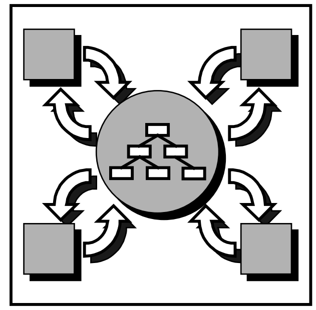


Software Design Description

Steve Wampler, Kim Gillies, Steven Beard, Andrew Johnson, Robert Laing, Christopher Mayer, Peregrine McGehee, Richard McGonegal, Pat Wallace

Gemini Software Design Team

Daresbury Rutherford Appleton Laboratories
Gemini Project
National Optical Astronomy Observatories
Royal Greenwich Observatory
Royal Observatory Edinburgh

8-M Telescopes

GEMINI Controls Group

Draft Document - Not for construction
Last Revision: 3/7/95

*We are continually being faced with fantastic opportunities,
brilliantly disguised as insoluble problems.*

Document Number: SPE-C-G0037
Issue Number: 3
Revision Number: 0
Date: June 6, 1995
Approved By: R.J. McGonegal
Systems: J. Oschmann
Science: M. Mountain
Project: D. Kurz
Proprietary: NO
Procurement Sensitive: NO
Disclosable: YES

Note to Critical Design Review Committee

Charge to the Software Critical Design Review Committee

The CDR Committee is asked to serve in an advisory role to the project and to submit a report to the Gemini Controls Manager. This report will review the Software Design Description contained herein.

We would appreciate the review committee making a recommendation based on a majority opinion. This recommendation should be one of the following:

1. recommended without conditions or reservations
2. recommended with conditions - conditions listed; conditions would be action items that need to be satisfactorily discharged.
3. recommended with reservations - reservations listed; reservations would be potential problems that require substantial additional work
4. not recommended - reasons listed

Materials to be Reviewed

There are 4 *books* supplied as part of this review

1. SDD part 1 - updated from PDR
2. SDD part 2 - updated from PDR
3. Gemini Scenarios - new from PDR
4. Interface Control Documents - new from PDR

Composition of the Software Design Description

The Software Design Description (SDD) is composed of the following logical sections (most chapters discuss changes from PDR):

- "Introduction" - chp.1-2
- "System Requirements" - chp.3
- "Principal Systems" - chp.4-9
- "Details of the Principal Systems" - chp.10-13
- "Details of the TCS Subsystems" - chp.14-21

- "Scenario Walk-Throughs" - chp.22
- "Formal Design Views" - app.A1-A2
- "Interface Description" - app.A3

Review Priorities

The Review Committee is asked to review all of the documents with the following priorities:

1. The interface control documents are new and should be reviewed in their entirety.
2. The scenario walkthroughs SimSch01-05 and SimTel01-09 should be the focus of review of the walkthroughs.
3. The Review Committee is also asked to comment about the coverage of the scenarios - it was not possible to do all 50+ for the CDR. However we felt that those not done currently can be postponed and will be part of the individual work package reviews. The scenarios will also be reviewed at upcoming system reviews.
4. The chapters up to and including details of major subsystems should be reviewed to the extent that they have been changed since the PDR.
5. The chapters on the details of the TCS subsystems - keeping in mind that each of these chapters will be the focus of a work package and will be expanded and reviewed during the course of that work package.
6. Formal design views

WORK PACKAGE PLANS

The Review Committee is also asked to comment on the plans for starting the Observatory Control, Telescope Control and Data Handling Work Packages consecutively following this CDR. Of particular interest is those areas which the Committee feels require more work before starting any or all of these Work Packages.

How to obtain this report?

This report is available from the Gemini Project Office in a variety of forms:

- FTP - This report is available via anonymous FTP from gemini.tuc.noao.edu, in the directory ~ftp/gemini/SDD. See the Read.Me in that directory for file contents and formats.
- Gopher - Alternatively, it is possible to obtain this report through the Gemini gopher server at gemini.tuc.noao.edu.

If none of these forms are suitable, please contact Steve Wampler {swampler@gemini.edu, (602) 321-3422} for alternative arrangements. Each reviewer is being sent a hardcopy version via Federal Express.

What happens after the CDR?

The first tasks will be to discharge action items arising from the July/94 system review.

The objective of the design team, after the CDR is finished and all actions have been closed, is to proceed with dividing the software development into the separate Work Packages. Each WP development will complete the detailed design for their respective portions of the system.

As part of that development each work package team will take over responsibility for the chapters in the SDD. For the principal system work packages (OCS, ICS, TCS, DHS) the WP teams will produce individual stand alone Software Design Descriptions as part of the process of leading up to the WP PDR and CDR.

As we shift from design work being done in Tucson to design work being done in the partner countries the work of the Tucson team will move to more of a managerial, review (yes, we will get to review someone else for a change!!), and change control responsibility.

Is this the last Software Review?

This is the last formal review of the overall software design. The review process after this can be split into 2 different types:

REVIEWS OF GEMINI

There will continue to be twice yearly system reviews of the Gemini Project by an external review committee. At each of these reviews an overview of the software design will be presented and reviewed.

REVIEWS BY GEMINI

All of the work packages will follow a formal System, Preliminary and Critical design review process. The review committee will be made up of Gemini project members and, for selected work packages, members of the SDD CDR committee. For any review Gemini has the option of inviting external reviewers to provide expertise and objectivity.

CONTENTS

FIGURES

TABLES

1 INTRODUCTION

1.1 PURPOSE

Any software system as large and complex as that for the Gemini telescope and instrument software requires a carefully thought out design structure to ensure successful operation, maintenance, and growth. The distributed nature of the Gemini software development further emphasizes the need for careful design. This *Software Design Description* (SDD) provides the fundamental framework for the design and implementation of Gemini control software.

It is intended to provide a reference suitable for use by all software developers, enabling each developer to understand not only the behavior of their code, but also how each part fits into the complete Gemini Control System (GCS). It *does not* provide detailed designs for the implementation of the various modules found in the software. This has been deliberately left for each Work Package Group to develop in conjunction with the Gemini Project Office. Rather, this document focuses on the infrastructure necessary to ensure seamless connection of the various systems that comprise the Gemini Control System.

1.2 ORGANIZATION

The overall design of the Gemini Control System is presented here using several different views. The body of the SDD presents a mixture of *informal* and *formal* presentation styles to help provide a thorough understanding of the philosophy of the design. By providing both styles together, the reader is able to integrate the design philosophy using the most effective method.

This initial presentation is organized into several layers:

- *high level descriptions* providing *user views* of the system, presented in Chapters 4-9.
- *detailed descriptions* concentrating on the *functionality* provided by the system, found in Chapters 10-21, and on the interfaces between the system components, found in Appendices 7-???.
- *operational scenarios* providing *behavioral* ‘snapshots’ of the system, found throughout the document and in Chapter 22 in particular.
- *formal design descriptions,* presented in Appendices 1-3.

1.2.1 High-level Descriptions

The first layer presented provides a user view into the Gemini Control System. Each of the principal systems within the GCS is presented, concentrating on the system behavior as viewed externally.

1.2.2 Detailed Descriptions

The detailed descriptions concentrate on the functionality of the principal systems and the interfaces between these systems, as well as the decomposition of each principal system into its component subsystems.

1.2.3 Operational Scenarios

These scenarios range from simple stare observations through complex spectral sky mapping and multiple object spectroscopy. Also included are scenarios for *house-keeping tasks*, such as daily start-up. By studying the behavior and responses of the control system with these scenarios, the reader can quickly gain a feel for the goals and effectiveness of the control system.

1.2.4 Formal Design Descriptions

For a *precise* and *accurate* understanding sufficient for using this document as a starting point for further development of the design, it is important that the formal descriptions of the project be readily accessible. For this reason, these formal design descriptions are presented as separate appendices.

Formal software design methods present a variety of views of a system, each intended to quickly and accurately convey the design of the system to a particular audience. The views that are presented are:

- a *decomposition description*, showing the overall design in a hierarchy of complexity, as systems are broken into subsystems,

- a *dependency description*, illustrating the connectivity between the various parts of the system,
- an *interface description*, providing details of the nature of the information that passes between the parts of the system, and
- a *detailed design description*, providing complete details of each portion of the design.

1.2.4.1 DECOMPOSITION DESCRIPTION

The decomposition description presents the design partitioned into various design entities that are then hierarchically divided into subordinate components. This view is intended to assist both designers and maintainers and can be used for such activities as determining the entity responsible for performing a given task, and for tracing design requirements into the actual design. It is also useful for project management, as it provides a means to assist in the planning and monitoring of the software development.

1.2.4.2 DEPENDENCY DESCRIPTION

The dependency description provides an overview of system functionality. It identifies the relationships between entities, the dependences among entities, and identifies required resources. This description is useful for assessing the impact of requirement and design changes. It is also useful to maintainers by assisting in the isolation of components that are causing bottlenecks or system failures. Finally, it is useful in developing a *system integration plan* and for *integration testing*.

1.2.4.3 INTERFACE DESCRIPTION

The interface description provides information essential to designers, programmers, and testers using the functionality of the software system. It provides details of external and internal interfaces not provided in the *Software Requirements Specification* and addresses the requirements found in the *Interface Requirements Specification*.

1.2.4.4 DETAILED DESIGN DESCRIPTION

The detailed design description provides the internal design specifications for all system components. It provides the information needed by programmers prior to component installation. It is also useful in the development of *unit test plans*.

1.2.4.5 FORMAL DESIGNS USED IN THIS DOCUMENT

This document presents the decomposition and dependency descriptions concurrently, using *data flow diagrams* that simultaneously display both the breakout and the connectivity of the system components. The interface description is provided by *entity relationship diagrams*, a *data dictionary*, and textual descriptions.

The detailed design description, as stated earlier, is *not* presented in this document. Rather, each Work Package group has the opportunity (*and responsibility*) to develop their own detailed design, working in conjunction with the Gemini Project Office. It *is important*, however, to note that the detailed designs are required to match the philosophy of design presented here.

1.3 SCOPE

This document presents the necessary information for the development of control software for the Gemini telescopes. It provides a means for communicating the software design. The SDD *must be viewed* as a document that is constantly evolving during the design of the software, while providing a consistent design philosophy throughout the design process. There is no attempt to fix all design decisions in stone.

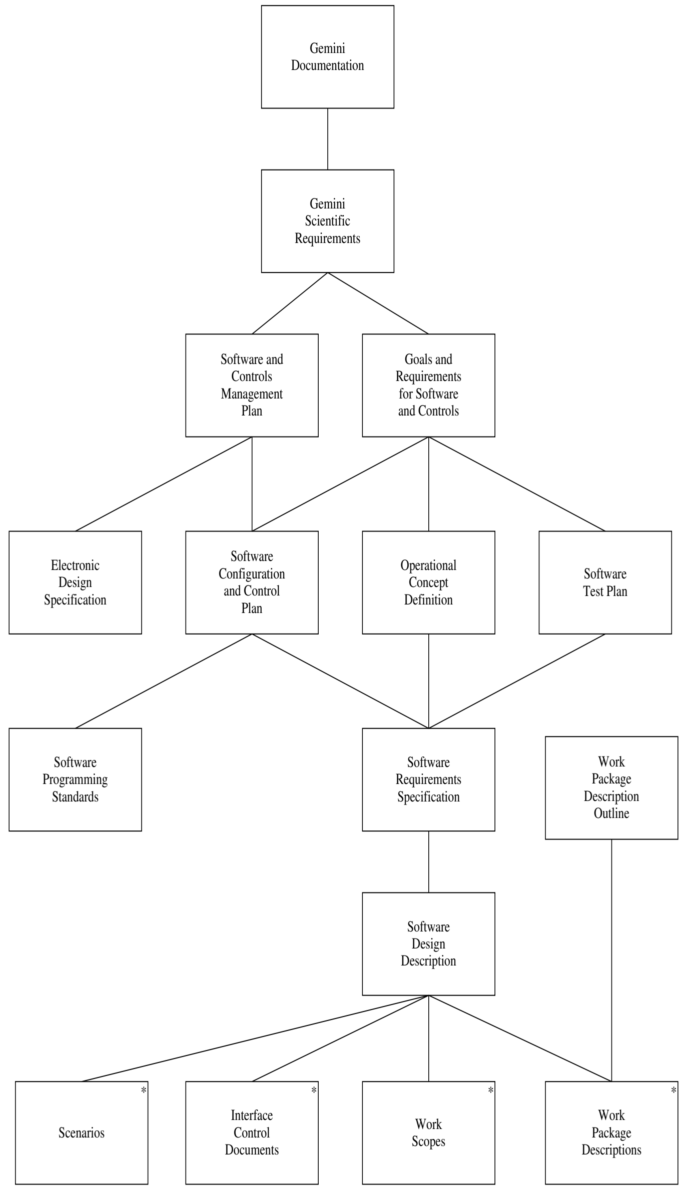
2

References

The Gemini Control System is a sophisticated system consisting of many components. Consequently, the documentation describing the system is also sophisticated and is composed of several documents.

Figure 2 - 1 presents a directed graph showing the relationships and dependencies among the various Gemini Software and Controls documents. The higher level overview documents are shown at the top of the diagram and the lower level, more detailed documents are shown at the bottom.

FIGURE 2 - 1 Gemini Software and Controls documentation dependencies



2.1 RELATED DOCUMENTS

The following is a list of the documents that are referenced in the SDD or the Interface Control Documents. The ideas of many of the authors of these documents have often found their way into this SDD. Any reference in this document can be traced to this reference list. The following is an example of a reference as they appear in the SDD: [2].

[1] AAO/DRAMA_GUIDE_3, *Guide to Writing DRAMA Tasks,* Tony Farrell, Anglo-Australian Observatory.

[2] AAO/DITS_SPEC_5, *Distributed Instrumentation Tasking System*, Tony Farrell, Anglo-Australian Observatory.

[3] AAO/SDS_07, *Self-defining Data System (SDS)*, Jeremy Bailey, Anglo-Australian Observatory.

[4] AAO/IMP_MANUAL_8, *Interprocess Message Passing (IMP) System*, Keith Shortridge, Anglo-Australian Observatory.

[5] Beard S.M. and Daly P.N., *CGS4 Data Reduction User's Guide,* Starlink User Note 27, Royal Observatory Edinburgh and Starlink Project, Rutherford Appleton Laboratory, UK, October 27, 1992.

[6] P.R. Bevington, 1969. *Data Reduction and Error Analysis for the Physical Sciences*, ISBN 0—07—005135—6, McGraw Hill inc., New York.

[7] V.Boutenko & B.Grundseth, 1994. *New Tool for PSF Measuring*, Canada France Hawaii Telescope Information Bulletin No.31, August 1994, page 10

[8] CAPFAST *"Electronic Circuit Design CAE User's Manual",* Phase Three Logic inc., Beaverton, Oregon.

[9] Malcolm Currie, Pat Wallace & Rodney Warren-Smith, *Starlink Standard Data Structures*, Starlink General Paper 38, Starlink Project, Rutherford Appleton Laboratory, UK, 1989.

[10] P.N.Daly, *QMAN — A Generic Queue Manager*, UKIRT Observer Note 16, Joint Astronomy Centre, Hilo, Hawaii.

[11] EPICS *IOC Application Developers Guide.*

[12] EPICS *Channel Access Reference Manual*, Jeffrey O. Hill, EPICS Group, Los Alamos National Laboratory.

[13] EPICS *Release Four Channel Access Functional Specification* (DRAFT - 23 September, 1994), Jeffrey O. Hill, EPICS Group, Los Alamos National Laboratory.

[14] EPICS *Alarm Handler User's Guide*, Martin Kraimer, Ben-Chin K. Cha, Mark Anderson, Janet Anderson, September, 19, 1991, Los Alamos National Laboratory.

[15] EPICS *Alarm Configuration Tool and Alarm Handler User Interface System Requirements Definition* (DRAFT), Janet Anderson, Ben Chin-Cha, August 30, 1993. Los Alamos National Laboratory.

[16] EPICS *IOC Record Reference Manual*, Janet B. Anderson, Martin R. Kraimer, October, 1992, Los Alamos National Laboratory.

[17] EPICS *Archiving Reference Manual,* Roger Cole, Los Alamos National Laboratory. [N.B. This document is in preparation. The Gemini project has a draft copy. Contact the author for up to date information].

[18] ESO *Graphical User Interface Common Conventions,* Doc. No. GEN-SPE-ESO-00000-0266, Issue 1.0, 10/05/93.

[19] GAOS-1-FUNCT, *The Gemini Adaptive Optics System Science and Functional Specifications DRAFT*, Glen Herriot, Bob McClure and René Racine.

[20] GEM/02.2/BDK/0993, *Initial Study of the Gemini Observing System*, Dennis Kelly, Royal Observatory Edinburgh, UK.

[21] GSCG.grp.001, *A Report on the Interface Model Issues*, Gemini 8m Telescopes Project.

[22] GSCG.grp.005, *Gemini System Interfaces*, Gemini 8m Telescopes Project.

[23] GSCG.grp.006, *Overview of System Interfaces*, Gemini 8m Telescopes Project

[24] GSCG.grp.013, *ICD 1 — The System Command Interface*, Gemini 8m Telescopes Project.

[25] GSCG.grp.014, *ICD 2 — Systems Status and Alarm Interfaces*, Gemini 8m Telescopes Project.

[26] GSCG.grp.007, *ICD 3 — Bulk Data Transfer*, Gemini 8m Telescopes Project.

[27] GSCG.grp.008, *ICD 4 — Logging Information,* Gemini 8m Telescopes Project.

[28] GSCG.grp.010, *ICD 5 — Wavefront Sensing Information Interface*, Gemini 8m Telescopes.

[29] GSCG.grp.011, *ICD 6 — ICS/TCS Direct Control Interface,* Gemini 8m Telescopes

[30] GSCG.grp.015, *ICD 7a — ICS Subsystem Interfaces*, Gemini 8m Telescopes Project.

[31] GSCG.grp.009, *ICD 7b — TCS Subsystem Interfaces*, Gemini 8m Telescopes Project.

[32] GSCG.grp.012, *ICD 8 — Non-Conforming ICS Interfaces,* Gemini 8m Telescopes Project.

[33] GSCG.grp.018, *ICD 9 — EPICS Time Bus Driver*, Gemini 8m Telescopes Project.

[34] GSCG.grp.020, *ICD 10 — EPICS Synchro Bus Driver,* Gemini 8m Telescopes Project.

[35] GSCG.grp.021, *ICD 11 — Event Bus,* Gemini 8m Telescopes Project.

[36] GSCG.grp.017, *ICD 12 — Interlock System*, Gemini 8m Telescopes Project.

[37] GSCG.grp.019, *ICD 13 — Standard Controller*, Gemini 8m Telescopes Project.

[38] GSCG.kkg.002, *OCS Behavioral Interface Model Report*, Version 001, 23/6/94, Gemini 8m Telescopes Project.

[39] GSCG.kkg.005, *Baseline Major Systems Interface*, Version 003, 28/6/94, Gemini 8m Telescopes Project.

[40] GSCG.sbw.008, *The Gemini Scenarios*, Gemini 8m Telescopes Project.

[41] GSCG.sbw.009, *Walkthroughs for the Gemini Scenarios,* Gemini 8m Telescopes Project.

[42] *Instrument Working Group Reports*, Revision 1, Gemini 8m Telescopes Project.

[43] Laing, R. A., *A Mathematical Description of the Control System for the William Herschel Telescope.*

[44] Laing, R. A., *Algorithms for Guide-Star Acquisition and Control of Mount and Optical Surfaces.*

[45] McNally, D. ed. 1988, Transactions of the IAU, *Proceedings of the Twentieth General Assembly*, Dordrecht, Kluwer.

[46] NOST 100-1.0, *Definition of the Flexible Image Transport System (FITS)*, NASA Science Office of Standards and Technology.

[47] NOST, *A User's Guide for the Flexible Image Transport System (FITS)*, NASA Science Office of Standards and Technology.

[48] OGIP. A proposal generated at the meeting of the Office of Guest Investigators (OGIP) FITS Standards Panel (OFSP) on June 16 1993.

[49] *Package Requirements Specification: EPICS Synchro Bus Driver*, Gemini 8m Telescopes Project

[50] *Pegasus Owner's Manual*, Canada-France-Hawaii Telescope Corporation, available from John Kerr.

[51] Percival, J.W., White, R.L, *Efficient Transfer of Images Over Networks*, in Astronomical Analysis Software and Systems II, Volume 52, Astronomical Society of the Pacific Conference Series.

[52] SPE-ASA-G0008, *Gemini Electronic Design Specification*, Garry Sedun, ASA Automation System Associates Ltd., 10114 McDonald Park Road, Sidney, BC, Canada.

[53] SPE-C-G0014, *Software Requirements Specification,* September 2, 1993, Gemini 8m Telescopes Project.

[54] SPE-C-G0023, *Standard Control System Work Package Description*, Gemini 8m Telescopes Project.

[55] SPE-C-G0026, *Goals and Requirements for Software and Controls*, March 5, 1993, Gemini 8m Telescopes Project.

[56] SPE-I-G0009, *Software Programming Standards,* Gemini 8m Telescopes Project.

[57] SPE-PS-G0001, *Gemini Scientific Requirements*, November 11, 1993, Gemini 8m Telescopes Project.

[58] STD-1016-1987, *IEEE Recommended Practices for Software Design Descriptions*, 1987.

[59] Straede, John. *Standard FITS Keywords*, Anglo-Australian Observatory, available from Keith Shortridge.

[60] SUN/33, *NDF — Routines for Accessing the Extensible N-Dimensional Data Format*, Rodney Warren-Smith, Starlink Project, Rutherford Appleton Laboratory, UK.

[61] *Trade Study: EPICS Event Bus Driver*, Gemini 8m Telescopes Project.

[62] *The Virtual Telescope Interface*, Jeremy Bailey, UKIRT System Note 6.3, 8 November, 1991.

[63] Wallace, P.T., *TPOINT - Telescope Pointing Analysis,* Starlink User Note 100, Starlink Project, Rutherford Appleton Laboratory, UK, 1989.

[64] Wallace, P.T., *Proposals for Keck Telescope Pointing Algorithms,* Starlink Project, Rutherford Appleton Laboratory, 1990.

[65] Ward, P.T. & Mellor S.J., *Structured Development for Real-Time Systems*, Yourdon Press, 1985.

[66] R.F. Warren-Smith, *NDF Routines for Accessing the Extensible N-Dimensional Data Format*, Starlink User Note 33, Starlink Project, Rutherford Appleton Laboratory, UK, 1991.

[67] VLT-SPE-ESO-17210-0001, *Central Control Software (CCS) Functional Specification,* ESO Very Large Telescope Project.

[68] VLT-SPE-ESO-17210-0554, *CCS Logging System Design Description,* ESO Very Large Telescope Project.

[69] *WHT ADAM/EPICS INTERFACE*, Version 0.2, Paul Martin, Chris Mayer, Phillip Taylor, 27 July, 1994. Available from the authors.

[70] WHT-PDF-1, Issue 2.2, *FITS Headers for WHT FITS tapes*, Steve Unger, Guy Rixon, & Frank Gribbin, November 1993. Available from the authors.

[71] WHT-PDF-4, Issue 2.0, *Collection of Data for the Observing Log at the WHT*, Guy Rixon, February 1994. Available from the author.

2.1.1 Reference Guides

The EPICS system is described in references produced by the various EPICS contributors in: [11]- [17].

The method of presentation in this document is based upon the guidelines found in: [58], while the formal design itself is modelled using techniques described in [65].

The *Electronic Design Specification* [52] is part of the Gemini Control System document, but provides stand-alone information on the electronic design.

2 References

Related Documents

Reference Guides

3 System Requirements

3.1 Operational Phases

There are countless ways that one might accomplish observing, with varying degrees of sophistication, flexibility, and efficiency. The Gemini system distinguishes between the following phases of operation:

- engineering/acceptance
- commissioning
- operational/maintenance

In all these phases, the system must retain support for interactive activities such as performing a focus sequence or recalibration. During the operational phase, the system must support *remote operations*, where much of the control activity takes place off-site.

Essential to the efficient operation of the Gemini telescopes after the commissioning phase is the science requirement for the concept of *planned observing*. This requires more preparation in advance of the observing session than is typically associated with classical observing; the observer has developed a *Science Program* that is submitted to the Gemini system *prior* to the actual performance of that task.This program might be as simple as a single science observation or as complex as a complete night (or several nights) of observing. Ideally, the interactive requirements have been minimized. The Gemini Control System has been designed to encourage and simplify planned observing. Planned observing is especially important during the operational phase and encompasses the following *observing modes*:

- queue-based observing

- service observing

Both of these, as well as other observing modes, are defined in “Operational/Maintenance Phase” on page 3 - 5 and described in detail in “Observing Modes” on page 5 - 46.

It must be recognized from the outset that the Science Requirement to provide planned observing presents unique challenges in the area of customer acceptance. If the Gemini Control System is to be seen as a benefit by the customers (astronomers) then it is essential that planned observing be seen as a natural and desired extension to the current modes of observing to which our scientific community is accustomed.

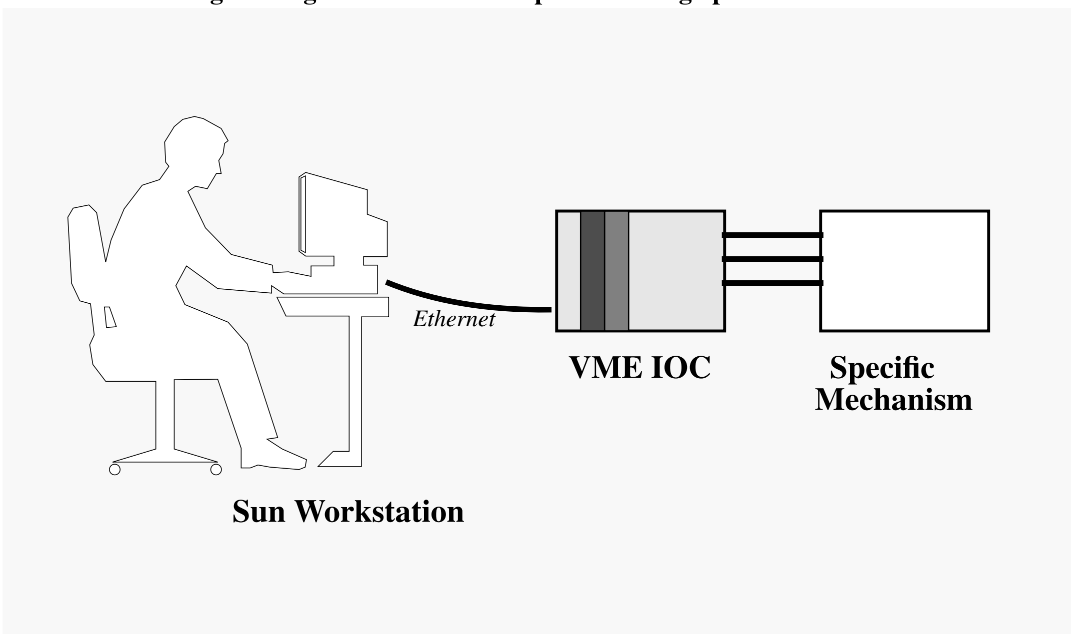
A fundamental precept of the Gemini control philosophy is that providing support for observing modes is a function of the *Observatory Control System* (OCS). As such, knowledge of the observing mode is *not* provided or needed outside the OCS. In particular, the *Data Handling System* (DHS), *Telescope Control System* (TCS), and *Instrument Control System* (ICS) have *no* knowledge of the current observing mode. Interaction between the OCS and these other systems is the same, regardless of the observing mode.

3.1.1 Engineering/Acceptance Phase

This phase is defined as the use of a subsystem completely disconnected from higher level and peer systems. The purpose of this phase is to allow checkout of a subsystem and to isolate systems to facilitate problem diagnosis and solution. Figure 3 - 1 illustrates this style of operation.

This phase is also used for acceptance testing during software development.

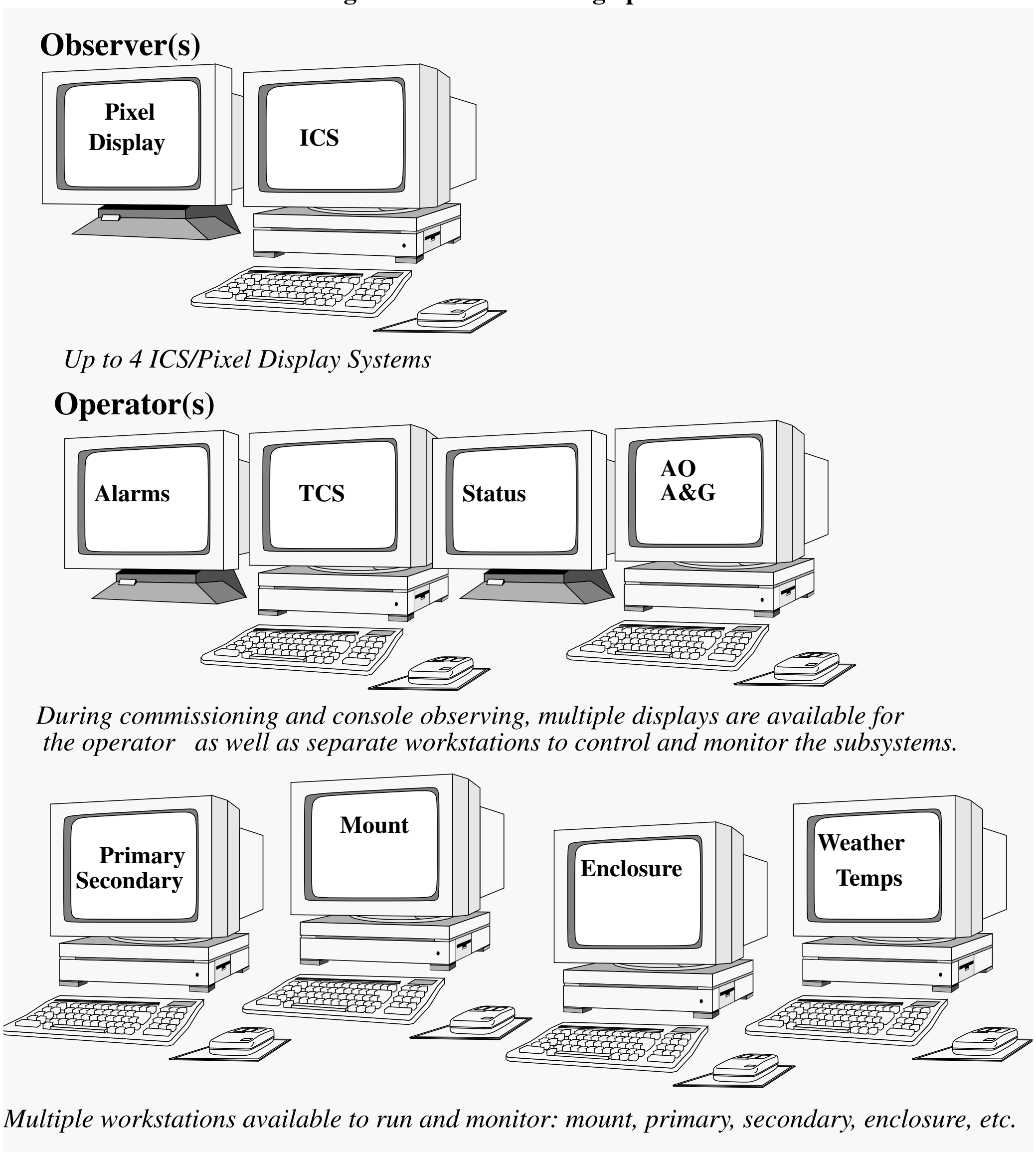
FIGURE 3 - 1 **Engineering/Maintenance/Acceptance testing operation**



3.1.2 Commissioning Phase

During this phase, the system supports the *interactive observing mode.* This mode is defined as an operational context that enables the observer to make decisions on the use of the system *interactively* and to be responsible for deciding what happens next in the system. This method of observing is heavily based on the traditional *one person, one night* philosophy. The observer might have done little preparation of an advance plan for using the telescope. In this mode, the telescope control system can do little to optimize system efficiency as there is no information available prior to the onset of the observing time. Figure 3 - 2 shows the system control consoles needed for interactive observing and during commissioning.

FIGURE 3 - 2 **Commissioning/Interactive observing operation**



In this phase the user acts in real time to sequence the different subsystems through the operational states that are needed. To use the system this way requires detailed knowledge of all subsystems and their interaction.

3.1.3 Operational/Maintenance Phase

In addition to supporting interactive observing and system maintenance, the operational phase provides planned observing.

Both interactive and planned observing may be used during remote operations.

In *queue-based observing*, the astronomer submits a science program to the Gemini system to be scheduled and run without significant interaction with any observer. (There may well be interaction between the system and an observer, but it is expected to be limited to *simple* interactions - alignment checks, focusing checks, etc.) Successful queue-based observing requires very carefully designed science programs and a well-engineered system that is well past the commissioning stage. As the detailed use of a scientific instrument cannot be completely predicted in advance, there is always some interaction required between the system and users. However, one goal of queue-based observing is to reduce such interactions to the minimum required for successful operation of the telescope.

Assuming a well-designed science program, it is possible that the astronomer would turn the observing over to an *on-site* observer for the observing session. This observing mode is called *service observing* and likely requires that the on-site observer and astronomer communicate prior to the observing session to ensure the successful collection of useful data.

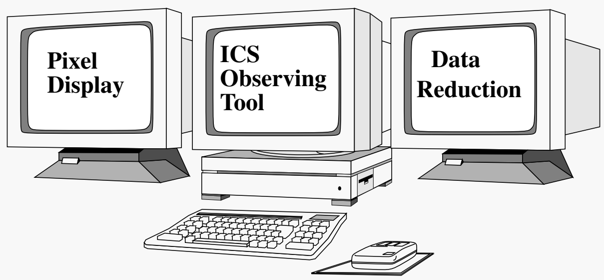
Observers may want to manage the progress of their science program *off-site*. The control system must provide for this *remote observing*, though off-site operation may be restricted to only a few pre-arranged sites (e.g. Hilo or La Serena). Aside from bandwidth restrictions imposed by the means of communication with the remote sight, there is no difference in the user view of the system from on-site observing.

An observer or an engineer may wish to monitor an observing session remotely in order to offer advice or supervision, or to troubleshoot a problem. It is possible to monitor the observing through a lower bandwidth communication, for example an Integrated Services Digital Network (ISDN) link over a telephone line. A low bandwidth connection may only allow a subset of the information available to the operator to be transmitted.

Figure 3 - 3 shows the typical observer station for planned observing, while Figure 3 - 4 shows the operator's environment.

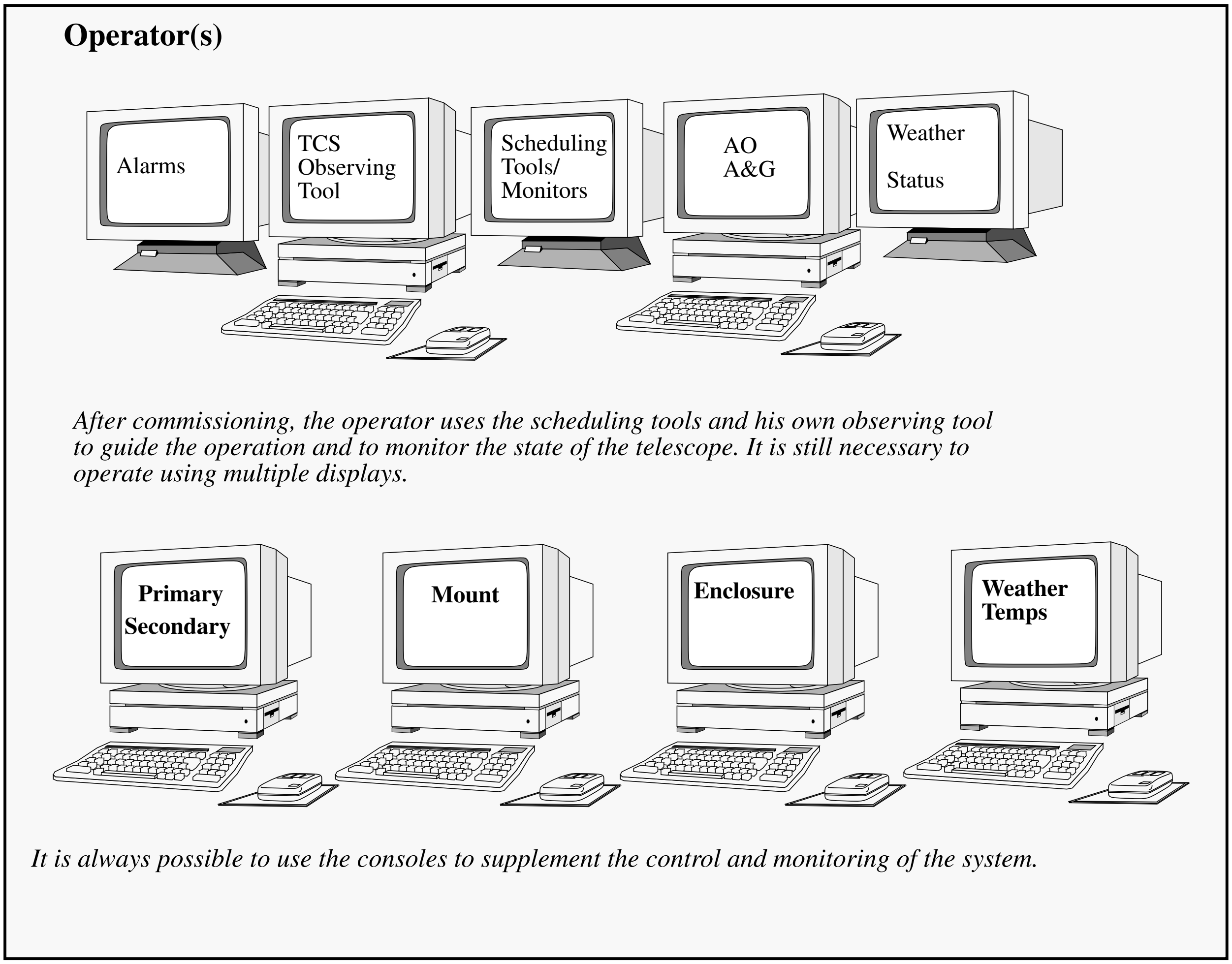
FIGURE 3 - 3 **Planned observing operation, observer station**

Observer(s)



Each observer station includes a control workstation with multiple displays. The observer can use the Observing Tool, Instrument Console, Quick Look, and the data reduction system. All the instruments must be accessible and it must be possible to operate them simultaneously.

FIGURE 3 - 4 **Queue-base observing operation, operator station**



3.2 EFFICIENT USE OF THE TELESCOPE

In order to make better use of the telescope it is desirable to be able to interleave observations from different observing programs. In addition, it is desirable to be able to switch observing programs and/or scientific instruments in order to make use of unique conditions such as good seeing and IR transparency. Finally, is should be possible to exploit concurrency by running system components as independently as possible.

3.3 TIME ALLOCATION MODEL

In order to understand the role and behavior of the Gemini Control System scheduling system it is necessary to make some assumptions about how the *Time Allocation Committee (TAC)* awards time on the telescope.

3.3.1 Granting of Telescope Time

The National TACs grades proposals as NN-G-xx where NN is the nationality identifier (AR,BR,CH,CA,UK,US), G is the grade (A,B,C,D), and xx is the priority within the grade. Thus BR-B-2 would the second priority B grade proposal from Brazil.

The Gemini TAC accepts the (A.B,C) proposals from the National TACs and has the opportunity to regrade them.

In addition, we have assumed that efficient operation has set minimum times that the telescope configuration and instrument complement remain unchanged. This is provisionally set as one month.

- The A proposals are those that *must* be done. The telescope configuration and instrument complement are scheduled in order to accomplish the A proposals. For the most part, the A proposals determine the telescope configuration and the 'straight through' instrument for month-long periods. The A proposals are prioritized for each particular system setup.
- The B proposals are those that *should* be done. The time available around the A proposals is filled in with the B proposals. In general, the B proposals fill in spare time on the 'straight through' instrument and determine which instruments are mounted on the side ports of the instrument cluster. The B proposals are also prioritized.
- The C proposals are a combination of lower priority B proposals and proposals that require only a few observations. These are also prioritized.
- The D proposals are those that are rejected by the TAC.

The proposals are awarded as either *interactive* or *planned* observing programs. If awarded as interactive programs, then specific time slots are allocated to the proposal; during that time that particular observer has sole use of the facility.

Proposals awarded as queue observing programs are put on a priority list for a particular telescope/instrument setup. During the period of time that this setup is available, observations from the A list are done in priority order. If at any time there are no A observations, either because they are all finished or because they are at too large an air mass, the observations from the B and then C lists are run. Rules need to be established that are based on efficient use of the telescope. For example, there might be nearby B&C observations that could be done efficiently even though a more distant A proposal is available.

If, at the end of a time allocation period (assumed to be six months), objects on the A, B, or C list have not been observed, they do *not* carry over to the next period. As is currently the custom, observing proposals that have been ‘weathered out’ must reapply for time. Now the concept of “weathered out” must be expanded to include the fact that the conditions, such as seeing or transparency, were never good enough for the observation to take place.

3.3.2 Tracking of Telescope Time

The concepts of the *Science Program* and *Observing Database* (explained in later chapters) allow the Gemini system to not only plan what should happen at the telescope but to track what actually did happen. With these tools the Gemini system can track (for instance) the amount of observing time per program, per observer, per partner country, etc. in order to produce statistics of interest. One possible application of this would be to adjust the priorities of science programs in order to satisfy the percentages of observing going to the partner countries.

3.4 SCENARIOS

In order to provide a better understanding of the Gemini Control System, it is useful to examine how the system behaves in specific situations. These *operational scenarios* provide both a means of describing the system’s behavior and a means of testing the behavior of the system.

The Gemini Software and Controls Group, in cooperation with the science community, has established a series of *reference* scenarios for this purpose. These scenarios may be considered *test cases* with which the functionality of the control system and subsystems can be measured against the system requirements. They also provide *examples* of the expected behavior for these systems and, in this capacity, assist the reader in understanding the GCS.

These reference scenarios are somewhat arbitrarily divided into several categories:

- *simple* scenarios intended only to illustrate broad functionality of the GCS
- *complex* scenarios that drive the design though extremes of operation
- *engineering* scenarios for examining acceptance testing and maintenance operation
- *commissioning* scenarios that address the special issues characteristic of telescope commissioning

- *detailed interactive observing* scenarios for examining the system behavior during interactive observing sessions

The set of reference scenarios described here is necessarily incomplete. It is fully expected that this set will be extended over time. These scenarios are used throughout this document to help describe the design of the various subsystems comprising the Gemini Control System.

A more complete description of all these scenarios, along with *walkthroughs* showing system behavior for these scenarios exist as separate documents [40] and [41].

3.4.1 Simple Scheduling Scenarios

These scenarios are considered simple because they describe situations that show basic scheduling features of the system without attempting to match actual operation of the Gemini 8-m telescopes. They help illustrate fundamental points in the control system design, but avoid detailed descriptions of real observations.

3.4.1.1 ONE ASTRONOMER, ONE DATA FRAME

The simplest scenario is one where there is a single observer using the telescope, taking a single frame of data with some instrument.

3.4.1.2 ONE ASTRONOMER, ONE COMPLETE OBSERVATION

The first scenario is not particularly realistic, as it typically takes more than a single exposure (‘*science data frame’)* to provide useful astronomical information. This second scenario assumes that multiple exposures need to be taken. For example, there need to be *biases, flats, calibrations, etc.* to accompany the science data, in order to permit useful data reduction. All of these are considered part of the same observation, however, as the gathering of useful astronomical information depends upon them all.

3.4.1.3 ONE ASTRONOMER, MULTIPLE OBSERVATIONS

Often, astronomers have more than one task envisioned as part of a program of study. This means that this program of study can be viewed as a *series of related observations.* This may involve examining several objects, using multiple filters, etc. In this scenario, we assume a program involving the observation of 15 different science objects.

3.4.1.4 MULTIPLE ASTRONOMERS

The next step up is to allow for the possibility of several astronomers, all scheduled to use the telescope during a given operational period (e.g. on the same night). In this scenario, we assume that there are several astronomers with individual programs of study drawn from the preceding scenarios.

3.4.1.5 INTERACTIVE OBSERVING

Some observations may require a high-degree of *interaction* between the observer and the system. While one goal of the Gemini telescopes is to use planned observing to increase the efficient use of the telescope, it is recognized that not all observing can be planned in advance. Interactive observing, where the telescope is under the interactive direction of the observer, must be fully integrated into the operation of the telescope. In this scenario, an interactive observation from one astronomer is added to the preceding multiple astronomer scenario.

3.4.2 Simple Telescope Scenarios

These scenarios are considered simple because they describe situations that show basic features of the telescope system in the context of a single observation.

These scenarios have been fleshed out as part of the UK TCS design study

3.4.2.1 CHOPPING

This scenario illustrates the different modes of synchronizing chopping between the instrument and secondary.

3.4.2.2 NODDING

This scenario illustrates the way in which nodding is synchronized between the mount and the instrument.

3.4.2.3 TIP/TILT CORRECTION

This scenario illustrates how the A&G, TCS, and secondary coordinate tip/tilt corrections.

3.4.2.4 CHOPPING AND NODDING

This scenario combines chopping and nodding.

3.4.2.5 DITHER AND TIP/TILT

This scenario combines tip/tilt with dithering.

3.4.2.6 CHOPPING, NODDING WITH TIP/TILT

This scenario is an example of 10 micron IR observing where all three are needed for optimum image quality.

3.4.2.7 SCANNING A GRATING

This scenario illustrates the level of coordination required between instrument and detector by doing a series of exposures as a grating is moved.

3.4.2.8 ASPECT

Area SPECTroscopy, an observation with unusual timing requirements, has been recast as a complex scenario.

3.4.2.9 SPECTROGRAPH WITH VERTICAL SLIT

This scenario illustrates a common mode of operation where the slit is held fixed as the telescope tracks.

3.4.2.10 SHIFT AND ADD IR IMAGER

This scenario illustrates the various modes of operation where the IR detector array electronics can do fast shift and add - and so the need for tip/tilting the secondary is removed.

3.4.2.11 NON-SIDEREAL TRACKING

This scenario considers an observation of a planet.

3.4.2.12 OCCULTATION EXPERIMENT

This scenario considers an occultation event to examine the capability for accurately time tagging data.

3.4.2.13 DOING A FOCUS RUN ON AN ECHELLE SPECTROGRAPH

This scenario examines the issues with using echelle spectrograph.

3.4.2.14 A MOSAIC OBSERVATION

A simple mosaic is performed.

3.4.3 Complex Control Scenarios

These scenarios attempt to depict situations that are likely to arise during *actual operation of the telescopes.* As such, they are often *considerably* more detailed than the simple scenarios and often include unscheduled events typical of real observing.

Because of the complexity of these scenarios, they are outlined here, with a complete description of each appearing as an appendix to this document.

