During the flight, scientists at Berkeley communicated over telephones with their counterparts at Wallops Island regarding the instrument and its operation.

4.3.11.1 Experiment Description and Summary

The BEARS experiment and its scientific objectives are described in detail in Chakrabarti et al., 1988d. The payload consists of:

1. a high resolution (approximately 0.5 angstrom) spectrometer to measure the EUV emissions (980-1360 angstrom) of the Earth's dayglow;
2. a moderate resolution (15 angstrom) EUV spectrometer (250-1450 angstrom), to measure the solar irradiation responsible for the photoelectron production; and
3. a hydrogen Lyman Alpha photometer to monitor the solar irradiance and geocoronal emissions.

Although at the time of the writing of this paper we had started our teleanalysis plans, they were not mature enough for inclusion in this paper. We hope to report these results in an upcoming publication.

4.3.11.2 Issues Investigated

The primary goal of the experiment was to involve scientists at their home institution in the interpretation of real time data. To our knowledge this was the first time a telemetry stream was decommutated in real time and spectra from the instrument, not just monitor values, were transmitted to remote locations for analysis. The final goal of this experiment is the development of hardware and software systems for real time teleanalysis, in conjunction with teleoperations.

4.3.11.3 Experiment Hypothesis

The primary hypotheses of the teleanalysis experiment are: real time teleanalysis even for short duration sounding rocket flights are useful from their scientific return. The technology is mature enough to support teleanalysis.

4.3.11.4 Method of Investigation

An IBM PC/AT based system equipped with special hardware was used to collect selected words from the telemetry stream. Specially written software used the data words thus obtained and formed spectra from the three spectrometers flown aboard BEARS. This was rather computer intensive, since the detectors employed were position sensitive detectors, and telemetered three 14-bit words describing the locations of each photon event recorded by the detectors.

The spectra generated were transmitted to the SSL from Wallops Island, using a pair of Microcom 9600 baud modems, using the MNP protocol for data compression and error correction. The spectra were displayed to the SSL real time data analysis team. Intensity ratios of selected spectral lines, background rates, spectral contents were all evaluated. For this experiment, we did not have any commanding capabilities. So the analyses were used to interpret the quality of the collected data and problems with the instrument.

4.3.11.5 Experiment Results

The teleanalysis experiment showed how valuable real-time teleanalysis can be. We did discover a problem with the instrument. Had we any commanding capability, we would have tried to interact with the instrument, to fix it. This is one of the prime goals of telescience: to give scientists an environment which "feels" laboratory-like. The reduction of telemetered words into physical form (intensity vs. wavelength) and their representation in familiar graphical form made the task of real time interpretation a success.

4.4 UNIVERSITY OF CALIFORNIA, SANTA BARBARA

Our main accomplishment during the month of August was our field experiment. The following briefly outlines the experiment itself and the work which will follow.

The telescience field experiment was coordinated with the overpass of Landsat 5 satellite on August 3, 1988 over the U.S. Forest Service Experimental Range Station at Coarsegold, California (path 42, row 34). Field work went from Aug. 1-3 and involved an effort to coordinate in-field data collection with a database and processing facilities at the home institution (UCSB). The goal of the field work was to collect data which could be used to study the depression of the thermal signal measured at the satellite due to varying amounts of transpiring tree canopy.

Field data collection involved sampling for percent canopy cover and ground surface brightness temperatures. Only a limited amount of the range station could be sampled from the ground, so area samples of percent canopy cover were regressed with existing video digitized photography to estimate this parameter for the entire range station. Canopy cover was estimated from the ground using the point-centered quarter method (as per "Methods in Plant Ecology", Moore and Chapman, 1986) for estimating tree density and convolving this value with the average measured tree canopy area. Sample values were recorded and

locations were marked on USGS 1:24,000 scale maps. After field samples had been taken, a MicroVAX II at the home facility was logged on to, using a modem and commercially available communications software. An imagery search was performed on the UCSB BROWSE database testbed, and online video digitized photography of the experimental range was located and transferred to the user account using the TCP/IP Telnet file transfer program. The data was subsequently processed using ERDAS image processing software. A ratio image of the near IR to red bands of the data was created. The imagery was acquired late in the summer when the grass layer had become senescent, so the ratio should have some relationship with the amount of photosynthetic activity in the tree canopy.

In order to coordinate the imagery with ground observations, the imagery needed to be rectified to the USGS maps used in field data collection. The IR/red ratio image was downloaded to the personal computer in the field using the KERMIT file transfer program. After transfer of the imagery, the data was displayed on "in house" developed image processing software for a personal computer with an EGA monitor. Ground control points were located in the image and on the map, and these values were used on the ERDAS software to rectify the imagery to a UTM projection for the study area. The data was resampled to a 120 meter resolution to coincide with the resolution of the TM thermal channel acquisition. Field measured values of canopy cover were transferred to the UCSB facility and the ratio values for coincident points in the rectified image were extracted. These values were then transferred to a VAX 11/750 computer at the UCSB facility for statistical analysis using the "S" software package. Ratio values were regressed against field measured values, and coefficients were returned to the MicroVAX. Once the regression coefficients were returned to the MicroVAX, they were applied to the ratioed imagery, to provide a new image converted to field measured units. If realtime data acquisition was available from the TM sensor, this image might be used to compare canopy density to the thermal signal. At the time of Landsat overpass on August 3, the field crew was measuring ground surface brightness temperatures in the field, using hand-held thermal radiometers. Soil surface temperatures were measured as well, under a number of conditions. Given an adequate characterization of this "background" thermal brightness and the canopy cover map generated the previous day, the effect of the transpiring canopy on the thermal signal at the satellite could be studied. The Landsat TM data has been ordered, and we are waiting for its arrival in order to complete the study.

Correction of atmospheric effects would be required in order to transform the effects of the canopy on the thermal signal to equivalent degrees of temperature. Acquisition of atmospheric data from the Solar Mesoscale Explorer satellite was arranged with the University of Colorado, Boulder. Data was requested for August 2, however, there were some thin high clouds observed in the area on this date, which did not seem to be present on August 3. In addition, the SME profiling through the limb of the atmosphere may not be of proper resolution for this study. We are currently waiting to receive the data acquired from the SME.

Problems which were encountered during the Telescience field experiment included the technical difficulty of finding a telephone at a remote study location. The next difficulty was that the telephones in that area were not mounted with the current standard modular jacks

which were required for the modem. However, a phone with a modular jack was eventually located. In terms of field data collection, difficulties were encountered in determining the best ground sampling procedure given the variation in slope, aspect, and wind exposure. The three man crew was also only capable of obtaining a limited amount of measurements to correspond with the satellite overpass, as this occurred at 9:30 AM when the ground surface temperature was rapidly changing.

4.5 UNIVERSITY OF COLORADO, LASP

4.5.1 Remote Access to Data for Teleanalysis

Objective

Establish and exercise a capability which will make it possible for others of the consortium to access the LASP SME data, and for us to access other members' selected databases.

Progress

1. improvement of the SME MENU system and the SME DATA database: Julie Lawe completed the changes to the MENU program to handle a modification of the data set structure. Barry Knapp completed the cleaning up and integrating of the Interactive Data Language (IDL) routines which are called by the MENU program. The SME data sets have been reprocessed during the summer and are being kept current. As a result of these actions, the MENU system is considerably smoother and more dependable in its operation and, therefore, more suitable for use by distributed users in a telescience environment. Also, all of the SME final science data products are accessible online, within a week or two of data acquisition.

2. preparation of User's Guide: Work has continued on an improved SME Science Data User's Guide, which will be suitable for use by scientists not as familiar with the SME data system as are the local researchers. Changes to the first draft are currently being made in order to reflect the changes in the system discussed above. Other background materials are being collected for incorporation into the guide.

4.5.2 Earth Science Telescience Field Campaign

Objective

The objective is to demonstrate the ability of selected telescience techniques and capabilities to improve the conduct of a scientific investigation in a field environment: specifically, use SME as a data source for a collaborative field campaign.

Progress

1. ***A Lesson Learned.*** The approach outlined in the last quarterly report to identify a

single major field test site and common scientific goals for multiple Earth Science testbed collaborators has been set aside. It turned out to be unrealistic in the context of the current TTPP project approach and resource level. It was simply not possible, in light of the scientists' ongoing workloads, to command their attention to the extent necessary to plan and conduct the campaign. To be successful the earlier envisioned approach would have to include identifiable support for the space science research component as well as the telescience technology component of any field campaign.

2. *Alternative Approach.* As an alternative to the earlier approach, still capable of achieving the basic objective, each Earth System Science Testbed collaborator has identified a task which is fundamentally local in nature and directly related to research already ongoing within his or her institution. Having done that, each collaborator is working with the other Earth Science collaborators to identify enhancements to the task which make greater use of non-local telescience techniques and capabilities.

3. *LASP Earth System Science Testbed Task.* Dick White is conducting an analysis of time series of solar line intensities measured for the full solar disk by the LASP UV spectrometer on SME. Dick, as a semi-retired former LASP researcher, and in collaboration with several current LASP scientists, is conducting most of this work from his home in south-central Colorado. He uses a personal computer at his home which is connected to the LASP VAX computers via telephone. He accesses the SMEDATA database, does remote computing in the VAX computers, transfers the graphical and tabular results to the PC, and does the final preparation of the data at the remote site. In this work, he makes full use of the SME MENU system and SMEDATA data sets. A report of this work, "SME as a Testbed for Telescience - A Case Study" is in preparation. It will include a discussion of the methodology, a description of the system, scientific results, and an evaluation of the approach (lessons learned). We expect to be in a position to demonstrate this approach widely to NASA managers, scientists, and engineers.

4. *Support of the UCSB ESS Testbed Task.* Jeff Star and Ken McGuire at the University of California at Santa Barbara conducted their first ESS field exercise on 1-3 August 1988. We provided access to the solar data from the SME spacecraft for their use in making atmospheric corrections to the Landsat data and to provide an indication of solar activity. These SMEDATA data sets are available to them electronically via the SME MENU system which was discussed earlier. In addition, LASP has been exercising the SME spacecraft to ensure that we can provide special operational support for field campaigns where instruments other than the solar spectrometer are needed. Elaine Hansen scheduled data acquisition from the IR Spectrometer (airglow instrument) and UV Spectrometer (ozone instrument) in support of the UCSB field exercise on 1-3 August, with concentration on obtaining additional spectral data in the 1.27 and 1.86 bands. Although this operation was unsuccessful on the first attempt due to TDRSS problems and the long lead-times required for SME repositioning and instrument temperature stabilizing, the measurements were made successfully in mid-August. This satisfactorily demonstrates the ability of the SME spacecraft and control center to support this type of special operation for future field exercises.

4.5.3 Telescience Technologies

4.5.3.1 Sun Porting

Objective

The objective is to make the OASIS teleoperation package available to the Sun community.

Progress

In July, a new release (V02.03.05) of the VAX/VMS version of the OASIS teleoperation package was made available. Among the many new features included in this release was a new version of the CSTOL parser implemented in Ada. In the previous version, the parser was implemented using a the VMS TPARSE utility and therefore, was not portable. The SUN workstation to be used in the development of OASIS was received at the beginning of August (several months late), with only 4 of the 16 megabytes of memory required. Porting of OASIS has started and will continue over the next few months. Progress is being slowed by the lack of memory. The rest of the memory is scheduled to be shipped in late October.

4.5.3.2 Reactive Control/Command Interlocking Study

Objective

A teleoperation system, with distributed users directly controlling their instruments on a spacecraft, requires that the instrument, the spacecraft, and its crew must be protected from accidental, erroneous commands and malfunction. The approach proposed for the Space Station consists of techniques to maintain the operation of an instrument within preallocated, scheduled sets of limits on the resources (often called resource envelopes) used or requested by the instrument. Instrument operations are maintained within the established resource envelope through a combination of lockouts on requested resources and reactions to out-of-conditions. The goal of this study is to prototype these protection techniques and to evaluate these technologies.

Progress

The Reactive control/command interlocking functions have been prototyped using the existing remote control of SME instruments testbed. This testbed (named TIGS) has been developed at LASP during 1987 under a separate NASA contract. The instrument Operation Management System (OMS) is provided by a downgraded version of OASIS package integrated into the TIGS instrument control software. We found that this design provides a fairly good match between the conceptual design of an instrument OMS and the actual simulation. The higher level of management is provided by the user to whom instrument interlock override requests are submitted and who can issue reactive control message to instruments. The prototype is now fully developed and is currently being tested.

The development of this prototype is raising the question of how a user teleoperating its instrument is made aware of actions taken by the higher level of management.

This development also shows that the distinction often made between hardware and software interlocks may not be 100% valid. In the prototype implementation, the higher level of management can force the instruments to request interlock override on some commands and can make some commands illegal to some instruments (the equivalent of a hardware interlock). This is because in our implementation the higher level of management controls a database that instruments have to access to get to their commands.

4.5.4 Astronomy Testbed

The astronomy testbed environment, consisting of a remote user operations interface, communications links, and the 16-inch telescope at the Sommers-Bausch Observatory on the University of Colorado Campus, is complete. This testbed has investigated the impact of distributing control and monitor operations of sophisticated scientific instruments by remote users. The remote user can be located anywhere on the local video network, which includes the entire Boulder campus of the University. The demonstration has involved scientists from all disciplines and has moved to several locations on campus. The suggestions from these scientists have been used to identify problems associated with remote operations, to determine how science instrumentation and experiment strategies can be enhanced to take advantage of telescience capabilities, and to learn how to take advantage of unpredicted observing.

In particular, the following features have been implemented during the last quarter:

1. The hardware necessary for remote control of the telescope focus has been installed and tested.
2. The user catalogs have been implemented to allow users to respond quickly to targets of opportunity.
3. A regular observing program has been instituted, with a new guest observer each week.
4. The telemetry format has been modified to enhance OASIS performance.
5. The help files have been completed on all OASIS items and a hardcopy help manual has been produced.

The following features are still being developed:

1. the investigation of simultaneous control by multiple users at separate locations;
2. weather prediction and monitoring to allow for efficient observations and safety;
3. efficiency testing to determine the ratio of search time to observation time; and
4. the User's Guide.

Preliminary Observations

1. There is a need to increase the menu network, so the observing can work in a

modeless environment.

2. The user interface should have an easy to use tutorial mode for the occasional users.
3. The performance of the remote user is limited by the ability of the local controller to transmit timely telemetry. Delays of more than one second seriously restrict remote operations.
4. The planning function is critical due to user fatigue during extended observations.

4.6 CORNELL UNIVERSITY

4.6.1 Current Status

An objective of the Cornell telescience effort is to perform remote access, reduction and analysis on data obtained with the University of Rochester infrared array camera. The data analysis software and data is located at Rochester. Based on our experience of trying to run IRAF remotely over the network with the University of Arizona, which demonstrated this is not feasible, we have decided that it is better to have analysis software at the home institution where the analysis is being done. The data can be transferred over the network (assuming it is not too large a data set) and software development can also be reciprocal in this fashion.

Work over the past several months has centered on developing software to enable us to handle the Rochester data. What becomes evident in this process is the need for standardization. Although this seems obvious, our experience emphatically demonstrates this. The University of Rochester experiment runs on a PDP 11/15 computer. This selection is a bit historical in nature, since at the time the selection was made limited computers were available that had sufficient processing speed and commercially available fast A/D boards to be able to handle the data rates at reasonable cost. The problem of handling real time operations was also an issue. As in many "early" astronomical applications, FORTH is used for both data acquisition and data processing.

The tasks are several fold:

1. transferring data between the Rochester PDP 11 computers and the Sun 3/60 at Rochester (purchased via the Telescience Testbed pilot Program);
2. transferring data to Cornell; and
3. analyzing the data here on a Sun 3/60.

These tasks have to be done while maintaining data integrity and performing the proper operations for data analysis. It should be pointed out that the data analysis procedure is nontrivial, involving many operations. Our selection for the simplest means to transfer data between the PDP 11 and Sun computer is to use Kermit. Because of problems with linking the Rochester Sun to their campus network (pointed out in an earlier report), we cannot yet link directly between Cornell and Rochester, and various means are being used to communicate data and code back and forth. This includes low speed modem links and hand

carrying tapes.

To perform the analysis at Cornell then, we are faced with the task of either rewriting the analysis software in a new high level language (C, Fortran), writing analysis routines to work under some already established image processing environment such as IRAF, or bringing up FORTH on the Sun and porting Rochester's current analysis software from the PDP 11 to the Sun. The first of these tasks was ruled out immediately because of the complexity of the task and the manpower required. Rochester has invested many man years of effort in the development of their software. As it turns out that IRAF is not well suited to handling many small array images (32x32 for the Rochester array) and does not have many of the routines necessary for the reduction process. After discussions with our programmer, Justin Schoenwald, it was decided to bring up FORTH on the Sun and attempt to port Rochester's software to the Sun. This work is now in process.

Our selection of FORTH as a programming language does not seem wise from an aesthetic viewpoint, however, from an implementation aspect it may logically be the quickest route and the code will then be portable to other Suns (and hopefully, to other UNIX based systems). After processing with the FORTH code, output will be written in a standard format, e.g., FITS, so that IRAF can be used for some final analysis, display, and plotting.

In summary, we conclude that having common hardware would have alleviated the problem of porting software but this is not a practical solution since it would not take advantage of current technology. The real solution nowadays is the selection of software and/or hardware which allows easy upward migration in the future as technology evolves. This seems to be a much more feasible alternative now than it was several years ago.

4.6.2 Future Plans

Over the next two months, our goal is to finish our porting of the Rochester FORTH software to the Sun and to begin analysis of the Rochester data. It is also hoped that the Rochester network link will finally be completed, so that we can have dynamic software development and data analysis occurring between Rochester and Cornell.

4.7 UNIVERSITY OF MICHIGAN

Teleoperations

4.7.1 Offline Guidance Requirements

The objective of this effort was to determine the system requirements for offline guidance of a remote technician.

4.7.1.1 Experiment Description and Summary Conclusion

We investigated the required *modus operandi* of a remote coaching system while in the offline state. This was accomplished through a collaboration between the project design team and selected personnel from the Space Physics Laboratory. Biweekly meetings were held to review the project design, to highlight any apparent limitations, and to suggest useful additions to the system.

4.7.1.2 Issues Investigated

What is the level of expertise of the user of the system? What methods of guidance should be provided? What are the most effective man/machine communication methods that can be employed?

4.7.1.3 Experiment Hypothesis

It was hypothesized that the level of expertise, and hence the training time, for a competent technician could be significantly reduced by providing an expert system with which he could interact. The technician is assumed not to possess all of the knowledge necessary to install and maintain sensitive instrumentation.

4.7.1.4 Method of Investigation

An expert systems programmer from the Robotics Laboratory and a research associate from the Space Physics Laboratory worked together to determine the requirements for offline guidance. Knowledge concerning the Fabry-Perot interferometer was obtained from the Space Physics Lab research and converted into a set of rules for the expert system by the expert systems programmer. The two met biweekly to discuss the operation of the system, to determine what rules were incorrectly coded, and to evaluate the utility of the current system.

4.7.1.5 Experiment Results

It was determined that three modes of operation should be provided by the system: inquiry, maintenance and diagnostic.

A novice user of the system uses the inquiry mode as a learning tool to provide a definition of terms used by the expert system and the physical appearance of different components of the instrumentation. The use of the inquiry mode diminishes as the technician becomes more experienced. The operator can make general inquiries about the operation of the remote coaching system, the specific instrumentation being maintained (A Fabry-Perot Interferometer in this case) and the procedures used for maintenance.

The maintenance mode is used after it has been determined that a specific component of the system is not working correctly. It leads the technician through a sequence of steps which are required to determine the cause of the problem. Each of the steps in the maintenance mode requires a number of substeps. Associated with each substep is an alignment rule which requests the technician to proceed with some action. After this action is completed,

and dependent upon the request made by the technician, the system either continues to the next logical step in the maintenance procedure or establishes a communication link to the real expert at the local site. To enhance the communications between the expert system and the technician, a picture of the device related to the current step is displayed on the video monitor. Text is displayed to inform the operator of the required actions and, finally, the system continues to the next step of the maintenance procedure.

The diagnostic mode is used to determine the cause of a problem. The diagnosis mode uses backwards chaining from the initial knowledge that the interferometer is not working correctly to determine the cause of the fault. As an example, the computer system, pressure system, mechanical alignment system, aperture system, and laser system must all be operational for a Fabry-Perot Interferometer to work correctly. First, the computer system is checked, then the pressure system, and so on. If one of these subsystems is found to be faulty, then the conditions for it to be operational are checked. For example, if the Aperture System is faulty, then the Input Aperture is checked to see if it is too large. This process is continued until every known cause of a problem has been checked.

The system was programmed using the CLIPS expert systems programming language[1]. It is designed for efficient forward chaining inference and incorporates the Rete Algorithm[3] for high speed pattern matching. The ability for the programmer to define new functions for the expert system greatly enhances the utility of the CLIPS language. The retrieve-image-rules rule, which is used to display images from the video database, is an example of the use of such functions.

4.7.2 Online Guidance

The utility of any expert system is limited by the amount of knowledge contained in the system. In anticipation of this lack of knowledge, we investigated methods by which the technician could place a call to the real expert for additional guidance.

4.7.2.1 Experiment Description and Summary Conclusion

An investigation into the methods which could be used for collaboration between the local expert and the remote technician were investigated.

4.7.2.2 Issue Investigated

What are effective methods of communications between the local expert and the remote technician?

4.7.2.3 Experiment Hypothesis

It was hypothesized that a technician will eventually require additional knowledge which is lacking from the expert system and that an effective means of communication with the expert is via normal telephone lines.