3.4.3.1 AREA SPECTROSCOPY

This scenario looks at the capability of the system to provide instruments with direct control of the telescope offsetting for time-critical synchronization to maintain efficient use of a detector. Although this is not a 'normal' method of operation, the scenario illustrates the flexibility of the control system design to handle unusual performance requirements.

3.4.3.2 LONG SLIT IMAGING INFRARED SPECTROSCOPY AT 12 MICRONS OF M82

This scenario describes an observation in which a spectral map is taken of a region of the Starburst Galaxy M82. The intent is to map a region of 20x30 arcseconds with the major axis of this rectangular region aligned with the major axis of M82. The equatorial coordinate system is rotated to align the x-axis with the major axis. The slit is aligned with the y-axis, with a position of PA=25 degrees. The observation requires both secondary chopping and telescope nodding along the slit.

Adding to the complexity of this scenario are a series of unexpected events that occur during the observation. The system design must deal effectively with these unscheduled events.

3.4.3.3 MULTI-OBJECT SPECTROSCOPY

This scenario involves surveying clusters of galaxies spanning a range of redshifts using a multi-object slit spectrograph. It involves sophisticated planning to (among other things) identify suitable candidates for spectroscopy as well as complex operations while running the telescope (to prepare, construct, and use a slit mask), including sophisticated interaction with the data reduction system.

3.4.3.4 PRE-EMPTIVE OBSERVING

This scenario will examine the impact of a decision to change the observing plan for the night in real time - in this case to observe a supernova.

3.4.4 Detailed Interactive Observing Scenarios

The early operation of the telescope is expected to involve quite a bit of interactive observing. These scenarios test the performance of the design during interactive observing. A selection of the simple scenarios will be used.

3.4.5 Preplanned Observing

These scenarios will look at the impact of the preplanning activity.

3.4.5.1 REMOTE PREPLANNING

In this scenario a observer will plan a science program, submit it for execution at a later date, and then monitor its progress through the system.

3.4.5.2 LOCAL PREPLANNING

In this scenario an observer will preplan a science program in advance of the run. This scenario will look at the issue of using items, such as filters, that are frequently changed.

3.4.5.3 REAL TIME PREPLANNING

This scenario will look at the impact of doing preplanning while an observation is being run.

3.4.6 Engineering & Commissioning

These scenarios examine the need to use the telescope for objectives not directly related to observing. A major consideration of the software design process is the development of a design that simplifies the connecting of the various system components into the final design. Acceptance testing and engineering are a critical part of this process. These scenarios detail and describe the activities that take place during acceptance testing and other stand-alone system operations.

The operation of the Gemini Control System during commissioning involves a great deal of coordination and system monitoring. Commissioning scenarios examine the behavior of the design during this crucial period.

3.4.6.1 OPTICAL ALIGNMENT

This scenario will examine how the initial optical alignment will be done and how it will be maintained.

3.4.6.2 METROLOGY

This scenario will look at the need to support new and temporary metrology.

3.4.6.3 NIGHTLY START-UP

Each night there are several steps that must be performed to bring the system up to operational performance levels. The process of cold starting the system at the beginning of the night includes:

- nightly recalibration of telescope points
- calibration of primary mirror figure
- twilight calibration data acquisition

These can be viewed as separate activities that the system must support. One of these activities is used as a reference scenario.

3.4.6.4 POINTING TESTS

This scenario involves the process of checking and updating the pointing information that is to be performed at the start of each observing session (i.e. at the start of each night and after changing instruments). This process consists of two fundamental operations:

- calibration of pointing origin positions
- calibration of collimation parameters, using three to five stars

This scenario is closely to tied to several other similar operations: full pointing tests, involving 50-100 stars, rotator axis calibration, etc.

3.4.6.5 TRACKING TESTS

This scenario will examine how tracking tests would be done.

3.4.6.6 FIGURE CALIBRATION

This scenario will examine how the *open loop* corrections to the primary mirror figure would be generated and applied.

3.4.6.7 MULTIPLE PROBE TESTS

In a number of engineering tests it will be necessary to have two or more probes simultaneously collecting data in the focal plane.

3.4.6.8 ENGINEERING LOGGING

It will be commonly required to allow engineers to collect data from the subsystems at high data rates. This scenario will examine debugging the secondary control system - where data may be needed at 200 Hz rates.

3.4.6.9 ENGINEERING ARCHIVING

In order to better calibrate the (for instance) tracking corrections there is a need to collect and archive data for future analysis.

3.4.7 Fail-safe Scenarios

These scenarios will examine common failure modes of the system and examine how the system will recover.

3.4.7.1 GLOBAL POWER FAILURE

In this scenario the entire power system is cycled.

3.4.7.2 SINGLE COMPUTER POWER FAIL

In this scenario a single computer will *die* and then be repowered.

3.4.7.3 SINGLE COMPUTER HANG

A common subsystem failure mode will be where, for instance, the mount control system hangs and keeps the demand velocity to the azimuth drive constant.

3.4.7.4 UNDERLYING HARDWARE FAILURE

This scenario will examine the impact of a servo amp failure and, separately, the loss of the high order bits of the azimuth encoding system.

3.4.7.5 NETWORK FAILURE

This scenario will look at the impact of different parts of the network halting or getting very slow.

3.4.7.6 INTRUSIVE ACCESS

This will examine the firewalls needed to prevent unauthorized access.

3.5 SCREENS AND DISPLAYS

3.5.1 Purpose

This section describes the general characteristics of the graphical user interfaces found in the Gemini Control System. The emphasis is on content, not form. While concentrating exclusively on the graphical user interface, the equivalent command-line interface functionality should be provided where appropriate.

3.5.2 Definitions

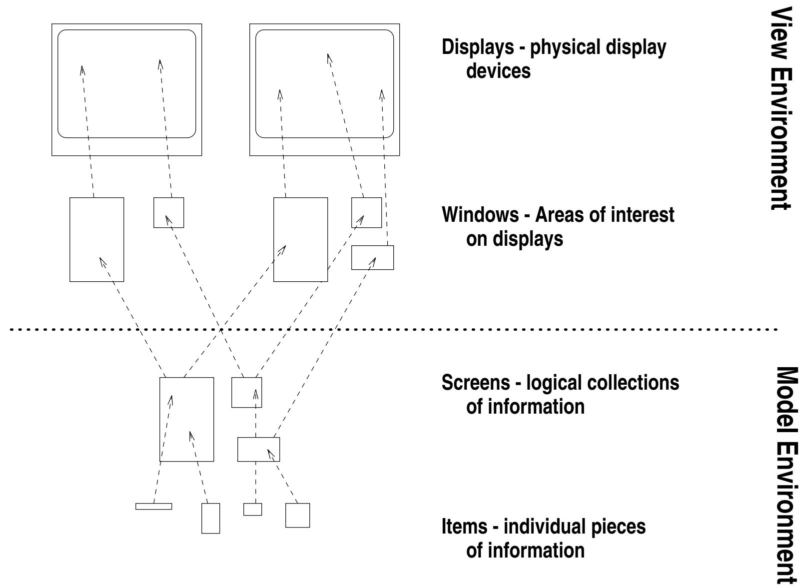
There are a few terms used here to describe the components of the user interface that need clarification. Both the *logical abstraction* and the *physical realization* for these components are defined here.

The display of information is split between the *model* and the *view*. The model provides the environment that applications run in and is composed of screens and items. The view is the physical representation of the model in front of the user.

The view is composed of *displays* and *windows*, while the model operates on *screens* and *items*.

- Display - A display is a physical display device, such as a monitor or a workstation. (This corresponds to the X-Windows meaning of a display.)
- Window - Information is reproduced on a display by mapping it to a window.
- Screen - A screen is a logical abstraction for communicating information. The contents of the screen represent information flows between the user and a process, and may be bidirectional in nature, though most screens in the Gemini control system provide for unidirectional flow. Screens are instantiated by mapping them to one or more windows on one or more displays.
- Item - Items are the individual components of a screen. They may be hierarchical in content.

FIGURE 3 - 5 **View and Model Environments**



Since content is more important than physical form in preliminary design, these sections focus on screens and items, not displays and windows.

3.5.3 General Features

3.5.3.1 PRIVILEGES

Privileges are determined by the user, not by the screen. Whether a user can perform control through a particular screen or not depends on that user's privileges (mode of operation), not the screen's functionality. A user's privileges are associated with all commands issued by that user and may be acted on by any process manipulating those commands. Thus a high-level process may choose to enforce privileges instead of passing the commands on to lower level processes. All subsystems must ensure that user privileges are followed when executing commands.

As a general rule, operators can do almost anything while observers can only control the science program contents, the data handling subsystem, and communications.

3.5.3.2 ON-LINE HELP

All screens include interactive on-line help facilities for the contents of the screens. Details of how the help facilities are implemented are considered part of the OCS work package design that are worked out between the work package responsible and the Gemini Software and Controls Group.

3.5.3.3 BEHAVIOR

The behavior of a screen is provided through the items in that screen. The items available in the Gemini Control System can be divided into several functional categories:

- Control - these accept user input on the control of some system. The type of control can vary both with the system and the level of access to that system. The implementation of a control item may include the features of an informative item (see below) - for example, an enclosure control item might display a schematic of the enclosure and allow the user to control the enclosure through actions to the schematic. (E.g., move the cursor to some portion of the schematic to obtain detailed status for the object represented by that portion of the schematic.) Control screens may be displayed in 'protected mode', where the user's ability to use the screen to manipulate telescope systems is restricted in some fashion. Screens that are protected are clearly marked when displayed. The exact restrictions placed on a protected screen are dependent on the functionality contained in the screen.
- Informative - these provide information to the user. They are subdivided further by the nature of the information:

1. Image display items that provide pixel-based image display. This includes display of science data, output of cameras, etc.
2. Status display items that provide ongoing feedback on system status checks. This can range from interlock conditions to seeing conditions to error budget activity.
3. Alarms that demand immediate attention of the user. They can be viewed as a special case of status display.

- Communication - items that allow dialogs among users (most commonly between observers and the telescope operator but also among collaborating observers).

The Gemini Control System provides a 'road-map' screen that determines which other screens are currently displayed in windows. Most screens are organized into a hierarchy to permit rapid access to varying degrees of detailed control and status reporting. However, some screens must always be available, regardless of the current state of the roadmap (status and alarm displays, interlock status, etc.). Other screens may be available outside this hierarchy, but need not always be available (e.g. dialog boxes).

The system normally allows users to move, resize, iconify, etc. windows, but there is a standard command always available that can be used to reset the display to a standard configuration. Some windows may be restricted in some fashion. For example, critical windows may not be iconified.

The system imposes no limits on the number of screens being displayed, nor where (which display) a screen may be mapped.

Managing the windows on the display is the responsibility of the user. Management of the screens (i.e. the contents of the windows) is the responsibility of the Gemini Control System system and is not available for user control.

If a screen contains items that are not available to the current user, then those items are clearly denoted as being unavailable in some fashion.

It is possible to map any screen to windows on any combination of displays to allow remote access, monitoring, and shared collaborations. For example, a remote observer may want to view the telescope control system consoles while the on-site operator is controlling the telescopes. As another example, an on-site observer may be working with a remote observer on an observation.

Access to displayed screens is controlled by the privileges of the user at the target display. As a result, it may be that two or more users might all have control access through multiple instances of the same screen. This capability is often selectively restricted, however.

4 High-level System Design

This chapter provides the framework for the Gemini Control System (GCS) when viewed from the highest level. The principal system components are defined, as are the principal interfaces in the system.

The remainder of this document expands this initial framework.

Appendices 1-3 of this document show the formal design views of the GCS.

The Interface Control Documents [26]-[30] provide details on the interfaces introduced in this chapter.

4.1 Principle of Operation

The Gemini Control System views the system as existing in a particular *state* at any particular moment. The transition of the system from one state to another is accomplished using a *configuration* that describes the controllable conditions for the new state. The GCS thus controls subsystems by providing a *target* configuration that subsystems are directed to achieve. The GCS monitors the performance of the system by comparing the target configuration with the *actual* configuration as reported by the subsystems.

4.2 PRINCIPAL SYSTEM COMPONENTS

At the highest level the GCS is comprised of the following components, each described more fully in later chapters:

TABLE 4 - 1 **Gemini Control System Components**

Principal System	**Subsystems**
Observatory Control System (OCS)	Visible User Interface Configurable Control System
Data Handling System (DHS)	Quick-look Display Servers Data Storage Server Data Reduction System Data Transport Manager
Instrument Control Systems (ICS)	1-5 micron imager control system 1-5 micron spectrometer control system 8-30 micron imager control system 8-30 micron spectrometer control system Multi-object spectrograph control system High Resolution Optical Spectrograph control system
	ARCON optical CCD controller ALICE infra red detector controller
	Plus other systems TBD…
Telescope Control System (TCS)	Telescope Master Control
	Adaptive Optics Control A&G (acquisition and guidance) Control Secondary Control Primary Support Control Mount Control Cassegrain Rotator Control Enclosure Control
	Interlock System

4.2.1 Observatory Control System

The Observatory Control System is the *overseer* of most GCS observing activities. The principal job of the OCS is to support and enable the astronomers and other users of the Gemini Telescopes to accomplish their work correctly and efficiently.

The OCS includes and provides application programs with functionality that enables the astronomers, operators, and other users of the system to accomplish their work.

The OCS also includes the system software that provides the infrastructure necessary to implement the functionality of the OCS application programs.

4.2.2 Data Handling System

The Data Handling System provides a data handling infrastructure for the OCS and the instrument control systems. Its primary functions, which are all related to processing and analyzing instrument data, are:

- to provide an interface between the OCS, instruments, and the quick-look subsystems.
- to provide a common, efficient data buffering, storage, and retrieval subsystem that is shared by all instrument and quick-look subsystems.
- to provide an interface to the external data reduction system.

Some components of the Data Handling System may be created on demand. For example, observers may have separate quick-look facilities for each instrument.

4.2.3 Instrument Control Systems

There can be multiple Instrument Control Systems concurrently running in the GCS at any one time. An ICS provides the functionality that is required to use a particular instrument.

Additionally, an ICS must interact with the other principal software components as specified in this design document to enable it to operate within the framework imposed by the GCS.

4.2.4 Telescope Control System

The Telescope Control System is a term used to describe the amalgamation of low-level systems that provide the upper levels with the functionality associated with the telescope mount, guiders, and other telescope subsystems.

4.3 EXTERNAL SYSTEMS

The Gemini Control System also interfaces to the following external systems:

TABLE 4 - 2 Principal External Systems

Observers
Operator
Output & Archive
External Databases (Guide star catalogs, etc...)

The interfaces to these principal external systems, as well as other minor external systems (those that interact with a single subsystem), are discussed first in Chapter 9, and then in the appropriate 'Details' chapters. A description of the 'generalized' interface design is provided in the Interface Control Documents..

In addition to the external systems listed above, the Gemini Control System relies on functionality that is provided by several software applications produced by commercial companies or institutions:

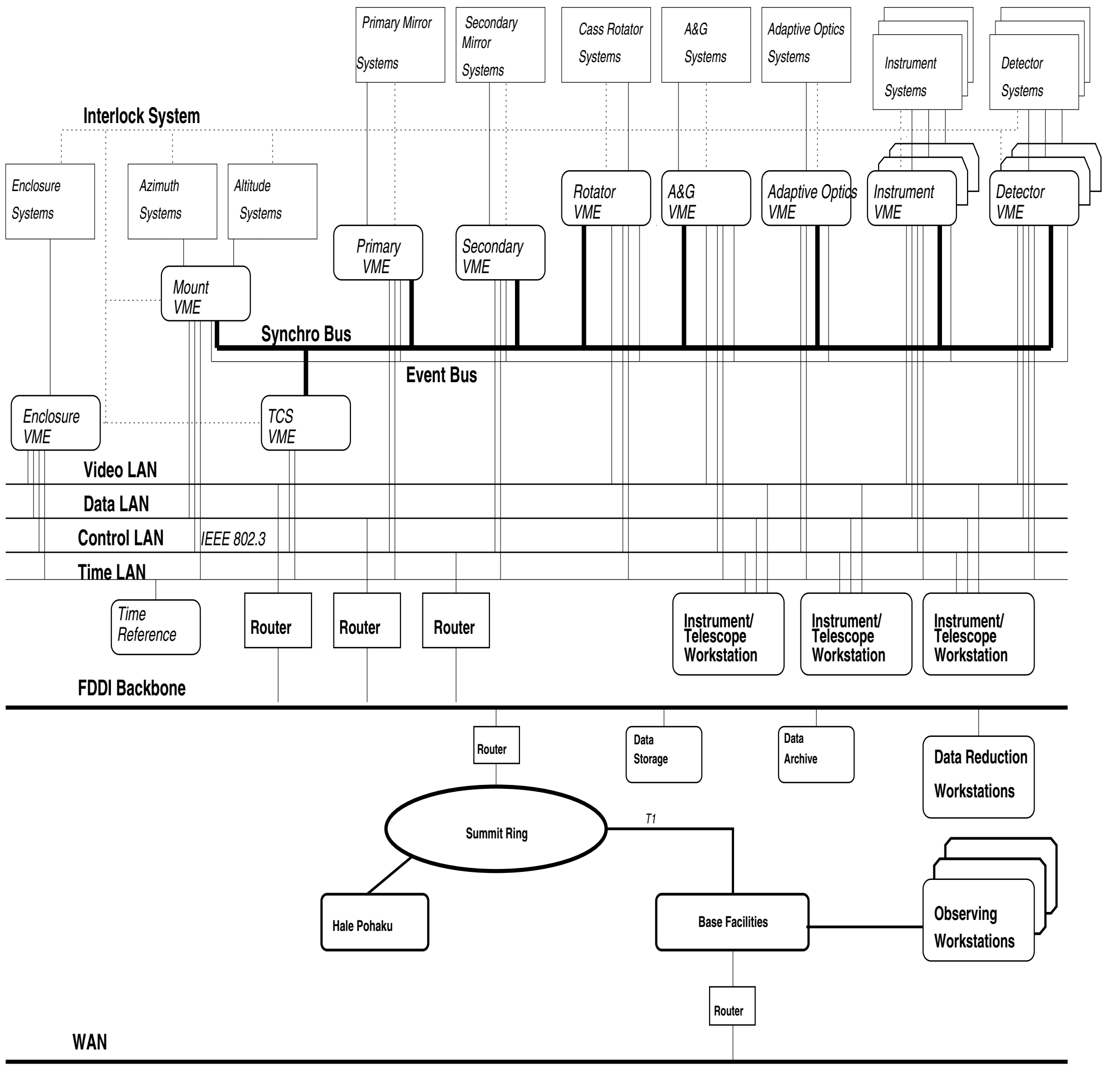
A Relational Database. A relational database is assumed to exist. It is used to store information relating to the observations, data, and the operational status of the Gemini telescopes and their subsystems.

An Astronomical Data Processing Application. Creating and maintaining an astronomical data processing system is outside the scope of the Gemini work. The Gemini telescope cooperates with a known data reduction system such as IRAF or ADAM. The GCS specifies interfaces to the applications as discussed in the Data Handling System chapters of this document.

4.4 HARDWARE COMPONENTS

The following figure shows the organization of the system hardware components as well as the physical connections between them.

FIGURE 4 - 1 **Gemini telescopes hardware configuration for Mauna Kea**



5 HIGH-LEVEL OBSERVATORY CONTROL SYSTEM CONCEPTS

This chapter discusses the framework and design issues of the Observatory Control System (OCS) when viewed from the highest level. The major system components are defined and discussed.

The Appendices of this document show the formal design views of the OCS. Interface Control Documents (ICDs) 1 [24] and 2 [25] describe the software interface between the OCS and the other principal systems. The information in these other documents may be of use when reading this chapter.

5.0.1 Changes since the Preliminary Design Review

Changes to this chapter have focused on fixing problems and incorporating suggestions that came out of the Preliminary Design Review and the System Design Review.

- Modify the presentation of health to guarantee that users will be notified when the number of contributors to a system's health severity changes.
- Improve the definition of the Science Program as the GCS document.
- Discuss the roles of observing and engineering consoles in the design.
- Add timing information and observer priorities to the information in a Science Configuration.
- Discuss the distinction between operators and observers in the SDD.

5.1 OBSERVATORY CONTROL SYSTEM OVERVIEW

The Observatory Control System is the *overseer* of most GCS observing activities. The OCS includes and provides application programs with functionality that enables the astronomers, operators, and other users of the system to accomplish their work correctly and efficiently. The OCS also includes the system software that provides the infrastructure necessary to implement the functionality of the OCS application programs.

The OCS is the principal system component of the GCS that controls and coordinates the activities of the other principal system components: the Data Handling System, the Telescope Control System, and the Instrument Control Systems.

5.1.1 OCS Requirements and Goals

Based on the scientific requirements in the Gemini document *Goals and Requirements for Software and Controls*[55] and the practical needs of the observers who will use the Gemini telescopes, there are a number of requirements that the OCS software must fulfil.

- It must be possible to operate instruments concurrently as long as no Gemini telescope shared resources are required. Shared resources consist of things like the telescope beam or the adaptive optics system.
- The OCS must provide appropriate user interfaces for all observing modes during the commissioning and operational/maintenance phases. The observing modes include interactive observing, queue-based observing, service observing, and remote observing. Service observing and queue-based observing are grouped together under the term *planned* observing. See “Operational Phases” on page 3 - 1 for more information on the operational phases and required observing modes.
- The OCS integrates planned and interactive observing. It is an OCS goal to make planned observing the preferred observing mode of the Gemini users.

5.1.2 OCS Software Structure

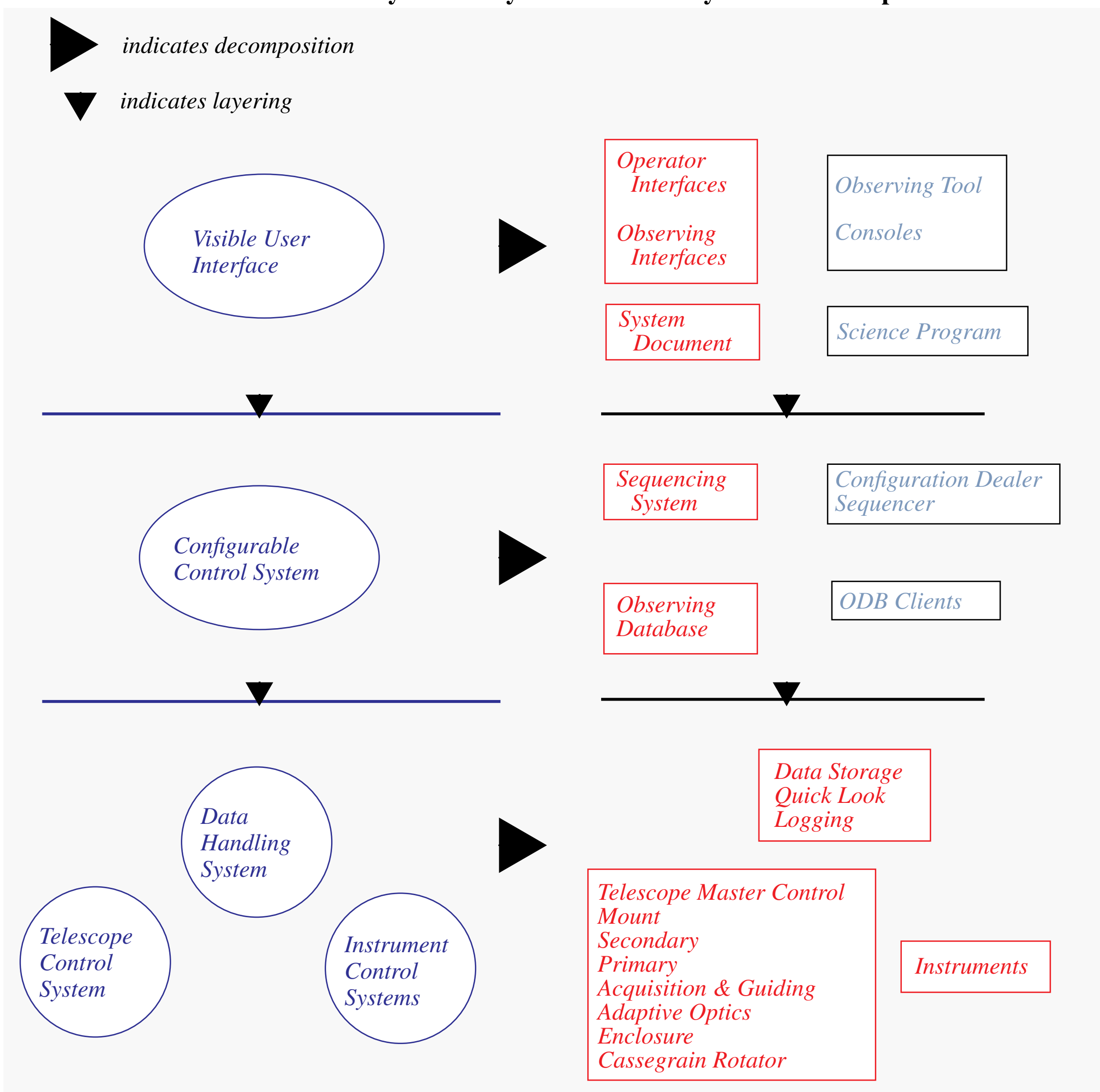
The Observatory Control System is the term used to describe all software entities that provide the observing capabilities required by the Gemini telescope users.

The OCS is decomposed into the Visible User Interface (VUI) and the Configurable Control System (CCS). The OCS controls and coordinates the activities of the other principal system components to allow the astronomer to complete his science goals or other work.

Figure 5 - 1 shows the software layering in the OCS and its relationship to the other principal GCS software system components. The decomposition of the OCS is shown by the large right-pointing triangles. Software layering is shown by the downward-pointing triangles.

At the highest level the OCS presents the capabilities of the Gemini telescopes. The graphical user interface programs and concepts are grouped into the Visible User Interface. The functionality of the Visible User Interface is implemented upon the Configurable Control System, which manages the other principal software system entities.

FIGURE 5 - 1 **Observatory control system software layers and decomposition**



5.2 OCS VISIBLE INTERFACES DESCRIPTION

The Visible User Interface (VUI) of the Gemini control system is made up of the applications, tools, documents, and concepts that are provided by Gemini and used by astronomers when they interact with the Gemini Control System. The VUI is more than the graphical user interfaces (GUI) presented to the users by the applications; it includes the way the telescope and associated hardware is presented to the users and the ways that scientists will do their work with the Gemini telescopes.

There are several fundamental concepts associated with the Gemini VUI. The next sections describe the VUI concepts, astronomer requirements and applications, and the operators' requirements and applications. This is followed by a discussion of the composition of *science programs*, and the structure of the *scheduling system* for manipulating planned science programs within the Gemini system.

5.2.1 VUI Design Assumptions

The Gemini telescopes are complex and the design of the VUI makes a few assumptions that simplify the OCS software design at this highest level. See "Screens And Displays" on page 3 - 17 for the definition of terms used here and a general discussion of user interface basics.

Workstation Class Machines. The OCS design assumes that the system is implemented on workstation class machines that will be adequately powerful and plentiful to accomplish the software requirements of the OCS.

Multiple Display Layout. Multiple displays are assumed to be available to operate both the telescope and instrument systems. The kinds of screens required by staff and observers will vary depending on the phase of telescope operations and the observing program. There is no requirement to produce an operator VUI that can be optimally utilized from a single display. See "System Requirements" in Chapter 3 for more information on the hardware and personnel configurations expected during the project phases.

Bandwidth Issues. The OCS design assumes that the workstations supporting the observing process are networked and that the bandwidth of the network connections of these machines is adequate to support the requirements of the observing process as described in this document.

Remote Observing. It is assumed that remote observing and monitoring activities use the same visible interface that will be used at the telescopes. Options for minimizing the amount of bulk data that must be transferred between sites will be available whenever appropriate. The Gemini software effort does *not* include the creation of new compression algorithms, but relies upon industry leadership in this area. Compression techniques, such as JPEG or the H-transform methods discussed by Percival and White [51], will be used to transfer images to remote sites whenever possible. The X11R6 release includes an optimized XLIB protocol for use with X11 programs executing over low-bandwidth connections. However, the OCS design does not rely upon long distance X11 connections.

Observers and Operators. The OCS makes a distinction between operators of the Gemini telescopes and their users, the observers acquiring data with the Gemini telescopes. The distinction in the document is merely to emphasize that these two kinds of Gemini users have duties and levels of expertise that are assumed to be different; the distinction does not indicate some fundamental assumption in the software design. Operators generally have more privileges and can access any of the Gemini tools and applications while observers can only access some operator tools if they have been given permission.

Operator Issues. It is assumed that during some operational phases it may be necessary to allow multiple operators to control the system at one time. It is assumed that the operators are *expert* users who have more familiarity with telescope systems than visiting users.

5.2.2 Observing Approaches and Issues

The changes that have taken place in the systems that control telescopes and their related hardware have moved astronomers away from hands-on access to the underlying hardware systems. Some time in the past astronomers stood in domes on cold nights peering through eyepieces and adjusting telescope and instrument controls directly. Gradually, technology progressed to the level that now allows astronomers to remain in comfortable rooms, possibly away from the telescope site, viewing telescope images on television monitors and interacting with the telescope and its systems with computer applications that control the hardware.

The new technology is popular because it allows astronomers to more easily manage the increased complexity of observing and more importantly, to focus on the issues of acquiring quality science data and efficient telescope usage.

Observing is the process of gathering the data needed to do the science. The science is generally the result of what the astronomer does with the data he has acquired during observing. The primary job of the OCS and the Visible User Interface is to facilitate the observing process.

Most modern astronomical software systems provide high-level control software for the instruments and telescope hardware systems required for observing. Graphical User Interface (GUI) applications are often used to present and simplify the functionality of these systems. These applications often present the capabilities of the hardware systems as control panels on the computer display, which are designed to remind the user of familiar hardware control panels.

With the kind of interfaces described in the previous paragraph, the observing process consists of operators and observers clicking on graphical widgets on computer screens to move and adjust the hardware in the telescope and instrument systems. Ultimately, an *observe* widget is pressed and the science data is acquired. At this point, the astronomer uses his astronomical training to analyze the data. Everything up this final point can be viewed as overhead that is required to get the science data. It is important to minimize this overhead whenever possible.

The Gemini telescopes are quite a bit more complicated than current telescope systems and this places new demands on the control system and the Visible User Interface. Meeting the image quality specifications requires sophisticated hardware systems with many controls that produce copious status information. To optimally observe a particular target with the Gemini telescopes in adaptive optics mode, there must be two properly chosen wave front sensor stars for the guiding system. The enclosure, primary, secondary, and adaptive optics systems must be tuned and operating correctly to meet the image error budget. The VUI must provide simplified telescope access for users in all operational modes.

5.2.2.1 INTERACTIVE OBSERVING

Interactive observing occurs when the *next* action in the observing process is determined by the individuals involved directly in the observing process just before the action is executed. Most current telescopes operate in some kind of interactive mode. The scenario described in the previous section with operators and observers pushing buttons and sequencing operations is a typical interactive observing session. See "Operational Phases" on page 3 - 1 for more information on interactive observing.

Interactive observing is an important part of Gemini observing during the commissioning and operational/maintenance phases.

5.2.2.2 PLANNED OBSERVING

In addition to interactive observing, the Gemini OCS must support the concept of *planned observing*. Planned observing means that the astronomer plans his observations before he goes to the telescope. It is a goal of the OCS design to make planned observing an attractive and desirable option for Gemini users. There are a number of reasons for believing that planned observing will eventually become the preferred observing mode.

- *System Complexity* — The Gemini telescope hardware systems are complex. To meet the image quality error budget requires enough target-dependent information (such as guide stars) that eventually astronomers will want to be able to minimize the amount of *setup* time associated with moving to new targets.

  Even the process of field acquisition and verification with Gemini telescopes is complicated because it may involve switching to an imaging instrument and then switching back to the astronomer's instrument.

- *Automated Setup* — An important advantage of planned observing is that it allows the system to automate the process of preparing the telescope systems for an observation. This minimizes errors, saves time, and should enhance the observer's confidence in the hardware allowing him/her to concentrate on the acquisition and analysis of science data.

  This becomes more important in scenarios with multiple observers using different instruments during a single observing session.
- *Site Conditions* — Both Gemini telescopes are located high on remote mountains. The combination of boredom, high altitude, and system complexity conspire to increase undesirable errors when observing interactively. Visitors will want to minimize observing errors.
- *Flexibility* — Planned observing will allow the staff to have alternate and backup plans available for telescope operators to be more responsive to variable weather conditions.
- *Increased Data Returns* — Observers submitting observations with demanding constrains are more likely to eventually get their data than under a planned mode when compared to interactively observing at the telescope over a short period of time. Planned observing can produce data for observers that might not otherwise be produced.

5.2.2.3 REMOTE OBSERVING

The design of the Observatory Control System provides support for remote monitoring of ongoing observations by astronomers. The creation, submission, and modification of science programs is also supported. Support for more interactive forms of remote observing will be considered by the OCS Work Package group. In the worst case, any X-based program can be run over a suitably quick network connection.

5.2.2.4 SUMMARY OF OBSERVING APPROACHES AND ISSUES

The OCS design is focused on the observing needs of the astronomers who will use the Gemini telescopes. It is the goal of the OCS to provide a state-of-the-art software system that simplifies the observing process.

Interactive observing is an important part of observing with the Gemini telescopes and it is supported throughout the design of the OCS. In addition, the OCS supports planned observing in a way that is a natural extension of interactive observing.

5.2.3 OCS Support for Operation Phases

There are three important operational phases in the lifetime of the telescopes and related subsystems: engineering/acceptance, commissioning, and the operational/maintenance phases. See “Operational Phases” on page 3 - 1 for more detailed information on the operational phases.

Engineering/Acceptance phase. The engineering/acceptance phase is supported by applications and software provided by the Work Package groups, not the OCS.

Commissioning phase. During commissioning, staff and astronomers use interfaces provided by the OCS group to control the telescope in the interactive observing mode. See “Commissioning Phase” on page 3 - 3 for a more complete description of the commissioning phase. The applications developed by the Work Package groups for engineering/acceptance are also available during the commissioning phase.

Operational/maintenance phase. To support the operational/maintenance phase, the OCS group provides applications and software that supports the planned observing modes in addition to the interactive observing mode. The software developed for the this phase is integrated with the commissioning phase software to provide the additional functionality required during the operational/maintenance phase. The applications developed by the Work Package groups for engineering/acceptance are also available during the operational/maintenance phase to aid the diagnosis of problems.

5.2.4 Visible User Interface Overview

The VUI must include applications and concepts that provide the functionality required by users at each phase of the project and for each of the possible observing modes. This means that the OCS must provide for interactive and planned observing, which includes queue-based observing and service observing. This section discusses how these two observing modes are supported.

5.2.4.1 VUI FOR INTERACTIVE OBSERVING

The OCS provides traditional graphical user interface control panels that allow interactive control of the telescope and related subsystems. These control panels are called *consoles* within the OCS design. The following are OCS console design guidelines.

- The OCS provides a consistent look and feel across all consoles. All graphical widgets are to be used consistently whenever they appear in order to limit user surprises. The layout of screens obeys consistent conventions. Currently, the baseline conventions are given in the ESO document, *ESO Graphical User Interface Common Conventions* [18]. The OCS Work Package group will modify this document as required for the Gemini Project.

- Consoles are separate programs rather than combined into one large program that does everything. This results in simpler program designs, limits interactions between systems, and aids long-term software maintenance.
- Users of interactive consoles are kept aware of the status of commands that are issued from consoles, and the consoles indicate errors or warnings that occur during an interactive command from a console.
- The consoles provide controls that are user-oriented. Console screens are not meant to replace engineering screens, but are focussed on the controls and information required for interactive observing.
- The OCS tries to make console interactive observing as simple as possible. This may require integrated, operator command and information screens.
- Display of observing information about subsystems is to be located in a few status screens rather than across several displays or screens.
- Interactive screens should represent system state pictorially whenever possible. A *schematic* is to be used to represent hardware systems and connections when reasonable.

The actual OCS screen examples appear in the detailed subsystem design parts of this document. For example, the secondary subsystem consoles appear in “Details of the Acquisition & Guiding System” in Chapter 19.

The OCS console prototypes are important because they indicate the kind of information required to adequately monitor and control the principal systems and their subsystems during the commissioning and operational phases. The principal system developers can look at the prototypes and see what information is expected to flow up from their low-level systems and design accordingly.

FIGURE 5 - 2 **An example OCS console**

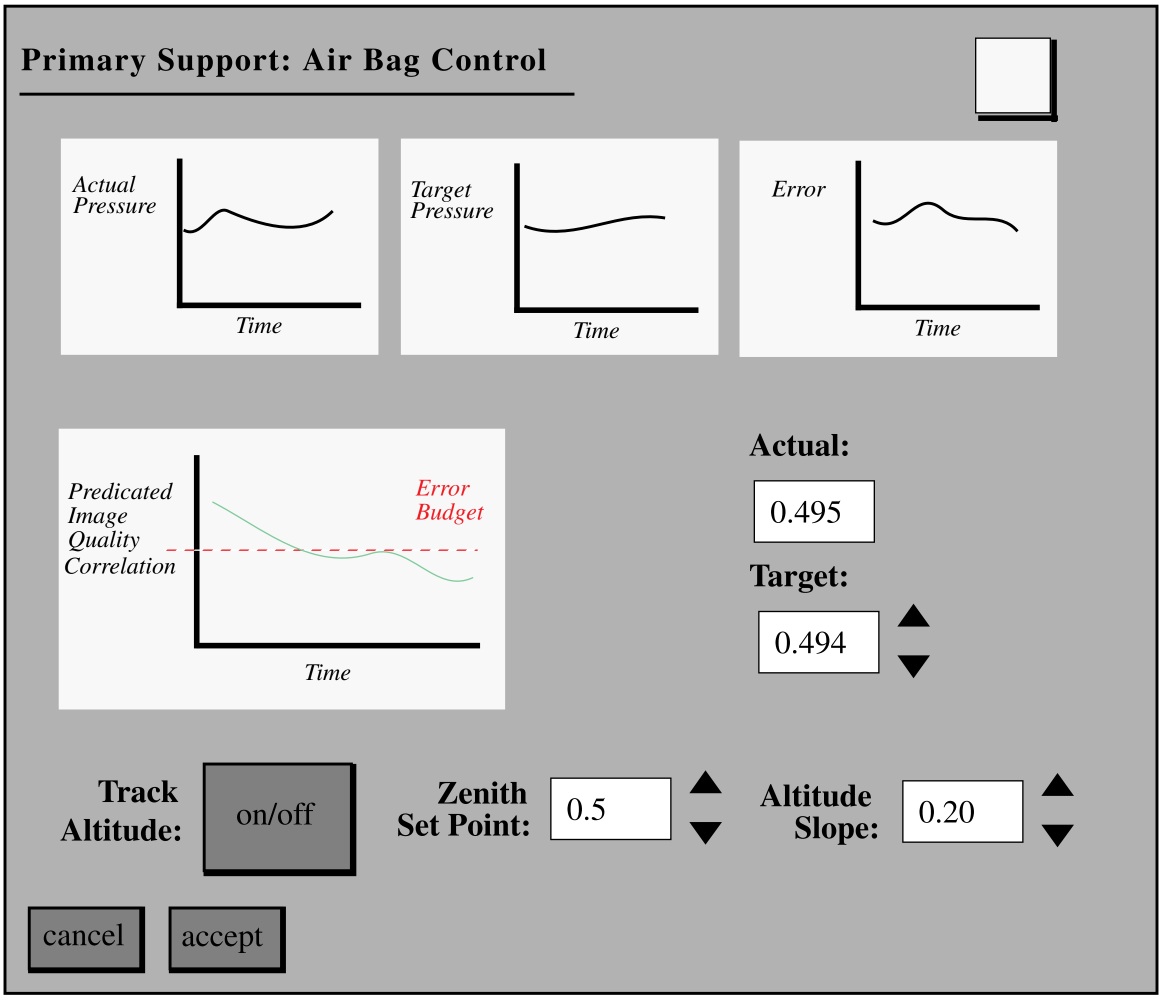


Figure 5 - 2 shows an example of an OCS interactive control console. The look and feel of this figure is not representative of any particular programming toolkit; it is merely showing the kinds of information and controls that are required in the OCS screens.

Across the top of the screen are a number of strip-chart items displaying the current and recent values for the primary airbag actual and target pressure along with the error. They are updated periodically with new values. The text boxes with up and down triangles indicate values that can be set. Text boxes with no triangles are read-only informational values. The grey rectangles are buttons; some contain state information like the *track on/off* button and some are momentary contact buttons like the *cancel* and *accept* buttons.

Most of the consoles are related to the control of the telescope systems and are used by the operators of a telescope; therefore, the console designs and controls can assume a higher level of understanding of the Gemini systems. Visiting astronomers will generally not interact with the telescope system consoles. They interact with the instruments they use in their science programs.

5.2.4.2 CONSOLE INTERACTIVE GRAPHICAL CONTROL SEMANTICS

One of the console design guidelines requires that the controls provide status and error information when they execute a control command. Typical GUI programs in control applications handle error and status information poorly. This section suggests a method the OCS console controls will use to present command progress, warnings, and errors. This is called *console interactive graphical control semantics*.

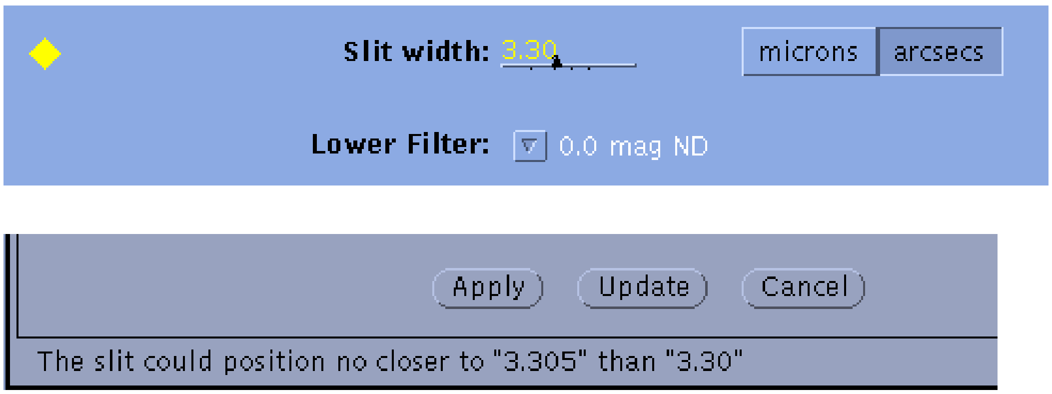
It is sometimes difficult to determine when a control command is executed with GUI interfaces. With a button, it is clear, but with an editable text item the correct time to change the value is ambiguous. All OCS screens solve this problem by delaying execution of commands until the user indicates *accept* by selecting the button in the lower left. Issuing commands is a two-step process. He/she is allowed to change more than one value, edit them and then execute them at a well-determined time by selecting *accept*. Selecting the *cancel* button removes any unexecuted changes and refreshes the display with the current values.

The following figures show how a graphical user interface item will indicate command progress. Figure 5 - 3 shows a text entry item that has been changed. When the user typed a value in the text area, the color changed and a change bar graphic appeared to the left of the item. This indicates the value has been changed but no command has been executed. Widgets in the normal state appear in the *normal* color. The lower filter in Figure 5 - 3 is in the normal state.

FIGURE 5 - 3 **A changed control item in a console screen**

Once the *accept* button is selected, all the changed controls can execute commands in the appropriate systems. The change graphic disappears when the command is accepted and the color of the text or control changes to a command-in-progress color.

FIGURE 5 - 4 **Command completion returning a warning**

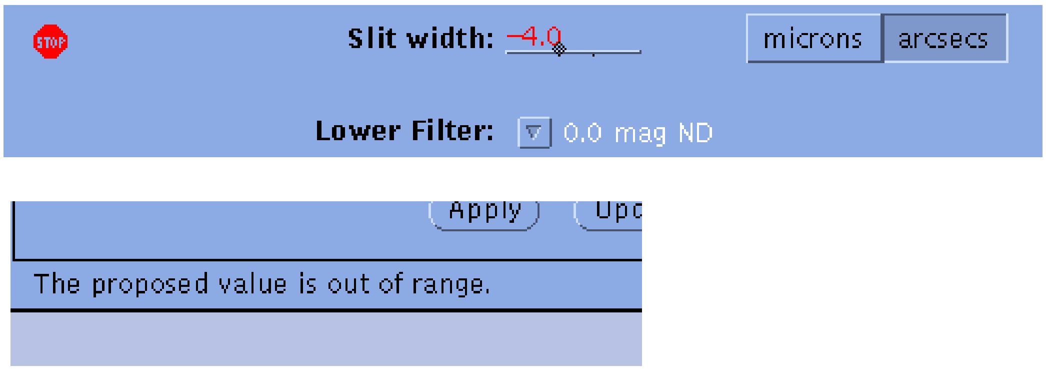


The commands complete with success, warnings, or errors. Error messages are not shown immediately because multiple commands in multiple systems may be in progress and messages could potentially be missed or overwritten. Command success is shown by the button or text returning to the normal state.

Figure 5 - 4 shows a command completion warning. The graphic on the left is now a yellow warning sign. Clicking on the warning sign shows the associated warning message in the message area and removes the graphic.

A command completion error shown in Figure 5 - 5. A stop sign error graphic appears on the left hand side. Clicking on the stop sign shows an associated error message in the message area, removes the stop sign graphic, and sets the text value to its current value. The stop sign remains until acknowledged by the user.

FIGURE 5 - 5 **Command completion returning an error**



This model for incorporating command progress, warnings, and errors is the prototype for the Gemini OCS interactive console screens.

5.2.4.3 VUI FOR PLANNED OBSERVING

The procedure used for changing values in the console screens can be extended for use in planning observations. As discussed in the previous section, two steps are used for interactive console control commands.

1. Make all changes in a screen.
2. Press *accept* to execute the changes in the screen.

These two steps can be slightly augmented to provide for planned observations.

1. Make all changes in a screen.
2. Save proposed values somewhere for execution at another time.
3. At some time in the future, execute the set of saved changes.

The contents of a screen can be viewed as a future *configuration* for the subsystem controlled by that screen. With interactive observing, the configuration is applied when the *accept* button is selected. Planned observations can now be accomplished by providing an application environment for creating, organizing, and storing the configurations that describe the future observations.

The OCS design calls this application the *Observing Tool*. The Observing Tool creates and reads documents called *Science Programs* that contain as much information about a group of observations as possible. The information for the observations comes from the interactive control screens and other more observing-related screens and forms. See "The Astronomer's Interactive/Planned Interface" on page 23 for more information on the Observing Tool. First, the idea of a science program is presented.

5.2.5 The Observer's Science Program

What is a science program? When an astronomer wishes to obtain astronomical data at most shared observatories, he must submit a proposal that outlines his science goals along with any observatory resources and special needs his proposal requires. The Gemini VUI extends the familiar proposal and uses it in the astronomer's observing environment and the operator's system control environment. The observer creates and submits an observing proposal, or *science program*, to the Gemini control system. The science program is used by the control system software to configure the control system, acquire any science data, archive, and possibly process the observer's data.