4.7.2.4 Method of Investigation

Several meetings were held to discuss the importance of different mechanisms for communication.

4.7.2.5 Experiment Results

It was decided that if the remote technician could not correct a problem with the knowledge provided by the expert system, then a mechanism should be provided to enable him to contact the expert for help. For this reason, during all three modes of operation, inquiry, maintenance, and diagnosis, the technician can ask for online guidance from the real expert at the local site. The system should then dial up the host computer of the local expert and the technician would log onto the computer in the normal manner. He can then determine if the expert is currently logged onto the computer and if not, he has the option of leaving a message for the expert using the expert's host computer electronic mail system. On the other hand, if the real expert is logged in, a dialog can be initiated between the real expert and the technician. This is done by the remote technician by invoking a communications program on the experts host computer. The expert must then proceed to a workstation equipped with a video monitor and start-up his communications program.

At this point, both the expert's and the technician's screens are divided into three windows. Both screens at the remote site and the local site are functionally the same. The window at the top of the screen displays the transcript of the expert system running at the remote site. The two windows at the lower part of the screen are used to display the text dialog between the technician and the real expert. When the dialog begins, both the remote technician and the local expert can enter text through the dialog windows. In addition, both can enter responses to the expert system program running at the remote site, display images from the video database, and capture live images.

Identical video images of instruments and expected sensor outputs are stored in databases at both the remote and local sites. When the technician explains a problem in terms of an object or a sensor output, he can simply display the related image from the video database at both sites. The objective of the design is to provide the expert with a view of the current state of the remote system and to provide effective methods of communicating with the remote technician.

A pointing device overlaid on the video monitor would significantly enhance the operation of the system. Both the real expert and the technician would have access to such a device. An example is a laser power supply containing several control knobs. If the expert wants the technician to turn a particular knob, he can simply point to it. Similarly, we have found that video images of devices are often cluttered with stray components, not immediately of concern. We have temporarily solved this problem by simply having someone point to the particular device with his finger before the picture is taken. The problem with this approach, is the requirement for many pictures of the same scene. A better solution would be to associate a position of the pointer with different rules when a picture is displayed in addition to the name of the picture. Thus, a picture containing several objects could be used by a

number of rules.

Workstations and Communications Infrastructure

4.7.3 Portable Workstations

This effort was to determine the feasibility of using a portable workstation at the remote site.

4.7.3.1 Experiment Description and Summary Conclusion

The required workstation environment for supporting remote coaching was investigated. We did this by reviewing the different manufacturers of portable computers which could satisfy the following requirements:

1. lightweight and easy for one person to carry;
2. a large variety of expansion boards, such as vision boards, should be available at a modest cost; and
3. a large amount of software, such as compilers, should be available.

4.7.3.2 Issue Investigated

What is the possibility of using a portable computer for the remote workstation?

4.7.3.3 Experiment Hypothesis

A portable computer could be used as the host computer at the remote site for remote coaching. This computer would have sufficient memory and expansion boards which are required to support all of the features of a remote coaching system.

4.7.3.4 Method of Investigation

A computer system was selected for use by the technician at the remote site. This system was evaluated in terms of the requirements for remote coaching.

4.7.3.5 Experiment Results

The Compaq Portable III personal computer was selected as the host computer for the technician at the remote site. This particular computer is an IBM AT clone. We found the Compaq to be completely adequate in most regards. The expansion boards for vision and serial I/O were readily available and performed as expected. However, we found two limitations which could be easily resolved.

The first limitation dealt with memory, both RAM and disk. We found that the IBM PC systems were originally designed with the notion that no user programs would ever require more than 640K of RAM. Though the Compaq was equipped with 640K of RAM, we felt that more memory would be needed to support extensions to the expert system. Hence, the use

of extended memory was explored. It was found that it is possible to store data in this type of memory, however, all program code must reside within the 640K barrier. Further, a program that wishes to store data in the extended memory space must be specifically written to do so. Since the CLIPS program was obtained from NASA, we could not take advantage of any extended memory.

The 640K RAM limitation was a severe restriction on the implementation of the remote coaching system. The CLIPS expert system program requires 200K to load before any rules are read into the system. The remote coaching rules which were developed for maintenance of the Fabry-Perot interferometer required another 500K of additional memory to load. Thus, as the system currently is designed, it is impossible to load the entire remote coaching system. Only smaller pieces of the system can be loaded at a time.

Another memory limitation is the number of video pictures which could be stored. Each picture requires about 250K bytes for storage. Thus, a 20 Megabyte disk can only hold about 80 pictures. A fully operational remote coaching system would require many more pictures.

The second limitation has to do with the DOS operating system used by the IBM PC type computers. The ability of the remote technician to continue working with the expert system while communicating with the real expert implies multitasking, one process for the expert system and the other process for communications. Not only is DOS not a multitasking operating system, it is not reentrant, which makes it difficult to develop multitasking software which uses the system calls provided by DOS. Such system calls are used by all software developed for the PC for such things as terminal I/O. Our temporary solution to this problem was to use a multitasking software package called DESQView. This software runs on top of DOS and performs task switching between system calls by monitoring all system calls made by the user's program. This method works fairly well for most programs, provided they do not directly access any system memory, such as the monitor display memory, or define interrupt routines for such things as serial port drivers which might be swapped out to disk.

Another indirect problem with using DESQView was the additional memory lost for running DESQView, which reduced the amount of memory available to run the remote coaching software. However, the DESQView system did allow us to get a portion of the remote coaching system operational, with communications from the local site to the expert.

Both of these problems could be addressed fairly easily. The memory problem with storing video images on a hard disk could be solved by simply using a video disk. Versions are available, which can easily store thousands of pictures. The problem with the 640K RAM limitation and the multitasking requirements can be solved by using a more powerful operating system such as a UNIX clone. The popular XENIX operating system would be a natural choice. With this system, the Compaq could support up to sixteen megabytes of RAM. This should be plenty of memory for even the most demanding remote coaching system.

4.7.4 Communications Architecture

The objective of this effort was to determine the requirements for communications between the remote and local sites.

4.7.4.1 Experiment Description and Summary Conclusion

We investigated the ability to present a view of the remote coaching workstation on the local experts workstation.

4.7.4.2 Issue Investigated

What kind of communications architecture is required to communicate between the local and the remote site?

4.7.4.3 Experiment Hypothesis

An expert could be presented with a view of the state of the system at the remote site. It could be made to look as though he were watching the operation of the technician as he interacts with the expert system at the remote site, giving him additional guidance, as needed.

4.7.4.4 Method of Investigation

Existing communications protocols were investigated to determine if any were applicable to the remote coaching. If not, a new communications protocol was defined to satisfy the requirements of this project.

4.7.4.5 Experiment Results

Several different serial communications protocols were investigated, such as KERMIT, XMODEM, YMODEM and ZMODEM. In each case, it was found they were all designed for file transfer from one computer to another. The remote coaching application is quite different. Not only is the ability to transfer files a requirement, but also the expert and the technician should be able to enter into a dialog while the expert system at the remote site is running. The expert should be provided with a window on his workstation showing the interaction between the technician and the expert system. Our solution to this problem was to divide the software at each site into four separate programs.

At the local site, where the expert is located, there are four programs running concurrently. The first program is a communications program used to direct information to and from the remaining three programs, through the serial port which is connected to the modem. The connection between these programs, is via TCP/IP based socket calls. This allows communication to a multitude of computers other than the one where the serial port is physically located. The second program receives text from the communications program and displays it in a window on the monitor screen. The third program, Prog3, receives text from the communications program, and displays it in a window on the monitor screen. The fourth

program receives text input from the technician and sends it to the communications program to be routed to a text window on the remote technicians workstation.

At the remote site, where the technician is located, there are four programs running concurrently. The first is the communications program, which is used to direct information to and from the remaining three programs through the serial port, which is connected to the modem. The second program receives text from the communications program and displays it in a window on the monitor screen. The third program receives text from the communications program and displays it in a window on the monitor screen. The fourth program receives input from the operator and sends it to the communications program to be routed to a text window on the local expert's workstation.

Thus, there are several windows located at the expert's workstation and several located at the technician's workstation. Text, which is entered at the technician's talk-local window, is displayed at the experts talk-remote window. Thus, text entered in one window of the technician's workstation is routed to different windows at the expert's workstation. To make all this work we had to define and implement our own communications protocol. The protocol enabled variable length packets of character information to be transmitted to and from processes running on the remote portable computer back to specific processes running on the expert's host computer.

References

1. NASA, CLIPS User's Guide, Lyndon B. Johnson Space Center, 1986.
2. Forgy, C. L., "Rete: A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem," *Artificial Intelligence* 19, 1982, pp 17-37.
3. Winston, P. H., *Artificial Intelligence*, Addison-Wesley Publishing Company, 1984.
4. Rauch-Hindin, W. B., *Artificial Intelligence in Business, Science, and Industry*, Vol. 1,2, Prentice-Hall, 1986.
5. "Pioneering the Space Frontier," U.S. National Commission on Space, NASA, Library of Congress, 1-211, May 1986.
6. Ride, S.K., "Leadership and America's Future in Space," NASA, August 1987.
7. Walker, M.W., Sheu, S-Y, Volz, R., Conway, L., "A Low Cost Portable Tele-Autonomous Maintenance Station," SOAR 88: A Workshop on Space Applications of Artificial Intelligence, Human Factors and Robotics, July 20-23, 1988, Wright State University, Dayton, Ohio.

4.8 MIT, ASTRONOMY

4.8.1 Status and Accomplishments of the Telescience Effort

1. application of concepts, methods, techniques, and information gained during the telescience testbed project; and
2. management of the collection of and rapid reduction Gigabit data sets from a fast imaging photometer.

Successful Science Experiment (The first occultation of Pluto's atmosphere using the KAO flying observatory Ref. Elliot et al.)

1. use of optical discs to archive images and programs used to acquire and process the images;
2. use of connectivity within the experiment;
3. use of connectivity for data collection;
4. use of connectivity for data archiving (accessing the optical disc); and
5. use of workstations capable of image analysis integral to the experiment.

Integration of the Computer Research Environment with the ATHENA Campus Network

1. Xwindows and utilities;
2. attaching directories to lab computers from other sources pluses and minuses;
3. use of research machines by students;
4. large programs kept in ATHENA 'disc lockers'; and
5. large data sets used by students also kept in the ATHENA lockers.

Use of a Cluster of Machines to Accommodate Parallel Data Analysis of Data Sets (similar to the use of many machines by a class of students)

1. good connectivity;
2. accommodates transfer of simpler programs under UNIX and 'C';
3. non-compatibility due to vendor differences;
 - a. IRAF not ported to IBM machines
 - b. data storage and programs not interchangeable on the executable level
4. exceptional ease to use many machines with the initial vendor (DEC) machines.

Use of the Xutilites (The use of a remote machine's terminal as if it is a local machine.)

1. Xdefine of the display: allows full use of the redefined terminal over an ethernet; and
2. i-mouse, graphics, and keyboard.

4.8.2 Experiments Under the MIT Astronomy Telescience Testbed

Experiment

Incorporating telescience concepts, methods, and hardware into a scientific observing program.

General Description

MIT Astronomy has conducted an experiment to use telescience methodologies in a very intense and successful occultation observation.

Summary

We have investigated the benefits and/or drawbacks of working in a real time environment, where the addition of telescience methodologies could be evaluated as to their usefulness in terms of the success or failure of the experiment itself. We believe this to be an acid test of the applicability of the methodologies.

We used the occultation of a star by the planet Pluto (a rare event) as the *SCIENTIFIC* experiment (vs. the telescience experiment) that would set the stage for this telescience test. The science experiment was very complex and required significant feedback to the investigators on a daily, and at some points, an hourly and minute by minute, basis. Communication with collaborators in Flagstaff, Arizona was also required. The *SCIENCE* experiment may be divided into three different parts for the purposes of this discussion:

1. predictions;
2. data collection; and
3. data reduction.

Description of Astronomy (science) Experiment

This phase consisted of determining where on the earth to deploy the KAO flying observatory to observe the occultation. In order to make the predictions, astrometry of Pluto and the star to be occulted was performed.

Data Collection

The data was taken with an imaging ccd photometer (MIT SNAPSHOT camera) placed aboard the KAO flying observatory. This consisted of two flights, the first being a calibration and check out flight.

Data Reduction/Analysis

The data reduction took place on the MIT campus, primarily on a VAX 750, and from there ethernetned to a MicroVAX II/GPX for archival onto an optical disc. Analysis started at this point between ourselves and collaborators.

Experiment Summary (science)

1. Software Development (June 1987 - January 1988)

From June 1987 to May 1988, a team of 10 people (3 MIT staff, 2 Lowell staff, 2 graduate students, and 3 undergraduates) worked part time to define the software, hardware, and data needs of the Pluto occultation project. The preliminary planning was done from June to September 1987, at both MIT and Lowell Observatory in Flagstaff, Arizona. From September 1987 to January 1988, all phases of the project were planned in detail: necessary software was identified and specifications were laid out, changes to the CCD and data

recording system were planned, and MKO was chosen as the observing site. During this phase of the planning, we had seven people working between 5 and 20 hours each per week .

2. Software Development (January - May 1988)

From January to May 1988, the software plans were finalized, the programs were coded, and the data reduction procedures were decided.

The data were in the form of spacewatch "strips": the telescope was locked into position, and the CCD was clocked out at the same rate at which the sky was passing by the chip. This results in a strip of data which is as wide as the CCD chip, and can be as long as the data recording system can handle. The data reduction "pipeline" was designed to handle these spacewatch data strips, which would be arriving from MKO at the rate of about 15 strips per night, with 8 Mb per strip. These data would enter the pipeline as strips, and after being processed by 10 programs, would exit as "star lists," row and column coordinates of each previously chosen network star in the strip. The end of the pipeline involved checking the star lists for any obvious problems, and then archiving the original data and the intermediate results onto OPTICAL DISKS. The planning, coding, and implementation of this pipeline was done by 3 people working 20 to 40+ hours per week .

3. MKO Observing

Set-Up

The CCD went out to MKO on 17 April 1988. The first data were taken on 23 April, and arrived at MIT on 28 April. Focus and coma problems were solved, and first astrometric data was taken on 1 May 1988.

Data Obtained

Data were taken on 31 of the 41 nights from 23 April to 2 June 1988, averaging 5 magnetic tapes per night (3 strips per tape). Tapes (160 tapes, 25 Mb per tape) were recorded for a total of 4000 Mb of data.

Observing Time Needed

Each night, observing activities lasted approximately 8 hours per night, with 7 hours spent at the telescope, and 3 hours observing the Pluto fields. After arriving at the telescope and opening the dome, a 1-2 hour wait was necessary to equalize temperature and improve seeing. There were 2-3 people observing at a time, but two people were sufficient after a routine was established.

4. MKO Data Analysis

Data Description

The relevant portion of the sky was divided into 11 areas, and data were taken, such that each strip fit almost exactly the boundaries of a single area. Areas I, III, V, VII, IX, and XI were abutting areas, and cover the portion of the sky in which Pluto was during 28 April - 9 June 1988. Areas II, IV, VI, VIII, and X overlapped the adjacent areas. A group of network stars (VJ~J13-15) in this area was chosen, and coordinates were determined at Lowell. Each strip contained between 5 and 7 of these network stars. The star network of areas II and III, both of which contained the occultation star, were expanded to provide better fitting. These areas contained 60 - 70 stars each.

Data Reduced

By 7 June 1988, the MIT team had reduced half of the data received: 2000 Mb. The reduction emphasized areas II and III because they had the greatest star density and because they both contained the occultation star. Pluto entered area III on 17 May, and entered area II on 25 May. Separate predictions were therefore possible from each area. The final area III prediction had 59 Pluto observations and 60 strips, and the final area II prediction had 15 Pluto observations and 36 strips.

Manpower

To reduce the 2000 Mb of data in six weeks required 2 people at 40 - 60 hours per week, and 2 people at 10 - 20 hours per week. There were 8 people working on data reduction at various times, both at MIT and MKO.

Compute Time

1. extracting stars from strips for fitting (MKO): 30 minutes/strip on GC(generic computer)
2. fitting for image centers (MIT-VAX-750):
 - a. areas II and III: 4 hours/strip
 - b. all other areas: 0.5 hours/strip
3. registering for predictions (MIT-VAX-750): 0.5 hours/area

5. KAO Observing

Test Flight, 7 June 1988

During the KAO test flight on 7 June, we took high-speed series images of Pluto in preparation for the occultation, as well as astrometric frames to update the prediction. Thirty-two frames were taken (1 minute exposures), centered on Pluto and the occultation star. These frames had about 15 network stars per frame.

6. KAO Data Analysis

Image Fitting and Registration

One person worked on processing the astrometric frames through the data reduction pipeline to get star centers. The network stars were extracted from the strips using the GC (generic computer) which was set up in the front of the KAO aircraft. These extracted data boxes were then transferred via ethernet to the DEC VS2000 computer installed in the back of the KAO. The pipeline software had been installed on this VS2000, to allow it to do the image fitting and registrations. Two people started extracting the network stars from strips using the GC, after the KAO returned to Hawaii, at about 2 a.m., local time. Another person worked the rest of the pipeline on the VS2000, and finished the reduction by 10 a.m. the next morning. Each frame required 15 minutes of processing time. Two people then worked on the registration of the strips to produce a prediction. This work required approximately 3 hours.

7. Conclusion

The occultation of a star by Pluto was observed successfully. The discovery of an atmosphere on Pluto being a fundamental planetary science discovery.

Issues investigated

Can a telescience environment be defined that is useful in a full blown active and intense research environment? If useful, what are the principal parts of this environment?

Experiment Hypothesis

We hypothesized that a net increase in the group's scientific productivity could be obtained by the incorporation and use of enhanced connectivity, standardization, computational power, computational flexibility, and data storage methods.

Method of investigation

1. Undertake a program to convince our group of the timeliness and applicability of the telescience concepts and methods to our environment.
2. Enhance our physical facilities with two MicroVAX II/GPX computers added to our environment through PROJECT ATHENA funding. This gives a firmer physical base of operation.
3. Identify the science experiment that would be the vehicle of this investigation.
4. Set up the optical disc archival program.
5. Plan details of the science experiment to capitalize on the new capabilities afforded by telescience:
 - a. software standardization to be able to move programs easily from one computer to another; and

- b. optimization of ethernet use for transfer of large files.
6. Coordinate this work with our collaborators at Lowell Observatory in Flagstaff Arizona.

Experiment result

1. We have successfully used many telescience concepts in our testbed.
2. Within our environment, we have standardized machines and software to make the use of additional machines on our network very fast and efficient:
 - a. By standardizing on the UNIX, 'C', environment and as it worked out, DEC machines, these setups and use of additional machines on a very short time scale for this type of operation became feasible.
 - i. The software was designed with archiving in mind (see below). The use of FITS headers and scripts allowed a pipeline approach to the data analysis and at the same time, the capability of reproducing the data reduction if necessary.
 - ii. The experiment used a distributed computational environment efficiently throughout most of the science experiment. Rapid transfer of image files to another workstation for further processing was a working reality during this experiment.
 - iii. Archiving on optical disc has been performed efficiently and in a straightforward manner. The discs are, in fact, stored in standard hanging files along with paperwork for the project. We believe this to be one of the most important aspects of our work, as it provides a method to remove the data, correspondence, and programs from the online (magnetic disc) storage, and thus ready the space for new work.
 - b. We have demonstrated that an upgrade in science capability can be obtained by incorporating telescience technology, methods and connectivity into our environment
 - c. We have demonstrated an effective interface of our local (group) environment with a wider network, the MIT ATHENA project of many distributed workstations.
 - d. Finally, we have done a remote observation (at Mauna Kea, Hawaii) and with the facilities at Cambridge, Massachusetts have been able to participate in that observing almost as if it was a local observatory. (We still need a better form of data transfer than tapes by commercial air. Note: this corresponds to a bit rate of 20 kbits sec⁻¹ averaged over 24 hours.)

4.9 MIT, LIFE SCIENCES

During the summer months, we began performing remote experiments utilizing voice, video and data. We performed 19 experiments investigating the trade-offs between frame rate, resolution, and gray scale on our ability to get useful information from the video during different phases of the experiment, and the effects of different voice protocols on the progress of the experiment.

The data link from Contel was successfully installed at the end of July, after a three month delay, and final programming efforts were completed at MIT and KSC. We extended the time period of our test sessions through the end of August to take advantage of this data capability.

The proximity of the Shuttle launch wreaked some havoc with the scheduling of NASASELECT (cancellation of video up to a few hours ahead of scheduled experiments), but we did our best to work around this. The relatively large number of people involved (10) and the requirement of off-hours operation (4-8 PM), plus our hope of using the same PI teams, made the dynamic scheduling more difficult.

We were forced by lack of funds and the commitment of resources (Project Athena) to other projects to stop our experiments at the end of August. We are now analyzing the data we obtained and working on preparing material for the final report.

4.10 PURDUE UNIVERSITY

The work during this past year to create a "telescience" database out of our Field Research database came to fruition during this last quarter. Researchers located at non-Purdue sites were able to log into the database, find the data that was needed, and transfer the data to their system via a remote login session. More details about the activities follow.

4.10.1 Equipment and Software

The equipment for the project, a SUN 3/60C, a Macintosh II, and a LoDown WORM (optical disk drive), was moved to a new building during the month of June. The down time was minimal.

4.10.2 Spectrometer/Multiband Radiometer Database

Information about Purdue's Spectrometer/Multiband Radiometer Field Research Database have been organized into two different versions:

1. HyperCard summary; and
2. Oracle summary and experiment data.

Oracle Version

The Oracle version of Purdue's Spectrometer/Multiband Radiometer Field Research Database came online during June. The database is implemented on a SUN 3/60. The user interface went through several iterations during July and August, thanks to input from the University of California, Santa Barbara (Jeff Star and Ken McGuire) and Kansas State University (Jim Hwang).