This section describes the composition of the observer's documents, which are used to guide the operation of the control system and the behavior and function of the observer's visible interface components within the Gemini control system. The intent is to finalize several points in the design:

- Terminology and the meanings behind the terms.
- Basic structure of observing documents.

5.2.5.1 STRUCTURE OF SCIENCE PROGRAMS

The science program is the anchoring concept in the Gemini Visible User Interface; it is the Gemini System Document — the implementation of the Science Program. The format and contents of the science program document will be published and accessible so others can use them and create software tools around it. It will be possible to save Science Program documents as disk files as permanent records of Gemini telescope sessions.

Observers create science programs and edit them to represent their system and environmental requirements. The science programs are submitted to the Gemini Control System. An interface program called the *Observing Tool* is provided as part of the Gemini software to assist the observer in configuring his science programs.

A science program is maintained as a hierarchy of objects along with associated attributes, status, and other information. The objects provide structure in the observing program. A science program hierarchy is made up of:

- Science Observations
- Science Configurations

Science Observations or *observations* are analogous to the traditional observation of classical-observing. They provide the observer with a way to logically group his observing activities in a natural and intuitive way.

A *science configuration* or *configuration* contains all the information required to configure the Gemini Control System to properly do a task or obtain observer data. Observing consists of the process of executing science configurations; the system moves from science configuration to science configuration while doing work on behalf of the user. A science configuration may consist of many of the screen configurations discussed in "VUI For Planned Observing" on page 5 - 14

The science observation and science configuration entities have a small number of defining characteristics. Science observations are *unordered* within a science program and scheduled independently (allowing the system to optimize telescope efficiency). Science configurations within the same science observation are performed *sequentially*, but may be interspersed with configurations from other observations in other science programs. It is reasonable for an observation to consist of a single configuration but an observation may consist of many related configurations.

It may be desirable to ensure that a group of configurations be executed consecutively as a block. An observer has the capability of indicating that some group of configurations within an observation execute *without interruption* (WI). This means that the observer wishes the entire group of configurations be executed together with no other configurations interspersed. The Gemini Control System schedules the group of configurations as a single block.

TABLE 5 - 1 **Characteristics of a Science Program and its components**

Document Component	Attributes
Science Program	contains observations contains program requirements defines an observer's science requirements
Science Observation	unordered in science program contains science configurations and any time constrains between the enclosed configurations (such as minimum times)
Science Configuration	configurations in same observation executed in order Configurations may be marked Without Interruption (WI)

It is possible for observers to specify some Observation/Science Configuration timing information. For instance, users can specify that configuration B follow configuration A by no less than a specified minimum amount of time or no more than a set maximum amount of time. There is currently no plan to allow observers to specify that a Science Configuration be executed at a particular absolute time and date since this would conflict with efficient scheduling and planning of the telescope.

There are cases when observations in a program need to be executed over several months. This might be a long all-sky program or a single observation that must be repeated as part of many observing sessions. The Science Program will allow observers to include this kind of information, but it will be up to the operations staff to ensure that long-term programs are executed appropriately with assistance from the OCS scheduling tools.

Observers can indicate priorities for the Science Configurations in their programs allowing them to emphasize portions of their program over others. This should increase telescope scheduling efficiency and increase the probability of getting the most important data to observers quickly.

Science programs, observations, and configurations can be viewed as information containers. Science programs contain one or more observations, which contain one or more science configurations. Science programs can vary greatly in size and complexity.

The simplest science program has one observation and one science configuration, but a science program could also be a complete night of service observations assembled by a staff observer from programs submitted by individual astronomers. The content of science configurations will be discussed elsewhere in this document.

5.2.5.2 SIMPLE EXAMPLES

The simple, hierarchical structure and characteristics of science programs, observations, and configurations offers a great deal of flexibility to the user when developing his observing session. A few simple examples can show how some common observing scenarios could be executed.

Case 1 — One astronomer, one configuration. The simplest case is where an astronomer wishes to take a single frame of data using a single instrument. In this case, a single observation is created consisting of a single science configuration.

Case 2 — One astronomer, one complete observation. In practice, case 1 is quite rare—even to collect a single science frame in a useful manner requires taking bias frames, sky flats, and so on. However, all of these configurations can be grouped into a single observation, with one configuration for each fundamental operation (biases, flats, science data, etc.). The ordering of the configurations in the observation will be the order in which the system will execute the configurations. Therefore if post-processing depends on calibration frames, the observer can guarantee they precede the science data.

Case 3 — One astronomer, multiple observations. It is more common for astronomers to have more complex requirements, increasing the complexity of the science program. The system provides a great deal of flexibility in how the astronomer might choose to use the telescope.

For example, assume there are 15 science objects to be observed as a part of a program. The most obvious ways of structuring the science programs are:

Scenario A. If the order of observing the objects isn't important to the observer, then the science program could be written as 15 observations, one configuration per object. This gives the scheduling system the greatest flexibility when scheduling observations, as it can arrange the tasks as needed for efficient observing.

Scenario B. If the order is important, but the timing between observations isn't, then the science program could contain one observation with 15 configurations, sequencing the 15 objects. (The system provides a mechanism allowing the astronomer to represent those 15 configurations with a single module that *iterates* across the object positions to produce the individual configurations.)

Scenario C. Finally, if the observer needs to ensure that all 15 objects are observed within a fixed period of time, the 15 configurations can be marked as requiring scheduling *without interruption (WI)*.

Since these methods respectively reflect a potential decrease in telescope efficiency, there may be some cost associated with moving from scenario A towards scenario C.

Case 4 — Multiple astronomers. Adding to the complexity of the Gemini system is the fact that there may be more than one science program available for operation during an evening. Several observers may have one or more active science programs during the course of an observing session. Since observations within programs are scheduled independently, adding additional programs merely increases the number of observations available for scheduling.

Case 5 — Classical/Interactive observing. Serendipity is an important part of the process of scientific discovery and there will be other times when an observer's science can only be accomplished through interactive observing. The Gemini control system supports classical observing through *Interactive Observing Observations*. For the purposes of scheduling, these observations behave in the same manner as other observations. When executing, they cause a special control system behavior allowing the Observer and Operator to work together to control the telescope in the traditional interactive fashion.

These five cases show how the few attributes of science programs, observations, and configurations can be used to construct both simple and complex observing sessions. System aid in the construction of science programs will be discussed later in this document along with several more observing scenarios.

5.2.5.3 THE CONTENTS OF SCIENCE PROGRAMS

This section describes the kinds of information that might be present in the different parts of the science program document.

Science Program. Observers submit science programs describing the observations they need to perform in order to obtain useful information from the Gemini Telescope. In the classic "one observer, one night" paradigm, the science program corresponds to the actions that the observer intends to carry out on the telescope during the night's observing. In the Gemini system, the science program may describe observations for a portion of a night, an entire night or several nights.

Science programs are developed by the observers themselves using an Observing Tool that is part of the Gemini control system. The observing tool collects information about the program that is useful in scheduling the program's observations. A science program identifies the telescope resources the program requires such as:

- instruments required by the program
- detector needs
- other hardware specifications such as required filter sets

A science program also contains information that is related more closely to the scientific goals of the science program such as:

- dark or bright time requirements
- minimal seeing conditions
- start and stop time limits (airmass specifications)
- expected program duration
- observer's personal science configuration priorities

General information on the program such as the names of the investigators, their institutions, and their various addresses would also be contained in the science program.

Observations. The information required in the observation is TBD but could include information required by the system to manage the observation such as the number of science configurations in the observation or the next unexecuted configuration. At this time we are requiring that any observation use at most one of the four instruments.

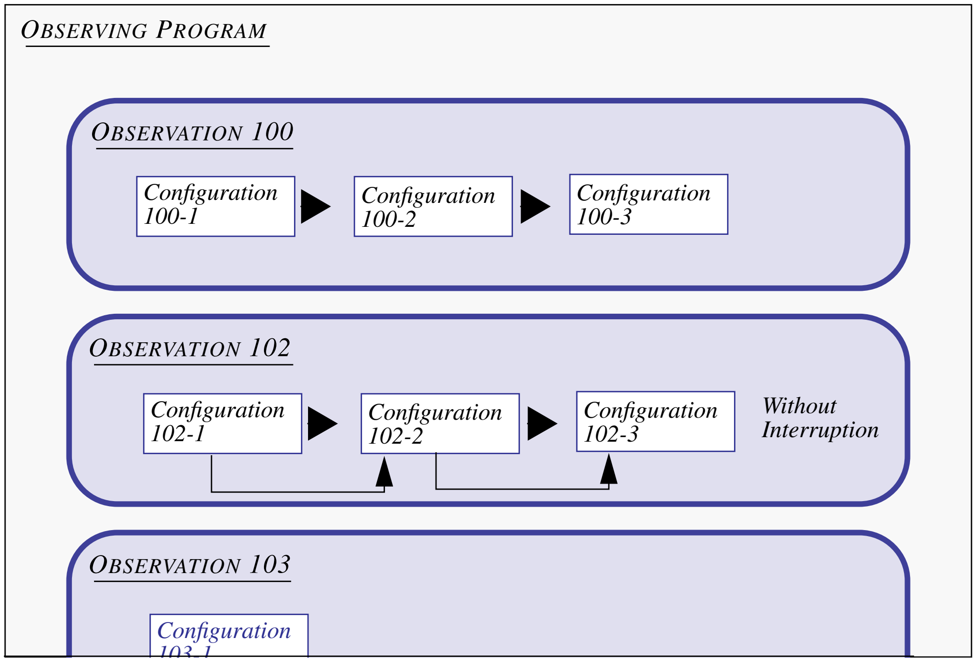
Information specifying time relationships or constraints between pairs of successive science configurations will exist in the observation containing the configurations.

Science Configuration. An observation contains one or more science configurations. A science configuration is a composite of subsystem configurations and configuration state information. A science configuration might include a telescope mount configuration, an instrument configuration, the positions of adaptive optics guide stars, or the name of a post-processing recipe to be run when the observation is completed.

A subsystem configuration is made up of pairs of attribute names and their values. The subsystem determines the names of the attributes and in most cases, they are hidden from the observer by the Observation Tool. The attributes can be viewed as fields in database records where the record is the desired science configuration.

Figure 5 - 6 shows the hierarchical science program structure for a science program with three observations. The observations are unconnected showing they are not ordered. The ordered processing of Science Configurations is indicated by the arrow between configurations. The second observation shows that its configurations are linked to be processed Without Interruption.

FIGURE 5 - 6 **Science Program and its components**



5.2.5.4 SCIENCE PROGRAM SUMMARY

The Science Program is a formal document in the Gemini control system with a published format. The document is created by the observer using and describes the work the user wishes to accomplish with the Gemini telescope.

5.2.6 The User and Operator Graphical User Interfaces

The following sections discuss and present prototype user interfaces for the applications used by the operators and observers. Not all of the features of these applications are laid out here and not all applications have prototypes. The application prototypes demonstrate what the system designers have in mind for these graphical user interfaces. In some cases, screen shots of the prototypes will be shown. In others, the user interfaces and their functions will be described as well as possible.

As discussed in “Visible User Interface Overview” on page 5 - 9, the OCS must provide interfaces for interactive observing and also the planned observing modes. This section discusses more specific high-level issues observer interface issues.

5.2.6.1 VUI PHILOSOPHY

The primary purpose of the VUI is to provide an efficient, usable system for doing astronomy at the Gemini telescopes. Often astronomy interfaces are designed around hardware or the limitations of the hardware systems. The OCS advocates a high-level observing-oriented VUI that puts the needs of observing first and removes as much of the hardware-orientation as possible. The primary guideline in the design of the user interfaces for the Gemini telescopes is simple.

> The high-level design of the OCS and the user interfaces provided by the OCS is user-centered based on the needs of observing and the astronomers and others who will use the system.

This means that OCS design decisions are made in favor of enhancing the efficiency and effectiveness of the observing process whenever possible and will not be directed by the hardware specifications.

5.2.6.2 INTERACTIVE OBSERVING WITH GEMINI

Interactive observing at Gemini telescopes is similar to other sites that use control panel graphical user interfaces to represent the functionality of the telescope and instruments. The operator and observer work together to sequence the operations of the subsystems, but most of the responsibilities for managing the system complexity will belong to the operator(s). It is a guideline of the OCS design to make interactive observing as simple and painless as possible. This is accomplished through an observing-oriented Telescope Control Console and sophisticated alarm, status, and error reporting. See Figure 5 - 17, “A prototype Gemini TCS console,” on page 5 - 39 and “Alarms, Errors, and Status Reporting” on page 5 - 40 for more information.

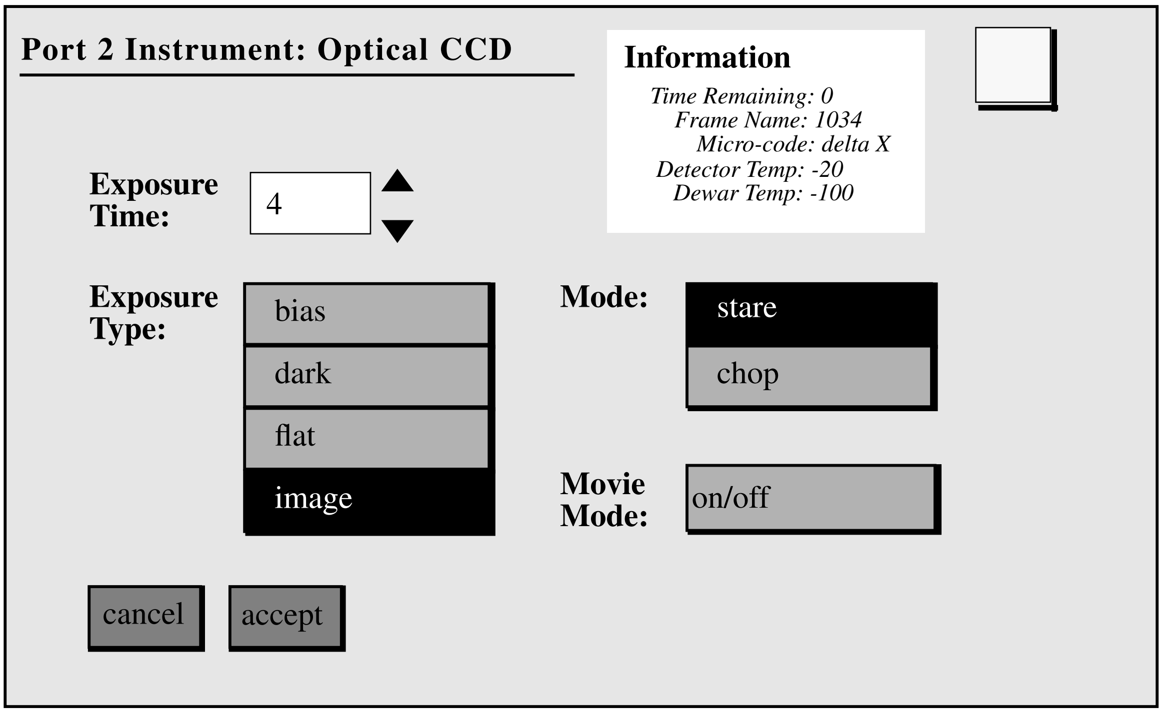
5.2.6.3 THE ASTRONOMER’S INTERACTIVE INTERFACE

The OCS high-level view of console/interactive observing was discussed on page 5 - 9. When observing interactively the astronomer uses instrument consoles provided as part of the OCS. The instrument consoles obey the same look and feel guidelines as the operator console screens.

Astronomers also have access to the Quick-Look and near-line analysis tools, which are part of the Data Handling System (DHS). These tools are used for data quality assessment and for observing related procedures such as auto-focusing and instrument centering, which must span the functions of the Instrument Control System, Telescope Control System, and Data Handling System.

The design and capabilities of the instruments are not well known at this time. Figure 5 - 7 shows a simplified console for an optical CCD showing the *cancel* and *accept* buttons. The instrument would be configured and a data frame generated when the *accept* button is selected.

FIGURE 5 - 7 **An example instrument console**



When console observing, some features of the Observing Tool and planned observing are not available.

- The data and information describing a science frame will only be present in the data files themselves. The external database will have no record of the activities associated with creating the data files.

5.2.6.4 THE ASTRONOMER'S INTERACTIVE/PLANNED INTERFACE

This section presents a high-level view of the OCS interface astronomer's use when doing planned observing. The user interface can also be used as an alternative to console/interactive observing providing some of the advantages of planned observing. We call this new mode *Observing Tool Interactive Observing.*

The Observing Tool (OT) is the OCS application that is used by astronomers and operators to create, store, and manage configuration information in the form of Science Programs, Observations, and Science Configurations. See “The Observer’s Science Program” on page 15 for more information on the Science Program, the Gemini document. The goals and uses of the Observing Tool are described in this section.

The following are objectives of the Observing Tool.

- Provide a single application that is the focus of observing activities.
- Provide an application that improves the observing process by focusing on the needs and requirements of observing.
- Allow the observer to organize his observing program intuitively and interactively.
- Provide a seamless user interface to the Gemini systems and the Scheduling System.

The Observing Tool and the operator’s Scheduling Tool use a data browser/outline metaphor to present the hierarchical structure of the science program, observation, and configuration information. The hierarchical structure of science programs was shown in Figure 5 - 6, “Science Program and its components,” on page 5 - 21.

The science program and its components are represented in the Observing Tool by small images that can be expanded and collapsed to show and hide the components they contain. Expansion of program entities can be continued recursively to explore or browse a program during any stage of its use. This is similar to “outline” mode in many popular word processing applications.

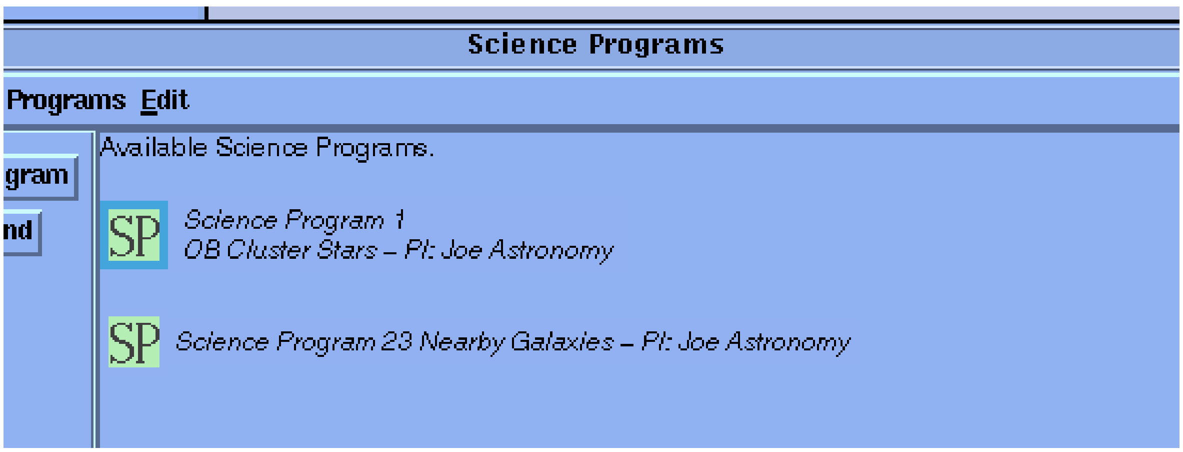
A formal Gemini document, the Science Program, has been created to allow prospective observers the flexibility to plan their observations at their home institutions at their own pace. Following the completion of planning a science program, they can save their observing program in a Science Program document for future reference or for usage as the start of a future science program. When ready, they can submit the Science Program to the Gemini Control System.

Observing Tool Prototype Tour. The OT concept is difficult to convey in words; an annotated example will now be presented. This example is based on a prototype that was created using the TK toolkit. Be warned that some of the details of the text are not consistent—the text in the figures is just for demonstration purposes. Some of the steps in a *real* system would no doubt be different.

In this example, an astronomer (who in the prototype goes by Joe Astronomy or Kim Gillies) has a partially completed Science Program that he has created and submitted to Gemini. He has more than one Science Program in the system, and the program he is interested in checking on is a service observing program. This example shows the steps he would take to check on the program.

He uses the Observing Tool to check on the progress of the program. He starts the Observing Tool at his home site, the OT connects, and the astronomer is presented with a list of Science Programs that he owns that are currently in the system. A view he might see is shown in Figure 5 - 8. This view is called the Science Program View.

FIGURE 5 - 8 **Science Program View shows science programs available to observer.**

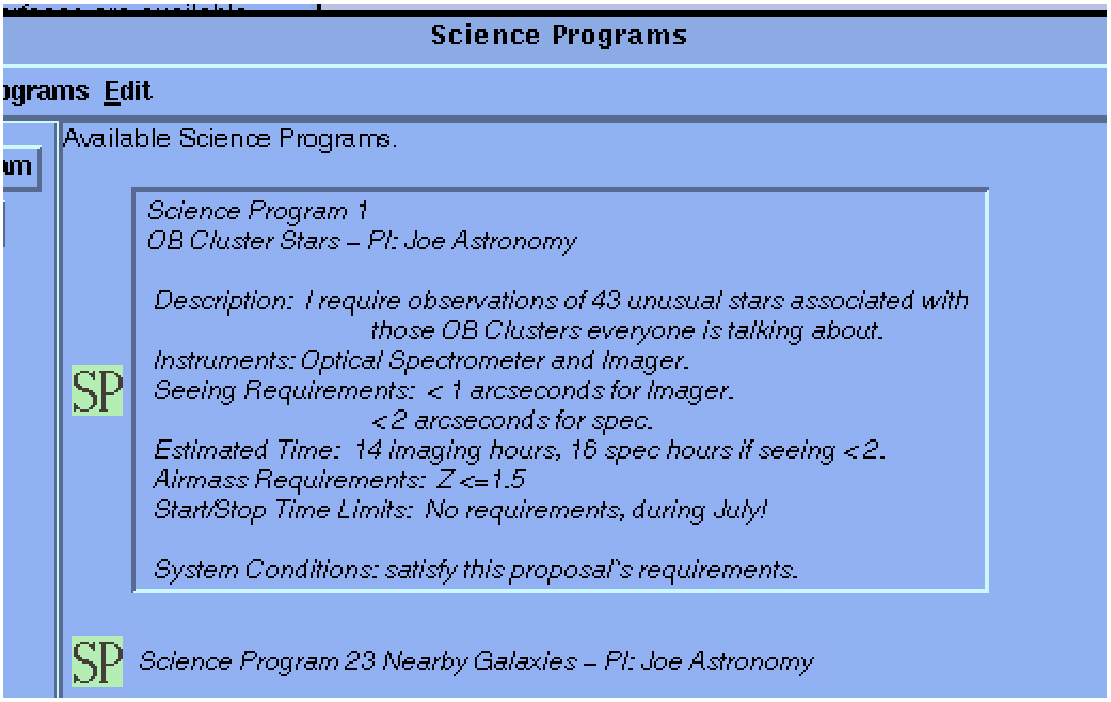


The view shown in this figure is the one that would be used to create, destroy, and cut and paste Science Programs. Science Programs can be saved and read from disk as documents in this window. Programs created locally can be submitted to the Gemini system and Programs already in the Gemini system can be retrieved, edited, and resubmitted to the Scheduling System.

In this view we see Joe Astronomy owns two Science Programs in the system. The collapsed view shows enough information to allow Joe to identify the program he wants. He is interested in finding out about the OB Cluster Stars program since he received e-mail stating that some of his observations were executed in the system recently.

Selecting and expanding the OB Cluster Star program, he looks at the observing requirements he specified for this program. Figure 5 - 9 shows the expanded program displaying example information. This information is stored in the Science Program. The observations in the program would not be scheduled until the requirements in the program are met.

FIGURE 5 - 9 **Expanded Science Program shows program requirements**



Now that he has refreshed his memory of the observing requirements, Joe is ready to take a look at the observations that were completed in the OB program. He double-clicks the OB program and the observation view for the program pops up. Figure 5 - 10 shows the Observation View. During the early prototyping phases, observations were called tasks; tasks are observations in the current nomenclature.

Joe chose to create the science needs of this program as 43 individual observations—one science object per observation. This choice was reasonable since each star was a different telescope target and the targets are scattered around the sky and the entire program will take months to complete. This choice makes it easier to schedule his observations because each observation is a relatively small time commitment and small blocks are easier to schedule than large blocks (a prototype scheduling policy).

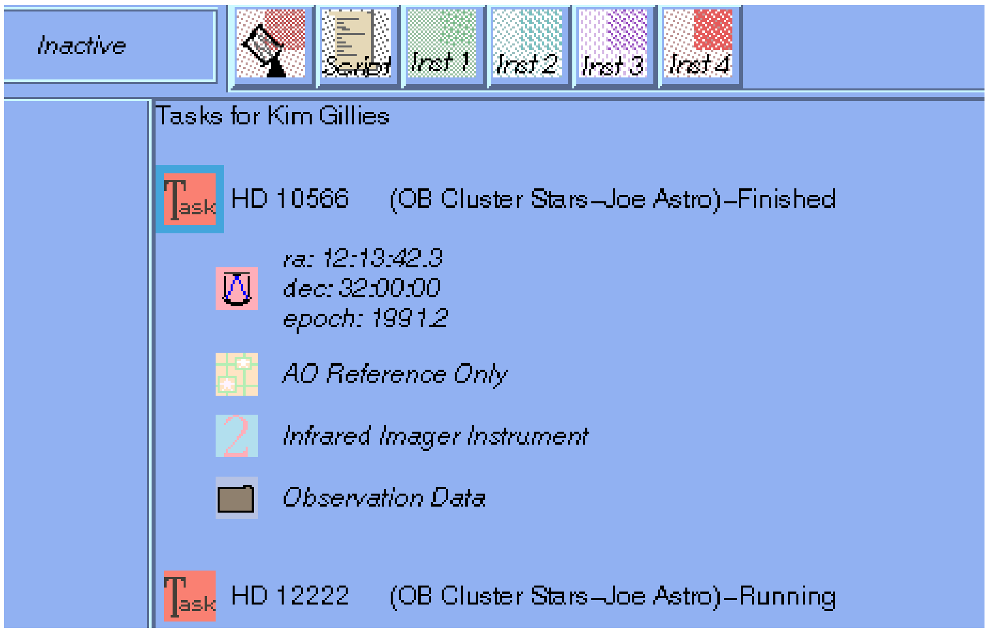
FIGURE 5 - 10 **Expanding the program shows the observations in the program**

The Observation View shows 7 of Joe's observations. HD10566 and HD 3259246 have been completed. HD 13222 is shown to be *submitted* meaning that the observation is scheduled to run (Joe is actually observing at another site so he's up all night.) HD 12222 is actually being observed at the time Joe connects. The others have not yet been observed.

The status messages Finished, Running, Submitted, and Not Submitted are just example status messages designed to show the different stages an observation might be in.

At this point, Joe wants to take a look at HD 10566, so he double clicks the observation icon to expand the observation just like he did the science program. Figure 5 - 11 shows the contents of the observation.

FIGURE 5 - 11 **Opening an observation shows the single Science Configuration**

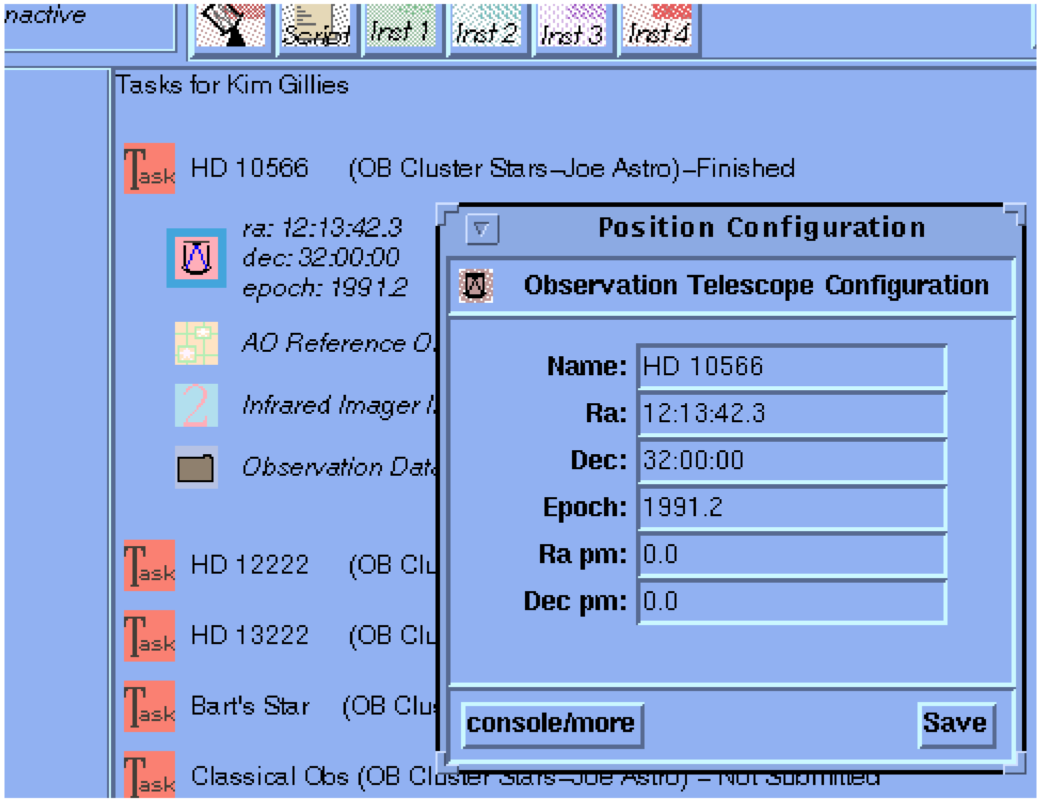


The observation expands to show the list of science configurations that it contains. In this case, the one example Science Configuration is made up of the telescope component, an adaptive optics component, and the infrared imager instrument component. The data that was acquired when the Science Configuration was executed is also represented by the folder image.

When expanded observations contain multiple Science Configurations, the observation is shown to contain a series of Science Configuration icons and descriptions. The Science Configuration descriptions might be library configurations such as flat, dark frame, or mosaic, etc. The example observations contain a single science configuration so the components of each observation are shown when the observation is expanded.

Figure 5 - 12 shows what happens when a configuration component is selected and expanded. The example shows the name and position information associated with the telescope configuration. A real telescope configuration would be more complicated.

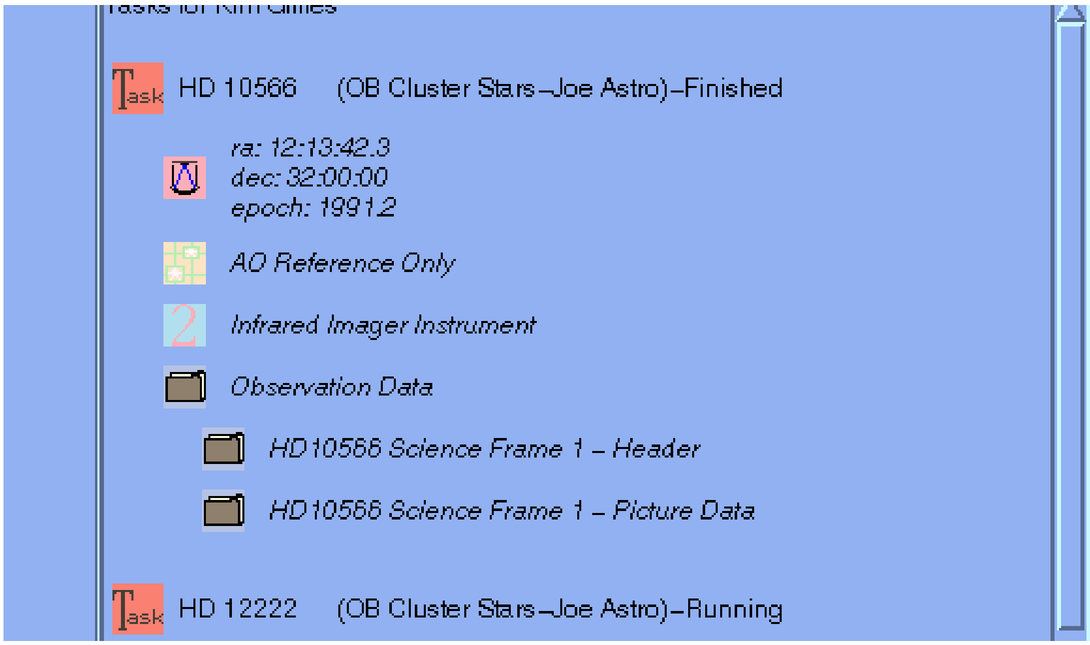
FIGURE 5 - 12 **Expanding a configuration shows the attributes of that configuration**



These component views are the screens that are completed to create or modify Science Configurations. New configurations can also be created by copying and pasting other observations and configurations or by including observations or configurations from a prepared library. Each of the component icons can expand to show its attributes when double-clicked.

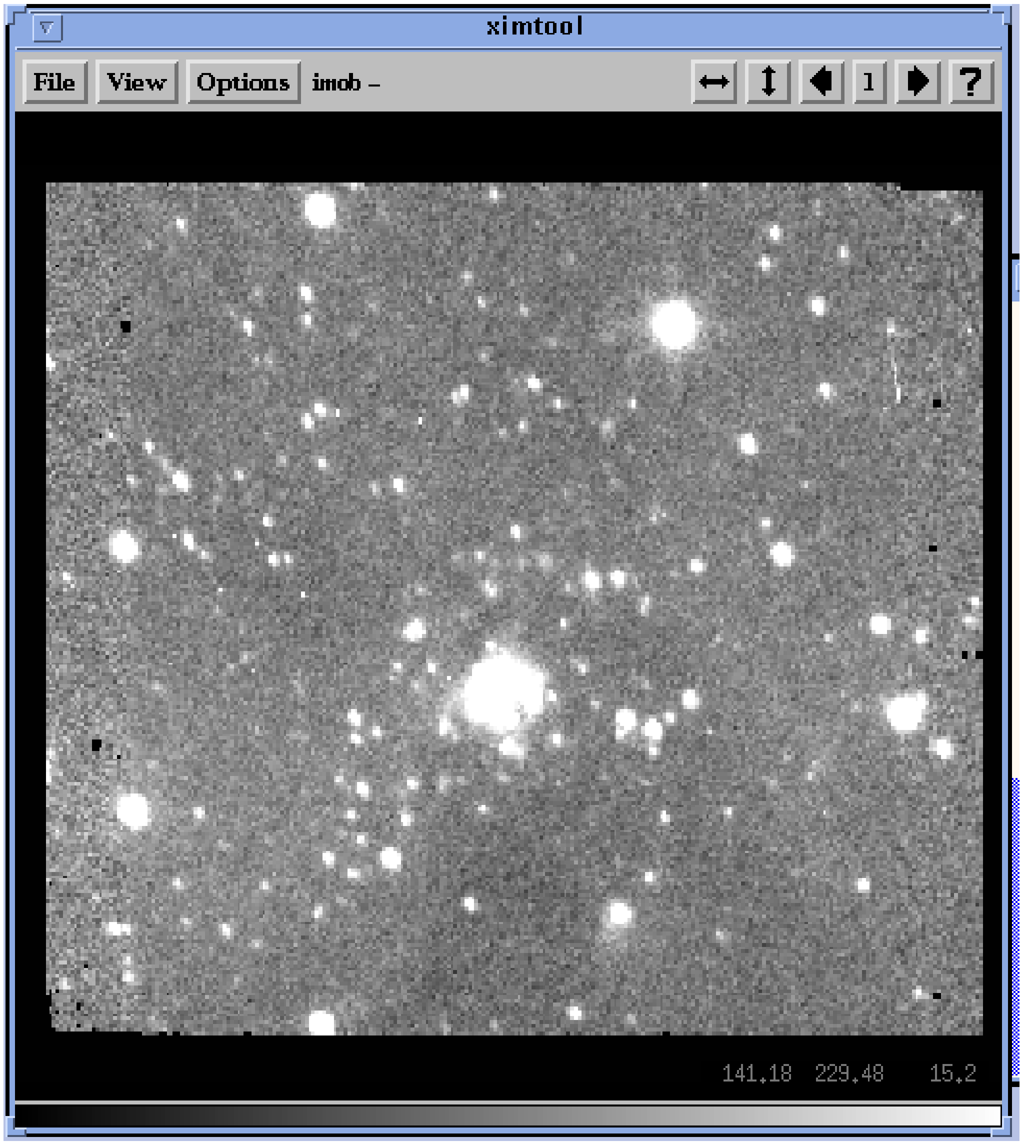
Users add the configuration components that are unique to their particular application to their Science Configurations. However, it will be possible to add components from any interactive control screen to a Science Configuration. This capability is represented in the prototype screens by assuming a drag and drop feature. Each interactive control screen, such as Figure 5 - 2, contains a drag site shown in the upper right by a shadowed rectangle (the drag site). Users can configure an OCS console, drag a component configuration from a console, and drop it into a Science Configuration.

FIGURE 5 - 13 **The observation is completed so its data is available in the data folder**



Joe Astronomy wants to review the data obtained and he double-clicks the data folder image. Figure Figure 5 - 13 shows that the HD 10566 data folder expanded resulting in a single data header with picture. Double-clicking the header or picture image displays the header or picture using a Data Handling System Quick-Look application or an image display application that is part of the data reduction system. The picture data would then be transferred to Joe's site taking into consideration the available bandwidth. Figure 5 - 14 shows the OB Cluster image displayed on a typical reduction system display. Joe now uses the facilities of the data handling or data reduction system at his home institution to examine his data.

FIGURE 5 - 14 **Clicking the data icon shows the associated image**



This completes the tour of the Observing Tool prototype. The point of walking through this scenario was to show how an Observing Tool program could work to represent a simple Science Program, how the OT is used, and how the OT interfaces to the interactive consoles.

Observing Tool Design Features. There are many features that should be available in the OT to maximize its usability and to take advantage of the experience users might have with similar PC-based programs. The following are features that the designers feel are important or essential to a good, usable Observing Tool. The actual feature list should be obtained by interacting with observers.

- The representations of a Science Program and its components can be created, selected, copied, deleted, and moved in the ways familiar to users of other similar computer packages. It should be possible to use copy and paste between Science Programs, Observations, and Science Configurations.
- Allow the user to determine what attributes will appear in his data headers. This is a natural feature that should be part of the OT. Users indicate which of the attributes in various configuration components should be saved. This feature allows users to even control what engineering data (such as voltages or temperatures) will appear in their headers if they desire. There will be a default group of attributes that appear in headers for any instrument/application so that only some observers will use this feature.
- The OCS uses the Science Program to create and define the observations and to record information concerning the progress of the observations in the Gemini Control System—the OT provides the observing log as well.
- Each Science Program created by the OT will be a stand-alone document that can be saved on disk and opened at a later time.
- The Observing Tool should understand as little as possible about the systems it is configuring, particularly with instruments since they will be provided by outside groups. The drag and drop protocol and the representation of configurations as database records allows this. See “High-Level Observatory Control System Concepts” on page 5 - 1 for more in-depth OCS information.

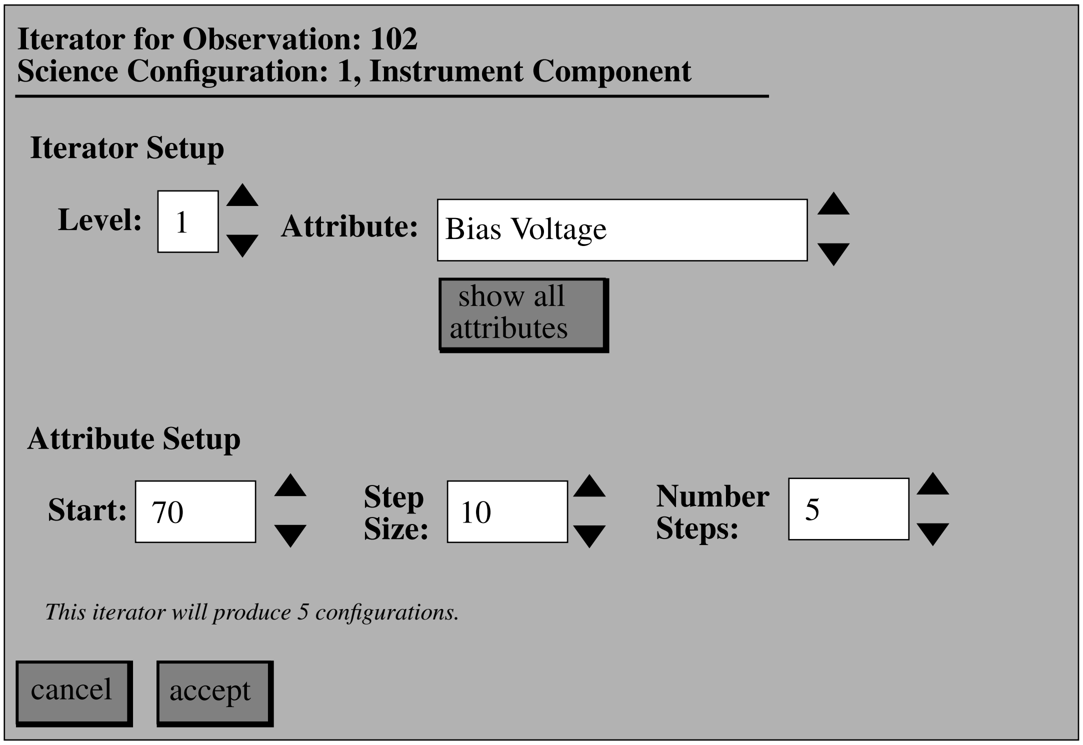
  The Observing Tool should know as little as possible about the Data Handling System implementation. Users will be able to include Quick-Look configuration components in their Science Configurations.

  Users can also include a data reduction component in their Science Configurations that defines any processing steps required for particular frames. These configurations are to be independent of any particular data reduction system to allow users to use the system with which they are familiar.
- The Observing Tool should allow the observer to create a *plan* much like the scheduling tool. This allows him/her to estimate how much time his/her observations will take.

Iterators. The Observing Tool provides the concept of a Configuration Iterator. An iterator is an entity that can modify one configuration to produce another, slightly changed configuration. Iterators operate on a part of a base configuration to produce a dynamic part of the base configuration For instance, an IR engineer might like to take a set of images while varying a detector voltage. He creates an base Instrument Configuration and operates upon it with an iterator. The iterator interface would allow him to set a range of values for the interesting value.

A prototype general iterator interface is shown in Figure 5 - 15. The purpose of this iterator is to allow any public attribute in the system to be varied while taking data. The *show all attributes* button commands the instrument to supply a list of all its public attributes. The *level* of an iterator can be set when more than one iterator is applied to a base configuration to order iterator changes. All the iterators with identical levels change values at the same time and higher levels change more rapidly than lower levels. This provides a nested loop capability. All second level iterators execute to completion before first level iterators change and once again execute all second level iterators. An image mosaic could be generated using a level one declination offset iterator and a level two right ascension iterator; more probable is a Mosaic iterator.

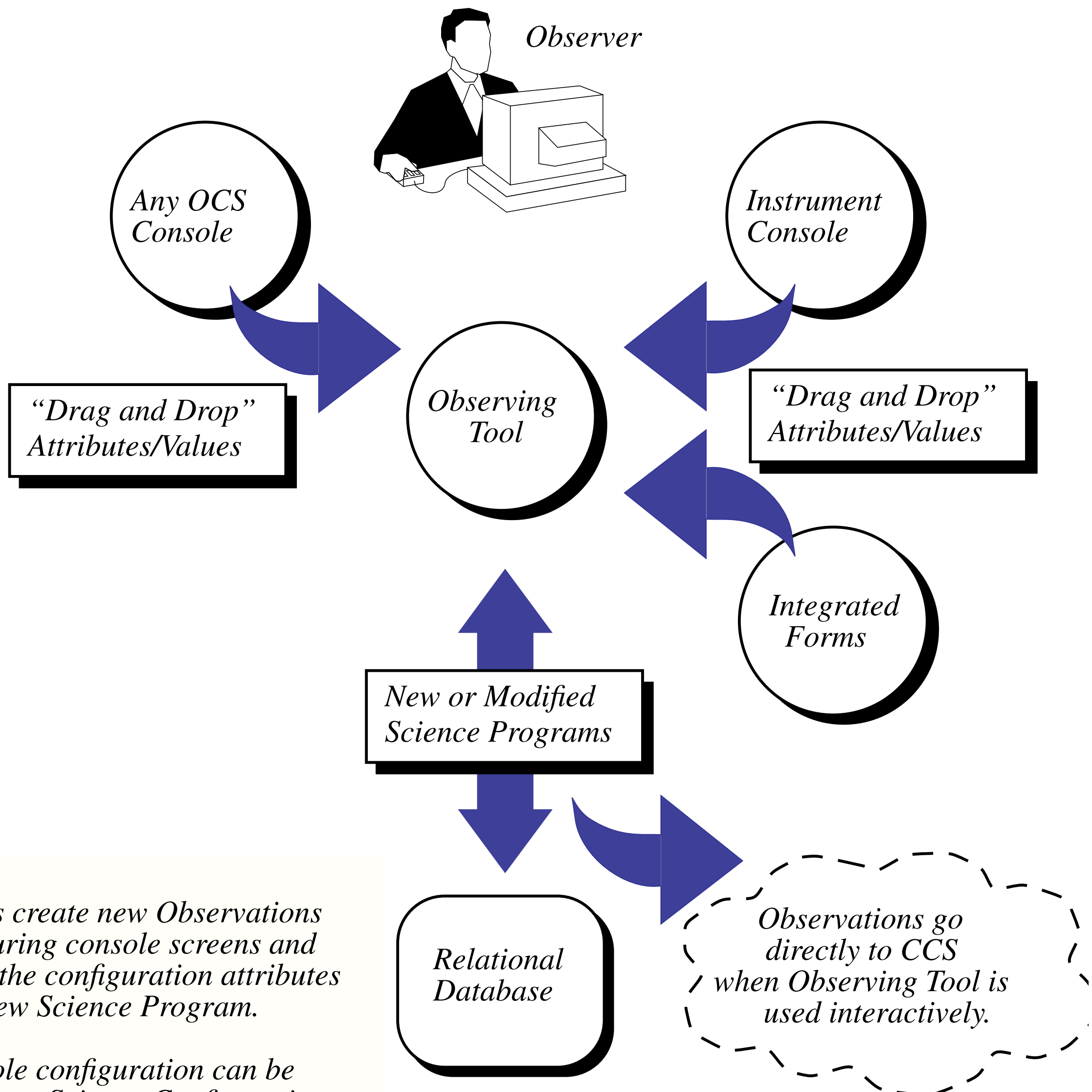
FIGURE 5 - 15 **An Observing Tool iterator prototype**



Observing Tool Interactive Observing. It is possible to imagine using the Observing Tool interface to observe interactively. In this scenario, users create a Science Program and create Science Configurations one-at-a-time immediately submitting them to the Gemini Scheduling System. This approach provides the astronomer with the interactive decision of what happens next along with the data management features of the Observing Tool. The user could queue the activities of the GCS by submitting more than one Science Configuration to the Gemini Scheduling System.