The database itself consists of about 200,000 high resolution spectra of soils and vegetation, and it contains, together with the spectrometer/multiband radiometer data, agronomic data, and meteorological data collected for around 200 experiments, from 1972 through the present. The Oracle database management system allows the user to view summaries of the experiments by experiment, instrument, location, and/or scene type. If the researcher finds the data from a specific experiment and instrument to be of interest, the researcher can transfer the data to his/her computer system via FTP or Kermit, along with the descriptive information about the experiment, instrument, and file formats.

The data for 80 of the experiments are on the disk, ready for online review and/or transfer. Data for another 20 experiments will be put on the disk during September as disk space allows. The experiment data loaded onto the disk are, hopefully, the most desired data. A mechanism is set up to notify us if data from an experiment is not on the disk desired. We can put that data on for a temporary period of time until the researcher has transferred the data.

Researchers from Kansas State University accessed the database during August to transfer some modeling data sets to their SUN system using FTP across the Internet.

The database can be accessed using the account: `guest@newton.ecn.purdue.edu` with a password of 'purdue', via either the Internet or dialup. To access the database via dialup, however, researchers need to contact Larry Biehl (`biehl@ee.ecn.purdue.edu`) to make prior arrangements. The phone in the room has to be shared.

HyperCard Version

Another stack of information was added to the HyperCard version of Purdue's Spectrometer/Multiband Radiometer Field Research Database Summary. The new file includes the detector descriptions for the instruments used for each of the field experiments. Some copies of the HyperCard version of the summaries were distributed to other researchers.

4.10.3 Field Research Database on Optical Disk

The field research database is presently stored on 40 magnetic tapes that are stored in the basement of Purdue's Library. The tapes have to be checked out for researchers to use them. We have obtained a LoDown 800 megabyte WORM (optical disk) drive for a Macintosh II to copy the tapes to. The data on the tapes are stored in IBM integer, EBCDIC character and floating point formats. These tapes are being copied to an optical disk for archival purposes. Also, an ASCII version of the data is also being created,

organized by experiment and instrument that researchers will usually use. (The data on the SUN 3/60C system accessible by Oracle are the ASCII versions.) ASCII versions of the data will also be archived on optical disk. This task will make the field research data much more accessible.

Eleven of the tapes were copied to optical disk during August.

4.10.4 University of Wisconsin's McIDAS System

We tested the revised version of the McIDAS software during August. The software worked well. Dr. Ralph Dedecker recently notified us that their McIDAS system can be accessed via FTP. We were able to access their system via FTP and transfer some McIDAS files to our system. We did not lose the connection, once established, and the transfer went smoothly and quickly. We are presently working with the University of Wisconsin to access AVHRR data. They recently developed the capability to handle the AVHRR data.

4.10.5 University of Colorado's OASIS

We have had long delays with this task because of the lack of a system on which to implement the OASIS software. We are working with another group at Purdue University, which obtained a VAXstation 2000 in April. However, the system lacked the proper networking software. The software problem was finally resolved in early August. We have since been in touch with Alain Jouchoux of the University of Colorado. They have agreed to send us a copy to install and use to try out testing remote control of the Solar Mesosphere Explorer satellite.

4.10.6 Recent Experience with the Internet

Researchers from Kansas State University were able to access our database via the Internet with little problem. They transferred 6 megabytes of data, averaging about 1 megabyte transferred every 45 minutes during the evening hours, before midnight. The response time during the interactive part of the transfer was somewhat slow but workable. There was one time during a transfer in the middle of the day, that a data transfer session was aborted before completion. The main limitation of the Internet appears to be throughput. But will we ever be satisfied?!

4.10.7 Other Notes

Larry Biehl has been asked by Dr. Vern Vanderbilt of NASA/Ames to give a seminar to his group on the Oracle and Hypercard versions of Purdue's field research database, during the TTPP meetings in October.

4.11 RENSSELAER POLYTECHNIC INSTITUTE

4.11.1 Experiment Title

The feasibility of remote operation of spaceborne Microgravity Materials Science experiments.

4.11.1.1 Experiment Description & Summary

To determine, experimentally, the applicability of the Telescience (remote operation) concept to the area of Microgravity Materials Science, RPI, in cooperation with the Lewis Research Center (LeRC) Microgravity Materials Science Laboratory (MMSL), planned to establish a testbed to remotely operate experiments in the areas of:

1. Glass Science,
2. Crystal Growth from the Vapor, and
3. Crystal Growth from the Liquid.

Drs. Robert Doremus, Heribert Wiedemeier, and Martin Glicksman, respectively, having previous experience in these fields, assisted in the program by advising on the type of experiments to be run, and on the data and control requirements to make the experiments viable. Existing equipment at MMSL was to be utilized, after suitable modification; communication links were to be established; and a remote control center was to be created at RPI.

The program was planned in three phases (one year each). The first phase was to establish the required facilities, the second to operate the experiments to determine the limitations of the facilities, and the third to upgrade the facilities and determine the facility requirements to allow the envisioned science return enhancement made possible by remote operation. These requirements would then be made available to Code S (Space Station) for possible use in the vehicle design.

4.11.2 Issue Investigated

The determination of the level of facilities required to provide a significant science return enhancement by remote experiment control. And, if possible, to determine a relationship between facilities level and science enhancement to allow a cost-effective determination of future equipment/facilities.

4.11.3 Experiment Hypothesis

In the opinion of the scientists participating in this program, based on previous experience, the scientific return of spaceborne experiments could be significantly enhanced if the Principle

Investigator (PI) could have a "hands-on" relationship with his equipment while it is operating. In this mode, he could make real-time adjustments to the experiment, based on real-time results. This program is intended to determine the feasibility of this approach and the complexity of the required facilities.

4.11.4 Method of Investigation as Planned

To establish a testbed to perform this experiment required three facilities:

1. space type experiment hardware;
2. suitable communication links; and
3. a simulated remote control site.

Experiment Hardware

The MMSL at LeRC was established to provide a facility containing a collection of spaceborne type materials science equipment which could be made available to potential P.I.s, for use in new experiments to determine the feasibility of future flight hardware. In this manner, new ideas could be justified for flight at a relatively low expenditure of time and money. These new ideas could then be considered for flight opportunities with a relatively high degree of success possibility.

In preliminary discussions with MMSL personnel, it was determined that generic type equipment existed in all of the science areas contemplated by RPI. Some modifications would have to be made to this equipment to allow remote control, and a communications facility would have to be established. Since the equipment existed, however, the cost of establishing this facility could be kept to a minimum. MMSL agreed to cooperate with RPI and was separately funded to complete this effort, as well as provide support during the actual experimental phase.

Communications Links

Three types of information flow are required for this experiment:

1. data flow from the experiment to the control center;
2. command flow from the control center to the experiment; and
3. video information from the experiment to the control center.

In the original concept for this program, it was anticipated that the first two types could be handled by existing data networks (BITNET, ARPANET, etc.) It was determined that the anticipated transmission delays would not be a problem. The video information flow was planned for the use of commercial communications satellites for high frame rates and land lines for slow scan transmissions. A major goal of this program was to determine the minimum amount of video data required in each of the three areas of science to be investigated.

Remote Control Center

RPI assumed the responsibility for establishing this facility. A small laboratory in the Materials Research Center at RPI was refurbished for this purpose. A Sun 3/60 computer was acquired for use as a control console and suitable video equipment procured for the display of the video information. It was anticipated that existing, user friendly, software programs (OASIS from the University of Colorado, for example) could be modified for our use. In addition, RIACS, as its contribution to the program, was to provide software support and additional programs as required. The requirements for software development at RPI would, therefore, be minimal. One of RPI's major responsibilities was to assure the compatibility of the MMSL software and the RPI software.

4.11.5 Method of Investigation

During the first four months of this program, many major problems were encountered which required several changes in program direction and caused significant delays. Among these were:

1. delays in acquiring the Sun 3/60;
2. problems with existing data networks, which precluded their use for this program;
3. incompatibility of existing software programs with the Sun UNIX operating system;
4. lack of video transmission capability for commercial satellites at LeRC;
5. equipment procurement delays at LeRC; and
6. requirements for software development at RPI beyond those anticipated, as well as the development delays caused by the Sun delivery delays.

Because of these problems, a revised plan of action was prepared to maximize the first phase results. This plan included:

1. The establishment, at RPI, of a complete end-to-end system. This required the acquisition of video equipment, computer software, and experimental equipment not originally anticipated. This change was made to facilitate the complete debugging of the system at RPI, to minimize the use of expensive land line charges, and to reduce the cooperative support needed at LeRC.
2. The procurement, at RPI, of a satellite receiving station to interface with the NASA teleconferencing system, which was to be used for the video transmission.
3. The development, at LeRC, of a computer controlled video frame acquisition and data compression system to allow frame transmission on dialup telephone lines at 9600 baud rates. A study of data compression, at RPI, indicated that compression of both resolution and brightness was possible without serious loss of useful data.
4. Reduction in the complexity of the experimental equipment at LeRC to be used in this first phase of the program.

5. The establishment of a communications protocol and software to allow the interfacing of two operating systems: UNIX at RPI and MS/DOS at LeRC.

4.11.6 Experimental Results

To restate the goals of first phase:

1. Establish a telescience (remote control) testbed specifically aimed at Materials Science requirements.
2. Obtain preliminary results from the operation of typical materials science experiments.

As of this writing (September 1988), one month remains on the original contract and the status of the program can be stated as follows:

Percentage of Completion

1. Establishment of a remote control center at RPI - 98%
2. Establishment of an experiment center at LeRC - 60%
3. Establishment of a communications protocol - 90%
4. Establishment of a communication link:
 - a. Command/Data - 90%
 - b. Video - 80%
5. Establishment of a remote control - 80% experiment @ RPI
6. Operation of a full-up system -50%
7. Performance of analytical studies - 90%

The following results have been obtained to date:

1. Existing communications capabilities, both land based and space based, are not adequate for real time experiment control.
2. New and more extensive data compression schemes are needed to support this concept.
3. The development of new, higher resolution, video equipment is required for many materials science experiments.
4. The concept of telescience for materials science should be separated into two specific areas. The first, remote control, requires moderate levels of communications but also has unique problems with controls on experiment operation to assure spacecraft safety, crew safety, experiment interaction, and resource allocation. The second, data acquisition and transfer, requires very high levels of communication capabilities and data acquisition capabilities. Both are important to telescience but the concept does not require both to be viable.

5. The concept of telepresence has been accepted by many of the experimenters in the field to enhance the scientific return. Previous attempts to enhance scientific returns by using highly automated equipment and/or hands-on crew assistance has been moderately successful but experience has shown that the expertise of the investigator cannot be transferred to either a machine or crewman under the practical limitations imposed by the space program. Often, rudimentary data and command flow capabilities can make the difference between an experiment's success or failure.

6. Establishment of adequate communications along with the appropriate equipment modifications are not incompatible with the Space Station guidelines.

7. Because of the level of complexity involved it is obvious that there are some planned materials science experiments that are not compatible with the telepresence concept. The majority, however, could have their science return increased significantly and their chance of success enhanced by the use of telepresence concepts.

8. Video requirements can be reduced to less than a 256 x 256 pixel resolution and less than 16 shades of gray and less than the standard 30 frame per second rate for many remote control operations without degradation of usefulness.

4.11.7 Microgravity Materials Science Summary

The University of Arizona has made good progress with development of a remote fluid handling testbed in support of the Microgravity and Life Sciences. Design of the software for the man/machine, machine/machine, and machine/instrument interfaces has been completed. Operation of a pH meter and electrophoresis apparatus, using a robot and local control computer, has been successfully demonstrated. A video tape of this demonstration will be made available to interested parties.

Implementation of the software for true teleoperation is nearly complete. As discussed in the University of Arizona section of this report, the goal of developing generic, modular software suitable for many different applications has been attained. A list of other important results is also contained in that section.

The phase one testbed will be completed and demonstrations scheduled by early October. Suggestions for future work include: compatible robot/instrument design, determination of type and amount of video feedback required, and provision for multiple remote experimenters and observers to interact with multiple experiments.

At Rensselaer Polytechnic Institute, effort continued toward the completion of the Remote Control Site. Software for the control program is nearing completion. The first experiment to be implemented was run under local control and a video tape made of the progress. For this experiment, time lapse videography is necessary and, therefore, a VTR compatible with our camera was ordered. Since experiment run time is approximately 30 hours, time compression of 60 to one is needed. Operation of this experiment at LeRC, with telephone line video, is still anticipated for late October.

4.12 SMITHSONIAN ASTROPHYSICAL OBSERVATORY

During the past quarter, we concentrated our efforts in two areas of the Testbedding activities: preparing for the first run of our 10 micron camera at Mt. Hopkins and monitoring of the network between the sites in Cambridge and Tucson.

As we stated in the last quarterly report, there have been difficulties in the development of the 10 micron camera. The first run with the camera was postponed from June until late September. We are still holding to that schedule. In addition to the hardware development for the camera, SAO has been devoting a substantial amount of effort to development of a Real-Time Instrument Control System for operation of the camera. This is a MicroVAX-GPX/Ultrix based software system for both control of the camera as well as data acquisition. The software has been developed entirely at the site in Cambridge for use at the various remote telescope sites. The TTPP has enabled us to link the actual instrument control computer to the WAN. We have now transferred and installed the entire Real-Time system from Cambridge to Mt. Hopkins over the network. Approximately 2.5 Mbytes of files were transferred and the system was rebuilt remotely. This took several hours and the link was lost twice during this time period. Actual transfer rates were about 1.4 kbytes/second. One problem occurred when the kernel was rebuilt: the microwave communications board was not configured properly and communication was lost. This required an onsite fix. However, future changes should be less difficult and less time consuming. Thus, we have accomplished one of our major objectives: demonstration of the use of networks for remote software development and installation. Future objectives we hope to meet in the next month are: to provide remote software debugging and enhancements based on user experience, and near realtime image transfer between the instrument computer and the image processing site.

During the last quarter, we continued to monitor the network performance between Cambridge and Tucson by running the "ping" program. First of all, the "ping" program we have been running consists of sending out 10 packets of 512 bytes of data every half hour, that is 332 packets per week. We record the number of packets that returned and the round trip time for the packet, similar to what one might expect for the response to a key stroke in Telnet or for a instrument command. We now have data covering seven months and have seen a significant improvement in both the success rate and in the ping times, beginning in the week of 12 June and continuing through August. The ping data can be divided naturally into three time periods of: the first ten weeks of the year for which we have ping data (24 January to 28 March); the second ten weeks (29 March to 11 June); and the third group of eleven weeks (12 June to 20 August). The mean number of times that 100% of the packets pinged was 63% in the first period, 39% in the second, and 72% in the third. The packet loss in the second period was dominated by three consecutive weeks when the network was almost dead, with completions of only 2%, 3% and 7% for those weeks. Although the packet completion rate was not substantially better than the earlier part of the year, the ping times have improved dramatically. The mean time for 10% of the packets to make the round trip (analogous to a key stroke echo) was 860 msec, 780 msec and 615 msec for the three time

periods, with the mean for the last three weeks being an amazing 440 msec. The most significant improvement was for the mean time for 90% of the packets to ping. The values were 2.6 sec, 2.9 sec and 1.03 sec. Thus, there appears to have been a substantial improvement in performance since 12 June.

Over the past year, we have also watched the transfer rate between our two sites over the network. As one might expect, a single user cannot, in general, realize the full network bandwidth. In tests involving files of about 1/2 Megabyte, the rates are typically on the order of 1.5 kbytes/sec = 12 kbits/sec and the rates have been about the same throughout the year. This is comparable to what one can expect from the best dialup modems. These tests have been run using the Internet, with no attempt to influence the routing. We have been awaiting (for nearly the entire period of the TTPP) the installation of the NSI 56k link to the University of Arizona, since SAO already has a 56k tail circuit. The word we have now is that this will not be in place until at least mid-October 1988.

Good network performance is vital to Telescience. We will continue to monitor the network performance, especially since the NSFnet support has come under operation of Merit.

4.13 UNIVERSITY OF WISCONSIN

Interfacing to NSFnet and McIDAS

An interface to the NSFnet has been installed for the establishment of an automated bridge between the NSFnet and the University of Wisconsin McIDAS system, enabling remote users to acquire McIDAS products. The bridge is currently being utilized and implementation approaches are being evaluated in light of the experiences in using the system, thus far. From the NSFnet side of the bridge, users of the McIDAS products access data via an FTP server that executes on the IBM PC bridge. The bridge accesses the McIDAS product database at scheduled intervals and provides several files of the latest McIDAS products to the FTP server. A remote user simply establishes an FTP connection to the bridge via the TCP/IP network and can then execute any normal FTP command, including disk directory queries and file transfer requests. The McIDAS products initially available include: sectorized GOES images over the US, in a format compatible with the IBM PC EGA adaptor. Additional products supplied will be dependent upon user needs. To date, access of the data by remote users has been very limited. Notification of the FTP server availability has just begun. It is anticipated that remote user access will increase. Users can access the bridge via *McIDAS.SSEC.WISC.EDU* or *128.104.83.40* for users whose name servers do not know the IP address. In addition to the FTP server function, the bridge hardware and NSFnet connection have been used for email transactions, remote terminal login to the UCSB browse system, and FTP access to the Purdue system.

Recent changes in the IP network configuration at the University of Wisconsin indicate improvements in reliability and response times have occurred with some sessions but degradation has occurred in others, as compared to the previous University of Wisconsin

network configuration. During the remaining Telescience contract period, we hope to have enough user feedback to evaluate the utility of the bridge and to supply additional products, as requested.

Access to McIDAS Products by Earth Science Users

IBM PC based software for the access and display of McIDAS GOES images and access to other McIDAS products has been distributed to Purdue, the University of California, Santa Barbara, and the University of Colorado in the past few months. The software allows access to McIDAS via asynchronous dialup. Software bugs were found by Purdue. Corrections have been made and distributed to the three university groups. Feedback from Purdue indicates that the software is now operating correctly, however, access of these data has been limited to a few test trials. Additional software is being developed at University of Wisconsin to allow users of the FTP server facility to decode the GOES images and merge these images with the display facility used for products acquired via asynchronous dialup.

Appendix 1 LOA

TTPP LOA

AAS	American Astronomical Society
AGC	Automatic Gain Control
AIPS	Astronomical Image Processing System
ALOT	Arc Laser Optical Technology
Andrew	Multimedia mail system with capabilities similar to Diamond; basis of Carnegie-Mellon EXPRES system
ARC	Ames Research Center
ARPA	Defense Department Advanced Research Projects Agency
ARPANET	Wide area data network supported by ARPA
AT	Astrometric Telescope
ATF	Astrometric Telescope Facility
AVHRR	Advanced Very High Resolution Radiometer; on the nimbus series of satellites. Run through NOAA
B&W	Black and White Display
BARRNET	Bay Area Regional Research Network
BAUD	not an acronym; a unit of signaling speed; refers to the number of times the state or condition of the line changes per second
BCE	Bench Checkout Equipment
CAS	Canadian Astronomical Society
CCD	Charge Couple Device; a technology for converting images into electrical signals
CCSDS	Consultative Committee for Space Data Systems
CDP	Command, Data, and Power interface unit; part of the EUVE instrument
CLIPS	A programming language for expert systems. A copy of the source program written in C was sent to U of Michigan from Johnson Space Center, Mission Support Directorate, Mission Planning and Analysis Division. It is in the public domain and has been installed on several systems, IBM PC -Microsoft C, Sun, Dec. The only restrictions to its use is that you can not sell it.
CODEC	Co-der/Dec-oder
CSDF	Commercial Space Development Facility
CUI	Common User Interface
DEC	Digital Equipment Company
DMIL	Direct Manipulation Interface Language
DSP	Digital Signal Processing
EPS	Experiment Payload Specialist
EUV	Extreme Ultraviolet
EUVE	Extreme UltraViolet Explorer
EXPRES	Experimental Research in Electronic Submission
FUV	Far Ultraviolet
FRICC	Federal Research Internet Coordination Committee
GETSCI	*
GOES	Geostationary Operational Environmental Satellite

GPX	Graphics Processor Workstation for MicroVAX II
GSFC	Goddard Space Flight Center
HCIG	Human-Computer Interface Guide
HIPS	Human Information Processing Laboratory's Image Processing
HRPT	High Resolution Picture Transmission
HUP	Human Use Protocols
IAPPP	*
IBM	International Business Machines
IDL	Interactive Data Language
IGBF	*
IGBP	International Geosphere Biosphere Program
IL	Intermediate Language
IMS	Instrument Management Services
Ingres	Database
IOMS	Instrument OMS
IPAC	Infrared Processing and Analysis Center at Caltech
IRAF	Image Reduction and Analysis Facility
IRAS	Infrared Astronomy Satellite
JPL	Jet Propulsion Laboratory
JSC	Johnson Space Center
JVNCC	John Van Neuman Computing Center
KSC	Kennedy Space Center
KSCBCF	*
Kermit	File Transfer Program
Kiwi	not an acronym; a "flightless bird" consisting of prototype EUVE electronics
LAN	Local Area Network
LASP	Laboratory for Atmospheric and Space Physics
LCC	Local Controlling Computer
LIB\$TPARSE	VAX/VMS library routine that provides a table driven parser. Used for the initial version of the CSTOL parser for OASIS
LSTB	Life Sciences Testbed
Magic/L	interactive programming language developed by Loki Engineering, Inc.
McIDAS	Man Computer Interactive Data Access System
MIT	Massachusetts Institute of Technology
MMSL	Microgravity Materials Science Laboratory
MMT	Multiple Mirror Telescope on Mt. Hopkins, AZ
MSFC	Marshall Space Flight Center
NASA	National Aeronautics & Space Administration
NASA SELECT	NASA run TV channel which carries NASA related events
NASCOM	NASA Communications- mission critical
NICOLAS	the inter-network gateway at Goddard Space Flight Center
NOAA	National Oceanic and Atmospheric Administration
NOAA-G	Name of the NOAA polar orbiting satellites
NRAO	National Radio Astronomy Observatory
NSE	Network Software Environment