This scenario, is in fact how the OCS implements one form of interactive observing. The observer's science program contains an Interactive Observing Observation that indicates his resource requirements and time requirements. When the observation is executed, the observer creates Science Configurations on the fly and submits them directly to the control system. Or, he interacts directly with the consoles and loses the data management features of the OT. Figure 5 - 16 on page 5 - 35 shows a summary of the activities observers and staff use to create and modify Science Programs in a block diagram form.

FIGURE 5 - 16 **Creation and modification of Science Programs within the OCS.**



Observers create new Observations by configuring console screens and dragging the configuration attributes into the new Science Program.

Any console configuration can be included in a Science Configuration.

Previously saved Science Programs can be modified from within the Observing Tool.

5.2.6.5 THE OPERATOR'S INTERACTIVE INTERFACE

The OCS high-level view of console/interactive observing was discussed on page 5 - 9. The OCS interactive operator interface consists of a number of subsystem console control screens, a TCS console, an Observing Tool, and the Scheduling Tool. Many of the OCS console screens are designed to provide control and status for the many telescope subsystems. They will be primarily for the operators of the telescopes and will be of little interest to most observers.

Observing consoles have different goals than engineering consoles and they present a different level of detail in their interface. Observing consoles provide access to the public, observing-oriented data provided by a principal system. Engineering consoles have access to features, information, and controls provided by a principal system for commissioning, testing, and diagnosing problems and failures. The design assumes that observing consoles will not be the same as the engineering consoles.

The prototype screens developed for the consoles are distributed throughout this document in their related subsystem chapters. Table 5 - 2 on page 5 - 37 shows the locations of the prototype consoles. Please refer to the referenced chapters to view the prototypes for the individual console screens.

Operating a Gemini telescope during the operational/maintenance phase will be a mixture of interactive activities mixed with times when the Scheduling System manages much of the telescope activity. The following section briefly discusses the main operator console since it is the central operator control console.

The TCS Console. The TCS console is designed to be the focal point of operator observing control. Unlike the majority of consoles that focus on the control of a single subsystem, the TCS console spans all principal systems with the goal of providing interactive control for the majority of the frequently-used telescope functionality. This console along with the operator's Observing Tool and Scheduling Tool constitutes the operator's primary tools.

The TCS console attempts to concentrate the major telescope controls on a main screen. Secondary controls, controls that are used only occasionally during an observing session, are located in a popup screen with a common user interface. Figure 5 - 17, "A prototype Gemini TCS console," on page 5 - 39 shows a prototype TCS console. This prototype is based on the actual control console for the 4 meter telescope at NOAO's Kitt Peak.

TABLE 5 - 2 **Prototype console screens and their locations in SDD**

Screen Name	Location
TCS Control Console	Details of the Telescope Control System
TCS WFS Signal Routing Console	Details of the Telescope Control System
Weather Monitoring Console	Details of the Telescope Control System
Structure Temperature Monitoring Console	Details of the Telescope Control System
Mount Control Console	Details of the Mount Control System
Cassegrain Rotator Control Console	Details of the Instrument Rotator Control System
Primary: Active Support System	Details of The Primary Support Control System
Primary: Passive Support System	Details of The Primary Support Control System
Primary: Air Support System	Details of The Primary Support Control System
Secondary: Alignment System Console	Details of the Secondary Control System
Secondary: Chopper System Control	Details of the Secondary Control System
Secondary: Tracker Control Console	Details of the Secondary Control System
A&G: Main Console	Details of The Acquisition and Guiding Control System
A&G: WFS 1 Console	Details of The Acquisition and Guiding Control System
A&G: WFS 2 Console	Details of The Acquisition and Guiding Control System
A&G: Calibration Sensor Console	Details of The Acquisition and Guiding Control System
Enclosure Console	Details of The Enclosure Control System

Across the top of the screen in Figure 5 - 17 are buttons to bring up an interface to an automated screen for opening and stowing the telescope systems. The *Inspector* button opens the standard interface for changing secondary parameters. It will not be discussed here.

Across the upper middle of the screen are the telescope position controls: the next target, the current target, and a menu of previous targets or position history. Coordinates are dropped onto these areas and once the arrow is selected, *go there* will send the telescope and any guide probes that have coordinates to their new targets.

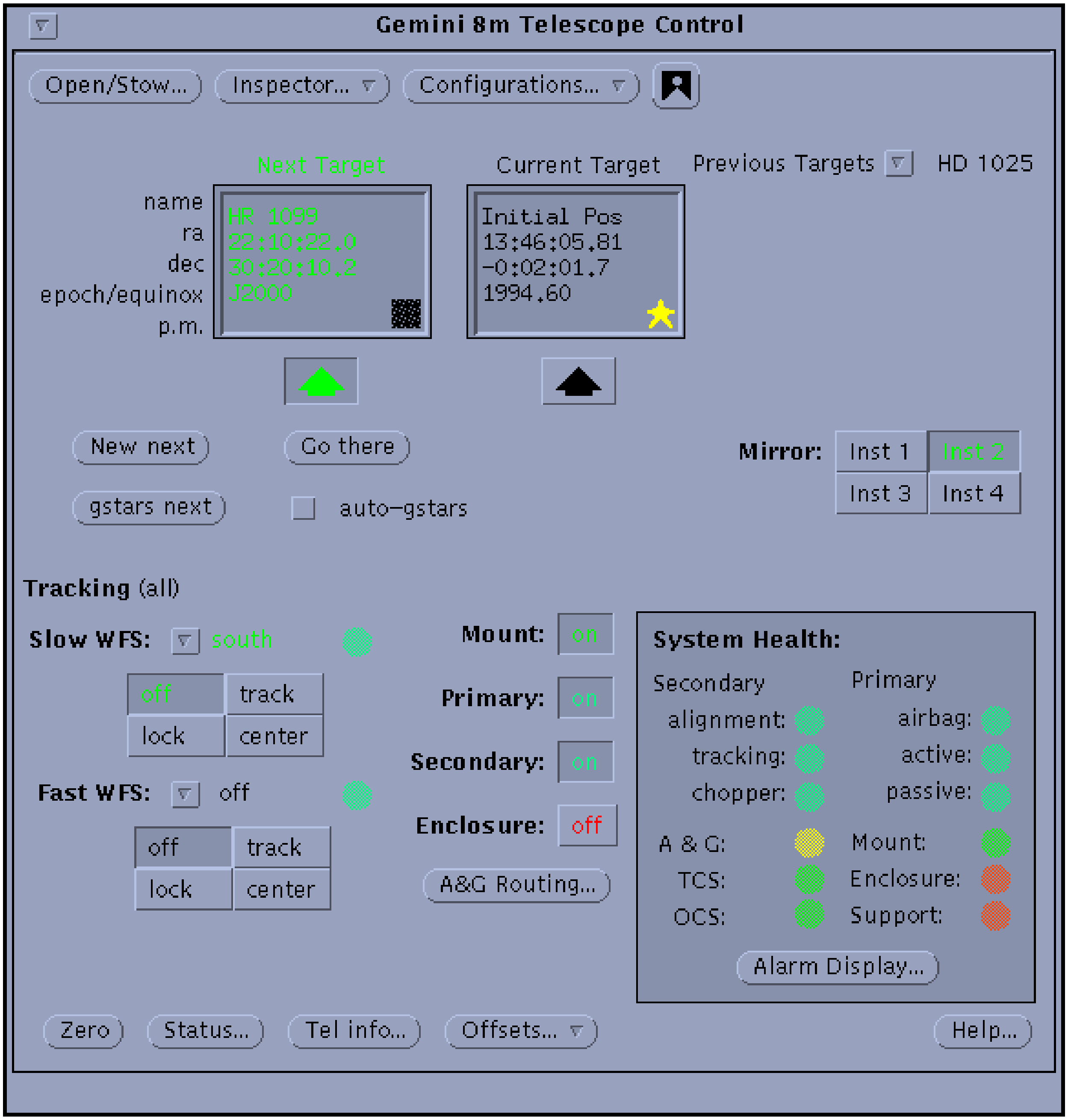
The yellow star indicates that there are guide stars associated with the telescope target. *Gstar next* and *auto-gstars* provide automated access to guide star catalogs when it is necessary to find guide stars for objects at the telescope. The TCS console can also accept coordinates from catalog programs and the Observing Tool.

In the central part of the screen is the control to move the mirror in the cassegrain rotator to one of the four available instrument ports. In the lower left corner are controls for the wave front sensors and controls for subsystem tracking. The colors enhance information on status as discussed in "Console Interactive Graphical Control Semantics" on page 5 - 12.

The "System Health" in the lower right of the TCS provides a simplified overall diagnostic view of the operating system. Subsystems report their overall *health* to the OCS. The operator must only check for green colors to ensure that the system is healthy. The information in this part of the screen is integrated with the error and alarm monitor screen. See "Alarms, Errors, and Status Reporting" on page 5 - 40 for further discussion of the OCS treatment of alarms and errors.

This prototype TCS console is an example of cross-subsystem control that will be present in the TCS, but not in the interactive consoles. More views of the prototype TCS can be seen in "Details of the Telescope Control System."

FIGURE 5 - 17 **A prototype Gemini TCS console**



5.2.6.6 OTHER OPERATOR FUNCTIONS AND APPLICATIONS

It is known at this time that there are a number of other features and functions that operators will need.

Access and Privileges. The Gemini Control System is a distributed system and observers may be at remote sites. This requires that portions of the software system be accessible from the Internet. The OCS must provide the ability to control access to the system. Since different observers will have access to the telescope and its systems during an observing session, access must be dynamic, changing access as the control system operates.

Unrelated users with simultaneous access will have a capability to interact with parts of the system but not with others. The capability is their privileges or permissions and will also change dynamically as the system executes.

There is no access/privileges user interface at this time.

Resource Allocation. The Gemini system is divided into several subsystems. Some resources, like the mount or telescope beam, are shared among the concurrent users. At one time there could be four teams working concurrently on a telescope; one team on each instrument port.

Managing resources is a separate issue from access and privileges. Resource management is taken care of by the Observing Database. See “The Configurable Control System” in Chapter 10 for more details.

Preventative Maintenance and Operator Reminders. We believe it is useful to provide operators and staff with the ability to post reminders. This could be a reminder to calibrate the primary mirror in 3 hours or a message to fill a dewar. This functionality will be provided in a TBD way.

5.2.6.7 ALARMS, ERRORS, AND STATUS REPORTING

The OCS provides a system-wide visible interface for alarms, errors, status, and health. The term *system soundness* will be used to refer to alarms, errors, health, and status together. The concepts behind these four terms are related but provide different functionality in the control system.

Health. Health is the highest level of status presented by the visible interface. It is defined to be a component’s well-being as determined by the component itself. Health can have one of three values: *good* means *I am normal, everything is okay*;

warning means *I'm operating but not normal*; and *bad*, which means *I'm not operational*.

Health is defined recursively—a component is in good health only if all the components that it manages and relies upon are healthy too. If a telescope is made up of and manages several subsystems and it can't work properly if they are not working properly, then the telescope is clearly not healthy when those items are not healthy. By logically combining the health of all systems it is possible to find out if the entire telescope system is healthy.

Severity. Alarms and errors can have one of two severity levels in the GCS. *Warnings* are alarms or errors that subsystems feel are important and should be brought to the attention of the user. A component should be able to continue working in spite of any events that cause warnings. Warnings can be acknowledged by users allowing operations to continue. Acknowledging a warning will keep the same warning from nagging the user with the same warning over and over (within reason).

A *failed* severity indicates that a component has moved to a state that will not allow it to continue operations. The user or operator must solve the problem before continuing operations. Failed alarms do not go away when acknowledged since by definition they must be attended to before observations continue.

The severity of the health of a system can only be one of three values, but the number of contributors to a system's health severity is also important information. Any status/alarm/health interface should keep track of the number of contributors to a system's health state and should notify the user when the number of contributors changes—even when the health severity itself does not change.

Errors. Errors can occur when components are requested to perform some activity and are posted to indicate that a problem occurred during the execution of a command or operation. The cause of a failure is a fault.

Each commanded item will have an error message that will contain the fault when possible as well as a description of the failure itself. Error messages should be meaningful and user-oriented; they should not be cluttered up with *programmer mumbo-jumbo* unless the failure is software related. Errors can have *WARNING* or *FAILED* severity.

Errors are not status and the two must remain independent throughout the system. If a motor is commanded to move, its status indicates whether or not it is finished through a *busy* or *done* value in an attribute. Once it is *done*, it can be checked for errors in an error attribute.

Successful command completion is indicated by an OK error value.

Alarms. Alarms are the asynchronous notification of occurrences in the control system that are critical or important to proper operation. For example, an alarm should occur if a power supply fails in the A&G system. Like the power supply, the items monitored by alarms are often not directly related to the observing activity but can have a profound effect on observing. As with errors, alarms can have *warning* or *failure* severity.

Status. Status consists of the values various hardware and software devices present to the other hardware and software systems. A component's *state* includes the attributes related to the purpose of the component, but will also include system related status attributes such as a component's health, operational state (busy, done), and error state.

Relationships. Health, alarms, status, and errors are not always independent although the GCS attempts to isolate them as much as possible. For instance, a component's health is related to alarms. The A&G health would be *bad* if a wave-front sensor power supply failed and generated an alarm. However, a component can be healthy and still generate a command error. If an A&G probe is requested to move to a position and it can't get there, it may produce a *failure* command error, but still be in good health.

To simplify the overall design, a health alarm must be defined in terms of the available set of principal system items that produce status and alarms. In other words, a bad or failed health determination must be traceable to one or more status items that are probably in an alarm state. This is not a limitation on the system; it merely means that all contributors to a system's health must be present as status items.

The details of these relationships are application specific. The developers of principal systems should use the definitions of health, alarms, and status as a guide to providing the OCS with information.

The actual status items for each principal system will be defined as part of each principal system Work Package and will be based on the control system needs and the kinds of information that must be available to operators and observers.

A Prototype Error/Alarm User Interface. Figure 5 - 18 shows a prototype OCS error/alarm monitor screen. This user interface is a prototype and only presents ideas on how the different parts of system soundness can be used

The health of the system is presented in the operator's TCS interface in Figure 5 - 17. The operator would notice health changes and move to the error/alarm monitor screen to check for the.

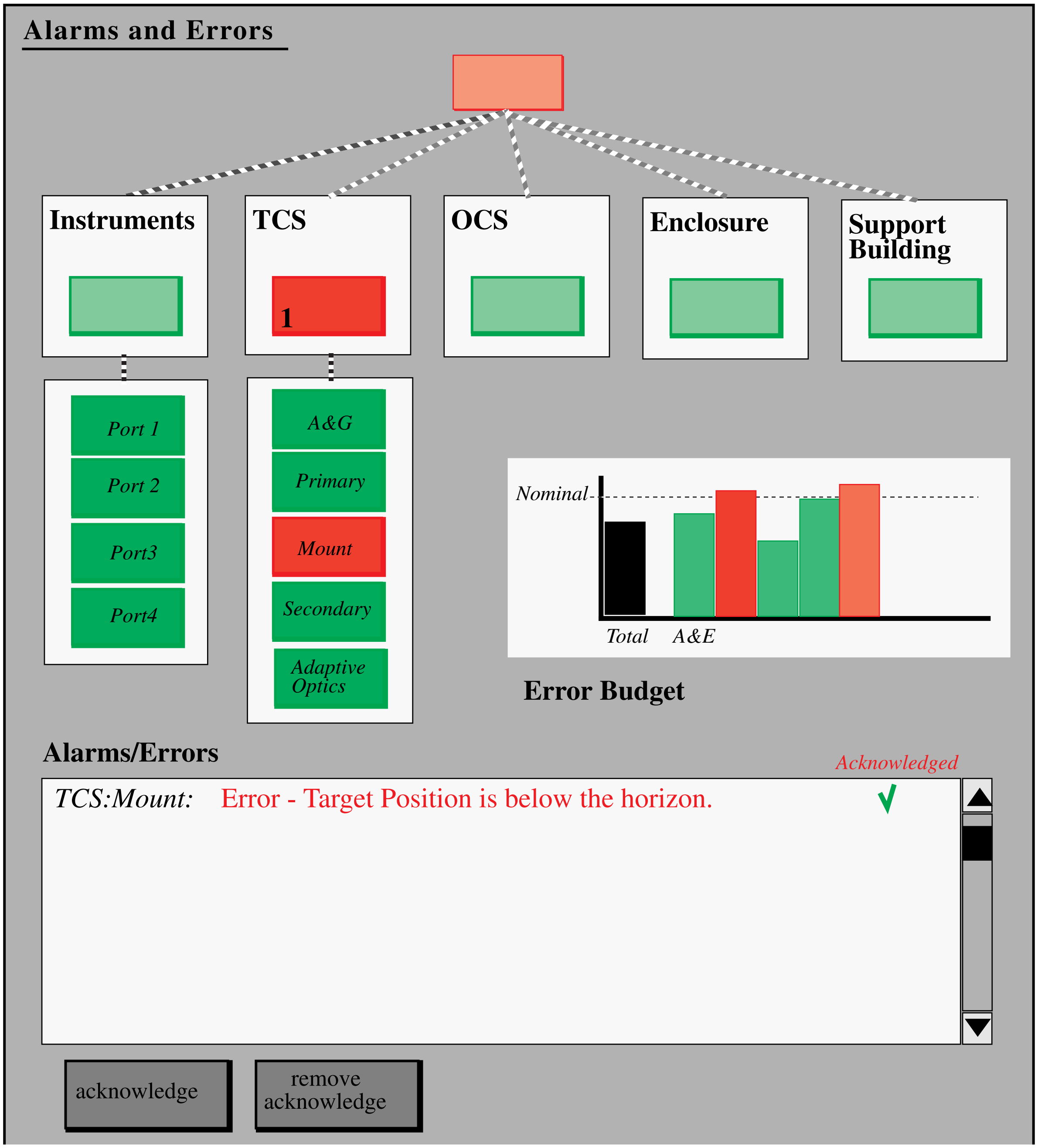
The top of the error/alarm screen shows a shallow representation of the system hardware hierarchy. The box colors indicate whether a component has posted errors or alarms the number inside a box is the number of contributors to the system's health state when that state is not ok. When components are made up of other components (like TCS) they are decomposed again.

Clicking on the rectangles (buttons) turns on or off the display of alarm and error messages from a particular subsystem. The text area below shows the messages and allows the user to acknowledge warnings.

In the middle of the screen is a prototype Error Budget monitor. The error budget total is comprised of amounts from several components. The histogram shows the components normalized to the values they must contribute for acceptable operations.

This interface is just a prototype. Careful integration and presentation of health, status, and alarms information will be part of the OCS Work Package.

FIGURE 5 - 18 **The prototype OCS alarm and error monitoring screen**



5.2.6.8 VISIBLE INTERFACE SUMMARY

This section of the high-level OCS document has presented the methods and concepts observers and staff will use when interacting with the Gemini Control System. Prototype interfaces and concepts related to the ways the user interfaces should work were presented with the goal of guiding future work.

5.3 A USER VIEW OF THE SCHEDULING SYSTEM

The Gemini *Scheduling System* is the part of the control system that provides the operator (in this section, an operator is any staff member charged with scheduling telescope operations) with the ability to choose and order configurations for execution in the Gemini system. The operator can choose one "next" configuration or he can choose entire observations and programs for execution. This discussion assumes a Gemini Control System that is operating in the operational/maintenance phase. In this phase, planned observing makes up a large component of the observing activities.

The scheduling system is not intended to provide fully automatic scheduling and running of the telescope. Instead, the scheduling system functions as a *technical assistant* to the operator, providing sufficient information to allow him/her to make the decision of what to do next. This means that the scheduling system functions as a *decision support tool*, not as an artificial intelligence system. The goal is to provide the operator with a tool that simplifies the job of scheduling the telescope while leaving the decisions on that control in the hands of the operator.

5.3.1 Observing Plans

For optimal telescope usage it is important for observers to design their programs in as much detail as possible ahead of the time their observations are to be made. The operator's job is to produce an optimal night's observing using the information provided by the observers in their programs and the real-time software and hardware systems. This operator activity is called *planning* and the result of the process of planning is an *observing plan*. It is possible to generate plans before they are executed in the actual hardware/software system as well as during the observing session. Observers can also generate plans but their plans would probably not be used for observatory observing plans.

The software tools must allow staff members to create telescope plans, monitor an executing plan's progress, and alter a plan based on changes to science programs or the real-time system information if the plan is executing. The process of choosing the "next" configuration is simplified when the operator or staff has provided an observing plan for an observing session.

The operator uses the scheduling portion of the software system to select one or more science programs for use during an observing session. The observations contained within the selected science programs are then candidates for the night's observing. The operator uses the information within the science programs and observations, and the recommendations of the scheduling system visible interface to schedule the configurations within the observations.

This section describes the design and operation of the scheduling system from the operator's perspective. The description of the scheduling system is based on the actions a telescope operator would be required to take in order to accomplish the goals of the science programs provided by users.

5.3.2 Observing Modes

The Gemini system must provide several observing modes, or ways for users to interact with the telescope system. These modes are called interactive observing, queue-based observing, service observing, and remote observing. See "Operational Phases" on page 3 - 1 for more information. The Scheduling System must allow all these modes and support them transparently.

Interactive Observing. Traditionally, astronomers use telescopes in a fully interactive manner with direct access to the telescope and instrument controls. In this mode, the observer determines the "next" system activity interactively in real-time. The scheduling system must allow interactive use of the telescope systems according to the will of the operator.

Queue-based Observing. In this mode, the sequence of science observations to be performed during a night of observing is planned. The scheduling system must be capable of executing an observing program automatically.

Service Observing. With service observing, the astronomer develops a science program in concert with an on-site Gemini science observer who then acts in place of an astronomer. This allows more flexibility in scheduling and executing the astronomer's program. The on-site observer can provide any human interaction that may be required during the execution of the program.

Remote Observing. Remote observing is different things to different people. Observers may wish to monitor the progress of their science program *off-site*. The scheduling system must not inhibit monitoring from a remote site or other remote observing activities.

5.3.3 Observing Pool

Observers create science programs and submit them to the *Observing Pool*, the repository of science programs within the Gemini System.

The planning system provides the operator with the capability of viewing and interacting with the set of science programs within the observing pool. The Observing Pool presents information on the science programs as indivisible objects—they can only be manipulated as entire science programs.

Rule sets that describe the criteria and policy used to determine science program applicability at a given time can be used to select programs for the *Selected Program Pool*. The rules sets might be based on:

- *Instrument Availability* — does this program use some instrument that is currently available?
- *Telescope Allocation Committee Priority* — is this a high priority project?
- *Objects Available* — will some targets used in the program be available during the night?
- *Program Open* — has the program been opened? Is it partially completed?

A qualified staff member will be able to override the recommendations of the rule set. It will be possible to find out why a program is not recommended.

As with all databases, the operator can view and ask questions about the contents of the Observing Pool that will help him/her select science programs. The operator may instruct the system to present or prioritize the science programs in the Observing Pool using a variety of possible criteria. This functionality is in addition to the rule set functionality. For example, individual queries might be based on:

- *instrument requirements* — show me all programs that use only the optical CCD.
- *seeing requirements* — show me all programs that don't require seeing better than 2 arcseconds.
- *policy requirements* — show me all programs that use only one instrument.
- *time requirements* — show me all programs that require observations at the beginning of the night.

The scheduling system itself may reduce the set of available programs by eliminating those programs that have no chance of being selected during a session. For example, science programs would be inactive if the current date wasn't within the date range assigned for a program by a Telescope Allocation Committee. The operator would have the capability of viewing all programs and any reasons they might be inactive. See "Time Allocation Model" on page 3 - 8 for a discussion of scheduling assumptions.

5.3.4 The Program Repository

A relational database will be used to store long-term information such as proposals, science programs, logs, and recent data. The Observing Pool will be part of the information found in the external relational database.

5.3.5 Selected Program Pool

Once a science program is selected, its observations become available for viewing, inclusion in plans, or execution in the telescope system. The *Selected Program Pool* is this dynamic set of observations.

5.3.6 Observation Planning

At the second step of the planning process, a staff member needs to choose and order observations from science programs. With service observing for instance, observations from different science programs need to be combined and ordered to produce a plan for an observing session ahead of time. A plan is very similar to a science program but might include additional plan-related information such as the estimated start times for the configurations in the plan.

As with science programs, the operator may instruct the system to present the observations using a variety of criteria such as:

- instrument requirements — what instruments are required?
- seeing requirements — what observations are okay with the current seeing?
- policy requirements — who knows?
- time requirements — are there any observations that should take less than 30 minutes?
- sky location — are there any observations near this position?
- observation owner — show me all of Joe Astronomer's observations.
- observations country of origin — show me all observations belonging to observers from Argentina.

The results of the view requests can be used to choose observations for inclusion in a plan or for immediate execution in the Gemini system.

When all observations have been added to a plan they can be saved for later use. For operators, the plans will appear in the Observing Pool along with observer science programs.

5.3.7 Validity Checking

Viewing observations according to various criteria is useful for selecting prospective observations, but it may be necessary for the planning process to allow more extensive analysis of potential observations and their programs for errors and inconsistencies. The operator must be able to ascertain that an observation can be successfully executed as suggested in the program at the planned time. The process of analyzing observations and programs is called *validity checking*. The set of validity checks that check for errors, inconsistencies, and appropriateness are called *static validity checks* since they are done once during the planning stage.

As with the science program selection process, rule sets would be applied to programs and observations to ensure that they were feasible at the suggested observation execution time. Any problems discovered during the analysis would be brought to the attention of the staff user during plan creation. Static validity checks would be analyzed for the current conditions and time or for the observation's estimated execution time in a plan.

An observation validity check might guarantee that selected guide stars are adequate for an observation or that observation constraints such as a zenith distance restriction be met. A program or observation could be analyzed to make sure that all required calibrations were present and acquired before any science configurations.

The rules of the validity checks could be complicated and will require that the rule creator know many things about many parts of the system. Therefore, it is probable that validity checking will become more important as experience with the Gemini telescope and instruments is acquired. The capability for static validity checking will initially be present.

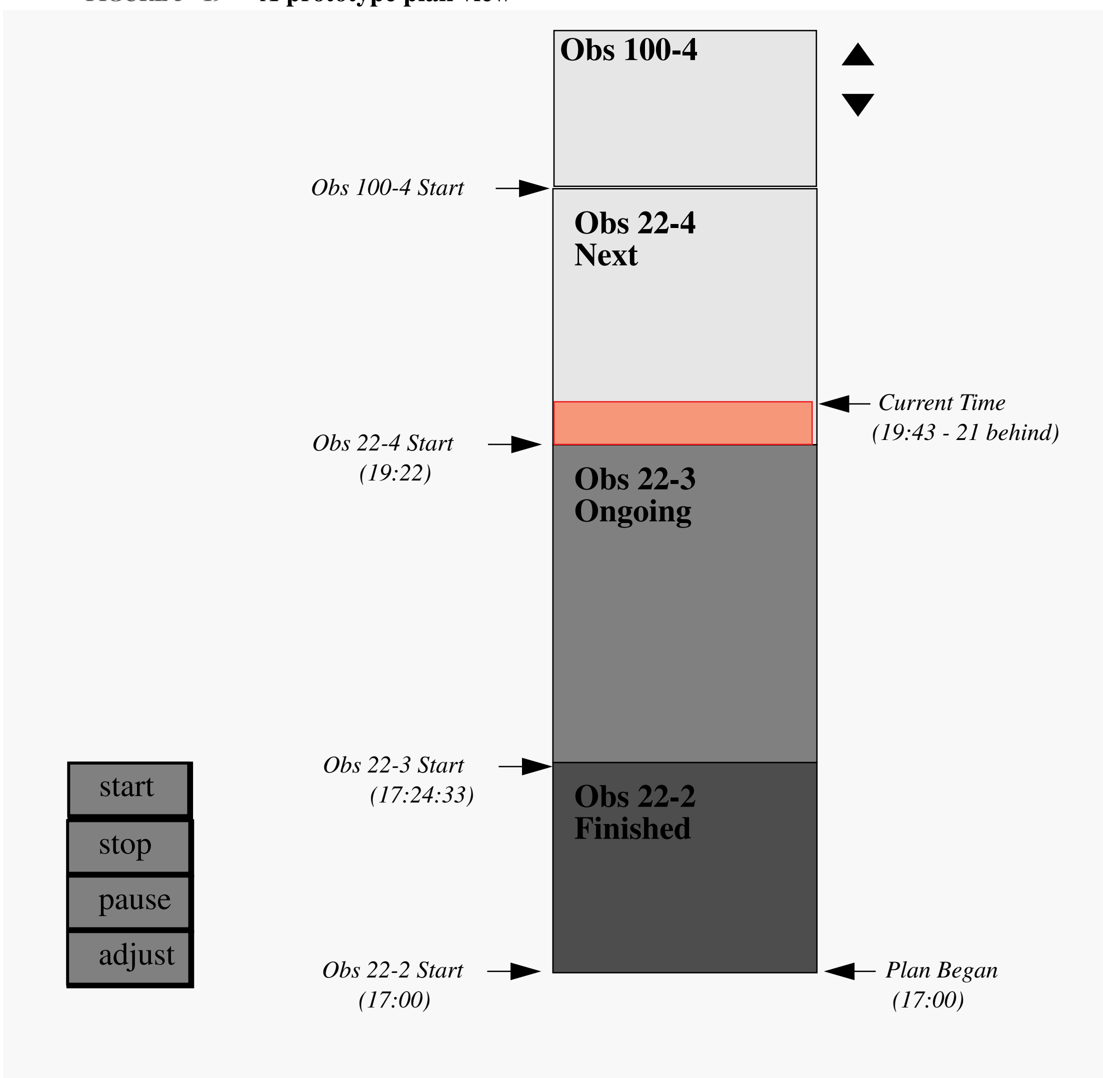
5.3.8 Plan View

The *Plan View* is a graphical representation of the current and upcoming operations of the Gemini System. The view shows the observations that have been scheduled and their estimated time requirements. It shows where the current time and how much operations are behind or ahead of the plan. The plan view is another way of tracking the system performance and efficiency—it is an information-oriented screen.

The presentation of the plan view will be part of the toolset provided by the OCS work group to monitor the activities of the Configurable Control System. A prototype look for the plan view is shown in Figure 5 - 19. The plan view shows the current position in the plan, what has been finished, and what comes next. The view shows that the system is currently 21 minutes behind the plan estimates.

A few controls are also shown in the view. The *adjust* control tells the system to update and correct the plan based on the current time. In Figure 5 - 19, selecting *adjust* at the conclusion of Observation 22-3 will move the start times for everything following Observation 22-3 ahead in time and remove the display showing the system to be behind in the plan. This might cause observations towards the end of the night to become impossible to observe. Plans must be adaptable to changes during an observing session.

FIGURE 5 - 19 A prototype plan view



5.3.9 Dynamic Validation

Observations that are already scheduled (in the Configurable Control System or CCS) and visible in the plan view were known to be valid observations when they were selected by the operator; the observations passed the static validation check. While they are in the CCS awaiting execution, observatory conditions may change such that an observation may no longer be valid. The checking of observations while they are in the CCS is called *dynamic validation*.

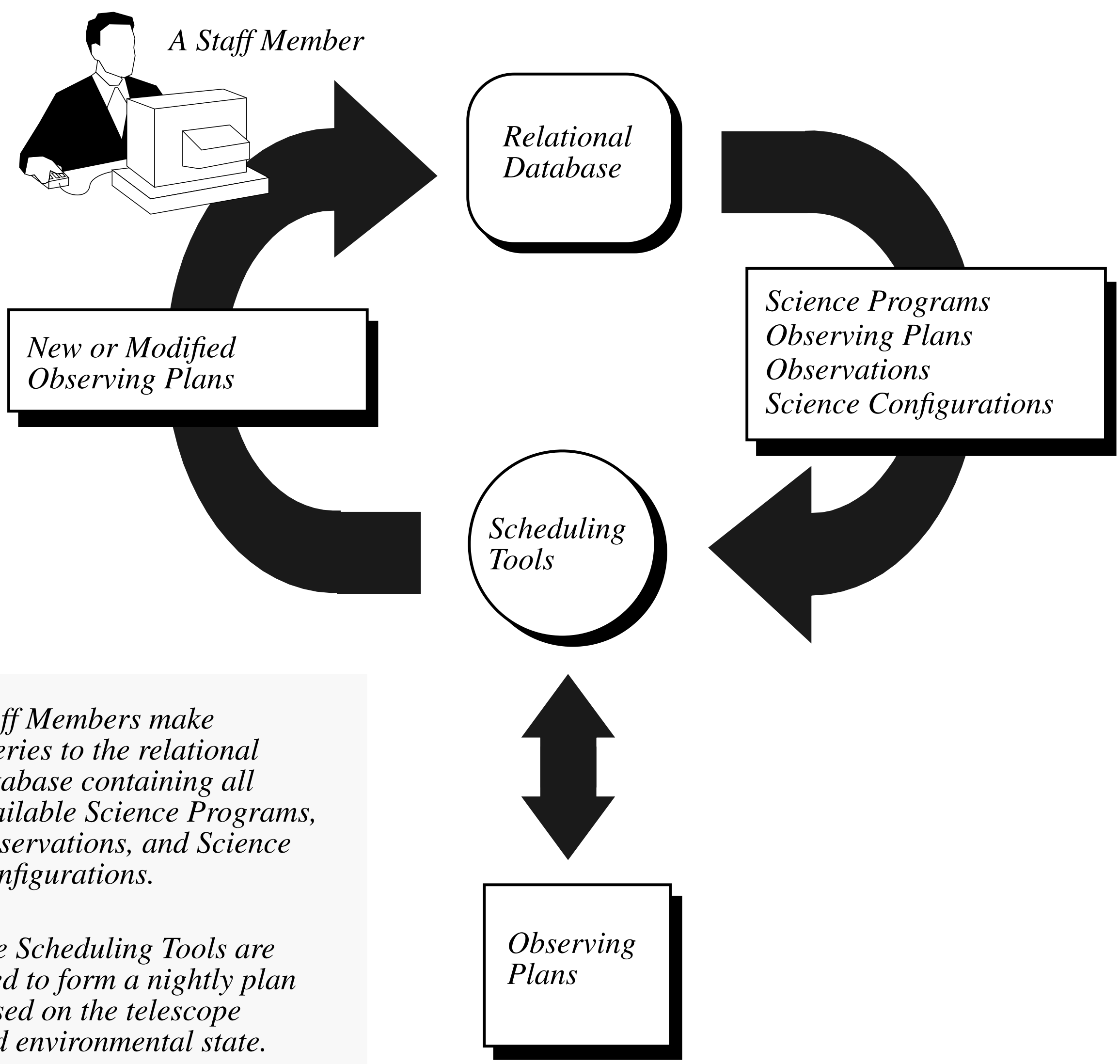
Dynamic validation of the observations in the CCS will take place periodically. The dynamic validation will be done through rules sets that operate on the observation configurations using the current observatory status information.

The view plan will be updated to present the current status of observations in the CCS and warnings or alarms will be sent to the operator when conditions demand attention.

5.3.10 Scheduling System Summary

Figure 5 - 16 on page 5 - 35 shows the process observers use to create, save, and modify Science Programs. Once created, the Science Programs are submitted to the GCS to be verified and scheduled. The following two figures summarize this section and the use of the Gemini scheduling tools.

FIGURE 5 - 20 **Scheduling Tools - Creating Observing Plans**

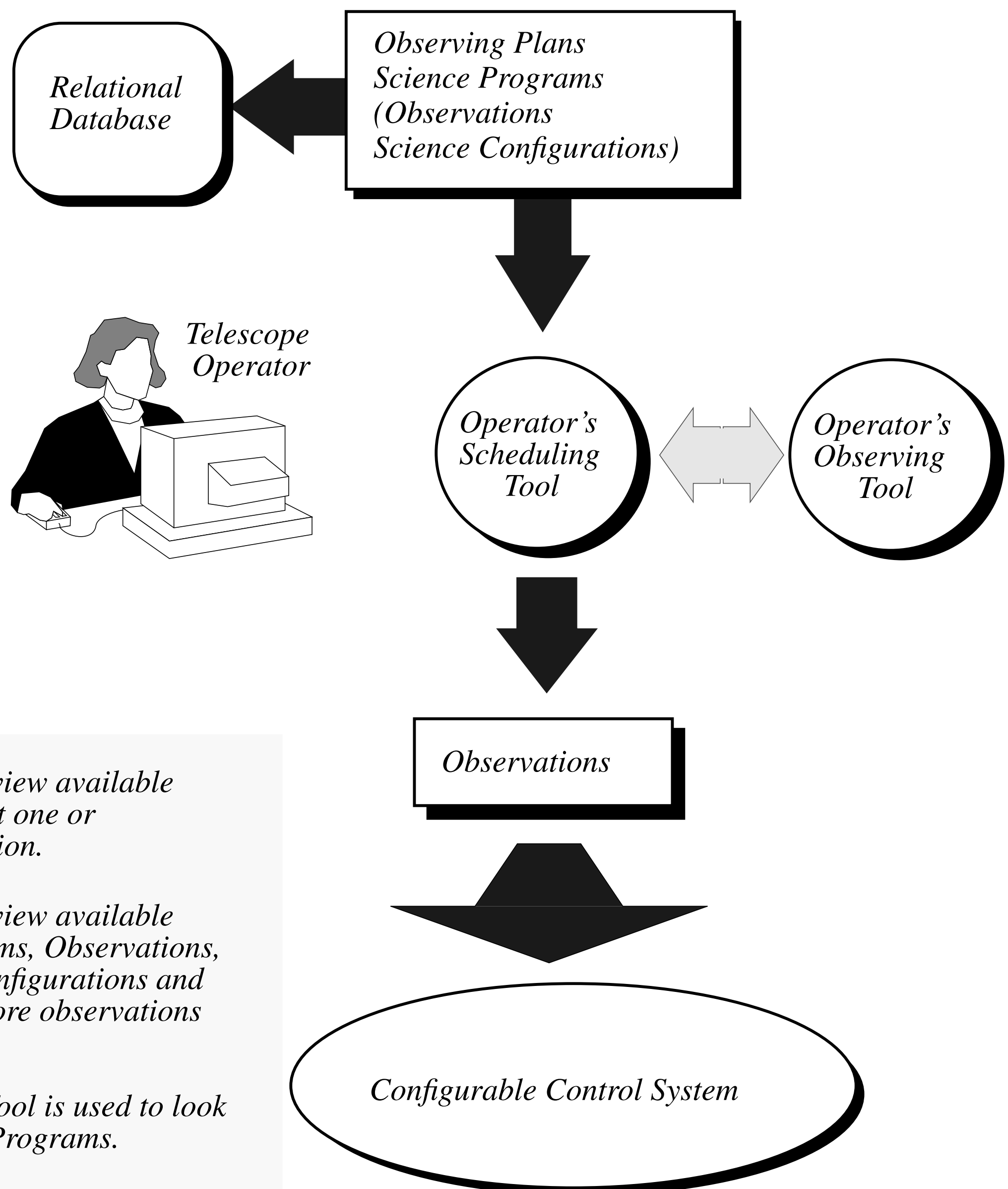


Staff Members make queries to the relational database containing all available Science Programs, Observations, and Science Configurations.

The Scheduling Tools are used to form a nightly plan based on the telescope and environmental state.

The Plans are stored back in the relational database for use during the night or future modification.

FIGURE 5 - 21 **Scheduling Tools - Choosing the next observation**



Operators can view available Plans and select one or more for execution.

Operators can view available Science Programs, Observations, and Science Configurations and select one or more observations for execution.

An Observing Tool is used to look inside Science Programs.

Observations are selected and sent to the Configurable Control System for execution.

5.4 THE CONFIGURABLE CONTROL SYSTEM

The Configurable Control System (CCS) is the hub of the OCS with many tasks that are related to coordinating the activities of the Gemini Control System during an observing session.

The high-level functionality of the CCS and its place in the overall design will be described in this section, but the majority of the CCS functionality will be described in “Details of the Observing Control System.” “Top-level Diagrams” on page A1 - 1 of this document show the formal design views of the OCS. This information may be of use when reading this chapter.

5.4.1 Configurable Control System Overview

The function of the CCS is to provide the functionality required by the VUI. In the software design it is the part of the OCS that interacts with the other software systems on behalf of the users.

It is called the Configurable Control System because within the design, the OCS views the observing process as a series of system configurations linked by periods of activity when the system moves from one system configuration to the next. As discussed in “OCS Visible Interfaces Description” on page 5 - 4 changes to the control system are generated in two ways. The control system moves to a new configuration based on the cumulative changes from interactive consoles or by large changes when a Science Configuration is executed in the system.

Like the low-level EPICS systems, the CCS is a database-driven system. Observations and their Science Configurations exist in the dynamic CCS database, called the Observing Database (ODB), as database records. It is a dynamic database because the observations themselves are dynamic; they can grow and shrink as the user needs change. The fields of the records are the attributes exported by the other principal systems that describe their state and status. The records that represent Observations and Science Configurations are created and destroyed as they move into the CCS, are executed, and finally leave.

The database model allows the design to take a high-level approach to commanding the other subsystems. From the point of view of the OCS, all it needs to do to prepare for a new observations is to say to all the subsystems, “Do whatever you need to do to match this configuration,” while handing it a Science Configuration. When the observer’s work is done and the cycle repeats. In actuality, it’s somewhat more complicated, but the database model is what allows the OCS to isolate itself from the details of the systems it must control.

The Observing Database is the most important part of the CCS. All other CCS functionality is implemented by applications that are *clients of the ODB*. An ODB client is a program that uses the information in the ODB to provide its own functionality and executes within the CCS. The other major component of the CCS, the Sequencing System are all clients of the ODB. The applications of the VUI are not formally clients of the ODB although they access ODB information.

The following are the important guidelines and goals of the CCS design.

- Strive to keep the details of the Data Handling System, Telescope Control System, and Instrument Control Systems in their respective systems. The OCS will become hopelessly complex and difficult to understand and maintain unless the divisions are at a high-level.
- Take advantage of the features of the low-level EPICS systems whenever possible.
- It must be possible to operate the four instruments simultaneously to the maximum extent possible. The CCS must provide this capability.
- Provide for flexible planned observing.

5.4.2 Major Components of the Configurable Control System

The CCS is decomposed into its two major parts: the Observing Database and the Sequencing System. The Sequencing System itself is a client of the ODB, but because of its integral place in the OCS design it has the same status as the ODB.

5.4.2.1 THE SEQUENCING SYSTEM

The Sequencing System is the component of the CCS that manages the execution of observations in the CCS. The Sequencing System is also decomposed into parts, and they are described in “Details of the Observatory Control System.” The following functionality is provided by the Sequencing System.

- It accepts Observations and their Science Configurations from the VUI applications and prepares them for execution.
- It manages the shared, unique resources of the Gemini Telescope and allocates them to executing Observations. The CCS must allocate the resources to the concurrently executing Science Configurations since there may be as many as four executing Observations. Shared resources include the telescope itself, the instruments, etc.
- It creates the executing observations, which are also clients of the ODB, and monitors their progress.

5.4.2.2 THE OBSERVING DATABASE

The Observing Database is the storehouse that holds the database records that represents the executing observations. The observations and their configurations exist in the ODB while they are executing in the Gemini Control System.

The ODB maintains the global system state as a dynamic database record. The system state is set of all current values for the attributes of the other principal software entities. This information is used by ODB clients to provide VUI functionality.

Here is a summary of the functionality provided by clients of the ODB.

- *Console display updates*. A client of the ODB will provide console displays with the current values of attributes in which they are interested. The client will watch for changes in attributes in the ODB global state and update the console screens as appropriate.
- *Access and privileges.* An ODB client will manage the dynamic access and privileges of Gemini users. The information about system users, their connections, and their permissions will exist as information in the ODB.
- *Executing Observations*. When an Observation's Science Configuration is picked to execute by the Sequencing System, the Observation's database record comes alive and a client dedicated to the Observation commands the other subsystems to match the Observation's first Science Configuration. It monitors the global status to check for compliance.
- *Alarms and Errors.* One or more ODB clients will be charged with monitoring the global system state for abnormal conditions. Abnormal conditions might be things like power supplies dying or dewars heating up, which are related to system health. Or they might look for variations between the global system state and the demanded configuration. This would alert the user if someone turned a crank on an instrument during the execution of an observation, for instance. Some problems will be related to OCS problems and these must be reported to operators and users too.
- *Executing Commands*. One or more ODB clients will execute individual commands on behalf of OCS consoles and DHS Quick-Look programs. See "The Command Processor" on page 56 for a complete description of this functionality.

5.4.2.3 THE COMMAND PROCESSOR

Many programs, including the OCS consoles and the DHS Quick-Look tools may need to interact with the other software subsystems to make small changes to the global system state (i.e. to issue commands).

The OCS implements this functionality in the CCS as opposed to having individual applications access the low-level systems directly. Commands will appear to the other subsystems to be *mini-configurations* and will be executed using the same methods as are used with configurations. The command functionality is implemented through the OCS

- To ensure a similar command “look and feel” for all required subsystem functions.
- To allow the OCS to track outstanding commands.
- To allow commands to use the same subsystem interface that exists between the subsystems and the OCS for configuration commands.

A TCL-based set of commands will be provided by the OCS Work Package group. Script creators will use the commands in the tools they create.

5.5 HIGH LEVEL OCS DESIGN SUMMARY

This concludes the presentation of the high-level design of the Observatory Control System. This chapter has focused on what the users of the system will see and do when observing with the Gemini telescopes.

The chapter, “Details of the Observatory Control System” will describe how the Configurable Control System provides the functionality required by the applications in the Visible User Interface.

6 HIGH LEVEL DATA HANDLING CONCEPTS

6.1 INTRODUCTION

This chapter gives a basic overview of the Gemini Data Handling System (DHS) and the various utilities included in the DHS work package. A detailed description of the design of the DHS is given in “Details of the Data Handling System” in Chapter 11.

6.2 CHANGES SINCE PRELIMINARY DESIGN REVIEW

The following changes have been made to this chapter since the Preliminary Design Review (PDR) on 20 April 1994:

- The OCS makes up a unique ID (formerly the “odometer number”) for each exposure rather than the DHS. The number is supplied to the DHS and the Instrument Control System (ICS) and is used to match data and header information together.
- The DHS makes available Quick Look and Data Storage servers, which an ICS can use for displaying and storing data. An ICS is responsible for storing its own data (using the Data Storage Server), and no longer acts as a server for the DHS.
- There is now a Synchronous Data Reduction server to allow direct feedback from the data reduction to the OCS for processing calibration runs.
- Data are passed from the ICS to the DHS using messages transmitted using IMP and built using SDS (IMP and SDS are part of the DRAMA system of the Anglo-Australian Observatory), rather than through a shared data buffer. The format of these messages is defined in an Interface Control Document (ICD/3 [26]).

- EPICS channel access is no longer used as a means of exchanging command and status information between the OCS and the DHS, and there is no longer a "symmetric database" model. Interface Control Documents (ICD) 1 [24] and 2 [25] describe how commands and status are now handled on the Gemini Control System.

6.3 THE ROLE OF THE DATA HANDLING SYSTEM

The Data Handling System fulfills the following roles:

- To coordinate the generation and storage of data:

  --By providing a server which can store the data generated by the scientific instruments;

  --By controlling the format of the files generated and the header information written to those files;

  --By maintaining a record of the files created and information on the status of those files (e.g. which have been reduced, which have been saved to permanent storage etc…).

- To allow the operator and observer to calibrate the Gemini telescope and instruments:

  --By providing a server which the Observatory Control System can use to perform "synchronous data reduction", i.e. an on-line data reduction process which generates information which can be used to calibrate the telescope or an instrument (an example is a focus run).

- To allow the observer to judge the quality of the data being generated and to take appropriate action if that quality falls below expectations:

  --By providing a quick look server which the scientific instruments (or other parts of the Gemini system) can use to display the latest frame (or part of a frame) from their detector.

  --By providing a separate, file-based quick look facility for instruments which do not have one of their own. (It is expected that most instruments will use the quick look server, so this additional facility will not be used very often).

- To allow the observer to make on-the-spot scientific decisions:

  --By providing a configurable on-line data reduction system which removes basic telescope and instrument effects and can display the data in a variety of ways;

  --By allowing the observer to interact with the on-line data reduction system through a "data reduction console".

- To generate a record of the observing session.

  --By generating an electronic observing log which the observer can take away;

  --By keeping a record of the observing activities and status of the various systems, which may be interrogated to obtain statistics or to track faults.

  --By providing utilities for the rapid collection of engineering data.

--By providing utilities for recording and dealing with system faults.

- To coordinate the permanent storage and archiving of the data:

  --By providing observers with a copy of their data;

  --By ensuring suitable calibration observations are made;

  --By sending the data regularly to an external archive site.

- To provide an interface to databases in the outside world:

  --By defining a standard database access protocol.

  --By providing libraries which other systems can use to access those databases.

6.4 WHAT THE DATA HANDLING SYSTEM CANNOT DO

The Data Handling System is designed to operate with data from a wide variety of different detectors and instruments and yet still be easily maintainable. It also needs to be flexible enough to work with a variety of external data reduction systems and cope with data from instruments we have not yet thought of. For these reasons, detailed knowledge of the internal workings of particular instruments or detector controllers needs to be kept outside the data handling system, apart from knowledge which can be contained within a data reduction recipe.

Here are the operations which the Data Handling System *cannot* do:

- It cannot provide a real-time display showing each individual pixel as it is read from the detector, nor can it show real-time video images at several Hz. Such displays (if required) need to be provided by the Detector Array Controller or Acquisition and Guidance unit as appropriate.

  The DHS quick look server *can* show a display building up from sub-frames (i.e. small rectangular subsets of the full area, see ICD/3 [26]), which in some circumstances may be sufficient.

- It cannot carry out any detailed pre-processing specific to a particular detector (such as the combination of frames from several detector chips at arbitrary orientations, or the removal of complex electronic offsets that can’t be removed by a simple BIAS subtraction).

- It cannot process data in stream mode. For example, the Data Handling System cannot accept a continuous stream of individual pixels and process each one on the fly (matching it up with its corresponding pixel in a BIAS, DARK, FLAT etc.…). The Data Handling System currently works only with whole images which can be stored in discrete files.

  However, the data storage server has been designed to allow an upgrade path to a stream mode of operation at some future date. It can accept a stream of sub-frames from which it builds up a single frame to be stored to disk. The quick look server too can build up a display from a series of sub-frames.

- It cannot accumulate, subtract and co-add the rapid pairs of frames taken by an instrument in CHOP mode at several Hz. The instrument must perform this function and generate one frame or set of frames.
- It cannot carry out any real-time processing on the data, such as “shift-and-add”. Such processing is the responsibility of the Detector Array Controller, the Instrument Control System, the Acquisition and Guidance subsystem, or the Adaptive Optics subsystem as appropriate.

Excluding these operations from the Data Handling System does not mean they cannot be carried out in the Gemini Control System. If they are necessary they can be carried out in the appropriate Detector Array Controller or Instrument Control System. Throughout this document the term *data reduction* will be used to refer to the data processing operations carried out by the Data Handling System. The term *data pre-processing* will be used to describe those operations carried out at a lower level by the detector or instrument control systems.

6.5 MAJOR SYSTEM COMPONENTS

Figure 6 - 1 on page 6 - 5 shows the various data handling and display screens which the Observer will have access to. The five screens shown are:

Real-time Pixel Display. This is not provided by the Data Handling System, but is included for completeness. Some instruments or detector controllers may provide their own real-time video display to supplement the quick look display. The display is optional and might not be present.

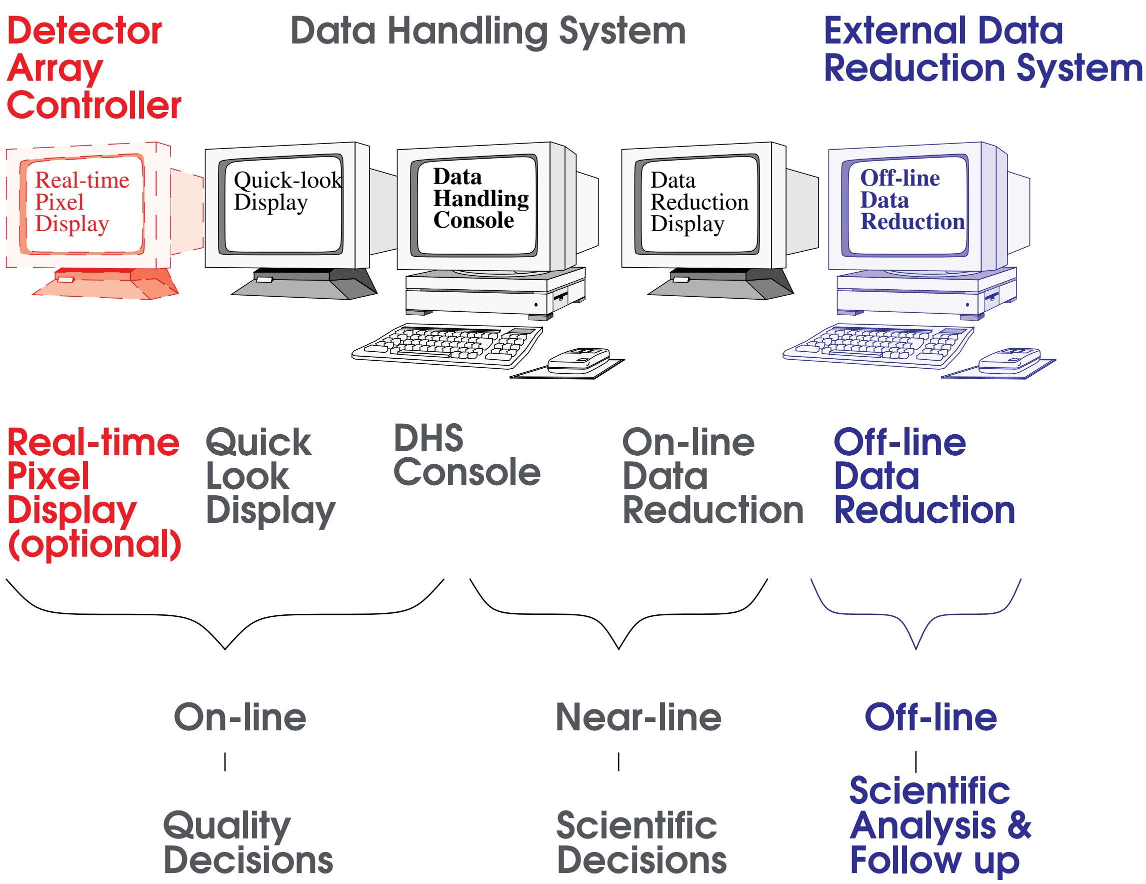
Quick-Look Display. A display showing the latest frame from the instrument, as described in “Quick look display server” on page 6 - 6. The display is described as “on-line” because it is synchronized to the data taking and always shows the very latest frame.

Data Handling Console. The console the Observer uses to control and interrogate the data handling system. In practise the data handling console is available from the Observing Tool (described in “Visible User Interface Overview” on page 5 - 9).

Data Reduction Display. The data display generated by the on-line data reduction system (see “On-line Data Reduction System” on page 6 - 16). The display is described as “near-line” because, although it is not synchronized to the data taking, reduced data should appear on this screen very soon (within a minute or so) after being observed. Any feedback from this display to the observing session always goes via the Observer.

Off-line Data Reduction. This workstation is used by the Observer or by an assistant to analyze earlier data using a favored off-line data reduction system (e.g. ADAM, IRAF, FIGARO etc…). Such off-line data reduction systems are the responsibility of the data reduction groups and are beyond the scope of the DHS.

FIGURE 6 - 1 **Data Handling on the Gemini System**



The "Data Handling System" is not a single process which the other Gemini systems make requests of. There are various pieces of the Data Handling System embedded in all the other systems, allowing them to carry out data handling tasks. The Data Handling System consists of the following collection of entities:

- A quick look display server (one or more per instrument).

- A synchronous data reduction server (one per instrument).
- A data storage server (one per instrument).
- A data storage database manager.
- Logging and problem tracking utilities.
- An on-line data reduction system (one per instrument).
- A data transport and archiving system.
- External database interface libraries.

6.5.1 Quick look display server

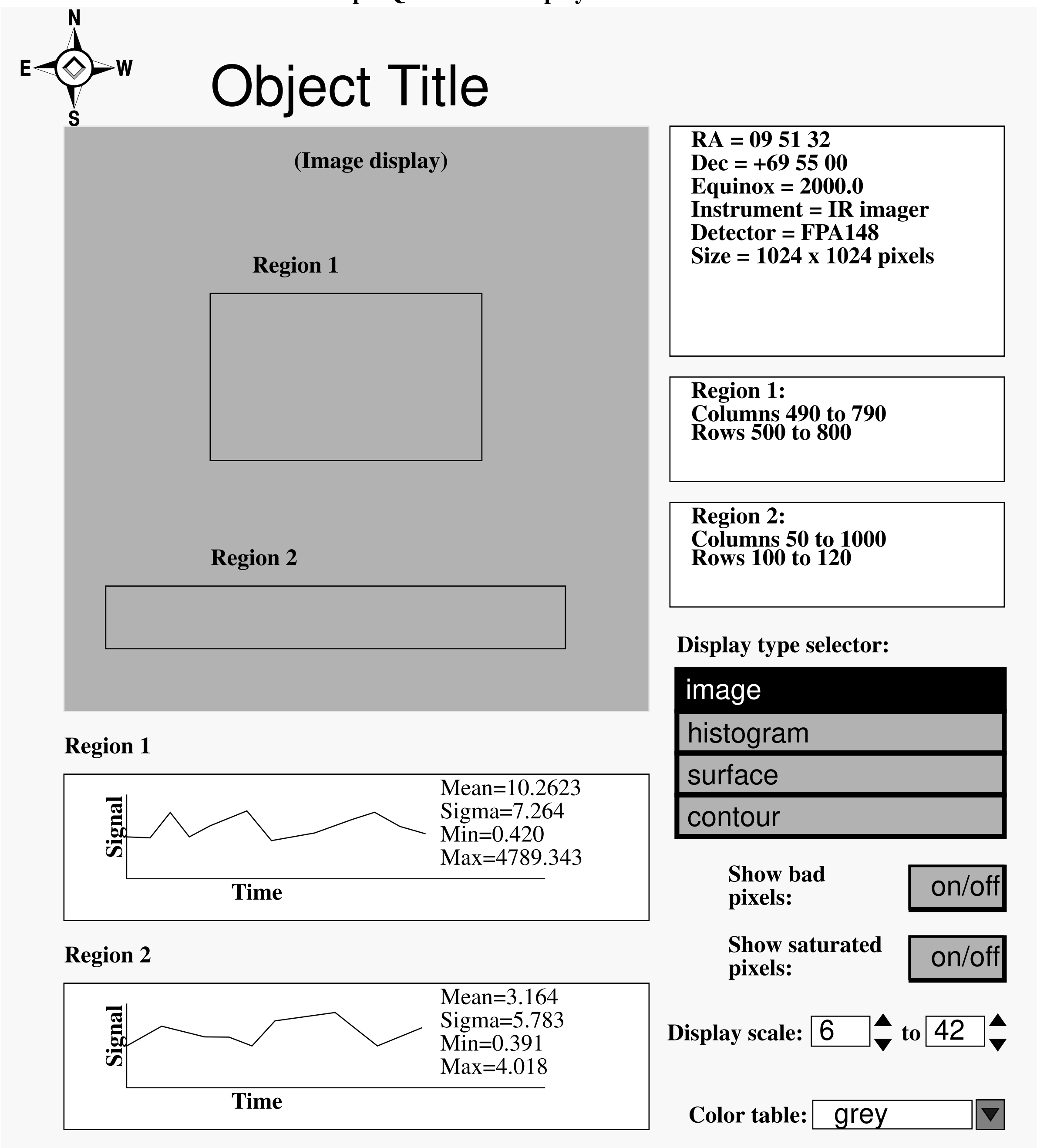
6.5.1.1 THE QUICK LOOK DISPLAY

The "Quick Look" display is supplemental to the field acquisition camera used by the telescope and any real-time video display generated by an instrument. It shows the latest frame of raw science data obtained from a particular instrument, giving the observer an indication of the quality of the data being generated. An example screen is shown in Figure 6 - 2 on page 6 - 7. In practise the display screen will have a lot more options and controls than shown (since it is likely to be based on PvWave). Wherever possible, the display should indicate the compass directions corresponding to the axes of the data, to allow the observer to match up the features on the quick look display to those on the field acquisition camera.

The display is updated continuously to give the observer a continuous report on what the instrument's detector is seeing. The display can be programmed to report various statistics in regions of interest, which can help when peaking up a signal (for example, the display shown in the figure shows the mean and standard deviation signal in two regions of interest).

The display can also be programmed to carry out simple arithmetic operations on the incoming data, or apply a previously downloaded calibration. Detectors which take a long time to read out can send their data to the Quick Look Server in small chunks, and the server will update its display to show the frame building up.

FIGURE 6 - 2 **An Example Quick Look Display**



The following simple processing operations may be carried out and the results reported on the quick look display:

- Addition of, subtraction of, multiplication by or division by a specified calibration frame.
- Calculation of simple statistics, such as the minimum, maximum, mean and standard deviation, in selected regions of interest on the detector's surface.

  It would be useful to have a graphical display (e.g. strip chart) of the changes in one or more of the statistics as a function of time.
- Summation of the data along all or a subset of its columns to generate a profile in the X direction.
- Summation of the data along all or a subset of its rows to generate a profile in the Y direction.

These processing operations provide information to help with the fine positioning of the telescope and the peaking up of the signal on the detector.

The observer is able to tailor the appearance of the quick look display, controlling the type of display, color table, scale and magnification, for example. Bad pixels and saturated pixels can be indicated on the display.

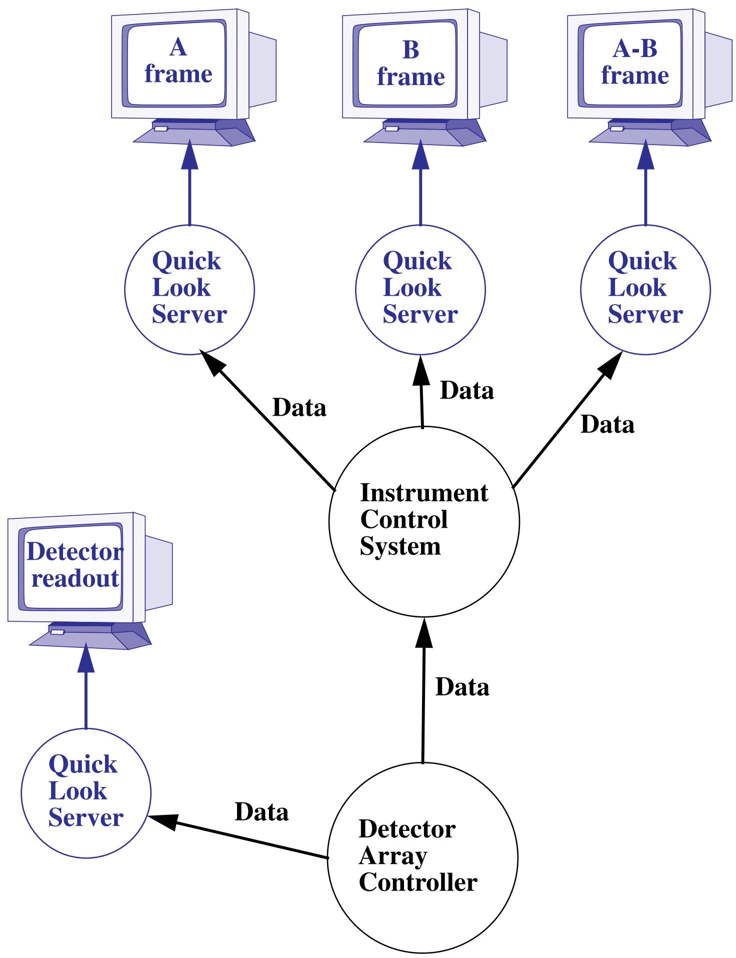
6.5.1.2 THE QUICK LOOK SERVER

The DHS provides a quick look server which instruments may use for displaying their data. Quick look server processes can be created and destroyed at will, and multiple processes may exist on the system at any one time. An instrument may create more than one display process, for example to display the "A" and "B" frames from a CHOP mode observation separately. If there is more than one instrument in use, each instrument can create its own separate quick look display or displays. Figure 6 - 3 shows a possible arrangement of quick look servers, where the Detector Array Controller is using one and the Instrument Control System is using three. This shows how a complicated arrangement can be set up by replicating one simple server.

Note that it is also possible for other Gemini systems (such as the Acquisition and Guidance subsystem) to display data using a quick look server.

It should be possible for a remote client to register an interest in the display being generated by a Quick Look Server and receive a copy of the display. This mechanism allows the operator or a remote observer to see a copy of any of the displays.

FIGURE 6 - 3 **A Possible Arrangement of Quick Look Server Processes**



The location of the quick look display screens is decided when an observer first registers to use an instrument. The quick look displays of a particular instrument are directed to the workstation of the observer registered to use it. Copies of the display may be routed to the Operator's workstation at the operator's discretion (for example if the observer's screen is at a remote location). More than one quick look display can appear on a given workstation. Each display will appear in its own window, which can be resized or iconized as desired.

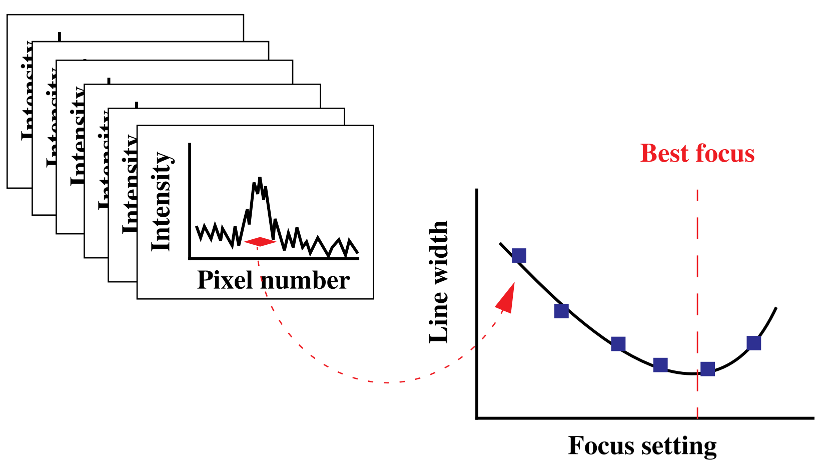
As far as possible, the quick look display server should be built from existing utilities. The current baseline is PVWave/IDL.

6.5.2 Synchronous data reduction server

The DHS provides a synchronous data reduction server which the Observatory Control System can use to process the data frames generated during a calibration run. Synchronous data reduction server processes can be created and destroyed at will, but there will only be one server for each active instrument.

Synchronous data reduction is used in any situation where it is not acceptable to wait for the on-line data reduction to process the data, and where information needs to be fed directly back to the Observatory Control System. An example is a focus run for a spectrograph where several exposures are made at different focus settings. The synchronous data reduction server processes each frame to determine the line width, and the best focus is determined by finding the minimum in a plot of line width against focus setting. Figure 6 - 4 on page 6 - 10 illustrates how a focus calibration curve is built up from several frames. The best focus setting is passed back to the Observatory Control System and used to change the spectrograph configuration.

FIGURE 6 - 4 **An Example Calibration Curve for a Spectrograph Focus Run**

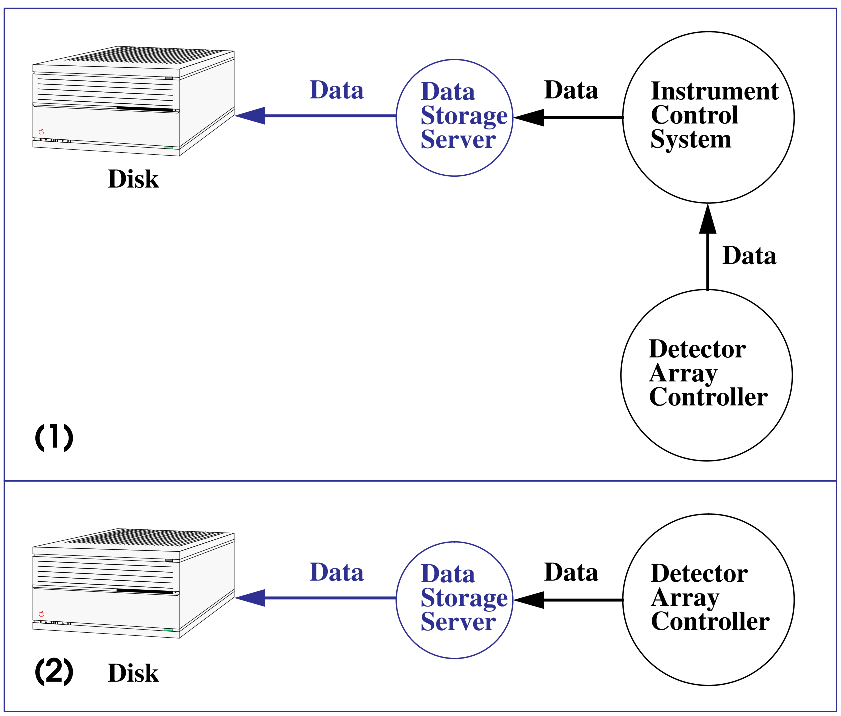


In practise the synchronous data reduction server is an enhanced version of the quick look server, based on PVWave with some extra IDL scripts for handling the various calibration modes.

6.5.3 Data storage server

The Data Handling System provides a data storage server which instruments use to store their data to disk on the host workstation. Data storage server processes can be created and destroyed at will, but there will only be one server for each active instrument. If there is more than one instrument in use, each instrument can create its own separate data storage server. Data may be stored to disk directly from the Detector Array Controller or passed through the Instrument Control System. Figure 6 - 5 on page 6 - 11 shows the possible data storage configurations.

FIGURE 6 - 5 **Possible Data Storage Configurations**



It is the responsibility of the Data Handling System to control the format of the files generated and the format of the header information written to those files (although it is the responsibility of the Observatory Control System to decide what the header items are and what actual values should be stored). For ease of maintenance, no other parts of the Gemini system may make any assumptions about the file format. For this reason all science data storage on the Gemini system must go through a data storage server. Instruments may store science data on their own private disks for internal use (e.g. for observing an occultation where speed of data storage is critical), but even these data must be transferred to permanent storage through a data storage server.

The names of the raw science data files are designed to be completely predictable from the unique integration ID (which is formed from a combination of telescope ID, science program number, observation number, configuration number and integration number, see "Details of the Observatory Control System" in Chapter 10).

6.5.4 Data storage database manager

The data storage database manager maintains a record of the files created on the system in the data handling database. At the end of each integration the OCS sends the DHS an ENDOBSERVE command, and the name of the new file created is entered into the data handling database by the data storage database manager. The file's entry in the database may be modified at a later date by the on-line data reduction system and the data transport manager. The data reduction system also uses this database to search for calibration frames.

6.5.4.1 THE DATA HANDLING DATABASE

The data handling database contains the following information for each file:

- The name of the file and its location on the disk.
- Its unique integration ID (as described above).
- The header information contained in the file.
- Information on which science program (and which observer) the file belongs to.
- The observation type for the file (BIAS, DARK, FLAT etc…).
- The status of the file:

--whether the exposure creating it has completed.

--whether it has been reduced.

--whether it has been saved to permanent storage.

--whether it has been saved to tape for the observer to take away.

--whether the observer has flagged it as "good" or "bad".

6.5.4.2 DATA HANDLING CONSOLE

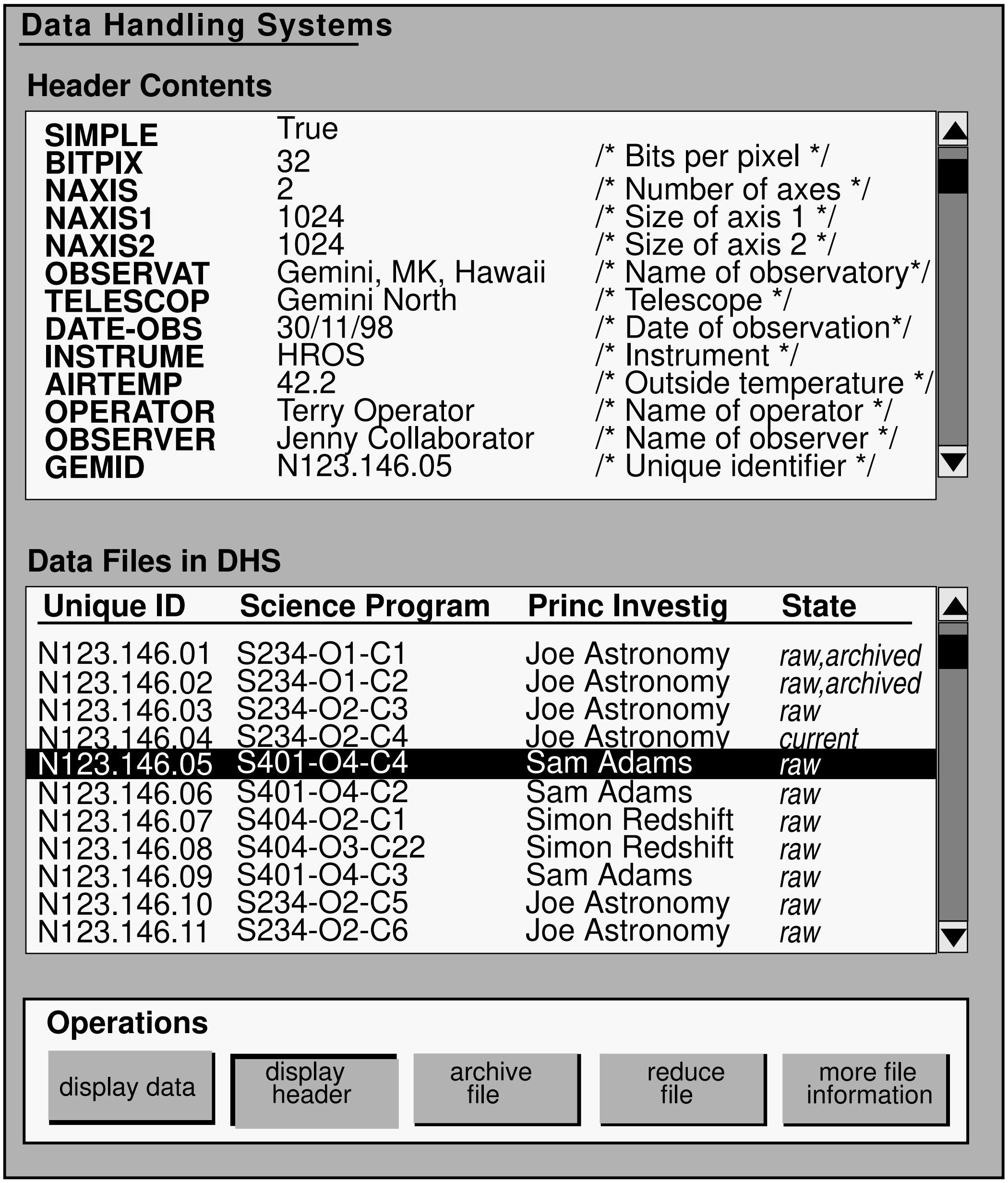
The observer may interrogate the contents of the Data Handling database by means of a Data Handling console, as show in Figure 6 - 6 on page 6 - 14. By using this screen the observer is able to:

- See the names of the files created previously during the observing session, together with information on which of these have been successfully reduced
- See the names of the files of reduced data created so far.
- Obtain information from a database of reduced observations.

  This database includes information on the instrument configuration used for each observation so the observer can, for example, select and examine all the "dark" frames taken so far with a particular exposure time, or find out what spectral standards have been observed so far etc...
- See the headers of the files created so far.
- Configure any quick look displays belonging to the Data Handling System.
- Ask the Data Transport system to save particular files to permanent storage.

In the example of Figure 6 - 6 the observer has asked to see the header information contained within a file.

FIGURE 6 - 6 **An Example Data Handling Console**



6.5.5 Logging and problem tracking utilities

The Data Handling System collects logging information generated by the Gemini Control System and stores it for future use. The following logs are maintained:

Electronic Observing Log. This log contains a list of observations made, together with observer comments, header information and information on the observing conditions when each observation was made.

Because the Observatory Control System allows flexible scheduling, more than one observer may obtain data on the Gemini telescope on any particular night. At the end of each night an observing log is generated for each observer from the headers of all the files belonging to that observer created during the night. Each observer is given a copy of his or her observing log to take away. A central observing log is also generated from all the observations made during the night, and this is copied to the permanent archive along with the data.

History Records. A record is kept of the science programs submitted to the system and the commands executed by the Observatory Control, Telescope Control and Instrument Control systems, together with miscellaneous comments logged by the observer and/or operator. Sufficient information is stored to allow an observation to be repeated at a later date. All alarms and error messages are also logged.

The Data Handling System provides a mechanism to allow the other systems to log a history of their usage, and it manages the output from those systems. However, the DHS cannot control the content of the messages, and it is up to the other systems to decide what information is logged and when.

These records are also stored in the permanent archive along with the data.

System Use Records. Statistics on the use made by the system is maintained in a database. System administrators can interrogate that database to determine, for example, the proportion of time being used by the various partner countries, or the percentage of time lost due to weather or mechanical problems.

Engineering Logs. These logs are used to track the state of the various system components at various times. It is configurable by engineers, so that quite specific pieces of information can be logged when necessary. Some pieces of information (such as the encoder errors on the secondary mirror) may need to be logged extremely rapidly in short bursts.

The engineering logs are used by engineers only, and they are not stored in the permanent data archive (although the system engineers will keep their own records).

In practise the EPICS archiving tool, AR, is used to log engineering information.

6.5.5.1 PROBLEM TRACKING

The Data Handling System also provides utilities for keeping track of problems. The operator and/or observer can log any problems they encounter, and the appropriate members of staff are notified. The day crew can fix the problems with the aid of the history records and engineering logs kept during the night.

The current baseline for a problem tracking utility is the GNATS package. The DHS will provide extensions to the GNATS package for the particular classes of problems relevant to an observatory.

6.5.6 On-line Data Reduction System

6.5.6.1 PRINCIPLES OF ON-LINE DATA REDUCTION

The objective of the on-line data reduction system is to remove the basic instrument, telescope and atmospheric signatures and generate sufficiently reduced data for the observer to be able to judge the data quality and make on-the-spot scientific decisions. Such a facility is especially important for infra-red observations where features may be swamped by background effects. The sort of scientific decisions on-line data reduction allows are:

- Do I have a sufficiently good signal to noise to meet my scientific objectives? For example:

  --Do I have sufficient recognizable features to calculate a redshift?

  --Have a detected the object or feature I am looking for unambiguously?

  --Does this spectral line have sufficient signal to noise for me to measure its FWHM?

  --etc.…
- Are the observing conditions good enough for this science program? Should I concentrate just on the brighter objects or should I switch to some other program?
- What should the next observation be? For example, the observer might wish to:

  --take another image to follow up an unexpected feature seen in an observation.

  --take another spectrum at a higher grating order to zoom in on a particular spectral line.

  --repeat an observation because of a mistake.

  --etc.…
- What should I say to my collaborators? The observer might wish to seek advice from collaborators or announce the success or failure of the program.

The on-line data reduction system does not produce publication quality reduced data for the following reasons:

- There are nearly always calibration frames taken at the end of the night which an observer needs to incorporate into the data reduction, so the reduction almost always has to be repeated.
- There may be crucial steps in the reduction (such as the wavelength or photometric calibration) which the observer may wish to check personally before committing the results to a scientific journal.
- There may be some steps which require personal intervention from the observer (e.g. deciding whether to reject an observation spoiled by temporary bad weather, or working around the result of deviations from the usual procedure).

The work is carried out by existing data reduction systems as much as possible, to save recoding any algorithms that already exist in these packages. Because they are beyond the control of the Gemini project these packages are regarded as being outside the Gemini Control System. See "Central DHS to External Data Reduction System" on page 11 - 8 for details.

Each file of reduced science data is given a name that is completely predictable from the name of its corresponding raw science data file. This enables the Observing Tool to display the names of available reduced files without having to know the details of the data reduction operations underway.

6.5.6.2 DATA REDUCTION STEPS

Basic data reduction may involve a number of steps, many of which can be carried out automatically without intervention from the observer:

- Application of a mask to indicate bad pixels, or a window to shut off certain portions of the detector.
- Subtraction of an electronic bias.
- Linearization of the response from the detector into units which are proportional to the number of incident photons.
- Subtraction of a dark frame to remove the contribution from any dark current.
- Division by an instrumental flat-field to even out any differences in response between the pixels of the detector.
- Division by a "sky flat" or "dome flat" to compensate for uneven illumination of the detector.
- Subtraction of a suitable background sky frame[1].
- Co-addition of consecutive exposures of the same object.

If the instrument is being used for spectroscopy there may also be the following steps:

- Calibration of the co-ordinates perpendicular to the slit direction into wavelength, with the aid of a calibration arc or night sky lines of known wavelength.
- Correction of the flat-field for the variation in illumination of the calibration source with wavelength.
- Division by a spectral standard to remove the variation of the transparency of the Earth's atmosphere with wavelength.

6.5.6.3 THE DATA REDUCTION QUEUE

Jobs are submitted to the on-line data reduction system via a "first in first out" data reduction queue. The observer can control the data reduction queue through a data reduction console by pausing it or altering the priority of the jobs within it (rather like a system operator managing a batch queue). There is one queue for each data reduction system (and one data reduction system per instrument).

The Gemini system does not wait for an observation to be reduced before moving on to the next, so any delay in the data reduction can never prevent further data being acquired. Data acquisition proceeds at full speed and the data reduction keeps up as best it can. Under normal circumstances the data reduction should not lag behind by more than a minute or so, so the data reduction is almost but not quite on-line ("near-line" as it is referred to in Figure 6 - 1 on page 6 - 5).

In some situations it may be necessary to control the data acquisition based on a decision made by the observer monitoring the on-line reduction. For example, an observer might request that a continuous series of exposures be taken which will be terminated when the on-line data reduction indicates a sufficient signal to noise has been achieved, or the observer might want to abort an observation in progress if the on-line data reduction shows the observation is inadequate. In all these cases the feedback from the on-line data reduction back to the observing process is made through the observer. If more rapid feedback is required a "synchronous" data reduction system can be used, as described in "Synchronous data reduction server" on page 6 - 10.

Note that in the above situation with the observer monitoring the data reduction display to decide when to terminate an observation, the delay between taking an obser-

1. In STARE mode only. When observing at infra-red wavelengths, successful subtraction of the sky background may only be possible by chopping rapidly between the object and nearby patches of blank sky (CHOP mode). These rapid pairs of object and sky observations are subtracted individually by the instrument controller.

vation and reducing it should not be too long, and it would be undesirable for data reduction to be held up if a large job just happens to have reached the front of the queue first. The DHS group need to consider possible solutions to this problem. Having two queues (low and high priority) and two data reduction agents is one possibility.

6.5.6.4 AUTOMATING THE DATA REDUCTION

The system reduces data automatically by classifying observations into different types. Some examples are shown in the following table. More are listed in “Data Reduction Data Types” on page 11 - 24:

TABLE 6 - 1 **Example Data Types**

Data type	**Description**
MASK	A map of “bad pixels” within an array which should be ignored.[a] A MASK is prepared in advance rather than being generated by the instrument.
BIAS	An very brief exposure made with the detector blanked off, used to record the electronic bias.
DARK	A longer exposure made with the detector blanked off, used to register the detector’s dark current.
FLAT	A long exposure of a calibration lamp used to measure the variation in sensitivity across a detector’s surface.
IMAGE	An exposure of an astronomical object made using an imaging instrument.
SPECTRUM	An exposure of an astronomical object made using a spectrograph.
etc.…	*etc.…*

a. Note that the frames generated by a detector controller should automatically have the detector’s bad pixels imprinted in their quality array. Using a MASK as a calibration frame during data reduction gives the observer a chance to apply an additional mask to the data (e.g. to mask off unwanted areas).

Every instrument has its own list of the data types it can produce, and these types are first defined whenever a new instrument is commissioned. Each data type has a ***recipe*** describing how it should be reduced. These recipes can vary from instrument to instrument and are again first defined when a new instrument is commissioned. The following table shows some example recipes. More are listed in “Data Reduction Recipes” on page 11 - 25:

TABLE 6 - 2 Typical Data Reduction Recipes

Data type	Typical Recipe
MASK	None. A MASK does not need to be reduced.
BIAS	Apply MASK.
DARK	Apply MASK. Subtract BIAS.
FLAT	Apply MASK. Subtract BIAS. Linearize. Subtract DARK. Normalize.
IMAGE (STARE mode)	Apply MASK. Subtract BIAS. Linearize. Subtract DARK. Divide by FLAT. Subtract Sky.
IMAGE (CHOP mode)	Apply MASK. Divide by FLAT.
SPECTRUM (STARE mode)	Apply MASK. Subtract BIAS. Linearize. Subtract DARK. Divide by FLAT. Subtract Sky. Calibrate into wavelength. Divide by SPECTRAL-STANDARD.
SPECTRUM (CHOP mode)	Apply MASK. Divide by FLAT. Calibrate into wavelength. Divide by SPECTRAL-STANDARD.
etc.…	*etc.…*

Note that it in order for the DHS to handle data properly it is crucial that these data types be reliable (i.e. they should be generated automatically by the ICS).

The recipe used for a particular data type may depend on the observing mode used with that particular instrument. In the infra-red the sky background may be cancelled out by observing in “CHOP” mode, where the secondary mirror is chopped and rapid pairs of object and sky observations are made. In this case some of the steps in the default recipe (subtraction of BIAS, DARK and SKY) are not required because they are cancelled out by the chopping. This is shown in the above table. Instrument groups need to define which steps in their recipe can be skipped when observations are made in CHOP mode.

Whenever an observation is reduced it is registered in the data handling database, and this database is searched each time a calibration observation is needed. Whenever a step in a data reduction recipe requires the use of a calibration frame a ***rule*** is used to find a suitable calibration frame in the database. As with a recipe, a rule can vary from instrument to instrument. It is up to instrument developers to define the data types, recipes and rules associated with the data generated by their instrument. Some example rules are shown in the following table. More are listed in “Data Reduction Recipes” on page 11 - 25:

TABLE 6 - 3 **Typical Data Reduction Rules**

Calibration Data type	**Typical Rule**
MASK	Match array size.
BIAS	Match detector name. Match array size.
DARK	Match detector name. Match array size. Match exposure time. Must be younger than <hhh> hours old.
FLAT	Match detector name. Match array size. Match filter. Match grating order. Match grating angle. Match slit type. Must be younger than <hhh> hours old.

In practise the various items may be matched to within a defined tolerance as (apart from the integer items) an exact match may not be possible.

The observer has the opportunity to configure the recipes and rules supplied with a particular instrument. For example it may not be necessary to apply a flat-field to a calibration arc observation (ARC), even though this is included in the default recipe. In general the following search path is used:

1. If the observer has configured a recipe for this particular observation use it.
2. If no recipe has been configured for this observation use the recipe the observer has configured for data of this particular type.
3. If the observer has not configured a recipe for this type of data use the default recipe as defined by the instrument suppliers.

6.5.6.5 DATA REDUCTION CONSOLE

The Data Handling System also provides a console to run and monitor the data reduction system. shows an example data reduction console screen. By using this screen the observer is able to:

6.5.6.6 OBSERVER INTERACTION

The observer interacts with the on-line data reduction through a Data Reduction console (an example of which is shown in Figure 6 - 7 on page 6 - 23) in the following ways:

- To configure the data reduction recipes and rules for a particular data type or for a particular observation (as described above).

  Note that the observer can only change a recipe by switching the various steps within it on and off.
- To interact with the data reduction queue. For example;

  --To examine the queue.

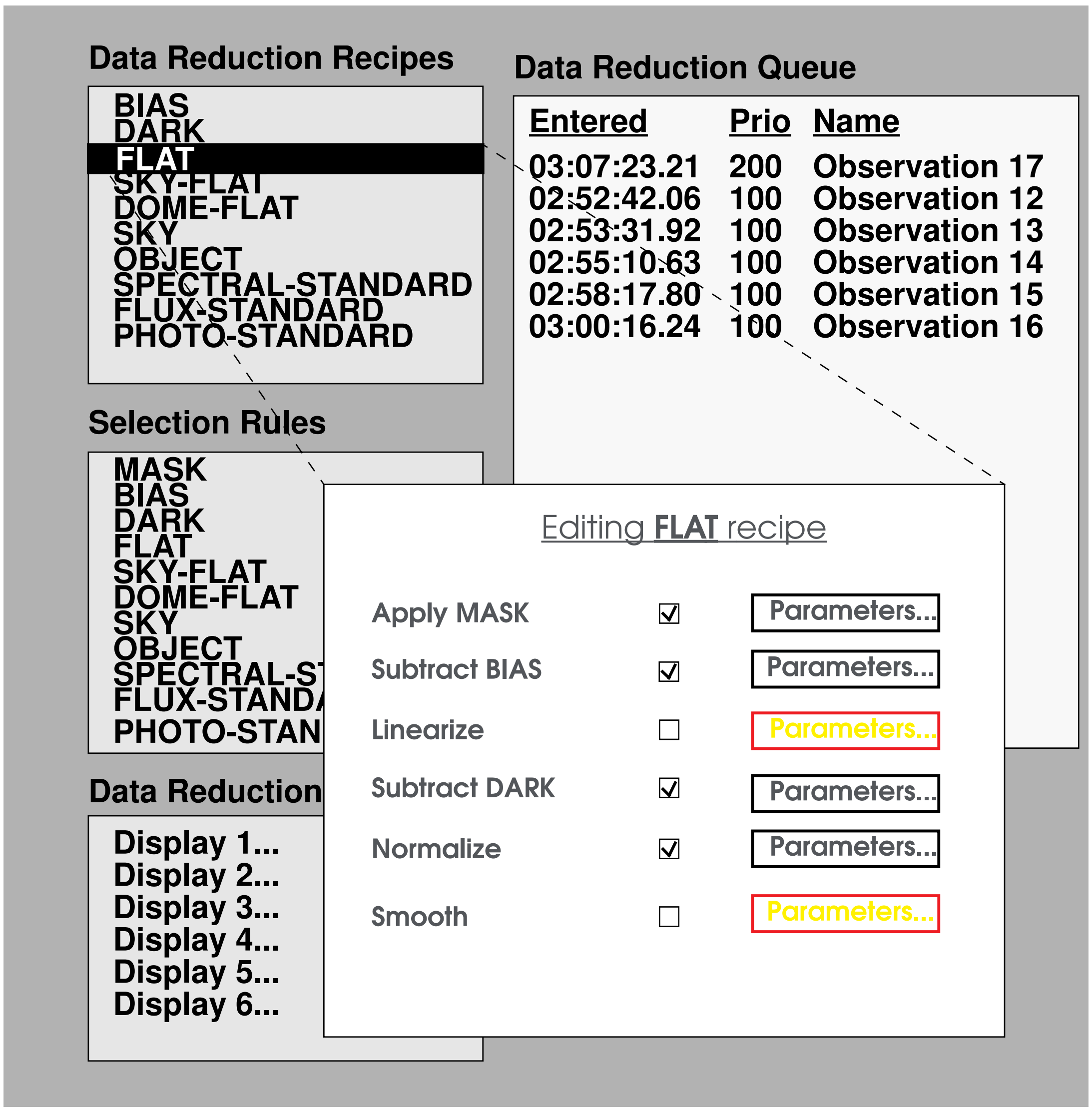
  --To pause the queue.

  --To add new commands into the queue (perhaps asking for a data reduction step to be repeated).

  --To rearrange any existing commands in the queue (perhaps to bring some high priority commands to the front).
- To change the appearance of the displays being generated, for example to alter the color table, scale, magnification and type of the displays. The display types available might include *image, surface plot, contour plot, histogram, linear extracted profile (or spectrum) display,* and *polarization vector map*, depending on the capabilities of the external data reduction system.

In the console screen shown in Figure 6 - 7 the recipe for reducing a FLAT observation is being configured. The recipe is configured by enabling or disabling the various steps. Some of the steps (such as the “smooth” step shown here) may be present in the recipe but disabled by default. The observer may enable these to switch on some additional optional processing. A step may have parameters, which the observer can also configure from the data reduction console (for example the "smooth" step might have a smooth scale length as a parameter).