NSF	National Science Foundation
NSFnet	Data network supported by NSF
NSI	NASA Science Internet
NSN	NASA Science Network; TCP/IP part of NSI
NTSC	Standard video signal format
OASIS	Operations and Science Instrument Support
OMS	Space Station Operation Management System
OMS/PMS	Operations Management/Platform Management
OMSS	Operation Management Services
OSSA	Office of Space Applications & Applications
PI	Principal Investigator
PSI	Payload Systems, Inc.
Project	MIT student network
RA	Research Assistant
RCC	Remote Commanding Computer
RFH	Remote Fluid Handling
RGB	Red, Green, Blue Video Display
RIACS	Research Institute for Advanced Computer Science
ROM	Read Only Memory
RS-232	Standard for serial data transmission
SAIS	Science and Applications Information Systems
SAO	Smithsonian Astrophysical Observatory
SCS	Software Control System
SIMBAD	Primarily a bibliographical cross reference database of 700,000 stellar and 100,000 non-stellar objects, which can be logged into remotely from around the world; set of identifications, measurements and bibliography for astronomical data
SME	Solar Mesosphere Explorer satellite
SOP	Science Operations Subgroup
SPAN	Space Physics Analysis Network
SPIE	Society of Photo-Instrumentation Engineers
SS	Space Station
SSE	Software Support Environment
SSIS	Space Station Information Systems
SSL	Space Sciences Laboratory at UC, Berkeley
SSP	Space Station Program
STScI	Space Telescope Science Institute
TAE	Transportable Applications Executive
TATS	Thaw Atmospheric Telescope Simulation
TCP/IP	Transmission Control Protocol/Internet Protocol
TDRSS	Tracking and Data Relay Satellite System
TeleWEn	Telescience Workstation Environment
Terracom	a company name
TFSUSS	Task Force on Scientific Uses of Space Station
TIF	Telescope Interface Unit
TIGS	Testbed at LASP

TMIS	Technical Management Information System
TTPP	Telescience Testbed Pilot Program
UC	University of California
UCB	University of California, Berkeley
UCSB	University of California, Santa Barbara
UIL	User Interface Language
Unify	Database
UofA	University of Arizona
USE	User Support Environment
USRA	Universities Space Research Association
UW	University of Wisconsin
VISTA	another image processing system
WAN	Wide Area Network

Appendix 2 TTPP BIBLIOGRAPHY

BIBLIOGRAPHY

"The '1990s' Control Centre - The New OCC," *Journal of British Interplanetary Society*, vol. 40, no. 6, pp. 251-255, Space Technology European Space Operations Centre, June 1987.

This article describes the concept of a workstation and the European Space Operations Centre in Darmstadt.

"EXPRES Project Completes First Year," *CSNET News*, no. 16, pp. 1-3, CSNET CIC, Winter 1988.

This article briefly covers the first year of the Experimental Research in Electronic Submission (EXPRES) project and information on UM EXPRES.

Bennett, Elliot, *HOLOP: A Case Study of Advanced User-interface Design for Interactive Control of Space Experiments*, DFVLR IB 333 - 88/5, 24 pp., DFVLR, Institute for Space Simulation, Cologne, West Germany, June 1988.

This paper details advanced concepts in user-interface design implemented in the computer program HOLOP Ops. HOLOP Ops was designed to provide a simple, easy, and fast user-interface for remote, interactive control the HOLOP facility aboard the D2 Spacelab mission. Such a user-interface is achieved by implementing full graphics capabilities (including pictures, icons, graphs, and mouse/cursor control) as well as full text displays and control in a transparent, integrated environment for experiment observation and control. The advantages and capabilities of this program's user-interface are described and analyzed for their ability to enhance space based science productivity in the Space Station era.

Bienz, Richard A. and Larry C. Schooley, "A Survey of Computer Networks," Technical Report TSL-001/87, 85 pp., Electrical and Computer Engineering Department, University of Arizona, Tucson, AZ, February 1987.

A summary of existing wide-area computer networks and their attributes and evaluation of their possible use in the University of Arizona TTPP testbeds.

Cellier, Francois, *Teleoperation of the Thaw Telescope at the Allegheny Observatory: A Case Study*, Technical Report TSL-004/87, 56 pp., Electrical and Computer Engineering Department, University of Arizona, Tucson, AZ, December 1987.

Primary purpose of this document is to define the intermediary language to be used for computer-to-computer communication between the local controlling computer at Allegheny Observatory and the remote commanding computer which will be located at the University of Arizona. Overview of plans leading to teleoperation the Thaw Telescope at Allegheny Observatory is also discussed as well as control loops, sensors, safeguard error messages.

Chakrabarti, Supriya, Carl A. Dobson, and J. G. Jernigan, "Telescience at the U. C. Berkeley Space Science Laboratory," *Bulletin of the American Astronomical Society*, vol. 19, p. 744, Space Sciences Laboratory, U.C. Berkeley, June 1987.

The University of California, Berkeley is a member of a University consortium developing methodologies for remote design, development and operation of space instrumentations, collectively termed *telescience*. We will discuss our efforts in extending an existing local software control system to allow the development and sharing of software between remote sites. We are developing a methodology for the remote operation of instrumentations utilizing networks such as the ARPANET. These techniques have already been demonstrated over a local Ethernet. These two areas of investigations address the teledesign and teleoperation components of telescience.

Chakrabarti, Supriya, Carl A. Dobson, George C. Kaplan, Herman L. Marshall, Michael Lampton, Roger F. Malina, Stuart Bowyer, and Jeff Star, "Astronomical Data Analysis from Remote Sites," *Astronomy from Large Databases, Scientific Objectives and Methodological Approaches, EOS Conference and Workshop Proceedings*, vol. 28, p. 295, Space Sciences Laboratory, UC Berkeley & Remote Sensing Research Unit, UC Santa Barbara (J. Star), Berkeley, CA, January 1988. Also available in Space Astrophysics Group Contribution Number 329.

Given the progress in the communication technology, it is expected that during the space station era the mode of instrument operation and data analysis will be dramatically different. A consortium of several universities and NASA centers are evaluating various aspects of design and operation of scientific instruments and data analysis over various computer networks from a remote site. Such a scheme has officially been termed *telescience*. We will report on the development of methodologies for teledesign, teleoperation and teleanalysis and the verification of these concepts using the Extreme Ultraviolet Explorer (EUVE), a satellite payload scheduled for launch in 1991. The EUVE telescopes will be operated remotely from the EUVE Science Operation Center (SOC) located at the University of California, Berkeley. Guest observers will remotely access the EUVE spectrometer data base located at the SOC. Distributed data processing is an integral part telescience. We will describe our experience with the Browse system, currently being developed at the University of California at Santa Barbara through a grant from NASA for remote sensing applications. We will discuss the suitability for its adoption for astronomy applications. Browse allows the examination of a subset of the data to determine if the data set merits further investigation. The examination can be as simple as looking for a specific data element based on its location, date of observation, quality indicator, spectral coverage etc. It also allows the viewing of data in various modes depending upon the available resources at the user's end (e.g., graphics terminal vs. dumb terminal), level of data compression applied, required display format etc. and its transmission over a network to a local graphics display station.

Gallagher, Maria L., *Telescience Testbed Kickoff Meeting Minutes*, RIACS TR 87.25, 26 pp., RIACS/USRA, Moffett Field, CA, September 1987.

The kickoff meeting for the Telescience Testbed Pilot Program was held on July 30-31, 1987 at NASA Ames Research Center. These are the minutes of that meeting.

Gallagher, Maria L., *Telescience Testbed Pilot Program Meeting II Minutes*, RIACS M88.1, RIACS/USRA, Moffett Field, CA, March 1988.

The TTPP II meeting was held on March 7-9, 1988 in Boulder, Colorado. These are the minutes of that meeting.

Gallagher, Maria L. and Barry M. Leiner, *Telescience Testbed Pilot Program First Quarterly Report*, RIACS TR 87.26, 35 pp., RIACS/USRA, Moffett Field, CA, September 1987.

The Telescience Testbed Pilot Program participants are required to issue reports to NASA Headquarters on a quarterly basis. This is the first quarterly report, covering the period April 28, 1987 through August 31, 1987.

Gallagher, Maria L. and Barry M. Leiner, *Telescience Testbed Pilot Program Second Quarterly Report*, RIACS TR 87.31, 57 pp., RIACS/USRA, Moffett Field, CA, December 1987.

The Telescience Testbed Pilot Program is an innovative activity involving fifteen universities in user-oriented rapid-prototyping testbeds to develop the requirements and technologies appropriate to the information system of the Space Station era. The Telescience Testbed Pilot Program is required to issue progress reports to NASA Headquarters on a quarterly basis. This is the second quarterly

report, covering the period September 1, 1987 through November 30, 1987.

Gallagher, Maria L. and Barry M. Leiner, *Telescience Testbed Pilot Program Third Quarterly Report*, RIACS TR 88.8, 77 pp., RIACS/USRA, Moffett Field, CA, March 1988.

The Telescience Testbed Pilot Program is required to issue progress reports to NASA Headquarters on a quarterly basis. This is the third quarterly report, covering the period December 1, 1987 through February 29, 1988.

Gallagher, Maria L. and Barry M. Leiner, *Telescience Testbed Pilot Program Fourth Quarterly Report*, RIACS M88.4, 69 pp., RIACS/USRA, Moffett Field, CA, June 1988.

The Telescience Testbed Pilot Program is required to issue progress reports to NASA Headquarters on a quarterly basis. This is the fourth quarterly report, covering the period March 1, 1988 through August 31, 1988.

Hack, B., *Man to Machine, Machine to Machine and Machine to Instrument Interfaces for Teleoperation of a Fluid Handling Laboratory*, Technical Report TSL-014/88, Electrical and Computer Engineering Department, University of Arizona, Tucson, AZ, June 1988.

The purpose of this thesis is the design and description of the software necessary for teleoperation of a remotely operated fluid handling laboratory. It does not include the implementation of this software. The laboratory for which it is designed is currently being developed at the University of Arizona, and is intended to be a small scale model of the fluid handling laboratory which will be aboard Space Station. The designed software includes a man/machine interface, a machine/machine interface, and a machine/instrument interface. The man/machine interface is graphical in nature, menu driven, and consists of high level commands which are independent of the devices in the laboratory. The machine/machine interface is also device independent. It consists of intermediary commands and maps the commands of the man/machine interface to low level, device dependent, commands and programs of the machine/instrument interface. Although the software is primarily designed for the model fluid handling laboratory, the needs and requirements of the operation of a similar laboratory aboard Space Station have been considered.