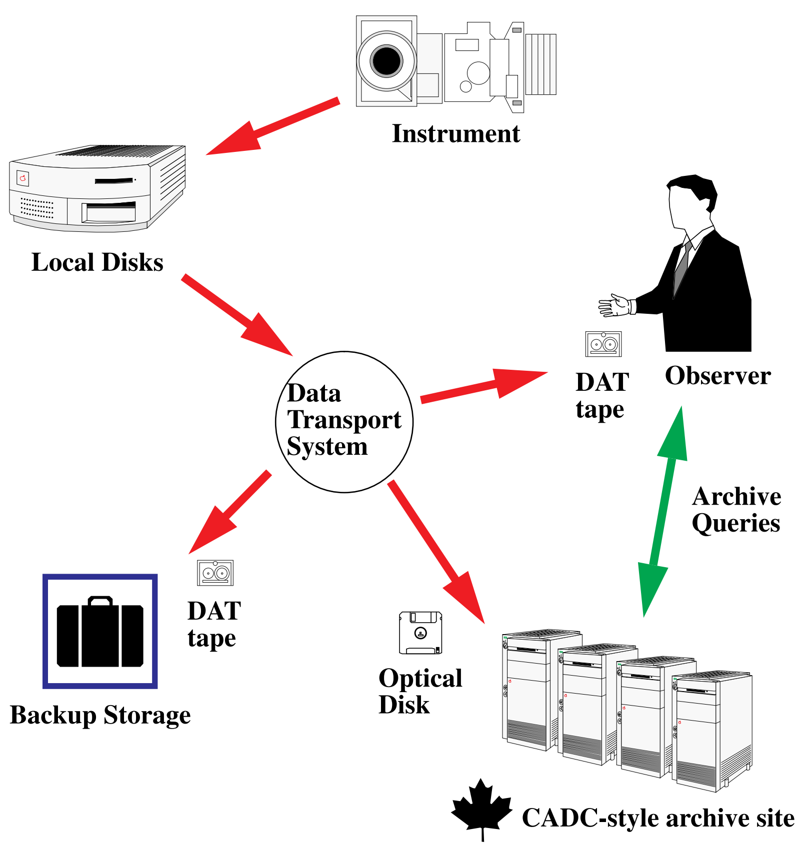
FIGURE 6 - 7 **An Example Data Reduction Console**



6.5.7 Data Transport and Archiving System

The local magnetic disks at the summit are only designed to hold a limited number of files (usually enough for 6 nights of observation). The DHS data transport manager saves these files to a permanent store (such as optical disk) at suitable moments during an observing session. The system may be configured to save the raw or reduced data, or both, to this permanent store. Two copies are made of the permanent store. One (on DAT tape or equivalent) is kept as a backup, while the other is sent regularly to an external archive site (the baseline is the Canadian Astronomy Data Centre, CADC) for incorporation into an archive. The CADC-style archive may be interrogated at some future date to find out, for example, if a particular object has been observed on the Gemini telescope. A schematic diagram showing the archiving of Gemini science data is shown in Figure 6 - 8.

FIGURE 6 - 8 **The Gemini Data Transport and Archiving System**



The Gemini system ensures that sufficient calibration and header information is recorded to make the archive useful to outsiders.

The DHS also copies the raw and reduced data generated for a particular science program onto a transportable medium (DAT tape or equivalent) for the observer to take away. The observer will have exclusive access to that data for a specified number of years.

Note that there will be two separate data streams, an observer visible stream and an observer inaccessible stream. The observer will be able to interact with the data in the observer visible stream, but the data in the other stream will be processed in a standard way for use by the archive center.

6.5.7.1 Policy for Archiving Reduced Data

Reduced data are stored in an archive to help the user of that archive make scientific decisions about the suitability of the archived data for a particular science program (in much the same way that reduced data obtained while observing are used to make scientific decisions).

When reduced data are created during an observing session the observer may have modified the recipe for his or her own purpose, so if the reduced data generated during the night are archived it would not be consistent with similar data recorded on other nights. For this reason all the data destined for the archive will be re-reduced using the *default* data reduction recipes (i.e. the ones supplied by the instrument builders unmodified). The default data reduction recipes used should be archived[1]. The data reduction recipes used by the observer should also be archived alongside the data so that an archive user may reproduce what the observer saw if desired.

This policy will only take effect when the data transport system is configured to archive reduced data.

Note that any calibration data reduced using the synchronous data reduction system will *not* be archived in reduced form, although the raw data will be.

6.5.7.2 Policy for Archiving Non-standard Detector Data

It is a requirement that *all* scientific data produced by an instrument on the Gemini telescope be included in the archive. However, some detectors may operate in a mode in which they dump data to a local disk much more rapidly that the Data Handling System

1. Note that the FITS header of reduced data should contain the name of the raw data file together with the name of the recipe used to reduce that data. See “Collection of Header Information” on page 11 - 33.

can cope with (See “Fast Data Output” on page 12 - 21). The detector array controller is responsible for storing and processing such data, but before the end of an observing session the data *must* be registered with the Data Handling System so it may be archived. The data must be stored in a format which the Data Transport System can read. One way of doing this would be to use a Data Storage Server to transfer the data from the detector’s local disk to the host disk off-line.

6.5.8 External Database Interface Libraries

Part of the Data Handling System work package is the provision of a subroutine library which other parts of the Gemini system may use to interrogate the external databases.

The Observatory Control System may need access to the following databases:

- A catalogue of guide stars.
- A catalogue of non-stellar objects.
- Ephemerides for the sun, moon and planets.
- A catalogue of spectral and flux standards.
- A catalogue of emission line nebulae (sometimes useful as wavelength standards).
- A catalogue of radial velocity standards.
- A database of previous observations made on the Gemini telescopes (to determine, for example, if an object has already been observed).
- *etc…*

The Data Reduction System may need access to the following databases:

- A catalogue of spectral and flux standards.
- A catalogue of arc line wavelengths.
- A catalogue of emission line nebulae wavelengths.
- A catalogue of the wavelengths of atmospheric lines.
- *etc…*

The current baseline for access to these catalogues is the ANSI standard Structured Query Language (SQL).

6.6 MANAGEMENT OF MORE THAN ONE INSTRUMENT

The Gemini Control System may have more than one instrument generating data at the same time (for example if an instrument does not need access to a shared resource it may be generating a calibration frame at the same time another instrument is observing the sky (See "The Configuration Dealer" on page 10 - 11 for details).

The DHS manages this simultaneous generation of data as follows:

- There is one central data transport manager and one logging system.
- There is one data storage database manager.
- Each instrument has its own:

--Quick look display server (any number)

--Synchronous data reduction server (one only).

--Data storage server (one only).

--On-line data reduction system (one only).

So the observer can have access to quick look and data reduction displays for each active instrument on the system.

6.7 THE PHILOSOPHY BEHIND THE DATA HANDLING SYSTEM

The design of the Data Handling System is based around the following philosophy:

- Data storage needs to be managed centrally and handled by one work package.

--To remove file format dependencies from the other Gemini systems and make them more flexible.

--To prevent other groups from duplicating the data storage routines.

--To ensure that files are given unique names.

--To ensure that files are stored and archived successfully.

- Data processing and display needs to be managed centrally and handled by one work package.

--To remove the need for the various instrumentation groups to write their own display software (with the exception of any real-time video display they may provide).

--To ensure the observer's display matches what is being observed.

--To ensure that files are reduced successfully, and in the correct order.

- The knowledge of how to handle different kinds of data needs to be kept in one place, where possible.

--To allow the OCS to have a uniform structure for all science observations.

--To allow the OCS to send the same sequencer commands regardless of data type.

One exception to this rule is the synchronous data reduction system, which the OCS needs to have some knowledge of, but that knowledge is kept to a minimum by means of a synchronous data reduction server.

- The internal workings of the DHS needs to be independent of software which is beyond the control of the Gemini project.

--A library is provided to allow Gemini software to access external databases. Only this library need know the exact format of these databases.

--A data reduction agent is provided between the Gemini software and the external data reduction system. Only this agent need know the exact file structure, commands and parameters required by the external data reduction system.

- Existing utilities are used wherever possible.

--The EPICS archiving utility is used for the collection of engineering data from the telescope and facility instruments.

--An existing utility is used for internal data storage and communication.

The current baseline is the “Self-defining Data System” (SDS) and “Inter-process Message Passing system” (IMP) used with DRAMA.

--Information is recorded using a commercial database.

The current baseline is SYBASE.

--An existing data reduction utility is used for the on-line data reduction system. An agent process is provided to convert file formats and coordinate the data reduction.

--Existing utilities are used for on-line data display.

The current baseline is PVWave/IDL

--Existing utilities are used to handle fault reports and software bug reports.

The current baseline is GNATS.

6.8 SUMMARY

The Data Handling System (DHS):

- coordinates the generation and storage of data, by providing data storage servers which the instruments may use to store their data to disk.
- maintains a record of the files created in a data handling database.
- allows the observer to judge the quality of the data being observed by providing quick look display servers.
- reduces data sufficiently to allow the observer to make scientific decisions, using ***recipes*** and ***rules*** to describe how each type of data should be reduced.
- saves the data to a permanent store and hence to a CADC-style archive.

- generates an electronic observing log, an engineering log and a history of each observing session.
- keeps track of faults reports.
- provides an interface to external databases.

Further details of the Data Handling System may be found in “Details of the Data Handling System” in Chapter 11.

7 High Level Instrument Control System Concepts

7.1 Introduction

This chapter gives a basic overview of the Gemini Instrument Control Systems (ICS) and their associated Detector Array Controllers (DC). A detailed description of the design of an ICS and DC is given in “Details of the Instrument Control Systems” in Chapter 12.

7.2 Changes Since Preliminary Design Review

The following changes have been made since the Preliminary Design Review (PDR) on 20 April 1994:

- The roles of the Instrument Control System and Detector Array Controller have been separated and more clearly defined.
- An Instrument Control System no longer behaves as a data server for the Data Handling System. The DHS makes available Quick Look and Data Storage servers, which an ICS can use for displaying and storing data. An ICS is now responsible for storing its own data (using the Data Storage Server).
- Data are passed from the ICS to the DHS using messages transmitted using IMP and built using SDS (IMP and SDS are part of the DRAMA system of the Anglo-Australian Observatory), rather than through a shared data buffer. The format of these messages is defined in an Interface Control Document (ICD/3 [26]).

7.3 THE ROLE OF AN INSTRUMENT CONTROL SYSTEM

Every instrument installed on the Gemini telescope needs an Instrument Control System (ICS) to control its functions. There can be several Instrument Control Systems active at any one time — one for each active instrument.

7.3.1 Categories of instrument

Instruments on the Gemini telescope fall into two broad categories:

Scientific instruments. are the ones used for astronomical research. Examples of these are the Infra-red camera and High Resolution Optical Spectrograph. At any one time there may be as many as four Cassegrain scientific instruments (one up looking, three side looking) and a single scientific instrument in the HIRES lab located in the telescope pier. It is these types of instruments which this chapter and the "Details of the Instrument Control Systems" in Chapter 12 will concentrate on.

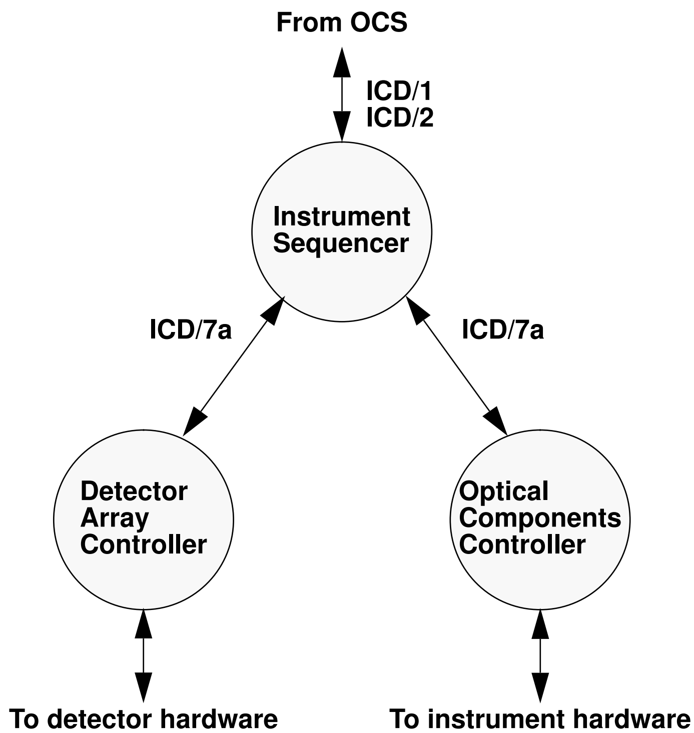
Ancillary instruments. are those parts of the Gemini system which behave like instruments, but which are used to perform some service. Examples of these are the wavefront sensors used by the Acquisition and Guidance unit. These are described in detail in the relevant telescope subsystem chapters (for example "Details of the Acquisition & Guiding System" in Chapter 19) but their control systems will have a lot in common with the ones described here.

7.3.2 The separation of instrument control and detector control

Every instrument has a detector inside it for integrating the light gathered by its optics. Different instruments may use the same kind of detector, so the roles of instrument control and detector control have been separated to allow different instruments to share the same detector controller (and to allow the systems to be provided by different work packages).

The Instrument Control System has two main subsystems separating the roles of instrument control and detector control, as shown in Figure 7 - 1 on page 7 - 3.

FIGURE 7 - 1 **The Main Subsystems of an Instrument Control System**



The labels on the figure show which Interface Control Documents (ICDs) control the flow of commands and status between the subsystems. The subsystems shown in the figure are:

Detector Array Controller (DC). This is responsible for sequencing the detector, reading and pre-processing the data from it. Some kinds of instrument, which contain more than one detector, may use more than one Detector Array Controller. For details see “The role of the Detector Array Controller” on page 7 - 6.

Optical Components Controller (OCC). This is responsible for controlling the actual mechanisms used to control the environment within and optical path through the instrument (filter wheels, shutters etc.…).

Instrument Sequencer. This is responsible for coordinating the actions of the above two controllers and tying the two together into one coherent unit. The Instrument Sequencer has an especially important role in instruments which need to synchronize detector read out with the movement of a grating, for example.

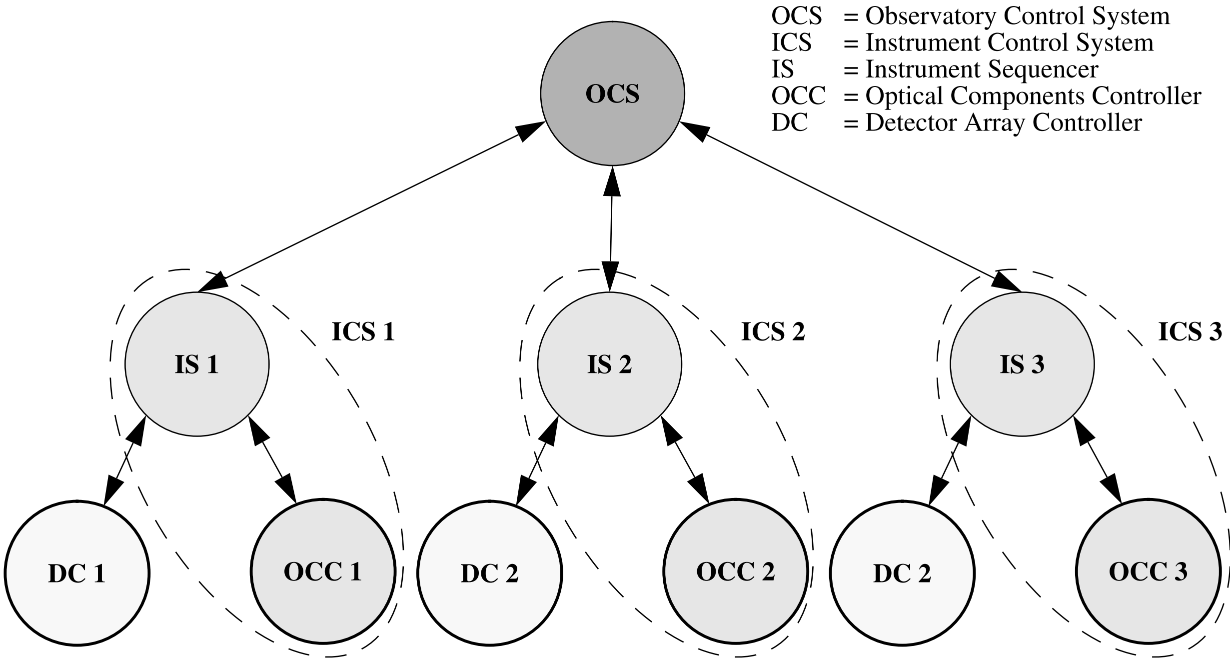
In practice an instrument group will be responsible for both the Instrument Sequencer and Optical Components Controller[1], which is why the two together are often referred to collectively as “the Instrument Control System”. See “The role of the Instrument Control System” on page 7 - 5.

The Detector Array Controller (DC) acts as a subsystem for the Instrument Sequencer in much the same way the Acquisition and Guidance unit and other telescope subsystems act as subsystems for the master controller of the Telescope Control System. The Detector Array Controller is subservient to the Instrument Sequencer, which has overall control.

Instruments using the same kind of detector will use *copies* of the same Detector Array Controller. They will not share the same actual controller. Figure 7 - 2 on page 7 - 5 illustrates the relationship between the Instrument Control Systems and Detector Array Controllers on the system. In this example there are 3 instruments on the system controlled by ICS1, ICS2 and ICS3. All three instruments are coordinated by the Observatory Control System (OCS), and the operation of each instrument is coordinated by an Instrument Sequencer (IS). Instruments 2 and 3 use the same kind of detector, controlled by DC2, so they both have their own separate *copies* of DC2.

1. The actual layout of an instrument control system may be more complex than show in Figure 7 - 1 (which gives a broad view of the main command components). A complete decomposition is shown in Figure 9, “Components of the Instrument Control System,” on page A1 - 10. All the processes shown there, except the Detector Array Controller and DHS server processes, are the responsibility of the instrument group.

FIGURE 7 - 2 **The Relationship Between Instrument and Detector Control Systems**



The ultimate role of an Instrument Control System/Detector Array Controller combination is to generate scientific data for the Observer. Each Instrument Control System acts as a data server for the rest of the Gemini Control System — it generates data in response to commands from the Observatory Control System (OCS). In order to achieve this an ICS and DC must also fulfil the following roles:

7.3.3 The role of the Instrument Control System

- To control various mechanisms which adjust the optical path between the telescope beam and the detector. The mechanisms might include shutters, moveable mirrors, filter wheels, slit wheels, gratings, and other devices.
- To provide the outside world with information on the current status of the instrument's mechanisms.
- To maintain an environment for a detector within an instrument to within acceptable limits of temperature and pressure etc.…
- To provide the outside world with information on the environment within the instrument and around the detector.

- To liaise with the Detector Array Controller for the instrument's detector
  --Commanding the detector controller.
  --Providing the detector controller with status information.
  --Obtaining status information from the detector controller.
- To accept data from the Detector Array Controller, preprocess it, and pass it on to a data storage server for storage to disk (optional).
- To pass data on to a quick look server, which will allow the observer to see the latest frame being processed by the instrument (optional).

The last two roles are optional because they vary from instrument to instrument. Some instruments will not need to preprocess their data, so the data are stored to disk directly by the detector controller.

7.3.4 The role of the Detector Array Controller

- To provide the detector with the correct electronic environment (bias voltages etc.…) to ensure it operates at its peak performance.
- To provide the observer with information on the status of the detector.
- To liaise with the Instrument Control System:
  --Accepting and obeying commands from that system
  --Providing status information back to the ICS.
- To sequence the detector with appropriate waveforms to cause it to reset, integrate and read out at the appropriate moments, which should be accurately synchronized with the needs of the instrument.
- To read out the data gathered by the detector and preprocess that data to remove any detector-specific features and reduce the image to a standard form. The data may be passed to the ICS or stored directly to disk via a data storage server.
- To provide the observer with a real-time display showing the latest frame from the detector. A DHS quick look server is available for this purpose, although a special-purpose, real-time video display may be more appropriate for some detectors.

7.3.5 Boundary Issues

Depending on the particular kind of detector and instrument being used, the boundary between the Instrument Control System (ICS) and Detector Array Controller (DC) may not be as clean cut as stated above. In these situations the instrument and detector groups should consult and decide where the boundary between their systems should be. Some examples are:

- A detector has an on-board heating device (such as resistor) used for fine control of its temperature. The resistor, being so closely integrated with the detector electronics, needs to be controller by the detector controller. In this situation the instrument cannot be solely responsible for the detector's environment. The instrument should provide the detector with a temperature slightly colder than ideal, and the detector can then use its on-board heater to tune its temperature accurately. Here it is the detector array controller that provides accurate information about the detector's temperature, not the instrument control system.
- The detector needs to be scanned across the sky in a way which is accurately synchronized with the readout from the detector. Here the instrument control system, or the telescope control system may need to be commanded directly by the detector array controller. These direct command issues are covered in ICD/5 [29].

In all the above cases the boundary between the ICS and DC should still follow the protocol described in ICD/7a [30].

Other common boundary issues which need to be resolved by instrument and detector groups are:

- Does the detector write its data directly to the data storage server, or does the instrument control system need to preprocess it?
- Who controls the length of an exposure, the instrument via a shutter mechanism or the detector controller by resetting and integrating the detector?

7.4 VARIETIES OF SCIENTIFIC INSTRUMENT

The scientific instruments placed on the Gemini telescopes come in two separate varieties:

- Conforming instruments
- Non-conforming instruments

The ancillary instruments are *all* conforming.

7.4.1 Conforming Instruments

These are instruments which are built or adapted specifically for long term use on the Gemini telescope. They include facility instruments and existing instruments which are adapted for Gemini.

A conforming instrument is built on as many Gemini standards as possible (e.g. the Gemini Electronic Design Specifications). The ICS should use one of the Gemini-supported hosts and operating systems (Currently Sun/Solaris or VME/VxWorks) and should be controlled in one of the following ways:

- The instrument may be controlled from a VME crate which is running VxWorks and EPICS. The VME hardware would be picked from the standard Gemini list of EPICS supported devices. See ICD/13 on the "Standard Control System" [37] for details.
- The instrument may be controlled from a Solaris host running a host-level EPICS channel access server. (Note that a host level channel access server is not currently available but may be available from Los Alamos National Laboratories soon on beta release).

In both cases the minimum requirement is that the ICS should act as a channel access server. If the instrument (in exceptional circumstances) needs direct access to information from the TCS database or needs to send commands to the TCS, it will also need to act as a channel access client.

7.4.2 Non-conforming Instruments

These are visitor instruments which are brought to Gemini for occasional use, or existing instruments which for some reason or other cannot be adapted to meet all the Gemini standards.

A non-conforming instrument is a self contained entity built to whatever standards the instrument team maintains internally. In order to be capable of being used on Gemini the instrument must be mechanically, electrically, and environmentally compatible. Potential instrument teams should read the Gemini Electronic Design Specifications for information on the electrical and environmental conditions. In order to be integrated into the Gemini Control System the non-conforming instrument has to support a command language and protocol generated by Gemini. It should either

- act as an EPICS channel access server; or
- accept and respond to commands sent using the DRAMA/IMP messaging system using the protocol of the "command layer server" described in ICD/1.

The detailed requirements for non-conforming instruments are described in ICD/8.

7.4.2.1 USE OF A GEMINI DETECTOR BY A NON-CONFORMING INSTRUMENT

It is possible for visitor instruments to make use of the facility Gemini detectors in one of two ways:

- A visitor "Instrument Sequencer" can communicate with a facility "Detector Array Controller" by obeying the protocol of ICD/7a.

- A visitor instrument can consist of an “Instrument Sequencer” and “Optical Components Controller” only. The facility Gemini detector will be available as another “Instrument Sequencer” and “Detector Array Controller”. In this situation, however, accurate coordination between detector and instrument will not be possible.

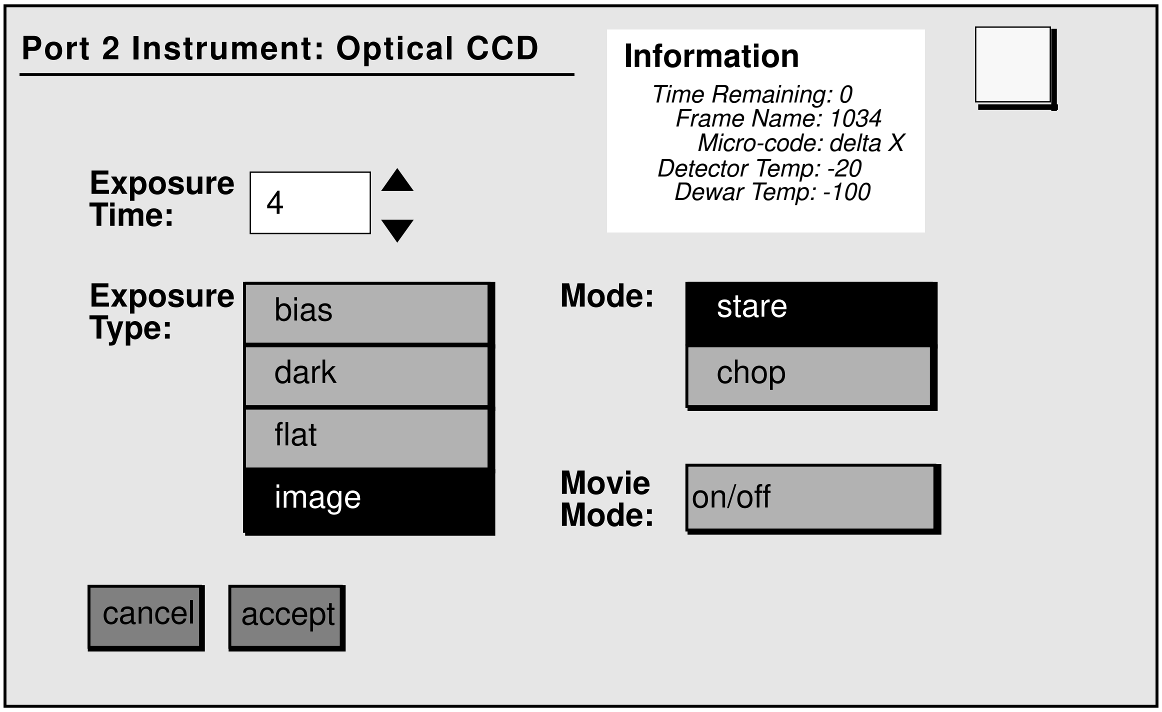
7.5 GENERAL INTERFACE TO SCIENTIFIC INSTRUMENTS

Most instruments will be treated by the Gemini Control System as data servers, and the ICS will generate data in response to commands from the OCS. In exceptional circumstances (such as the generation of a map in which the telescope and instrument need to be accurately synchronized) it may be necessary for the ICS to send commands to the TCS. In these circumstances the ICS will need to act as a channel access client. The ICS to TCS direct command interface to use for this purpose is described in ICD/6.

The OCS provides an instrument console which an observer can use to set up an instrument, in much the same way the Telescope Control System consoles are used to set up the telescope. Figure 7 - 3 on page 7 - 10 shows an example instrument console. See also “The User and Operator Graphical User Interfaces” on page 5 - 21.

The observer first creates a *science observation*, which consists of a series of *science configurations* necessary to carry out a scientific observation. Each *science configuration* describes one configuration of the various subsystems of telescope and instrument (See “Structure of Science Programs” on page 5 - 15 for details). The Observatory Control System (OCS) looks after sequencing the movement of the system from one configuration to the next as required. The Instrument Control System (ICS) is responsible for deciding exactly how the instrument should respond (e.g. which motors to move etc…) whenever the OCS requests a new configuration.

FIGURE 7 - 3 **An example instrument console**



In addition to this console, an instrument may also have a “visualization display”. This will show the observer a cartoon layout of the instrument, indicating whether shutters are open or closed, mirrors are in or out, etc.… The visualization display will be read-only for the observer, and will just report the status of the instrument. A good example of an instrument visualization display is the MIMIC display produced for the WYFFOS instrument on the William Herschel Telescope.

Configuration changes are requested using a standard set of commands (listed in “The Instrument Sequencer” on page 7 - 19). These commands are communicated from the OCS to the ICS using messages send using the DRAMA/IMP message system. An ICS running on a Unix host may intercept these IMP messages directly. If the ICS is running under VME/VxWorks, a Unix host agent process (provided as part of the OCS package) translates the commands into EPICS channel access calls. The protocol for the communication of commands is described in ICD/1 [24].

During the "engineering/acceptance" and "commissioning" phases, as well as during periods of maintenance during the "operational/maintenance" phase (See "Operational Phases" on page 3 - 1), a much more real-time environment is needed for checking out and testing the instrument. For conforming instruments the EPICS Graphical User Interface (GUI) tools (such as EPICS DM) are used. These GUI tools require an external Sun Workstation. The goal is for all of the functionality of a conforming instrument to be available through the EPICS interface. A non-conforming instrument uses the instrument team's native interface (i.e. whatever they use when they are not at Gemini).

7.6 REQUIREMENTS FOR NEW INSTRUMENTS

When a new instrument is developed the developers need to supply the following information to allow the instrument to be slotted into the Gemini Control System:

- A unique name for the instrument (which will be appraised by the Gemini project office).
- A list of parameters that define all the modifiable attributes for that instrument. For example the sort of information contained in this table would be…

Name	**Description**	**Type**	**Restrictions**
SHUTTER	Shutter position	BINARY	[open | closed]
FILTER	Filter combination	CHAR	Choice defined at setup.
SWIDTH	Slit width (microns)	FLOAT	5.0-5000.0
SLENGTH	Slit length (microns)	FLOAT	1000.0-250000.0
GORDER	Grating order	INTEGER	1-4
GANGLE	Grating angle (degrees)	FLOAT	0.0-165.0
etc.…	etc.…	etc.…	etc.…

The exact parameters will vary from instrument to instrument. The names should correspond to those of records in the instrument's EPICS database. These parameters may be used by the OCS to form configurations for that instrument.

- A list of the parameters which the instrument will provide as status information. The subset of these which are to be displayed on the instrument console in the OCS, and the different subset which should be included in the FITS header of data taken with this instrument should be identified, together with the appropriate FITS keywords. For example the sort of information contained in this table:

FITS keyword	Location	Type	FITS header?	Instrument console?	FITS Comment
UTSTART	UTSTART	CHAR	✓		HROS Time when shutter opened.
UTEND	UTEND	CHAR	✓		HROS Time when shutter closed.
ELAPSED	ELAPSED	FLOAT	✓		HROS Elapsed time (seconds).
DARKTIME	DARKTIME	FLOAT	✓		HROS Dark current accumulation time (seconds).
EXPTIME	EXPTIME	FLOAT	✓		HROS Exposure time (seconds).
	COUNTDOWN	FLOAT		✓	HROS Countdown to end of exposure (seconds).
HRS-FOCS	FOCUS	FLOAT	✓	✓	HROS Focus position.
HRS-FILT	FILTER	CHAR	✓	✓	HROS Filter combination.
HRS-FW1	FILTER1	CHAR		✓	HROS Filter wheel 1 position.
HRS-FE1	FENCRD1	INTEGER		✓	HROS Filter wheel 1 encoder position.
HRS-FW2	FILTER2	CHAR		✓	HROS Filter wheel 2 position.
HRS-FE2	FENCRD2	INTEGER		✓	HROS Filter wheel 2 encoder position.
HRS-SW	SWIDTH	FLOAT	✓	✓	HROS Slit width (microns).
HRS-SL	SLENGTH	FLOAT	✓	✓	HROS Slit length (microns).
HRS-GO	GORDER	INTEGER	✓	✓	HROS Grating order.
HRS-GA	GANGLE	FLOAT	✓	✓	HROS Grating angle (degrees).
etc.…	etc.…	etc.…			etc.…

The columns in this table are as follows:

FITS keyword. Is the FITS keyword to be used when this item is included in a FITS header. This column should only be blank if the item will *never* be included in a FITS header (for example the exposure countdown). All other items should have a FITS keyword standing by, even if their “FITS header?” column is blank. If possible there should be a sensible relation between the name of the EPICS record and the name of the FITS keyword, but this may not always be possible. Note that a prefix based on the instrument name has been included in the instrument-specific FITS keyword to ensure the keyword names are unique. Including the instrument name in the FITS comment helps identify the source of the information as well. See “Standard FITS Keywords” on page 11 - 38 for a list of keywords which an instrument is expected to provide.

Location. The location in the instrument’s EPICS database where the value may be found.

Type. The type of the value (CHAR, INTEGER, LOGICAL, FLOAT).

FITS header? This column contains a ✓ if the item has been selected to appear in the FITS header. Instrument groups should supply a list of the default items to appear in the header. The other items can be written into the header by configuring the OCS to read them.

Instrument console? A ✓ in this column indicates an item that the instrument developer would like to see displayed on this instrument’s console within the Observing Tool.

The information in the above table is sometimes referred to as the “Packet Description File” (PDF) as it describes a way of obtaining a “packet” of FITS information. All the above information must be updated by the ICS before signalling the end of an exposure.

See also Table 11 - 7, “Standard FITS keywords provided by the ICS or DC,” on page 11 - 43.

- A list of the types of data the instrument is capable of producing, and a description of the recipes required to reduce those types of observations. See “Automating the Data Reduction” on page 6 - 19 for more details.
- A description of the way in which the instrument needs to be calibrated by the Synchronous Data Reduction system, and some IDL scripts for performing the calibration. See “Synchronous data reduction server” on page 6 - 10 for more details.
- A setup file (ASCII recommended) containing information such as

  --A list of available filter combinations

  --A lookup table converting filter combination into actual filter wheel positions.

  --Lookup tables for converting filter wheel positions, grating position etc… into motor encoder readings.

  --The addresses of the various ports used, and other constants.

  --etc…

  The ICS is responsible for decoding the contents of this file. The name of the file can be provided by the OCS as part of the instrument configuration at start-up.
- An ICS command interpreter which understands at least the basic sequencer commands listed in “The Instrument Sequencer” on page 7 - 19.

7.7 REQUIREMENTS FOR NEW DETECTOR ARRAY CONTROLLERS

The requirements on a new Detector Array Controller are similar to those of a new instrument controller. The developers need to supply the following information to allow the controller to be slotted into the Gemini Control System:

- A unique name for the controller (which will be appraised by the Gemini project office).
- A list of parameters that define all the modifiable attributes for that detector controller. For example the sort of information contained in this table would be…

Name	Description	Type	Restrictions
BVOLTAGE	Bias voltage (volts)	FLOAT	0.0-10.0
BGAIN	Bias amplifier gain	INTEGER	1-40
NREADS	Reads per exposure	INTEGER	1,2,4,8,16,32,64,128
RDINT	Time interval between reads (milliseconds)	FLOAT	10.0-10000.0
XSTART	X coord of start of region of interest on detector	INTEGER	1-1024
YSTART	Y coord of start of region of interest on detector	INTEGER	1-1024
XEND	X coord of end of region of interest on detector	INTEGER	1-1024
YEND	Y coord of end of region of interest on detector	INTEGER	1-1024
etc.…	etc.…	etc.…	etc.…

The exact parameters will vary from detector controller to detector controller. Just as with an instrument, the names should correspond to those of records in the detector controller's EPICS database. These parameters may be used by the OCS to form configurations for that detector controller. These configurations are treated by the OCS as part of the instrument configuration, but they will be forwarded to the detector controller by the ICS.

- A list of the parameters which the detector controller will provide as status information. The subset of these which are to be displayed on the instrument console in the OCS, and the different subset which should be included in the FITS header of data taken with the instrument using this detector should be identified, together with the appropriate FITS keywords. For example the sort of information contained in this table…

FITS keyword	Location	Type	FITS header?	Instrument console?	FITS Comment
UTSTART	UTSTART	CHAR	✓		ALICE Time when exposure started.
UTEND	UTEND	CHAR	✓		ALICE Time when exposure ended.
ELAPSED	ELAPSED	FLOAT	✓		ALICE Elapsed time (seconds).
DARKTIME	DARKTIME	FLOAT	✓		ALICE Dark current accumulation time (seconds).
EXPTIME	EXPTIME	FLOAT	✓		ALICE Exposure time (seconds).
	COUNTDOWN	FLOAT		✓	ALICE Countdown to end of exposure (seconds).
AL-BVOLT	BVOLTAGE	FLOAT	✓	✓	ALICE Bias voltage (volts).
AL-BGAIN	BGAIN	INTEGER	✓	✓	ALICE Bias amplifier gain.
AL-NREAD	NREADS	INTEGER	✓	✓	ALICE Reads per exposure.
AL-RDINT	RDINT	FLOAT	✓	✓	ALICE Time interval between reads (milliseconds).
etc.…	etc.…	etc.…			etc.…

The columns have the same meanings as those used for an instrument. Note that the FITS keywords UTSTART, UTEND, ELAPSED, DARKTIME and EXPTIME have been used in these examples by both the ICS and DC. Information about the exposure time could come from the ICS or the DC, depending on the properties of the instrument or the detector (because an exposure could be started by the instrument opening and closing a shutter or by the detector controller resetting and integrating the detector). The instrument and detector groups need to agree on who provides exposure time information.

All the above information must be updated by the detector controller before signalling the end of an exposure.

- Information on the maximum size of the data array generated by the detector[1] controller, and the ideal environment for the detector e.g. (temperature and pressure) which an instrument must try and maintain.

1. The actual size of any array generated by the detector controller is contained within the data structure, as described in ICD/3 [26]. The maximum size is needed so the Data Handling System can estimate if there is sufficient disk space for the next observation.

- A setup file (ASCII recommended) containing information such as

  --A list of available detector waveforms and their uses, together with pointers to other setup files containing the waveforms.

  --The addresses of the various ports used, and other constants.

  --etc…

  The DC is responsible for decoding the contents of this file. The name of the file can be provided by the OCS (via the ICS) as part of the detector configuration at start-up.

- An ICS command interpreter which understands at least the basic sequencer commands listed in “The Instrument Sequencer” on page 7 - 19 and described in ICD/7a [30].

  The Detector Array Controller may have a series of special-purpose commands, in addition to the basic sequencer commands, which an ICS may wish to use. It is up to the instrument and detector groups to liaise together to agree on these commands.

The details of the interface between a Detector Array Controller and Instrument Control System may be found in ICD/7a [30].

7.8 INTERFACE TO ONBOARD WAVE FRONT SENSORS

There is a requirement for some scientific instruments to support an onboard wavefront sensor. This wavefront sensor is a logically separate control system which must have access to the reflective memory bus in order to pass this information to the other systems.

The onboard wavefront sensor may be a clone of the A&G wave front sensors. The only difference may be that some of the commands for manipulating the probes etc… may have to come from the instrument controller. The configuration of any wave front sensor is regarded as part the instrument configuration.

The details of the way in which wavefront information is exchanged between Gemini systems is described in ICD/5 [28].

This subject is under discussion.

7.9 INTERFACE TO CHOPPING AND NODDING

Chopping and nodding require accurate synchronization of the instrument and telescope. Chopping is achieved by moving the secondary mirror, and can be maintained at a high frequency (see Table 18 - 2, “Chopper Functions,” on page 18 - 13). Nodding is achieved by moving the whole telescope mount (See “Details of the Mount

Control System” in Chapter 14), and is therefore a much slower operation. Chopping and nodding movements can take place alone, or they can be combined together. If combined with chopping, nodding will take place after a whole number of chop cycles.

7.9.1 Chopping with Gemini Instruments

Three kinds of chopping are supported by the Secondary Control System (SCS, see “Details of the Secondary Control System” in Chapter 18 for details):

- Two position, i.e. A-B-A-B…
- Three position, i.e. A-center-B-center-A-center-B…
- Saw tooth chopping, in which the chopper is drifted from A to B and back at a constant rate.

The chopper mode, throw, frequency and position angle are included in the SCS configuration sent by the TCS. Once the SCS has been set up with this information a change in chopper position can be demanded by:

- A digital signal generated by the SCS on the event bus.

  The SCS uses its own internal clock to make chopper changes at the specified frequency. It makes a digital “chopper in position” signal available on the event bus when the chopper has achieved the intended position and has settled down. See ICD/11 [35] for details of the event bus.
- An analogue (probably square wave) signal from anywhere (e.g. the DC or ICS) on the event bus.

  The ICS or DC (usually the DC) can move the chopper by applying a signal to the event bus. Only the frequency of this signal is significant. The amplitude is ignored, and the signal is sensed as being negative, zero or positive (the zero being used for the center of the three position chop mode). The SCS will still provide a “chopper in position” signal on the event bus. The OCS will allow the SCS to regain control of the chopper when another instrument requires the telescope beam.

  For a saw tooth chop, the change in state of the event bus is used to reverse the direction of motion of the secondary.

Instrument developers are encouraged to use the first method where possible, and allow the TCS/SCS to control the chopper.

The exact details of the chopping mechanisms are under discussion.

7.9.2 Nodding with Gemini Instruments

While nodding, the telescope can be cycled through a list of arbitrary (RA,Dec) positions, allowing any pattern on the sky to be sequenced. At least two positions should be supplied. Once the last position in the list is reached the telescope steps back to the first position, and a new cycle begins. A telescope beam change can be demanded by:

- Commands sent to the TCS from the OCS as part of a sequence recipe. An example sequence recipe would be:

```
Issue APPLY
For each (RA,DEC) in the list
    Move telescope to the next position.
    For each observation at this position
        Issue OBSERVE
        Issue ENDOBSERVE
    Next observation
Next nod position
```

The exact recipe depends on how large the telescope movements are, and whether a new guide star needs to be acquired after each movement. This is the normal way in which nodding will be handled on the Gemini system.

- Direct commands sent to the TCS by the ICS.

During times where accurate synchronization between the ICS and TCS is critical, the OCS can temporarily relinquish control of the TCS and allow TCS commands to be sent directly from the ICS. In this mode there is great flexibility. The ICS can request telescope beam changes or it can request other more unusual operations, (such as a drift between two positions at a constant angular rate).

Note that this mode should *only* be used where accurate synchronization is essential. The OCS will take control of the TCS again once the observation has completed.

For conforming instruments the commands should be sent using EPICS channel access. Non conforming instruments can use IMP (if they are not based on EPICS); however the performance penalty[1] they would get by doing this is such that there is probably no justification for direct TCS control by non-conforming instruments.

1. Note that IMP is a fast messaging system. The performance penalty arises from having to ascend from the real-time layer to the command layer in order to send the messages.

Each time it is commanded to move, the TCS makes available a “telescope moving” and a “telescope in position” signal on the event bus, which instruments can monitor. See ICD/11 [35] for details.

Nodding across the zenith should be avoided wherever possible, as should nodding close to any zone of avoidance (such as the horizon).

The exact details of the nodding mechanisms are under discussion.

7.10 THE INSTRUMENT SEQUENCER

The Observatory Control System (OCS) has its Observing Database loaded with the configuration information required by each active instrument — this information comes from the instrument configuration part of the *science configuration* which is driving the observation. The OCS provides the instrument with the science configuration and then issues a sequence of the following commands in order to run the instrument All instruments therefore need to respond at least to the following high level commands:

- TEST; test all components.
- INIT; set up the instrument and read the hardware configuration file.
- RESET; change to the start-up configuration.
- CHECK; check a target configuration is feasible.
- APPLY; move to a target configuration.
- GUIDE; activate any on-board wavefront sensor.
- ENDGUIDE; deactivate any on-board wavefront sensor.
- OBSERVE; make an exposure.
- ENDOBSERVE; tidy up after an exposure.
- PAUSE; pause an exposure.
- CONTINUE; continue an exposure.
- STOP; cancel an exposure.
- FLUSH; complete any pending data processing operations and store the data generated by an exposure.
- ABORT; emergency stop.
- PARK; adopt a safe configuration.

See “Standard Commands” on page 12 - 5 for more details on these commands. In addition to the above, each instrument may have its own set of unique commands which are specific to its functions.

7.11 DELIVERABLES

7.11.1 What is Provided by Gemini

Gemini (and the work packages developed on behalf of Gemini) provides the following resources to instrument developers:

- Documentation, as listed in “References” in Chapter 2:

  --This Software Design Description.

  --Interface Control Documents (ICDs) [24] to [37].

  --Scenarios and Walkthroughs.

  --Electronic Design Specifications, [52].

  --Programming Standards, [56].

  --*etc.…*

- A standard control system consisting of VME crate/CPU modules with EPICS and VxWorks. We provide a hardware list, software and hardware installation instructions, as well as consultation etc… See ICD/13 [37] for details.

- Standard buses:

  --Event bus — chop, nod/dither, shutter status, ICD/11 [35].

  --Synchronization bus — wavefront information, ICD/10 [34].

  --Time Bus — UTC from GPS, ICD/9 [33].

  --Interlock System — safety, ICD/12[36].

  --Control LAN — command/status.

  --Data LAN — bulk data.

- Standard EPICS:

  --Standard IOC records [16].

  --Channel access [12].

  --GUI tools [18].

  --Alarm manager [15]

  --Archiving and logging tools [17]

  --*Plus other things* (see [11]).

- EPICS driver creation and development via a subcontract. We do not expect to develop many drivers, so it makes sense to concentrate that expertise in 1-2 people. See ICD/13 [37] for details.

- EPICS extensions, such as the proposed CIO record for handling commands, [24], the SIR record for handling status and alarms, [25], and the LOG record for logging messages, [27]. These will be developed by the Gemini project office.
- A "Command Layer Agent" process for converting IMP messages into EPICS channel access. A conforming instrument only needs to present a channel access interface to the rest of the Gemini system. See ICD/1 [24] for details.
- A quick look server for data display (DHS work package).
- A data storage server for storage of data to disk (DHS work package).
- Standard software libraries and tools containing code from previous work packages. These include:

  --Libraries for packaging and transmitting data using SDS and IMP (DHS work package).

  --Libraries for reading and writing the FITS and SDS format files created by the data storage server (DHS work package).

  --Data processing libraries (DHS work package and external data reduction systems).

  --Tools for creating instrument consoles (EPICS consoles — "Standard Control System" work package and other ICS groups; TCL/Tk consoles — OCS work package).
- A core instrument controller skeleton. This is a template which runs on top of the standard control system containing sufficient code to enable it to receive the standard sequencer commands from the OCS. There will be two forms of core instrument skeleton:

  --An Instrument Control System skeleton.

  --A Detector Array Controller skeleton.

  An instrument developer will be able to edit this template, add new commands, and add the code for actually controlling the instrument. Developers will not be forced to use this skeleton. It will be provided as a useful example.

  The details of the core instrument skeleton *are yet to be decided.*

7.11.2 What an Instrument Developer Should Provide

Instrument Control System developers should provide:

- An EPICS system based on the core instrument skeleton (when available) and built using the standard drivers and hardware provided by Gemini. The EPICS system should be capable of driving all the mechanisms and electronics associated with the instrument and (if necessary) preprocessing the data from that instrument, in response to OCS sequencer commands. The ICS must propagate any data quality and variance information provided by the Detector Array Controller.
- A list of parameters defining the modifiable attributes for that instrument (see "Requirements for New Instruments" on page 7 - 11).

- A “packet description file” containing a list of parameters which the instrument can provide as status information and/or header information. (see “Requirements for New Instruments” on page 7 - 11).
- An engineering console for testing and commissioning the instrument, based on the EPICS GUI tools (e.g. EPICS DM).
- A list of the data types the instrument can generate, together with a

  --data reduction recipe

  --search rule

  for each one.

7.11.3 What a Detector Array Controller Developer Should Provide

Detector Array Controller developers should provide:

- An EPICS system based on the core instrument skeleton (when available) and built using the standard drivers and hardware provided by Gemini. The EPICS system should be capable of sequencing the detector and reading it out. The system should also preprocess the data from that detector, removing any electronic signatures that cannot be removed by a simple BIAS or DARK subtraction, and removing knowledge of the structure of the detector, producing a whole array in standard format.
- A map of the bad pixels on the detector, in the form of a data quality array transmitted with the data.
- An estimate of the variance of the data (if appropriate), in the form of a variance array transmitted with the data.
- Information on the environment needed by the detector — the mechanical, electrical and physics (temperature and pressure) environment required.
- A list of parameters defining the modifiable attributes for the detector (see “Requirements for New Detector Array Controllers” on page 7 - 14).
- A “packet description file” containing a list of parameters which the instrument can provide as status information and/or header information. (see “Requirements for New Detector Array Controllers” on page 7 - 14).
- An engineering console for testing and commissioning the detector, based on the EPICS GUI tools (e.g. EPICS DM).
- A contribution to the data reduction recipes and rules for instrument’s using that detector. Instrument Control System developers can incorporate these into their recipes and rules, but they have the final say in exactly how data from their instrument should be processed.