Kaplan, George C., "EUVE Contributions to Telescience," *EUVE Technical Bulletin*, no. 4, p. 2, Space Sciences Laboratory, UC Berkeley, Berkeley, CA, September 1987.

Brief Overview of UC Berkeley's telescience experiments for the TTPP.

Koch, David, Terry Herter, John Stauffer, and Erick Young, *Telescience Applied to the Space Infrared Telescope Facility*, 8 pp., Smithsonian Astrophysical Observatory (Koch); Department of Astronomy, Cornell University (Herter); NASA/Ames Research Center (Stauffer); Steward Observatory, University of Arizona (Young), September 1987.

In the future, the approach to the conduct of scientific space missions will be substantially different from the approach that has been used in the past. A more distributed approach will be taken with the scientists conducting operations and analysis, remotely from their home institutions, making major use of standardized software and compatible hardware. Key to this approach have been the rapid adoption of the use of local and wide area networks, the use of standardized software tools and the plethora of powerful workstations. These developments will be applied to the Space Infrared Telescope Facility project in the space station era. A number of telescience testbed activities are being undertaken to develop experience and to determine the applicability of telescience methods.

Leiner, Barry M., *Telescience Testbed Pilot Program*, RIACS TR 87.12, 42 pp., RIACS/USRA, Moffett Field, CA, May 1987.

The Telescience Testbed Pilot Program is an innovative activity to address a number of critical issues in

the conduct of science in the Space Station era. Several scientific experiments using advanced information processing and communications technologies will be carried out and the results evaluated to determine the requirements and their priorities. This will provide quantitative evidence as to the relative importance of different functions in the SSIS and their required performance. Furthermore, it will allow a set of scientific users to gain experience with advanced technologies and their application to science. This report is based on the proposal from USRA to NASA for the establishment of the Telescience Testbed Pilot Program. It describes the motivation for the program, the structure of the effort, and several strawman scientific experiments that constitute the heart of the activity.

Leiner, Barry M. and James R. Weiss, *Telescience Testbedding: An Implementation Approach*, RIACS TR 88.2, 9 pp., RIACS/USRA (Leiner) & NASA/HQS (Weiss), Moffett Field, CA, February 1988.

Telescience is the term used to describe a concept being developed by NASA's Office of Space Science and Applications (OSSA) under the Science and Applications Information System (SAIS) Program. This concept focuses on the development of an ability for all OSSA users to be remotely interactive with all provided information system services for the Space Station era. This concept includes access to services provided by both flight and ground components of the system and emphasizes the accommodation of users from their home institutions. Key to the development of the Telescience capability is an implementation approach called rapid-prototype testbedding. This testbedding is used to validate the concept and test the applicability of emerging technologies and operational methodologies. Testbedding will be used to first determine the feasibility of an idea and the applicability to real science usage. Once a concept is deemed viable, it will be integrated into the operational system for real time support. It is believed that this approach will greatly decrease the expense of implementing the eventual system and will enhance the resultant capabilities of the delivered systems.

Leiner, Barry M., *Telescience Testbed Pilot Program Interim Report*, RIACS TR 88.6, 16 pp., RIACS/USRA, Moffett Field, CA, February 1988.

The Universities Space Research Association (USRA), under sponsorship from the NASA Office of Space Science and Applications, is conducting a Telescience Testbed Pilot Program. Fifteen universities, under subcontract to USRA, are conducting a variety of scientific experiments using advanced technology to determine the requirements and evaluate trade-offs for the information system of the space station era. This report represents an interim set of recommendations based on the experiences of the first six months of the pilot program.

Lichtenberg, Byron K., "Concepts for Crew Experiment Interaction-Future Space Flights: Workstation Design and Requirements," p. 3, Payload Systems, Inc., June 1988. Submitted to the International Conference on Environmental Systems to be held July 1988.

Current space lab and space shuttle workstations are inadequate for the next generation of space experimentation. The capability of the current experiment computers is severely limited, the maximum sample rate that can be acquired and processed for on board display is 10 samples per second, and the displays have a maximum of 750 word storage associated with them. Second, the ability to modify experiment operations real time is non-existent unless it was programmed in approximately 18 months before flight. The appearance of new generations of computers will alleviate these problems, but acceptance by the space engineering community is still limited. This paper will discuss the concepts and requirements for future workstations and capabilities that should be inherent in the next generation of space craft.

Marchant, Will, Carl A. Dobson, Supriya Chakrabarti, and Roger F. Malina, "Telescience -Concepts and Contributions to the Extreme Ultraviolet Explorer Mission," *SPIE Proceedings*, vol. 851, p. 173, Astronomy Dept. & Space Sciences Laboratory, UC Berkeley, Berkeley, CA, November 1987. Also available in Space Astrophysics Group Contribution Number 315.

A goal of the telescience concept is to allow scientists to use remotely located instruments as they would in their laboratory. Another goal is to increase reliability and scientific return of these instruments. In this paper we discuss the role of transparent software tools in development, integration, and postlaunch environments to achieve hands on access to the instrument. The use of transparent tools helps us to reduce the parallel development of capability and to assure that valuable pre-launch experience is not lost in the operations phase. We also discuss the use of simulation as a rapid prototyping technique. Rapid prototyping provides a cost-effective means of using an iterative approach to instrument design. By allowing inexpensive production of testbeds, scientists can quickly tune the instrument to produce the desired scientific data. Using portions of the Extreme Ultraviolet Explorer (EUVE) system, we examine some of the results of preliminary tests in the use of simulation and transparent tools. Additionally, we discuss our efforts to upgrade our software "EUVE electronics" simulator to emulate a full instrument, and give the pros and cons of the simulation facilities we have developed.

Pan, Ya-Dung, *Teleoperation of Mechanical Manipulators Aboard the U.S. Space Station*, Technical Report TSL-002/87, 74 pp., Electrical and Computer Engineering Department, University of Arizona, Tucson, AZ, December 1987.

This study presents a new analytical controller design strategy for the teleoperation of mechanical manipulators aboard the U.S. space station. This controller design strategy emphasizes on the stability of a closed-loop control system involving time delay. Simplified dynamic equations of the Stanford arm are considered as the manipulator model. A local linearizing and decoupling control algorithm is applied to linearize and decouple the dynamic equations. Once the linear form of the manipulator is obtained, a model prediction control loop is constructed and implemented as a digital controller to provide the predictive states information, and a particular model reduction method is applied to yield a reduced-order digital controller. This reduced-order digital controller is a highly self-tuned controller which can control the closed-loop system with time delay by following a specified performance.

Pennisi, Liz, "Computers on Long-Distance Research," *LQP*, p. 8, December 7, 1987.

Magazine article on the use of computers on long-distant research at the University of Arizona.

Raize, Efraim, *Computer Interface for Electrophoresis Apparatus*, Technical Report TSL-011/88, 28 pp., Electrical and Computer Engineering Department, University of Arizona, Tucson, AZ, May 1988. The author is a visiting scholar at the University of Arizona.

This report summarizes the considerations required for an adequate interface, lists the electronics design and shows the drawings and procedures for operation and maintenance of an Electrophoresis machine in an automated laboratory.

Raize, Efraim, *Syringe Driver Assembly for Automatic Fluid Handling Laboratory*, Technical Report TSL-012/88, 17 pp., Electrical and Computer Engineering Department, University of Arizona, Tucson, AZ, May 1988. The author is a visiting scholar at the University of Arizona.

This report describes the design and implementation of a driver assembly for an automated fluid handling laboratory.

Schmerling, Erwin, "The Interaction of Users with Instruments and Databases in Space," *Information Systems Newsletter*, pp. 12-18, NASA Headquarters, January 1988.

This article discusses the Telescience Testbed Pilot Program's objectives, planned contributions and

defines the term Telescience.

Schooley, Larry C. and Francois Cellier, *Telescience Testbed Pilot Program Quarterly Report*, Technical Report TSL-003/87, 16 pp., Electrical and Computer Engineering Department, University of Arizona, Tucson, AZ, December 1987.

First quarterly report for the University of Arizona's Telescience Testbed Pilot Program.

Schooley, Larry C., Richard A. Bienz, and Francois Cellier, *Basic Research in Telescience Testbed Program Final Report: NASA Grant NAGW-1073*, Technical Report TSL-005/88, Electrical and Computer Engineering Department, University of Arizona, Tucson, AZ, January 1988.

Final report for NASA grant NAGW-1073.

Schooley, Larry C., Don G. Schultz, and Francois Cellier, *University of Arizona Presentation, Second Telescience Testbed Pilot Program Meeting*, Technical Report TSL-007/88, 50 pp., Electrical and Computer Engineering, University of Arizona, Tucson, AZ, March 1988.

The set of transparencies presented by the University of Arizona at the second TTPP meeting held in Boulder, CO on March 7-9, 1988.

Schooley, Larry C. and Francois Cellier, *Telescience Testbed Pilot Program Quarterly Report For Winter 1987-88*, Technical Report TSL-008/88, 15 pp., Electrical and Computer Engineering Department, University of Arizona, Tucson, AZ, March 1988.

Second quarterly report for the University of Arizona's Telescience Testbed Pilot Program.

Schooley, Larry C. and Francois Cellier, *Telescience Testbed Pilot Program Quarterly Report for Spring 1988*, Technical Report TSL-013/88, 10 pp., Electrical and Computer Engineering Department, University of Arizona, Tucson, AZ, June 1988.

This is the third quarterly report for the astrometric telescope, remote fluid handling, and technology development projects at the University of Arizona. It does not include the UA involvement in the SIRTf project.

Schultz, Don G. and R. Fardid, *An Automated Remote Fluid Handling System*, Technical Report TSL-015/88 (videotape), Systems and Industrial Engineering Department, University of Arizona, Tucson, AZ, July 1988

Starks, Scott, David Elizandro, Barry M. Leiner, and Michael Wiskerchen, "Computer Networks for Remote Laboratories in Physics and Engineering," 1988 Annual Conference of the American Society for Engineering Education, June 1988, (also available as RIACS TR 88.13, 7 pp., April 1988).

As we embark on a new era in engineering education, we must exploit technological advances which offer opportunities for improving the educational process. One area of technology which offers opportunities for enhancing the manner in which research is conducted and ultimately affects scientific and engineering education is computer networks. As computer hardware has become less expensive, more numerous and more capable, individuals and organizations have developed a keen interest in connecting them together in order to form networks. This in turn has had an impact on the manner in which laboratory research is conducted. This paper addresses a relatively new approach to scientific research, telescience, which is the conduct of scientific operations in locations remote from the site of central experimental activity. A testbed based on the concepts of telescience is being developed to ultimately enable scientific researchers on earth to conduct experiments onboard the Space Station. This system along with background materials are discussed in this paper.