8 High Level Telescope Control System Concepts

8.1 Changes Since Preliminary Design Review

There have been no changes of substance to this chapter since the PDR.

8.2 The Telescope Control System

The task of the TCS can be stated in general as “Take the target position, specified in a variety of defined astronomical coordinate systems, and calculate the mount, instrument rotator, and optical surface positions required for the target to be imaged perfectly at a given point in the focal plane.

The positional and optical quality loops (below) are essentially the same, given that some actions affect both image position and shape. The key difference is between operation with and without wavefront sensor error feedback - all we are doing is generalizing from a system which only worried about the first two Zernike polynomials (tip and tilt) to one which has to deal with a larger set.

The telescope control system has the following roles.

8.2.1 Pointing Error Budget

The Telescope Control System (TCS) must take the target right ascension (RA) and declination (DEC) and calculate the encoder values to use to have that particular RA and DEC fall on the pointing reference axis of the telescope.The TCS must point the telescope such that the r.m.s. pointing error after correcting for all repeatable and predictable errors is less than 3 arcseconds. This requires an accurate model of these errors.

In order to maintain the pointing, the TCS coordinates the positions of three different subsystems; the mount, consisting of altitude, azimuth, and instrument rotator; the primary mirror, consisting of tip, tilt, translation, and piston; and the secondary, consisting of tip, tilt, translation, and piston.

8.2.2 Tracking Error Budget

The TCS must keep the telescope line of sight within the error budget allocated, currently about 30 milliarcseconds r.m.s. This requires both an accurate model of the repeatable and predictable tracking errors as well as access to a focal plane reference (autoguide) in order to make real time corrections.

8.2.3 Image Quality Error Budget

The TCS must also maintain the image quality at the same time as it maintains the telescope line of sight. This requires open loop correction of the optical alignment and primary mirror figure, as well as environmental control. The enclosure control system, with access to the wind blinds, wind gates and ventilation systems, is available to the TCS for adjusting environmental conditions within a limited range.

In order to keep the system within the error budgets above the TCS can be viewed as a number of different parts working towards the above goals:

- a pointing, open loop tracking manager, responsible for obtaining and maintaining the telescope line of sight given a target RA/DEC.
- a closed loop tracking control manager, responsible for maintaining the telescope line of sight given a focal plane guide star
- an optical surfaces control manager, responsible for maintaining the optical alignment and mirror figure given a focal plane wave front sensor for closed loop, open loop with mirror model (LUT) tied to pointing, temperature, etc.
- an environment control manager, responsible for maintaining the enclosure, telescope, and mirror environment in order to minimize the effect of temperature and wind on the image quality

8.3 TELESCOPE CONTROL OVERVIEW

The Gemini control system can be run in a number of different modes depending on which servo loops are closed. It is important to note that, on an alt-azimuth telescope with control of the position of the primary and secondary mirrors, it is possible to correct for changes in incoming wavefront with more than a single control motion. Conversely, given a change in the incoming wavefront, it is not always possible to decide *a priori* which control system to use to apply a correction.

Consider the case where the secondary translates relative to its ideal position. This manifests itself as an increase in the tip and coma terms in the wavefront. If one was not watching the coma term then tipping the secondary corrects the tip component but increases the coma component. In order to make the correct decision requires monitoring both components.

8.3.1 Operating Modes of Telescope

For the purpose of this paper the generic term "look up table" (model) is used to refer to a combination of analytical model, polynomial fit, and a look up table used to predict the position of a component based on the current state of the system.

The telescope operates in the following modes:

- raw mode

  System runs off of the encoders with no corrections. System does not run in this mode once initial calibration work done during commissioning has finished.

- open loop mode

  No guide star available, all control systems run off look up tables (which are also known as "pointing maps").

- closed loop mode

  Guide star available but either too faint for tip/tilt correction or tip/tilt not supported/in use for this focus (for instance at f/6). Star position is used to inject slow frequency corrections (< 0.1 Hz) into alignment of mount and optics. Limit on mount correction bandwidth is imposed to minimize induced vibrations in the structure.

- tip/tilt mode

  Guide star (within isokinetic patch if atmospheric turbulence as well as wind shake is to be removed) is sampled at up to 200 Hz for tip/tilt and information is used for fast steering of secondary with a bandwidth of up to 40 Hz (adjustable in use). Either DC tip/tilt offset of secondary or filtered tip/tilt signal is used to correct mount position. If two guide probes are in use then difference of the signals can be used to correct for rotation errors in the instrument rotator.

- fast focus mode

  Fast tip/tilt sensor supplying tip/tilt information also supplies focus information at a up to the tip/tilt sampling rate (200 Hz). Focus information is used to correct the secondary focus position at a bandwidth of less than or equal 40 Hz.

- active optics mode (normal operating mode of telescope)

Fast tip/tilt and focus star is integrated long enough (~60 sec) to remove effects of seeing (no need for this object to be within isokinetic patch). Coma component may be used to correct translation of secondary mirror. Astigmatism and higher components may be used to correct figure of primary mirror. DC offset of the primary figure may (TBD) be used to correct primary-secondary alignment.

- adaptive optics mode

  Used when a bright star is available. Deformable mirror with tip/tilt optic used to extend tip/tilt correction to higher frequencies and wavefront correction to more modes. DC position of tip/tilt optic (if present) or low pass filter of tip/tilt signal may be used to correct or drive secondary tip/tilt position. DC focus term of deformable mirror or low pass filter of focus term may be used to correct secondary focus. DC figure terms of deformable mirror or low pass filter of input wavefront may be used to correct primary figure.

8.3.2 Normal Operating Mode of the Telescope

The normal operating mode of the telescope is with the tip/tilt loop, fast focus loop, and active optics loop closed.

8.3.3 Control Systems Available

The following control systems are available for use in the different modes of operation:

- mount altitude drives (4 motors)
- mount azimuth drives (8 motors)
- cassegrain rotator (4 motors)
- secondary slow 5 axis articulation
- secondary fast tip/tilt/piston
- some scientific instruments can do fast shift and add which can provide an equivalent control system to the secondary fast tip/tilt system
- primary passive pneumatic support providing 80% of axial support at zenith with controlled pressure
- primary passive hydraulic wiffletree providing 20% of axial support with controlled tip/tilt/piston
- primary passive hydraulic lateral support with controlled translation
- primary active axial pneumatic support for control of primary figure
- adaptive optics deformable mirror and (TBD) internal tip/tilt mirror
- enclosure windscreens (both upper and lower)

- enclosure wind gates
- enclosure ventilation systems (dome skin, flushing fans, etc.)

8.3.4 Sensors Available

The following sensors are available for use:

- Acquisition and Guidance (A&G) fast wave front sensor - provides low spatial order, tip/tilt and focus, error at up to 200 Hz in parallel with science observing
- A&G slow wave front sensor - medium spatial order (~ 30 spots or equivalent) read out at 5-200 seconds to provide tip/tilt, focus, astigmatism, and coma in parallel with science observing (the fast and slow A&G sensors may be identical devices)
- A&G high resolution wave front sensor - provides high spatial order (~400 spots or equivalent) map of wavefront errors; precludes science observing while in beam
- scientific instruments which are read out quickly (such as long wavelength infrared cameras) can provide the equivalent of tip/tilt information which could be used to drive the secondary or other control systems.
- on board wavefront sensors - for some scientific instruments there is a need to share the focal plane with the wave front sensor in order to obtain objects nearly on axis. These wave front sensors are functionally equivalent to the A&G fast wave front sensor.
- adaptive optics wave front sensor - provides medium spatial order map of wavefront errors at very high rates (< 1 KHz) {This may be an onboard WFS or a facility WFS (TBD).}
- tape and friction encoders, fiducial system on altitude, azimuth, and instrument rotator axes
- position transducers and temperature measurement sensors for Primary and Secondary.
- laser measurement of secondary position, gyros, tilt meters, accelerometers, strain gauges (all TBD)

8.3.5 Models Available

In order to correct the predictable and repeatable errors of the telescope (aka pointing errors - but no longer limited to just pointing) a number of models have to be developed. These models use physical and geometrical terms based on the appropriate functions wherever possible and only resort to look up tables and/or polynomial expressions for the predictable residuals from the physical model.

In general these models are based on Altitude and Azimuth grids on the sky.

8.4 OPERATING MODES OF THE TELESCOPE

8.4.1 Open Loop Mode

8.4.1.1 POINTING

In open loop mode there is no guide star and all the systems run from the appropriate models. The TCS uses these models to command the next position of the servo system. If required there can be temperature corrections for the models as well.

8.4.1.2 OPTICAL SURFACES

The dominant contributions to image centroid motions during open loop mode are wind shake and telescope flexure. In open loop mode there is no star on the wavefront sensor and the primary active and passive system run from models. The TCS uses these models to command the next position set of the active actuators and the passive system in order to maintain the primary figure at a nominal performance.

The dominant contributions to image degradation during open loop mode are gravitational warping of the mirror cell and higher order (focus and above) atmospheric effects.

8.4.2 Closed Loop Mode

8.4.2.1 POINTING

In closed loop mode a guide star is available for making corrections but there is no fast tip/tilt of the secondary available. The position of the guide star in the focal plane is used to make relatively slow corrections to the position of the mount, primary, secondary, and rotator which are required due to (a) errors in the models and (b) non-repeatable errors such as hysteresis and wind shake.

During this mode all corrections are performed by slow tip/tilting of the secondary. If required the DC offset of the secondary slow tip/tilt mechanism can be used to make corrections to the mount.

8.4.2.2 OPTICAL SURFACES

During closed loop mode the models are used to feed forward position and velocity information to the servo system. The guide star information is used to make corrections to the model.

This servo improves the windshake performance but the major contributors to image centroid motion are still windshake and atmospheric tip/tilt.

8.4.3 Tip/Tilt Secondary Mode

In tip/tilt mode a guide star within the isokinetic patch is used by an on board WFS or a peripheral guider to generate corrections to the tip/tilt of the secondary in order to maintain the image centroid. The main sources of error to be corrected are wind shake of the telescope/optics and atmospheric turbulence induced motion of the image centroid. This information can come from a dedicated X/Y guider or it may come from the wavefront or curvature sensor. In this mode of operation the adaptive optics system is not operating.

During this mode all corrections are made by tip/tilting the secondary. If required the DC tilt offset of the secondary can be used to inject corrections to the slow secondary “5 Degrees Of Freedom” (5 DOF) mount and the telescope drives. The models are used as in closed loop mode.

It is the goal of the models to model the repeatable errors such that DC corrections are unnecessary. It is the goal of the telescope design that it is stiff enough that dynamic errors and non-repeatable errors are small enough such that the tip/tilt corrections for image centroid line of sight do not cause appreciable image degradation.

We model this as a closed loop servo with a bandwidth of 40 Hz and a sampling rate of 200 Hz. The bandwidth is driven by the requirement to reduce the windshake to fit within the error budget. The sampling rate is driven by the requirement to have a sampling rate that is at least 5 times the servo bandwidth in order to have a stable servo system.

Once the tip/tilt system is activated the dominant contribution to encircled energy diameter is the higher order effects of the atmosphere.

8.4.4 Fast Focus Mode

In this mode the star which is being used to generate tip/tilt information is also used to provide focus information at or less than 200 Hz). This focus information is used to make corrections to the focus position of the secondary by making offsets in the actuators of the secondary tip/tilt system.

We model this as a closed loop servo with a bandwidth of upto 40 Hz and a sampling rate of upto 200 Hz. The effects of a changing focus do not have large effects in open loop or closed loop mode because these modes are dominated by wind shake. The improvement in image quality due to the tip/tilt system results in the focus effects becoming a significant contributor to image quality.

The major impact on telescope focus is thermal changes in the optical support structure - these can be adequately reduced by thermal sensors and calibration - however extra margin is introduced with the fast focus system.

Wind buffeting and atmospheric higher order effects (beyond tip/tilt) cause a significant focus component degradation to image quality relative to the error budget. These effects can be reduced substantially by the use of a fast focus system with a few Hertz closed loop bandwidth.

8.4.5 Active Optics Mode

In this mode an off-axis star is observed with a wavefront sensor and the information derived from that signal is used to make corrections to the primary figure. The adaptive optics system is not used in the active optics mode. It is necessary to integrate long enough to remove average out the effects of seeing.

During this mode all corrections are made by altering the figure of the primary. If required the DC offset of the primary figure can be used to make corrections to the 5 DOF mounts of the primary and secondary - but it is the goal of the appropriate models to model the repeatable errors such that corrections are unnecessary. It is the goal of the primary support system that it is stiff enough that dynamic errors and non-repeatable errors are small enough such that corrections are unnecessary.

The primary axial model is used during this mode to feed forward corrections to the primary figure. The wavefront sensor information is used to make corrections to the model and to update its current value. The model is capable of working in both an absolute mode and in an incremental mode.

All tilt and wavefront information from the Acquisition and Guide unit (A&G), On Board Wavefront Sensors, and Adaptive Optics Wavefront Sensors are rotated to apply to the fixed reference frame of the primary mirror. This is handled by the Telescope Control System as it collects all of the relevant position and orientation data.

8.4.6 Adaptive Optics Mode

In this mode a star is used as a probe of the wavefront errors at a high sampling rate. This information is used to manipulate an internal tip/tilt mirror (optional, TBD) and a deformable mirror provided with the AO system.

During this mode all tip/tilt corrections may be made with the AO small internal tip/tilt mirror. If required the DC offset of this mirror can be used to make corrections to the tip/tilt position of the secondary. This in turn can make corrections to the mount if needed. It will be possible to operate without an AO tip/tilt optic and to rely on the secondary or to use a Scientific Instrument's shift and add capability.

During this mode all wavefront corrections are made with the deformable mirror. If required the DC offsets of the deformable mirror can be used to make corrections to the primary figure, primary 5 DOF system, and the secondary 5 DOF system.

The appropriate models are used in this mode to feed forward position and velocity. However the secondary tip/tilt system is not used other than as a follower to the adaptive tip/tilt mirror DC position (if in use).

The primary axial support model is used in this mode to feed forward corrections to the primary figure. The sensor information is used to make corrections to the model position and to update its current value. The model is capable of working in both an absolute mode and in an incremental mode. Figure 8-1 shows an overview diagram of the control system in its active mode. It is important to note that system design and architecture allows any sensor (or combination of sensors) to be routed to any of the control systems.

8.5 MODEL FOR THE GEMINI TELESCOPES

8.5.1 Pointing/Tracking Model

The TCS must accept target and guide star positions and calculate servo positioning information for the various control subsystems. The process for target position transformation can be described as a hierarchy of processing loops with increasing performance requirements as shown in Figure 12-1. Transformations on guide star position information is quite similar, with the lower, faster steps omitted. The TCS treats the position of the telescope line of sight and the guide probes as separate pointing models.

When a focal plane guide star is available the TCS must use this information (or direct this information) to the appropriate subsystems in order to improve the tracking performance of the telescope.

8.5.2 Optical Surfaces Model

The TCS must generate target wavefront and translation information for the primary mirror in order to maintain the primary figure and the optical alignment within acceptable limits.

When a focal plane star is available the TCS must use this information to improve the look up tables used when a guide star is not present.

The TCS will also generate target translations, piston, and tilts for the secondary mirror in order to maintain optical alignment.

8.5.3 Environment Model

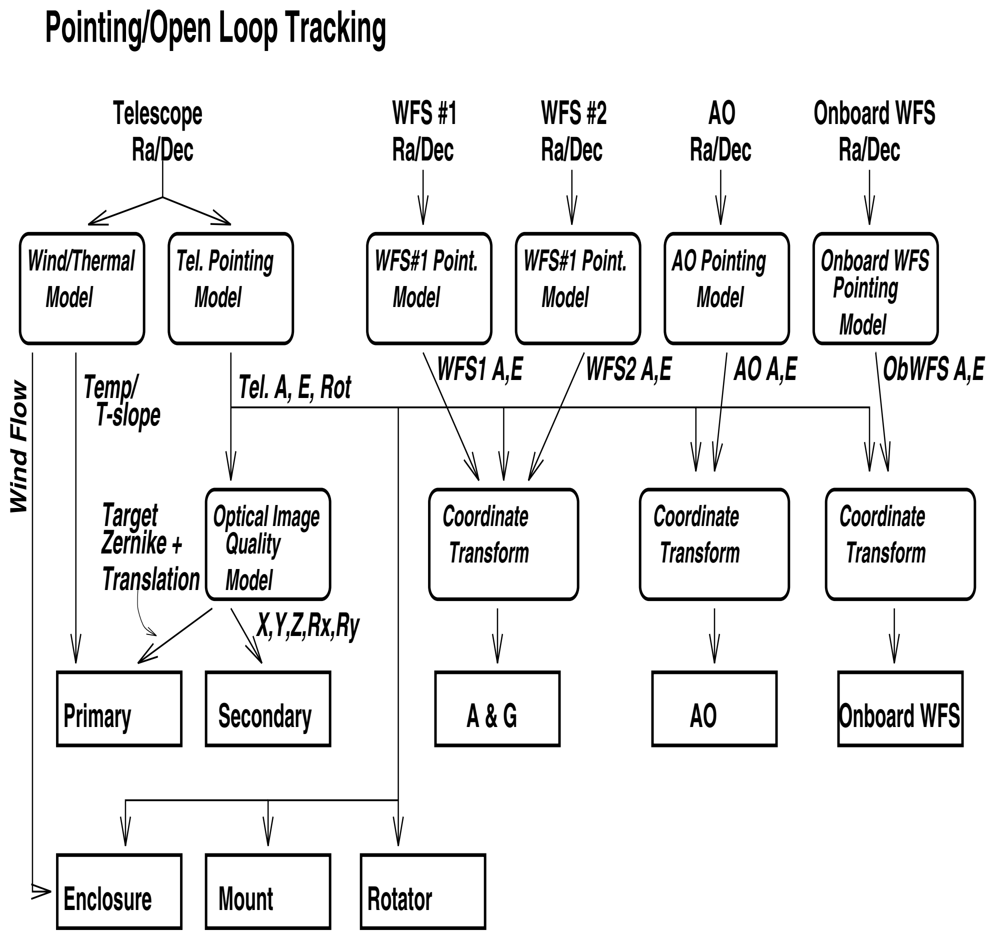
The TCS must also accept thermal requirements and transform these into specifications for adjusting the windblinds, windgates, and other enclosure control systems.

The TCS must also use these thermal requirements in order to control the temperature of the primary mirror in order to minimize mirror seeing.

The TCS will use a variety of models to predict this, including:

- carbon copy - tonight will be like last night
- prediction - use a more sophisticated model based on temperature, humidity, pressure and forecast

FIGURE 8 - 1 **TCS Pointing / Open Loop Tracking Model**



The main difference between the telescope and WFS pointing models will be the probe flexure models.

The information passed to the probes will be {Probe Position + Expected Detector position of Cross Hair}

8.6 TELESCOPE CONTROL SUBSYSTEMS

The TCS is responsible for managing and sequencing the different subsystems which make up the Gemini system. These subsystems are:

- mount control
- cassegrain rotator control
- primary support control
- primary thermal control
- secondary control
- acquisition and guiding (A&G) control
- enclosure control
- adaptive optics control

8.6.1 Mount Control

See “Details of the Mount Control System” in Chapter 14 for more details.

8.6.2 Cassegrain Rotator Control

See “Details of the Cassegrain Rotator Control System” in Chapter 15 for more details.

8.6.3 Primary Support Control

See “Details of the M1 Control System” in Chapter 16 for more details.

8.6.4 Primary Thermal Control

This subsystem has been merged with the Primary Support Control. See “Details of the Primary (M1) Thermal Control System” in Chapter 17 for more details.

8.6.5 Secondary Control

See “Details of the Secondary Control System” in Chapter 18 for details.

8.6.6 A&G Control

See “Details of the Acquisition & Guiding System” in Chapter 19 for more details.

8.6.7 Adaptive Optics Control

See “Details of the Adaptive Optics Control System” in Chapter 20 for more details.

8.6.8 Enclosure Control

See “Details of the Enclosure Control System” in Chapter 21 for more details.

9 HIGH-LEVEL SYSTEM INTERFACES

The following sections introduce the high-level interfaces within the Gemini Control System. First the principal system interfaces are identified and described. Several possible *interface models* are then presented along with a discussion of their advantages and disadvantages along with the selections chosen for the GCS design.

The interfaces are descibed in detail in the Interface Control Documents [22], et. al.

9.1 GEMINI CONTROL SYSTEM OVERVIEW

9.1.1 External Interfaces

The GCS has the following major external interfaces:

- Observer/Observatory Control System
  - Planning and Observing Instructions
  - Graphical Information Display
- Observer/Data Handling System
  - Optional DHS Quick Look Display
  - Reduced Science Data Display
  - Interactive Data Reduction Commands
- Observer/Instrument Control System
  - ICS Real Time Display
- Operator/Observatory Control System

  - OCS Control Commands
  - Scheduling Commands
  - Scheduling Information
  - Graphical Status Display
- Operator/Telescope Control System
  - Handset Control Commands
- External Data Reduction System/Data Handling System
  - External Data Reduction Commands and Parameters
  - External Raw Science Data
  - External Reduced Science Data
- Output & Archives/Data Handling System
  - Status Information and Data
- External Databases/Observatory Control System
  - External Observing Data
- External Databases/Data Handling System
  - External Data Reduction Data
- Environmental Monitoring/Telescope Control System
  - External Environmental Information

In addition, there are interfaces from low-level systems to:

- Telescope hardware
  - Telescope Control Signals
  - Telescope Hardware Status
- Instrument hardware
  - Instrument Control Signals
  - Instrument Status and Data
- Hardware setup information
  - Telescope Setup Information
  - Instrument Setup Information

The interfaces to the telescope hardware, the instrument hardware, and the hardware setup information are specific to individual subsystems within the GCS and not considered further here.

What follows are brief descriptions of the major external interfaces. Details can be found in the chapters for the subsystem that is responsible for each external interface.

9.1.1.1 OBSERVER/OBSERVATORY CONTROL SYSTEM

Typically, observer interaction is through an *Observing Tool*, provided as part of the OCS. In addition, observers use *system consoles* to furnish configuration information and perform interactive observing.

9.1.1.2 OBSERVER/DATA HANDLING SYSTEM

The quick-look facility/observer interface has two components:

1. a graphical-user interface for common operations
2. a command-line interface for entering quick-look scripts

Both of these interface components are provided using PV-WAVE+IDL.

9.1.1.3 OPERATOR/OBSERVATORY CONTROL SYSTEM

The operator interfaces to the OCS using both the system consoles, a *Control Tool,* and a *Scheduling Tool* for control of the Scheduling System. The Scheduling Tool allows the operator to examine Science Programs using a variety of criteria to develop an operational plan for each night's observing. These interfaces are all provided as graphical user interfaces.

9.1.1.4 EXTERNAL DATA REDUCTION SYSTEM/DATA HANDLING SYSTEM

As mentioned earlier, the interface required of a particular DRS is not under the control of the Gemini system. Instead, a *DRS agent process* must provide this interface, with the DHS connecting to all DRS agents in a uniform, but TBD interface. The DHS treats the DRS agent process as a *Data Reduction Server.*

9.1.1.5 DATA HANDLING SYSTEM/OUTPUT & ARCHIVES

The DHS provides science data to the archive using FITS file format. Engineering data is archived by the DHS in a format that is suitable for ease of retrieval and for constructing time flows of events. This interface is TBD, but is likely to be compatible with formats used by EPICS logging applications.

9.1.1.6 EXTERNAL DATABASES/OBSERVATORY CONTROL SYSTEM

The OCS may require access to external databases for activities such as guide star selection/verification, etc. The external databases have interfaces that are not under the control of the Gemini Control System. Consequently, *External Database agent processes* are used to provide access to these interfaces. These agent processes present an SQL interface to the OCS.

9.1.1.7 EXTERNAL DATABASES/DATA HANDLING SYSTEM

This is a similar situation to the above. The External Database agent process presents an SLQ interface to the DHS.

9.1.1.8 ENVIRONMENTAL MONITORING/TELESCOPE CONTROL SYSTEM

Since the external system for Environmental Monitoring is still under development, and includes design from sources external to the Gemini Project, an agent process is needed to handle this interface. This agent process provide information to the TCS database.

9.1.2 Prototype Screens

This document uses *prototype display screens* to help describe the information that must be passed along system interfaces. These screens, found throughout the document, show the information and control for various system components that must be available to users of the system. Systems must support the operations described in each screen.

9.1.3 External Bus Connections

The GCS is connected to the following bus:

- Internet

The integrity of the control system needs to be protected from the Internet through a *firewall* of some form. However, the need to support remote operations makes the design of the firewall difficult. Care must be taken to provide a secure system without compromising remote operation of the telescope.

9.1.4 Internal Bus Connections

The GCS contains the following internal buses as shown in Figure 4 - 1 on page 4 - 5:

- Control LAN

- Data LAN
- Video LAN
- Time LAN
- Synchro bus
- Event bus

9.1.5 Internal Data Stores

All internal data stores are within the principal subsystems and are described more fully in the appropriate 'Details' chapters.

Briefly, the following are the principle data stores in the Gemini Control System:

- Science Program Store -- this is a relational database holding Science Programs. Observers interact with this store through the Observing Tool to construct, manage, and monitor Science Programs. Operators use the Scheduling Tool to select and order Observations to be performed the GCS. This store is part of the Observatory Control System, which is responsible for updating the contents as the telescope operates.
- Observing Database -- also part of the OCS, this database contains the target and actual system configurations, queues for holding pending configurations, and other items required to sequence the principal system components.
- TCS Database -- this is the EPICS database controlling the operation of the TCS and its subsystems.
- Image Database -- managed by the Data Handling System, this database holds images needed by the Data Reduction and Output & Archive Systems.

9.1.6 Internal Interfaces

The internal interfaces in the Gemini control system can be categorized into the following classes:

- *primary internal* interfaces between the principal internal systems
- *secondary internal* interfaces between each internal system and its subsystems

There are also connections between systems using the Synchro Bus and Event Bus. These are not considered further here, but represent additional internal interfaces.

9.1.7 Primary Internal Interfaces

Commands and configuration parameters must be passed from the OCS to the other principal systems: *Data Handling System, Instrument Control System,* and *Telescope Control System.* There is also status information that must flow back from these systems to the OCS. In addition, interfaces exist between the DHS and ICS, and the DHS and TCS. Descriptions of the primary internal interfaces are found in the 'Details' chapters on the principal systems in the GCS. However, there are some general design issues that are discussed below.

Interfaces for commands and configuration parameters are modelled using the messaging facilities from the DRAMA system developed by the Anglo-Australian Observatory, specifically SDS format on top of the IMP messaging system. For the ICS and TCS systems implemented on VxWorks, processes exist to convert messages from this interface layer into EPICS Channel Access operations.

The OCS issues *directives* to the other principal systems, which obtain any necessary information from the Observing Database and then respond appropriately. There are *primitive directives* that are principal system specific as well as more general directives that all principal systems must properly handle. These general directives correspond to the principle actions that occur during telescope operation and represent activities that might occur as the control system moves from one configuration to another:

- **TEST**

  The system should perform self-testing and power-up activities, making the results available to the OCS.
- **INIT**

  The system should perform any necessary start-up initialization. This might involve reading some setup files.
- **RESET**

  The system should reset itself into its start-up state (without reading the setup files).
- **CHECK**

  The system should check the target configuration for validity.
- **APPLY**

  The system moves to the target configuration in preparation for an exposure. At this stage the Data Handling System may need to establish a 'quick-look' process.
- **VERIFY**

  The system pauses to allow the users to examine status information and use consoles to make any adjustments to the current configuration.
- **ENDVERIFY**

The system should leave the verification state.

- **GUIDE**

  Each system performs any actions necessary for accurate guiding and monitors subsystems for adherence to the error budget. This primarily affects the Telescope Control System, though instruments with on-board wave front sensors may need to perform special actions. All systems should monitor themselves and report any alarms.
- **ENDGUIDE**

  The system should leave the guiding state.
- **OBSERVE**

  Perform the actions necessary to take an exposure.
- **ENDOBSERVE**

  Tidy up at the end of an exposure.
- **FLUSH**

  Perform the actions necessary to obtain the data gathered during an exposure.
- **PAUSE**

  Suspend an exposure.
- **CONTINUE**

  Resume an exposure.
- **STOP**

  Cancel an exposure.
- **ABORT**

  Abort all activities.
- **PARK**

  Park moving parts, save any pertinent state information and prepare to be switched off.

9.1.8 Secondary Internal Interfaces

Each principal system is comprised of a number of subsystems. The interfaces here function similarly to the interface between the OCS and the principal systems, with commands and parameters flowing from the principal system to each subsystem and status information flowing back from the subsystems. Additionally, some subsystems may need to transmit and receive (possibly large) data sets. Secondary internal interfaces are described with each subsystem.

The interfaces between a principal system and its subsystems are specific to the detailed design of that principal system. Information describing these interfaces appears in the appropriate ‘Details’ chapter of this document.

9.2 INTERFACE MODELLING ISSUES

This section discusses issues and design trades that arise when attempting to model various interfaces. Not all interfaces are presented here as many of the interface models are straightforward.

9.2.1 External Interface Models

9.2.1.1 USER/SYSTEM INTERFACES

The OCS is a *database-driven* system, with two principle databases:

- the *Observing Database*, holding the all system configuration information
- a *TBD relational database,* holding Science Programs - accessible via SQL

Observers communicate to the OCS through both databases using a facilities provided as part of the OCS. Operators use similar interfaces. The interfaces between users and the system are *Graphical User Interfaces*. Special consideration must given to the *commissioning/interactive use* consoles. While the interface between these consoles and the users is straightforward, there is also an issue on how these consoles interface to the system. This issue is discussed further under the Primary Interface Models section below.

Given the complexity of the Gemini telescopes, *visual tools* are heavily relied upon in the user interfaces. Both PvWave and Tk/Tcl are used as appropriate to provide clear and easy access to both status and control information. Animations of mechanisms, control behavior, etc. are expected throughout the user interfaces. Visual queues should draw user attention to critical activities.

However, not all interfacing between users and the system is best accomplished through visual techniques. Visualization techniques for control do not always work well with unanticipated or highly repetitive actions. Consequently, the GCS provides methods for *programmatic* control using programs written in the TCL programming language. More information on this interface can be found in the Observatory Control System discussions.

9.2.1.2 INTERFACING TO THE EXTERNAL DATA REDUCTION SYSTEM

The Data Reduction System (DRS) is considered external to the Gemini control system. Interfacing to a specific DRS requires an *agent process* to be written that implements the interface required of that DRS.

9.2.1.3 HARDWARE INTERFACES

Interfacing to the hardware is primarily through EPICS device drivers. Systems that cannot or do not support EPICS require custom interfaces that are beyond the scope of this document.

9.2.1.4 HARDWARE SETUP INFORMATION

Many hardware systems may require special initial configuration information. While the Gemini Control System provides a means of storing and downloading such information, details of this information is specific to individual hardware systems and not considered further here.

9.2.1.5 EXTERNAL DATABASES

The Gemini system expects to be able to use star catalogues, etc. provided from diverse sources in varying formats. These databases typically require agent processes to convert Gemini Control System *SQL database queries* into a form appropriate to each external database.

9.2.2 Primary Internal Interface Models

Here, there are some choices that can be made. Consider the interface between the OCS and the *Telescope Control System (TCS).* Both are database-driven systems. There are two principal issues that must be addressed:

- Where do the commissioning/interactive-use consoles connect into the system - through the Observing Database in the OCS or through the EPICS databases in the TCS?
- How does information flow between the Observing Database and the EPICS database in the TCS?

In the Gemini Control System, the commissioning/interactive use consoles connect through the Observing Database and the interface between the OCS and TCS is done using the ‘Symmetric database model’ presented below.

The remainder of this section presents an analysis of the alternative models that were considered for these issues. Although the issues here address the TCS specifically, they apply to all the primary internal interfaces.

9.2.2.1 CONSOLE INTERFACE MODELS

Connecting the consoles directly to the TCS system has following advantages:

- no need to complicate the OCS design with console details

- these consoles may be used independently of the OCS, providing a second stand-alone interface to the TCS and associated subsystems (the other interface is through the EPICS engineering screens)

However, there are also some disadvantages:

- it is difficult to maintain a consistent look and feel across also consoles when they are developed by diverse groups, unless the OCS development group implements them - in which case this group is forced to learn numerous details of the TCS that are not otherwise required.
- it is difficult to provide the OCS with the ability to track the status of commands, responses, etc.
- in general, designing a system with many entry points is undesirable, as it introduces additional complexities and unexpected feature interaction
- since there are already stand-alone engineering interfaces, the utility of a second stand-alone interface is questionable
- these directly-connected consoles increase the number of EPICS *channel-access* connections, with the associated performance hits to the real-time portions of the control systems
- consoles that require access to multiple subsystems in order to function properly impact multiple subsystems and require sharing of information across these subsystems

Switching these consoles to interact through the Observing Database in the OCS reduces or eliminates most of these problems. While it does require the Observing Database to obtain information from the subsystems, most of this information is required to be available to the OCS for system health monitoring and most of the details of the subsystems remain hidden from the OCS. Furthermore, the consoles can be built and tested before the TCS system is available.

9.2.3 Secondary Internal Interfaces

These interfaces are divided into the following classes:

- between two EPICS systems
- between an EPICS system and a non-EPICS system
- between two non-EPICS systems

Most of the details of these interfaces are design issues that are to be worked out between the Gemini Project Office and the individual Work Package Groups. A few general comments may be made, however.

9.2.3.1 EPICS/EPICS INTERFACES

In general, these interfaces are between systems involved in *real-time* control. Furthermore, it is likely that many of these interfaces are between a relatively sophisticated system (such as the TCS) and a less sophisticated subsystem (such as the mount). Consequently, the number of parameters that need to be passed are both reasonably small and well-defined. These interfaces are therefore typically controlled by the master system, using EPICS channel access to direct the subsystem.

9.2.3.2 EPICS/NON-EPICS INTERFACES

These require the addition of an *agent process* to translate between EPICS channel access and the non-EPICS system. Each agent process is unique to the particular system/subsystem.

9.2.3.3 NON-EPICS/NON-EPICS INTERFACES

This case is similar to the previous. An agent process is used to handle communications between the two systems.

10 Details of the Observatory Control System

The “High-Level Observatory Control System Concepts” chapter described the Visible User Interface of the Observatory Control System. That chapter emphasized what observers and staff would see and do when observing with the Gemini telescopes. This chapter continues the description of the Observatory Control System by developing more of *how* tasks are accomplished rather than *what*. This chapter assumes you have read the “High-Level Observatory Control System Concepts” chapter.

The demanding challenge for the Gemini telescopes and the Observatory Control System is planned observing and the majority of the details in this chapter focus on the design details that must be present to support the planned observing modes. The functionality required by interactive observing is based on the functionality supporting planned observing and is described later in the chapter. This chapter starts with a in discussion of the Science Program structure. The Configurable Control System details and finally, the details of the Visible User Interface are discussed.

Whenever the words program, observation, and configuration are used in this chapter they refer to the OCS Science Program, Observation, and Science Configuration, respectively, unless otherwise noted.

10.0.1 Changes since the Preliminary Design Review

Changes to this chapter have focused on fixing problems and incorporating suggestions that came out of the Preliminary Design Review and the System Design Review.

- Purged the requirement that principal systems controlled by the OCS use the state model of the Sequence Executor.
- Improved and refined the definitions of OCS Sequencer Commands.

- The design of the status and alarm system has been included.
- An EPICS system has been included as part of the OCS, to handle status, alarm and logging information from the rest of the observatory.
- The requirement for a host-level Channel Access Server has been removed.
- An phased approach for OCS development is included.

10.1 SCIENCE PROGRAMS AND OBSERVATIONS

This section describes the details of science program, observation, and science configuration structure. This information is required to appreciate the details of the execution of observations.

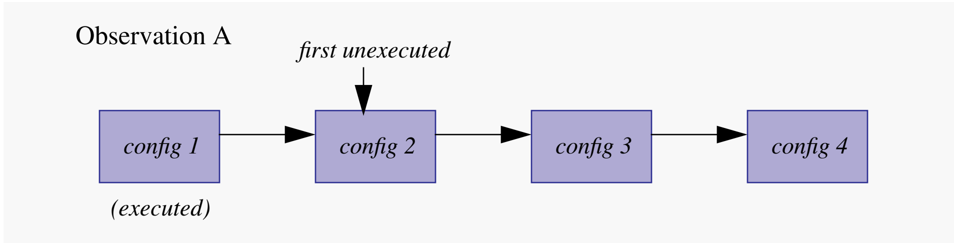
When using the Observing Tool, the astronomer creates a science program document that contains one or more observations, one or more science configurations, and other related information. The science program is submitted to the Gemini Control System where it resides in an external database until scheduled by the operator.

A Science Program is an unordered group of Observations. Typically, the observations are grouped together in a program because they share setup information, calibration requirements, and a common scientific goal.

The observations in a science program can be *partially executed* or *completed*. A partially executed observation has one or more unexecuted configurations. A science program is completed when all of its observations are completed. The completion status of a program and each of its observations is part of the information in a science program.

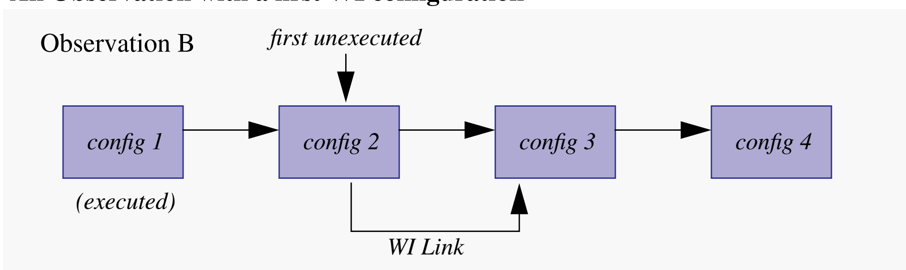
An observation is viewed as a chain of science configurations. Some of the configurations in an observation may have been previously executed. Any previously executed configurations in an observation are found at the beginning of an observation because of the properties of the observation—configurations in an observation are executed in order (although observations in a program are not necessarily executed in order). If an observation is not completed, it must have an unexecuted first configuration. Figure 10 - 1 shows a typical observation with multiple configurations. In this case, the observation is partially complete since the first configuration has been previously executed.

FIGURE 10 - 1 **An Observation is a chain of configurations**



When the first configuration within an observation is the start of a group of configurations marked *without interruption*, the operator must consider the entire group rather than just the first configuration when scheduling the observation. Figure 10 - 2 shows an observation with a *first configuration* that is the head of a chained series of configurations.

FIGURE 10 - 2 **An Observation with a first WI configuration**



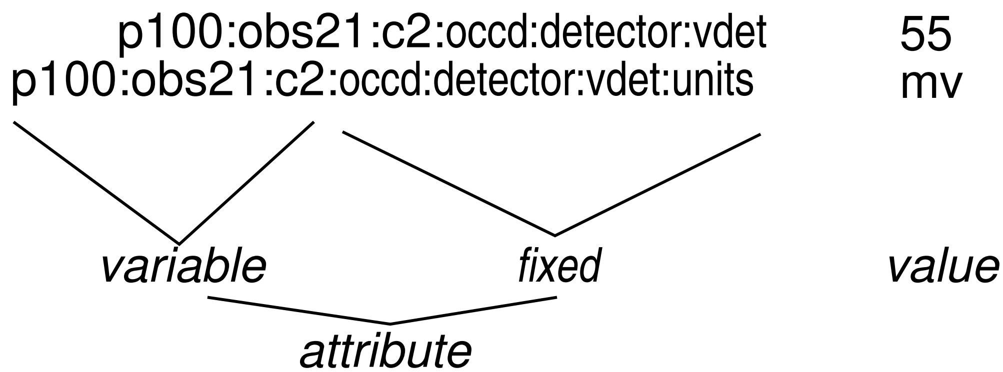
It may be that by default all observations created in the Observing Tool execute without interruption. This would mean that by default observations are executed *completely* rather than one configuration at a time requiring fewer scheduling decisions. Astronomers may find it more natural and intuitive to execute entire observations. This default behavior is to be determined at a later time—the software system supports both behaviors.

The information describing any temporal requirements between successive configurations can be located either in the containing observation or in the science configurations themselves. If *config 2* in Figure 10 - 2 is constrained to execute at least 30 minutes after *config 1*, it will not be possible to schedule *config 2* until the constraint is satisfied. If the two configurations are linked WI and *config 1* is scheduled, the telescope will block waiting until the timing constraint becomes satisfied. The total execution time for the observations will take into consideration any WI links and temporal constraints.

10.1.1 Attributes and Values

The information in configurations is represented as a set of attributes and values. An *attribute* is a description (probably textual) of an item in some device that needs to be controlled or monitored. The value is the data that should be used to set the attribute or the data that is read from the attribute. Figure 10 - 3 shows the structure of an attribute in detail.

FIGURE 10 - 3 **The structure of attributes**



The attribute is made up of the *variable* part, which describes a science program, observation, and science configuration, and the *fixed* part, which describes a controllable item or command in an instrument. In Figure 10 - 3, the attributes describe a commandable item called *vdet* that is part of the optical ccd detector device within science program 100, observation 21, and configuration 2. The detector voltage, *vdet*, is a commandable item within the optical CCD system. The attribute sets the value of the item to 55 and sets the units of the item to milli-volts.

The principal systems the OCS must control are Unix-based (DHS) and VxWorks-based (TCS, some ICS) and the mapping of the commandable items in the attributes to commandable items in a principal system is done for each system. Interface Control Document 1 (ICD1 [24]) describes the method used in EPICS-based principal systems.

10.1.2 The Structure of Science Configurations

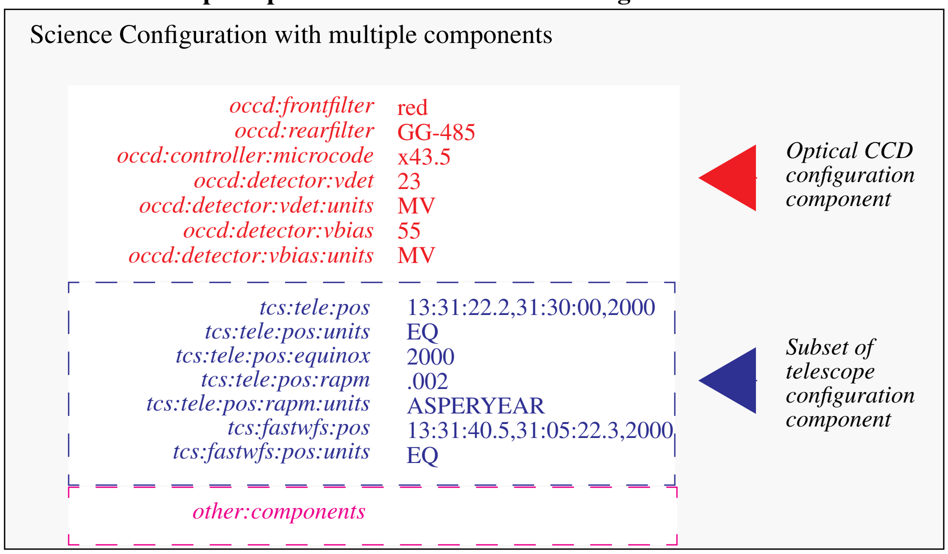
Configurations are an amalgamation of configuration components describing the portions of the Gemini Control System that the user wishes to customize for his/her own purposes. The work done at a particular system configuration is also described by a configuration component. It is more difficult to describe the work done as part of science configuration than it is to describe the hardware setup.

Configuration components consist of a number of attribute value pairs similar to a Motif toolkit widget or an EPICS record description. Figure 10 - 4 shows an example of a portion of a science configuration. In this figure, the attributes appear on the left and the attribute values appear on the right separated by a gap. The configuration shows two configuration components but most real configurations contain more. The example shows simplified configuration components for an optical CCD and the Telescope Control System.

There are a number of important points shown in this figure.

- The configuration information forms a tree. The parts of the attribute name (separated by colons) are nodes and form a path to the leaf or node that is to be set with the value. The left most part of the attribute name is closer to the base of the tree.
- The configuration component attributes form a hierarchy that matches the hierarchy of the principal system and item being controlled. The hierarchy promotes encapsulation and allows identical names for identical parts of related or identical systems (filter1.power, filter2.power). This makes it easier to remember attribute names.
- A principal system wishing to access its component within a configuration only needs to look at the left-most parts of the fixed part of the attribute. Principal systems must have unique names; for instance, *occd* for optical CCD and *tcs* for the Telescope Control System.
- The configuration size is not bounded and can contain as many configuration components as it needs to include. There are no restrictions on the configuration components other than that they have unique left-most parts of their attributes.

FIGURE 10 - 4 An example representation of a Science Configuration



- There is no restriction on the number of attributes/value pairs that can be included in a configuration component. This is crucial if the Observing Tool is to have access to *all* of the configuration components of a subsystem. An engineer might decide he wants to vary the detector bias voltage during a series of test frames; something that might not interest the astronomer. It would not be in the configuration component if it wasn't needed.

The principal systems use the right-most part of the attribute names to perform actions on behalf of the commanding system. See [24] for more information on the mapping of attribute names to functions in principal systems.

10.1.3 The Representation of Programs, Observations, and Configurations

Science Programs and their contents exist in databases for their entire lifetime within the Gemini Control System. The preceding description of science configurations points out that a configuration is just a bag of attribute/value pairs. A database can easily represent this kind of structure as a variable sized record. The same approach can be used to represent observations and science programs.

The Gemini Control System must manage several concurrently executing configurations. The attribute/value structure works equally well when defining multiple observations. Figure 10 - 5 shows how observations are an extension of configurations.

This figure shows two observations *obs21* and *obs25*. The only differences between obs21 and obs25 are the values of a few attributes and the left-most portion of their attribute names.

Figure 10 - 6 shows a science program containing the two observations of the Figure 10 - 5 example. The example program is denoted by the attribute component *p100*. Each science program is identified by a unique odometer value across both Gemini telescopes in the same manner as instrument data frames. With this representation, it is easy to denote all attributes that belong to a program (p100) or an observation within a program (p100.obs21).

FIGURE 10 - 5 An example representation of an Observation

obs21:occd:frontfilter	red
obs21:occd:rearfilter	GG-485
obs21:occd:controller:microcode	x43.5
obs21:occd:detector:vdet	23
obs21:occd:detector:vdet:units	MV
obs21:occd:detector:vbias	55
obs21:occd:detector:vbias:units	MV
obs21:tcs:tele:pos	13:31:22.2,31:30:00,2000
obs21:tcs:tele:pos:units	EQ
obs21:tcs:tele:pos:equinox	2000
obs21:tcs:tele:pos:rapm	.002
obs21:tcs:tele:pos:rapm:units	ASPERYEAR
obs21:tcs:fastwfs:pos	13:31:40.5,31:05:22.3,2000
obs21:tcs:fastwfs:pos:units	EQ

Observation 21 Optical CCD and Telescope components

obs25:occd:frontfilter	blue
obs25:occd:rearfilter	GG-485
obs25:occd:controller:microcode	x43.5
obs25:occd:detector:vdet	23
obs25:occd:detector:vdet:units	MV
obs25:occd:detector:vbias	55
obs25:occd:detector:vbias:units	MV
obs25:tcs:tele:pos	18:41:22.2,31:40:00,2000
obs25:tcs:tele:pos:units	EQ
obs25:tcs:tele:pos:equinox	2000
obs25:tcs:tele:pos:rapm	.001
obs25:tcs:tele:pos:rapm:units	ASPERYEAR
obs25:tcs:fastwfs:pos	18:43:40.5,31:40:22.3,2000
obs25:tcs:fastwfs:pos:units	EQ

Observation 25 is different in the tcs position and occd filter

10.1.4 Operations on Programs, Observations, and Configurations

The VUI applications, particularly the Observing Tool, as presented in “High-Level Observatory Control System Concepts” require that a number of operations be possible on the representation of Science Programs and their contents. The following is a minimal list of required operations.

- *creation* - The ability to create science programs, observations, and science configurations is required. These components would be empty after creation.
- *initialization* - It is necessary to initialize new components with some attribute values (odometer values, etc.) after creation; this is initialization.
- *copying* - The ability to copy one program’s components to another is required. A copy of a component copies all the components contained in the source component to the destination component. A shallow copy copies only the structure of a component hierarchy. A deep copy does a shallow copy and then recursively copies all attribute values.

FIGURE 10 - 6 An example representation of a Science Program

p100:obs21:occd:frontfilter	red
p100:obs21:occd:rearfilter	GG-485
p100:obs21:occd:controller:microcode	x43.5
p100:obs21:occd:detector:vdet	23
p100:obs21:occd:detector:vdet:units	MV
p100:obs21:occd:detector:vbias	55
p100:obs21:occd:detector:vbias:units	MV
p100:obs21:tcs:tele:pos	13:31:22.2,31:30:00,2000
p100:obs21:tcs:tele:pos:units	EQ
p100:obs21:tcs:tele:pos:equinox	2000
p100:obs21:tcs:tele:pos:rapm	.002
p100:obs21:tcs:tele:pos:rapm:units	ASPERYEAR
p100:obs21:tcs:fastwfs:pos	13:31:40.5,31:05:22.3,2000
p100:obs21:tcs:fastwfs:pos:units	EQ
p100:obs25:occd:frontfilter	blue
p100:obs25:occd:rearfilter	GG-485
p100:obs25:occd:controller:microcode	x43.5
p100:obs25:occd:detector:vdet	23
p100:obs25:occd:detector:vdet:units	MV
p100:obs25:occd:detector:vbias	55
p100:obs25:occd:detector:vbias:units	MV
p100:obs25:tcs:tele:pos	18:41:22.2,31:40:00,2000
p100:obs25:tcs:tele:pos:units	EQ
p100:obs25:tcs:tele:pos:equinox	2000
p100:obs25:tcs:tele:pos:rapm	.001
p100:obs25:tcs:tele:pos:rapm:units	ASPERYEAR
p100:obs25:tcs:fastwfs:pos	18:43:40.5,31:40:22.3,2000
p100:obs25:tcs:fastwfs:pos:units	EQ

Observation 25 is different in the tcs position and occd filter

- *deletion* - The ability to delete program components is required. It must be possible to delete components from within other components without influencing the remaining components.
- *addition* - The ability to add components to already existing components is required.

The operations on programs and their compounds are required to be accessible as TCL-based commands.

10.1.5 Summary of Science Program Structure

This section has presented the design details of Science Programs and their components Observations, and Science Configurations. A design for attributes was presented and possible implementations given.

10.2 THE CONFIGURABLE CONTROL SYSTEM

The “High-Level Observatory Control Systems Concepts” chapter of this document described the purpose of the Configurable Control System (CCS) and its place in the software design as the hub of the run-time operations of the OCS. The CCS is the software layer that provides support for the functionality of the observer and operator VUI; it is layered below the VUI and above the other principal systems in the software design as shown in Figure 5 - 1, “Observatory control system software layers and decomposition,” on page 5 - 4. The major task of the CCS is the coordination of the activities of the Gemini Control System during an observing session. This function is the focus of most of this section.

The principal components of the CCS are the Observing Database and the Sequencing System, which itself is made up of The Configuration Dealer and Sequencer. Additional entities exist in the CCS to provide important system functionality such as operator reminders, access and privileges, alarms, status, and error handling. These entities are grouped together as *clients of the Observing Database*.

“Top-level Diagrams” on page A1 - 1 of this document show the formal design views of the OCS. This information may be of use when reading this section.

10.2.1 Sequencing System Details

The discussion of the Scheduling System in “A User View of the Scheduling System” in Chapter 5 described how staff and observers create programs to direct the actions of the Gemini Control System. The previous section of this chapter showed how observations are structured and that they are represented as database records of attributes and their values.

The Sequencing System of the CCS is made up of two principal components: the Configuration Dealer (CD) and the Sequencer. The job of the Configuration Dealer is to accept observations (the containers of science configurations) for execution and allocate for them any resources required for their execution. The Sequencer accepts observations that are *ready* to execute from the Configuration Dealer, executes them and monitors their progress in the system.

The impact on the OCS design of supporting planned observing must be stressed. The requirement that observations be specified and then executed at a later time drives the design of the Configurable Control System. This chapter focuses on planned observing but points out any specialized interactive observing related information. The details of these software components are now presented.

10.2.2 The Configuration Dealer

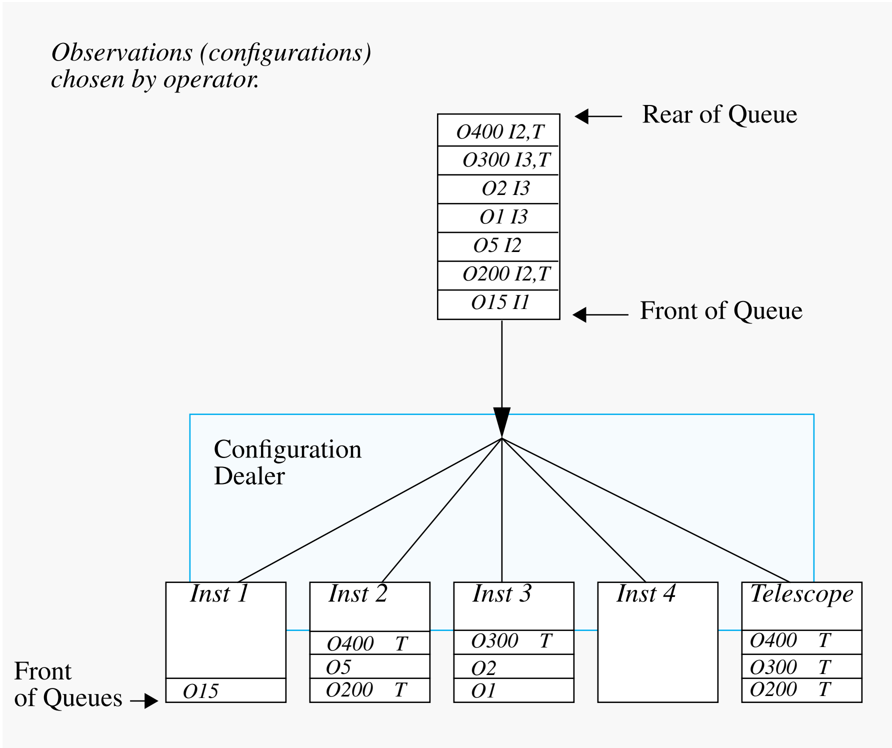
To optimize the use of the telescope it is important that activities that are independent take place at their own pace without concern for other independent activities. There can be up to four working instruments in place on a Gemini telescope at once and it is conceivable that one individual might be running tests on one instrument while observing is taking place with another instrument. Even with a single instrument it is possible that two observations might need to execute concurrently; for instance, an observer would want telescope setup for a science object to proceed while he was executing an instrument bias configuration.

A Gemini system *resource* is defined to be a shared hardware or software system that may need to run independently and can be used by at most one executing observation. The telescope beam is viewed as a telescope resource that must be shared among the instruments (instruments are also resources) and allocated by the operator. The telescope beam is granted to observations in the order in which the operator selected them and submitted them to the Configuration Dealer or the order they appear in a plan.

The Configuration Dealer (CD) is the front-end of the Gemini Sequencing System. The Configuration Dealer provides a simple, transparent means for managing the Gemini system resources. This document considers the resources of the Gemini system to be the telescope beam and the four instruments. The set of resources may change since it is tied to the physical design of the telescope. It may be impossible to use some hardware subsystems independently with some physical layout choices and the final hardware decision will determine the system resources. The efficiency and concurrency of the instruments is tied to the final set of resources. The Configuration Dealer is designed to work with a set of resources and that set can change.

Each resource is represented in the Configuration Dealer by a queue. The observations containing the “next” configurations are selected by the operator, a plan or in some cases, the observer, and are placed in the CD. The observation at the front of each queue is the observation which currently has possession of the queue’s resource. Figure 10 - 7 and Figure 10 - 9 present an example of Configuration Dealer operations. Figure 10 - 7 shows a series of observations that are chosen to execute by an operator. The top half of the figure shows the order of the observations and the resources that each requires. For instance, Observation 400 requires instrument 2 and the telescope resource.

FIGURE 10 - 7 **Example observations as entered by an operator**



The Configuration Dealer takes these observations and splits them up as shown in the lower half of this figure. The CD inspects the first unexecuted configuration in an observation and any other configurations chained to it, and creates entries in the appropriate resource queues. The observations at the front of the resource queues are *ready* to run. Observation 15 in at the front of the instrument 1 queue, Observation 200 is at the front of the instrument 2 and telescope queues, and observation 1 is at the front of the instrument 3 queue.

The Plan View (Figure 5 - 19 on page 5 - 50) is one way of presenting the data in the Configuration Dealer. When executing a plan, the operator is loading an entire night's observing session into the configuration dealer.

The observations are now ready for execution in the Sequencer.

10.2.3 The Sequencer

The Sequencer is charged with taking individual science configurations (and their containing observations) from the Configuration Dealer queues and carrying out the work specified in the configurations. The observation and its configurations stay together in the Sequencer, but the sequencer executes science configurations. The sequencer must support the concurrent processing of configurations when possible.

A science configuration specifies two things: the configurations of the various hardware systems and the observing work that is to be done with the hardware setup as in that configuration. *Processing a configuration* is defined to encompass the following high-level steps:

1. Acquire resources required by the first configuration and any linked configurations.
2. Set the hardware systems as defined in the first configuration.
3. Wait for all systems to complete motions and check for errors.
4. Verify and/or adjust configuration.
5. Do any work required.
6. Do any post-processing required for the just completed work.
7. Check for a linked configuration and if there exists a linked configuration go to 2.
8. Free all resources used in the configuration(s).

The format of observation processing is embodied in the configuration's *sequence recipe.* The tasks in this list are embodied in the actions taken in a configuration's sequence recipe. The term *execute an observation* means process an observation's first configuration and any configurations linked to that configuration.

10.2.3.1 SEQUENCER OPERATION

The attributes of an observation and its configurations are moved to the Observing Database from the external database when the operator submits an observation to the Configuration Dealer.

The Sequencer must manage the execution of multiple observations. The Sequencer cycles through the Configuration Dealer queues one at a time repeating the same pattern over and over to make sure it eventually processes all ready observations. The pattern might be: inst1, inst2, inst3, inst4, telescope. The repeating pattern is required to protect against livelock discussed in "Resource Allocation" on page 10 - 17.

At each queue, the Sequencer pauses and executes the simple algorithm of Figure 10 - 8 and Algorithm 10 - 1. The algorithm makes references to an entity called a *sequence executor*; an Observing Database client that has the job of executing an observation.

ALGORITHM 10 - 1 Sequencer algorithm steps

1. If the queue is empty, go on to the next resource queue and start at 1.
2. The queue is not empty so check to see if the observation is *executing*. If it is not executing, go to algorithm step 5.
3. The observation is executing so check to see if the observation is *done*. If it is not done, go to the next resource queue at start at 1.
4. The observation is done so destroy the sequence executor, do any required cleanup, go on to the next resource queue and go to algorithm step 1. Set the observation to *not executing*.
5. The configuration is not executing so create a new sequence executor to execute the observation. Tell the executor to start executing it and mark it *executing*. Go on to the next resource queue at start at algorithm step 1.

FIGURE 10 - 8 The Sequencer Algorithm as a state diagram.

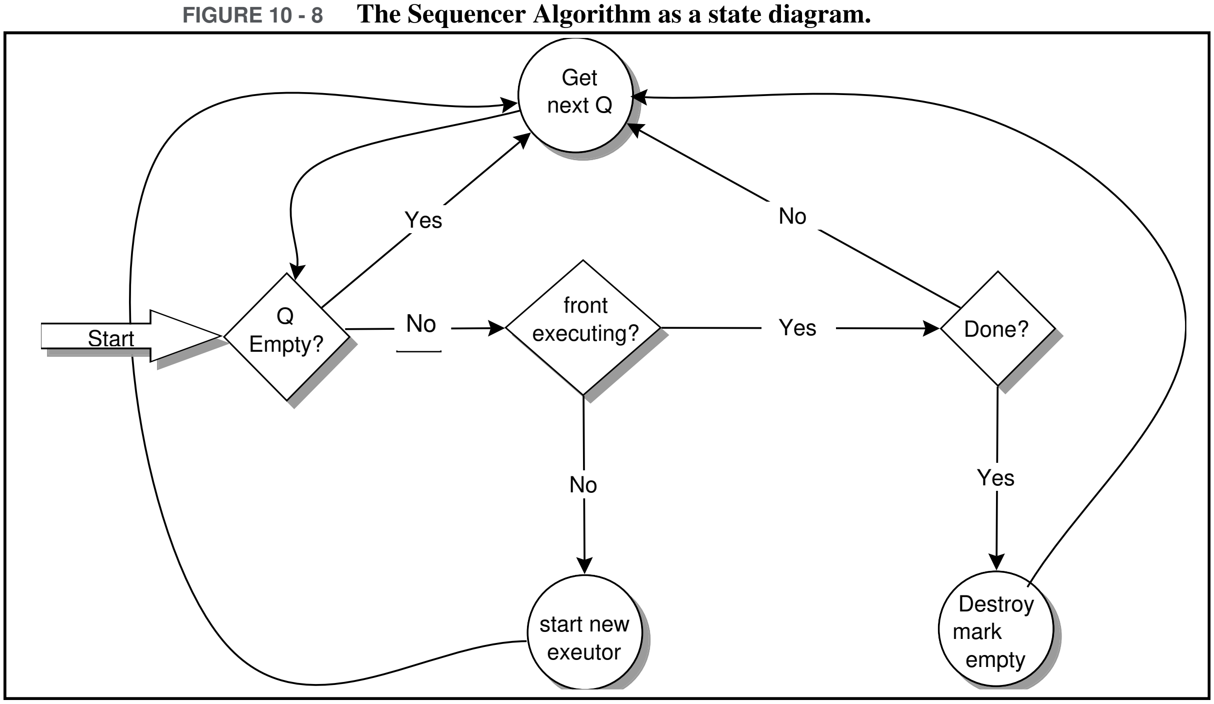


FIGURE 10 - 9 Configuration Dealer, Sequencer, and executing Observations

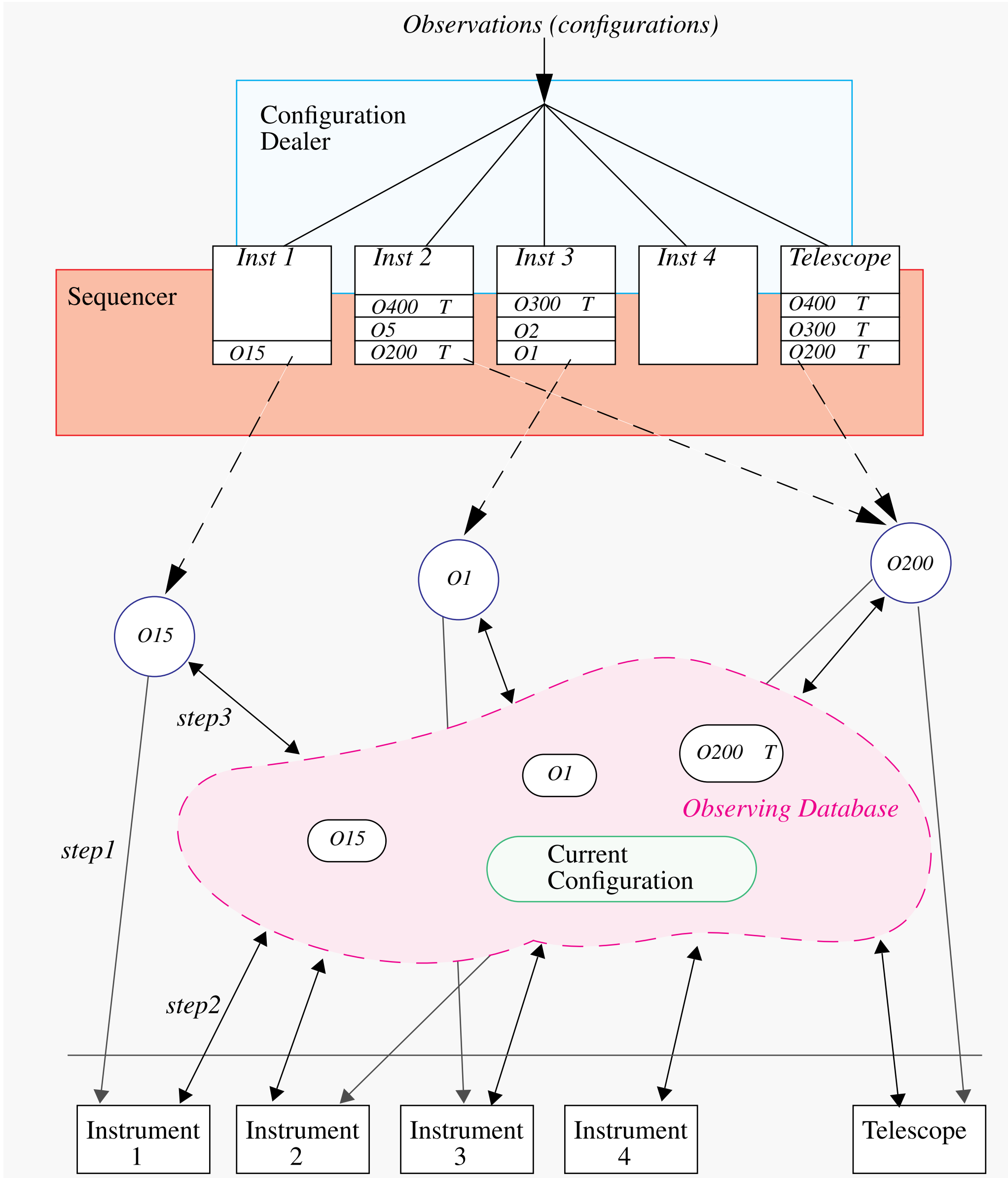


Figure 10 - 9 shows what happens when the Sequencer runs its algorithm on the observations of Figure 10 - 7. There are three concurrently executing observations: O1, O15, and

O200. The Sequencer first looked at the Instrument 1 queue and created a new executor to execute Observation 15. It then moved on to the Instrument 2 queue and started Observation 200, which also uses the telescope. It then goes on and starts Observation 15 for Instrument 3. After this step, the sequencer finds all the queues executing and waits until one is done.

Executing configurations reside in the Observing Database (ODB), another part of the CCS discussed in The Observing Database section of this document. For now, consider the ODB to be the environment holding observation configuration information during the time the observations are executed and the processes (executing observations, etc.) that use it. The Sequencer looks in the configuration data in the ODB for an observation's execution condition.

Figure 10 - 9 shows a several other details of the CCS Sequencing System. The Instrument 4 queue is empty and awaiting something to do. The arrows from the front of each queue point to the Observing Database client that is executing the observation. An observation's sequence executor connects directly to its Instrument Control System and the Observing Database. Each Instrument Control System listens to its sequence executor and can read and write to the Observing Database.

The telescope queue shows that Observation 300 is the next observation to receive the telescope resource. Observation 2, which doesn't need the telescope, not Observation 300, is the next observation to be executing with Instrument 3. When Observation 200 finishes, Observation 300 will begin to execute and do as much as it can and then wait until Instrument 3 is finished.

The Sequencer continues to execute observations as long as the operator keeps the Configuration Dealer queues non-empty. The first configuration and any configurations WI linked to the first configuration are passed to the Sequencer one at a time and are allowed to complete. The Sequencer then looks for the next observation or the control system becomes idle. The Gemini Control System remains busy as long as there are observations with unexecuted configurations in the Configuration Dealer.

The operator can interact with the observations in the Configuration Dealer. These interactions are discussed in "Configuration Dealer Interactions" on page 10 - 59.

At this point in the discussion, a number of observations have been picked to execute and based on their resource requirements some of them are ready to execute their observations. The sequence executor is discussed after a discussion of resource allocation.

10.2.3.2 RESOURCE ALLOCATION

The instruments and the telescope are what are called permanent resources in the field of computer science[1]. A permanent resource can be used repeatedly by many processes. In our case, up to one instance of each resource is present and available at any given time.

Deadlock in a computing system occurs when two or more processes are waiting indefinitely for conditions that will never hold. An observation can require at most two resources (at this time): the telescope and an instrument. A deadlock would occur in our system if an observation owning instrument 1 is waiting for the telescope resource that is owned by an observation that is waiting for instrument 1. This is called circular waiting and is one of the requirements of successful deadlock.

These statements assume observations execute without encountering problems that would cause them to block indefinitely. If a process blocks indefinitely, it is not deadlock, it's an error. The operator must intervene and remedy the situation.

The control system must ensure that deadlock does not occur. In our case, this is fairly easy. The instrument queues award their instrument to the observation at the front of their queue in first-come-first-serve fashion. An instrument can be acquired by an observation in no other way than to enter the instrument's Configuration Dealer queue and wait until it gets to the front. The same is true for observations requiring the telescope resource. A resource is deallocated by the observation at the front of its queue before the next observation moves to the front; the resource is always free at that time. As discussed in the earlier section, the sequencer looks at the Configuration Dealer queues one at a time and starts the execution of the observations if needed. Serializing process execution prevents deadlock from occurring.

Livelock occurs in a computing system when two processes are ready and able to run but because the scheduling of the processes is not fair one will never run. Since we have specified in the Sequencer discussion that the Sequencer looks at the Configuration Dealer queues in a fixed pattern, all Configuration Dealer queues will eventually be examined and their observations will execute. No livelock can occur.

10.2.4 The Sequence Executor

The Sequence Executor (SE), or Executor, is a process with the job of executing a particular observation. The Sequencer creates an executor and tells it which observation in the Observing Database it should execute. A major guideline of the CCS is to isolate the OCS

1. Holt, R. C., "On Deadlock in Computer Systems," Cornell University, Ithaca, New York, 1971.

from the details of the instruments, telescope, and data handling systems. The problem the executor must solve is how to control and sequence these software subsystems without knowing anything about them!

> Rather than creating and destroying Sequence Executors, it may be simpler, more efficient, and economical to run a Sequence Executor for each instrument at all times. The OCS Work Package group will examine this issue.

10.2.4.1 SEQUENCE RECIPE MOTIVATION

The observing process is a mixture of routine, repetitive procedures and also decisions that *must* be made interactively at the time an observation is executed. The conformation of the observing process is relatively static observation after observation. Interspersed within the fixed observing form are incidents of interactivity where decisions must be made by the humans involved in the observing process. Someone watching a classical/interactive observing session and recording the activities that take place would probably come up with a list that would include the following items and notes:

What I saw at the Telescope Over and Over Again

1. The operator waits for the observer to choose his next target.
2. The observer decides which science object should be observed and relays the information to the operator.
3. The operator moves the telescope to the proposed target. Some related hardware is sometimes initialized during acquisition.
4. The observer considers the upcoming observation and possibly prepares the instrument for the observation.
5. The operator and observer work together to verify the target and, if necessary, fine tune the telescope position for the observation. The observer may need to do a variety of tasks at this point to verify the correct telescope position. Sometimes the observer must do other operations unrelated to verifying the target before he is willing to continue on with an observation.
6. Telescope guide systems are setup and activated.
7. Observation begins.
8. Observer and operator patiently wait for the observation to end. The observer may occupy himself with analysis of previous observations during this time.
9. The operator reads a book or sleeps if the observation takes a long time.

10. The observation ends and the observer examines the result. The analysis of the science data is often used to determine the next observing activity.
11. They go back to item 1 if they are happy. Sometimes they go to step 7 to make a new observation at the current target.

The items in this list are representative of operations at typical classical/interactive observing sites. Of the items, 2 and 5 are interactive steps and the remaining observing/procedure steps are fixed. The order of the operations and the amount of concurrence in the operations varies, but the division of the observing routine into static and interactive components is authentic.

10.2.4.2 SEQUENCE EXECUTOR CONTROL

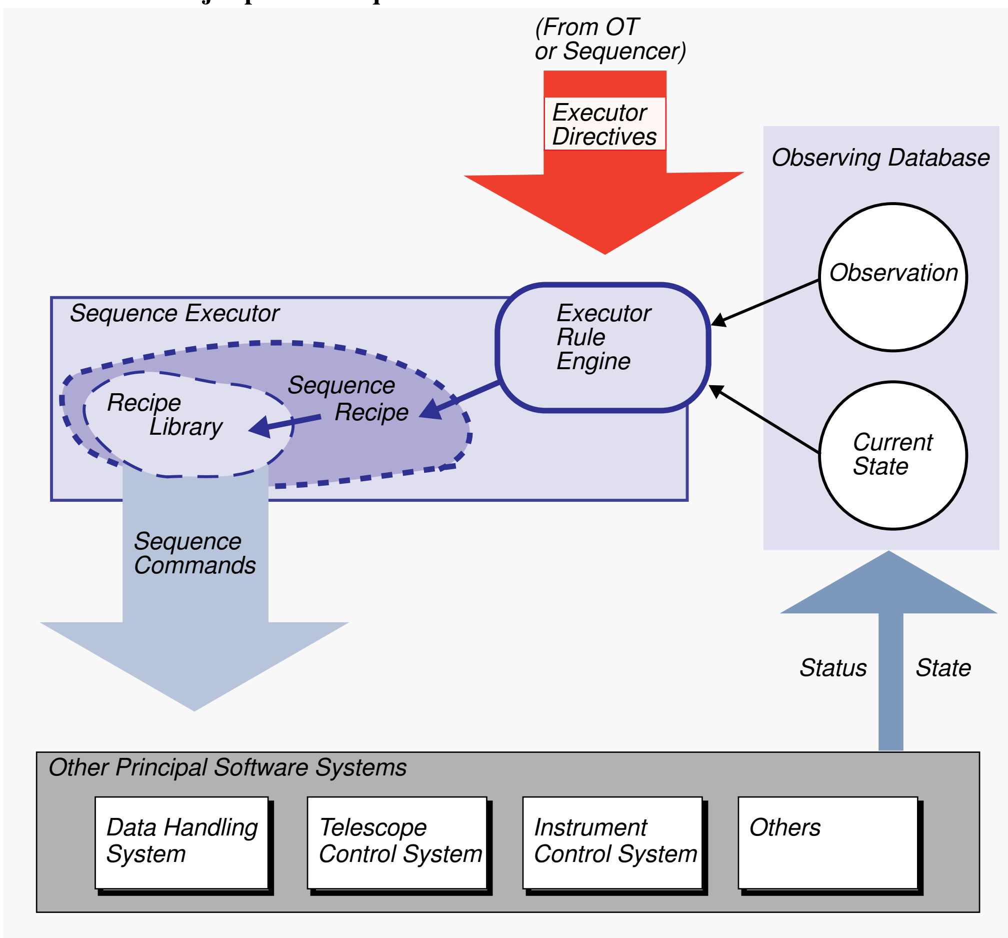
The OCS takes advantage of the division of fixed and interactive activities to provide support for the planned observing modes required in the Gemini project. Figure 10 - 10 shows the software entities and concepts used in the executor to control the software subsystems. Each part of the figure is discussed in subsequent paragraphs.

The Sequence Executor uses a *sequence recipe* and its companion *recipe library* in order to process a configuration. The recipe orders, controls, and synchronizes the other principal software systems to provide a consistent framework for the observing process and the predictable observer/operator interactions. The recipe makes calls to the recipe library that uses high-level, system independent *sequence commands* to control the principal software systems.

Each science configuration references a TCL script that is the configuration's sequence recipe and library. Many recipes can be used during an observing session, but there will probably only be a few in regular use. The Observing Tool sets the name of the appropriate recipe for a configuration in the configuration itself or, in some cases, the user's program may require a special recipe constructed by staff at the telescope.

At this point it is necessary to discuss the Sequence Commands, the directives that flow from the sequence executor to the principal software systems.

FIGURE 10 - 10 **Major parts of Sequence Executor control**



10.2.4.3 SEQUENCE COMMANDS

The Observatory Control System defines a small set of high-level OCS Commands that it can send to any of the other principal software systems, and those systems *must* react or respond correctly and appropriately based on the definitions of the OCS Commands and the duties each system is to perform. At this time there are two groups of commands: configuration commands and status commands. The Sequence Commands are a subset of the OCS configuration commands. The other OCS commands are defined in “Run-time Dynamic Headers” on page 10 - 59.

To handle these commands, the principal systems are required to receive configuration information as a set of attributes and values and process them in a manner appropriate for their system. The interface used to pass sequence commands to the other software systems is described in "OCS Software Interfaces" on page 10 - 53 and Interface Control Document 1 [24].

Table 10-1 shows the current revision of the sequence commands. The commands describe the structure of the observing process. The sequence executor itself runs as a set of state machines and assumes that the principal systems do not run in lock-step. The Sequence Executor sends its commands based on its own state but does not force the state machine model on the principal systems it controls. At any time, a principal system must accept any of the commands in Table 10-1; it may then process or reject the command. It is the job of the sequence executor to make sure the software systems are correct before progressing in its recipe.

A few comments regarding the sequence commands are listed here.

- Commands **config(verify)** and **config(endverify)** are present to indicate to systems that operators and observers are involved in verification of the system state. Systems may choose to do something appropriate during verify, but are never allowed to turn-off interactive command capability.
- Movie-mode capability, a mode where frames are generated as quickly as possible, viewed, and thrown away, is no longer controlled through Sequence Commands. Rather, a Science Configuration is constructed indicating that a detector should be operating in movie-mode. The event-driven Data Handling System accepts the data from the Instrument Control System and makes it available as quickly as it arrives.
- Commands **config(test)**, **config(init)**, and **config(reset)** all require the target system to perform an initialization procedure. **config(reset)** can be used quite often to bring a major system to a known state, **config(init)** is more severe and would be used when problems occur in a principal system. **config(test)** is the lowest level initialization and requires the system to perform extensive internal testing.

TABLE 10 - 1 **Sequence Command definitions (revision 2.0)**

Event (Configuration Command)	Command Arguments	Action/ Definition
config(test)		A system should assume it has just been switched-on and perform self-tests to guarantee it is healthy. The command completes successfully when it passes its tests or it fails returning the fault.
config(init)		The system should execute its most complete initialization sequence. This can include rebooting and reloading any internal setup files (a *hard* init). The command completes successfully when the system is initialized or it fails returning the fault.
config(reset)		A system should do whatever is needed to reset its internal system state to the state it had at start-up and become ready for new commands (a *soft* init). It should NOT reboot or re-read any setup files. This initialization command is less severe than **config(init)**.and **config(reset)** should be the final phase of **config(init)**. The command completes successfully when the system is initialized or fails returning the fault.
config(park)		A system should adopt an internal configuration in which it can be safely switched off. This will occur at the end of an observing session. The command completes successfully when the system is ready to be powered down or fails returning the fault.
config(check)	a configuration	A target system checks that the configuration is valid and that it can process the configuration. It returns an error if either condition fails.
config(apply)	a configuration	First, the target system checks that the configuration is valid and that it can process the configuration. It returns an error if either condition fails. Then, the system moves to the requested configuration, positioning any motors, mechanisms, etc.as requested. The command completes successfully when the system has successfully completed all motions or actions required to match the configuration.

Event (Configuration Command)	Command Arguments	Action/ Definition
config(verify)		This command indicates to a system that verification of configurations is underway by the OCS, operators, and observers. A principal system must be capable of executing changes to its state during a verify, and it must also update its status and state in the ODB. Interactive commands must *always* be accepted by a system. This command only provides information for principal systems and requires no special actions. A principal system should successfully complete immediately after noting the **config(verify)** command.
config(endverify)		This command indicates to a system that verification of configurations is finished. This command can be executed at any time. This command only provides information for principal systems and requires no special actions. A principal system should successfully complete immediately after noting the **config(endverify)** command.
config(guide)		This command indicates to the telescope that it should start guiding. The telescope should do whatever is indicated for guiding in the configuration. A principal system must monitor its configuration and report any failures while guiding and should execute any system particular behavior that should occur while the telescope is guiding. The command completes successfully when the **config(guide)** actions have begun successfully. Systems that choose to ignore the **config(guide)** command should immediately complete successfully.
config(endguide)		This command indicates to the telescope that it should stop guiding. A system should execute any particular behavior that should occur when the telescope stops guiding. Systems that choose to ignore the **config(endguide)** command should immediately complete successfully.

Event (Configuration Command)	Command Arguments	Action/ Definition
config(observe)		This command indicates that data acquisition should begin in an instrument system based on its current internal values. Instruments executing a **config(observe)** remain busy until they have completed the configured observation. The sequence executor uses completion of **config(observe)** to determine when an observation completes. Systems other than instruments should complete **config(observe)** immediately either successfully or with failure returning the fault.
config(endobserve)	Filename, integration ID, and header info (DHS only).	This command indicates to systems that the current exposure has been completed by the instrument(s). Systems should complete **config(endobserve)** immediately either successfully or with failure returning the fault.
config(pause)		This command indicates that a system should do whatever is appropriate for it to pause data acquisition. Pause indicates to the principal system that the user intends on continuing at a later time. Instruments must mark in their global state whether or not they can be paused. The command completes successfully when the **config(pause)** actions have begun successfully. Systems that choose to ignore the **config(pause)** command should immediately complete successfully.
config(continue)		This command is the reverse of pause. A system should do whatever is appropriate for it to resume data acquisition. The command completes successfully when the **config(continue)** actions have taken place successfully. Systems that choose to ignore the **config(continue)** command should immediately complete successfully.

Event (Configuration Command)	Command Arguments	Action/ Definition
config(stop)		This command indicates that a system should stop the current data acquisition process normally, as if it were the end of the data acquisition period. The command completes successfully when the **config(stop)** actions have taken place successfully. Systems that choose to ignore the **config(stop)** command should immediately complete successfully.
config(abort)		This command indicates that a system should stop the current data acquisition process immediately and discard any data. The command completes successfully when the **config(abort)** actions have taken place successfully. Systems that choose to ignore the **config(abort)** command should immediately complete successfully.

10.2.4.4 CONFIGURATION COMMAND STATUS AND ERRORS

The sequence executor must track the progress of the principal systems and any problems or errors that may occur during the execution of an observation. The Sequencer Commands behave and are implemented as defined in "ICD 1 - The System Command Interface" (ICD1 [24]). All commands are executed asynchronously; a command always returns immediately having either accepted the command or rejecting it. The calling system can monitor the execution of the command or be notified when a command is completed. The following information is available to a sequence executor for each Sequence Command that is executed.

Sequence Command Busy. All commands must set their busy state, which can have one of two values: BUSY and DONE. Once a system accepts a Sequence Command, it sets its the command's *busy* attribute to BUSY before acknowledging acceptance of the command. When a principal system is finished executing a Sequencer Command, it sets command Busy to DONE. The OCS uses this attribute to monitor principal system activities.

System Configuration Command Error. Command Error is an attribute indicating a system's success or failure when executing a Sequence Command. A system should set this attribute before setting Configuration Busy to DONE. All OCS commands result in OK, WARNING, or FAILURE. See "Alarms, Errors, and Status Reporting" on page 5 - 40 for more information on OCS errors.

System Configuration Command Error String. If a Sequence Command results in WARNING or FAILURE, this attribute is set to be a text message indicating the most probably cause of the problem in such a way that an operator can understand it and act upon it.

10.2.4.5 THE SEPARATION OF COMMANDS AND ALARMS AND STATUS

The OCS Sequencer Commands are system commands that result in activities in a principal system that may change its state as described by its status information. In the GCS, commands and their completion status, and a principal system's status information are separate. A Sequence Executor doesn't use a system's status information to determine when a command has completed and whether or not it has succeeded or failed; the command completion information for a particular command is determined by the commanded principal system and passed back to the commanding system (ICD1 [24]).

The GCS status and alarm interface and implementation is discussed in..... and defined more fully in "ICD 2—Systems Status and Alarm Interface" [25].

10.2.4.6 SEQUENCE EXECUTOR DIRECTIVES

The Sequence Commands are what comes out of a Sequence Executor. There are also commands that can go into an SE. These commands are directives to the SE issued by the parts of the OCS that can control a SE; currently, the Sequencer and the Observing Tool.

TABLE 10 - 2 **OCS Sequence Executor directives**

Sequence Executor Directive	**Command Arguments**	**Definition**
se(start)	an observation name	A Sequence Executor is directed to start execution of the observation given as the argument of the command.
se(pause)		The Sequence Executor should pause execution of its observation. Pause implies that execution will be continued at a later time.
se(continue)		The Sequence Executor should continue execution of its observation.
se(stop)		The Sequence Executor should stop execution. Stop means terminate the current observation.normally with config(stop).
se(abort)		The Sequencer Executor should abort execution - this implies any ongoing observation will be sent config(abort).
se(done)		The Sequence Executor is told that it is about to be destroyed by the Sequencer. It should do any required clean up immediately.

The se(start) directive indicates that the executor should begin executing the observation indicated in the command argument. Under some observing circumstances, the observer is allowed to restart a Sequence Executor with a configuration he has created. This occurs in situations where the operator and the observer have decided that *one-more* exposure is needed before completing the execution of the observation.

The se(pause), se(continue), se(stop), and se(abort) directives are available to allow the operator or observer to interact with an ongoing observation.

The se(done) directive tells the executor that it is about to be destroyed.

10.2.4.7 THE SEQUENCE EXECUTOR RULE ENGINE

Once the Sequence Executor receives a directive it must decide how best to execute the observation. The job of the Sequence Executor Rule Engine or Rule Engine is to examine the next target science configuration, compare it to the current configurations of the principal systems that appear in the target science configuration, and to decide how to bring the principal systems into compliance with the target configuration. The Rule Engine will be part of each Sequence Executor.

This decision capability is implemented through a rule set, a hopefully simple set of tests that will analyze the target and current configurations and generate the correct series of Sequence Commands. The actual rule set is TBD by the OCS Work Package group. The following are situations that must be properly treated by the Rule Engine.

- If an observation was previously executed, and one or more of the principal systems still match the previous configuration, the Rule Engine will only send **config(apply)** to the principal systems that need adjustment.
- The executor must properly treat the “one-more” requests that may come from the Observing Tool. One proposed operating scenario allows astronomers to create and submit new configurations with the Observing Tool directly to a Sequence Executor during the execution of one of their observations. The Rule Engine must decide which new configurations will be allowed and which ones will be rejected and forced to through the Scheduling System.

The Rule Engine rule set need not be complex. The rules will generally be based on comparisons between the components of the target and current configurations. The following are simple rules which can make correct recommendations for the problems given above.

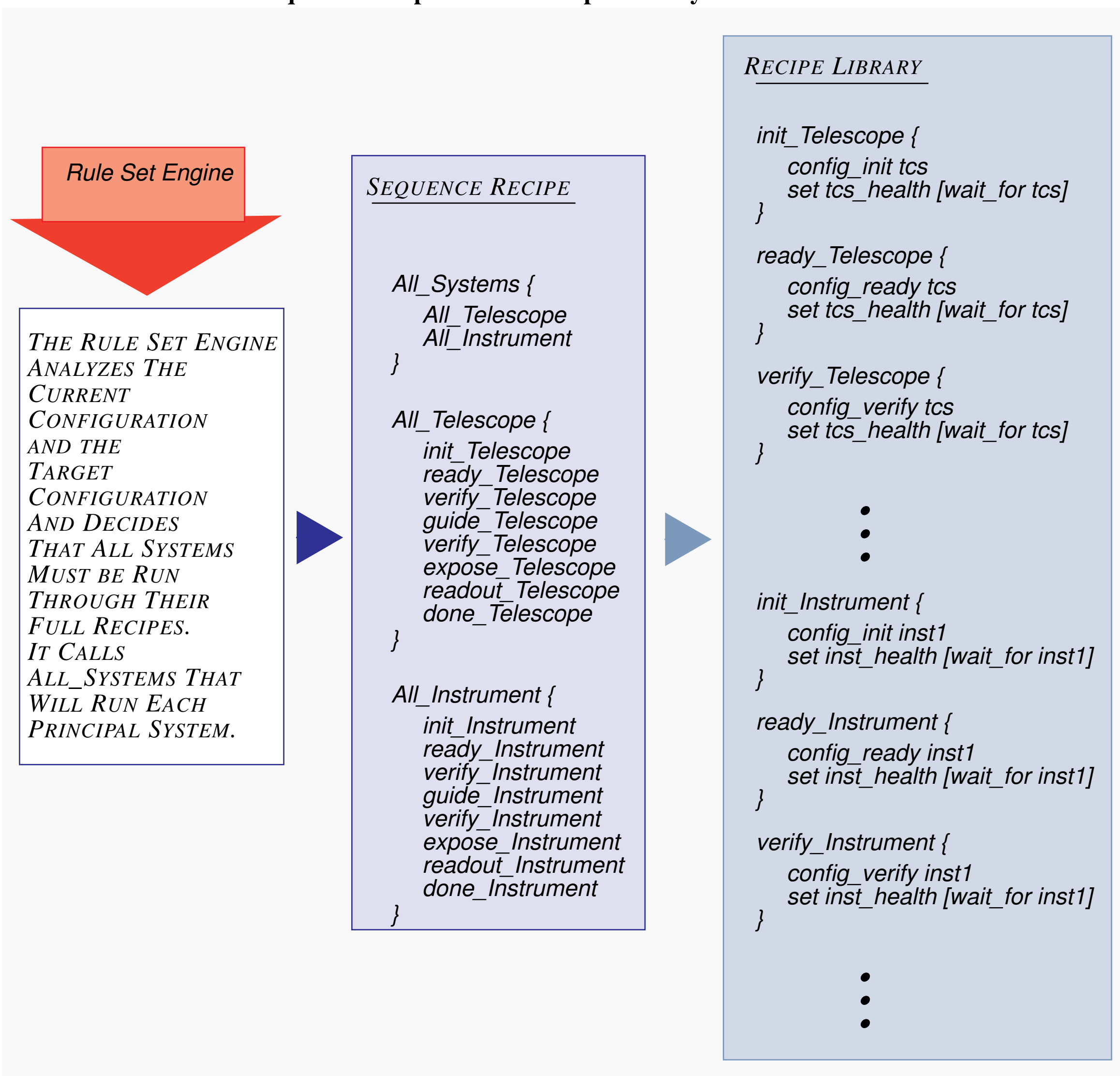
- After an observation a system is left ready for a **config(observe)** or a **config(apply)** Sequence Command. If the next configuration is identical to the current configuration for one or more of the configuration components, the Sequence Executor need issue no commands. This test is done for each principal system.
- Rules specify the policy describing how “big” of a change is acceptable for observers to issue without going through the Scheduling System. For instance, different instrument configurations may be fine but different telescope configurations may require submitting the configuration through the operator/Scheduling System.

  A finer grain control might specify that telescope configuration changes are allowed but not large telescope position changes (i.e. no new slew targets).

10.2.4.8 THE SEQUENCE RECIPE AND RECIPE LIBRARY

The final parts of the Sequence Executor are the Sequence Recipe and the Recipe Library. The Executor Rule Engine decides what needs to be done to match the current configuration and a target configuration, and then does it by executing procedures with standard names that reside in the Sequence Recipe and Recipe Library. The Sequence Recipe presents a slightly higher level of abstraction of the Recipe Library to the Executor Rule Engine.

FIGURE 10 - 11 **Sequence Recipe and the Recipe Library**



This idea is illustrated in the simplified example of Figure 10 - 11. This example is meant to show the roles of the Sequence Recipe and the Recipe Library. The Recipe Library in Figure 10 - 11 shows example low-level communication primitive procedures written for each of the Sequence Commands—a set for each principal system. Each primitive procedure sends a Sequence Command to a principal system, waits for it to complete, and then sets an error condition.

The Sequence Recipe implements a higher level, complete recipe in procedure *All_Systems*. This procedure calls other procedures that sequence each of the principal systems (CCD and Telescope Control System) in the configuration. These procedures then call procedures in the Recipe Library. Figure 10 - 11 is only an example, and the actual recipes will be more complicated and will use errors to do recovery and inform users of problems. This example actually results in the telescope and instrument running completely asynchronously. The example does show how the Sequence Rule Engine will make calls to the appropriate parts of the Recipe that will then call low-level procedures in the Recipe Library.

It is required that the Observing Database be kept up to date with the state of the principal systems during the processing of configurations allowing the Sequencer Recipe to determine the status and progress of the principal systems.

A Sequence Recipe will present a standard set of procedures that can be called by the Sequence Rule Engine. Figure 10 - 11 shows *All_Systems*, *All_Instrument*, and *All_Telescope* as example recipe procedures. Others are required to implement common requirements. For instance, when the observer is repeating an observation, the Rule Engine calls just the exposure-related procedures in the recipe. The definitions of procedures in the recipes and the recipe library itself are TBD by the OCS Work Package group.

Recipes and parts of Recipe Libraries will be written in TCL. Primitives for the TCL scripts will be written in a compiled language to keep the scripts as simple as possible and to speed execution. A Science Configuration's recipe is an attribute in the configuration itself and can change for different programs if necessary.

10.2.4.9 SYNCHRONIZATION AND VERIFICATION

Figure 10 - 11 showed a simplified example that ignored some real world problems. Real recipes will be required to run principal systems asynchronously for the first few executor states and then provide rendezvous points where a recipe will wait until all the required principal systems are in states that allow the recipe to continue. This is primarily because of the desire to execute multiple observations in parallel to optimize telescope efficiency. Also, at appropriate steps in the execution of a recipe, a Sequence Recipe needs to wait for operator/user input or verification before continuing. This section discusses how this functionary is provided.

FIGURE 10 - 12 **Flow in an example Sequencer Recipe**

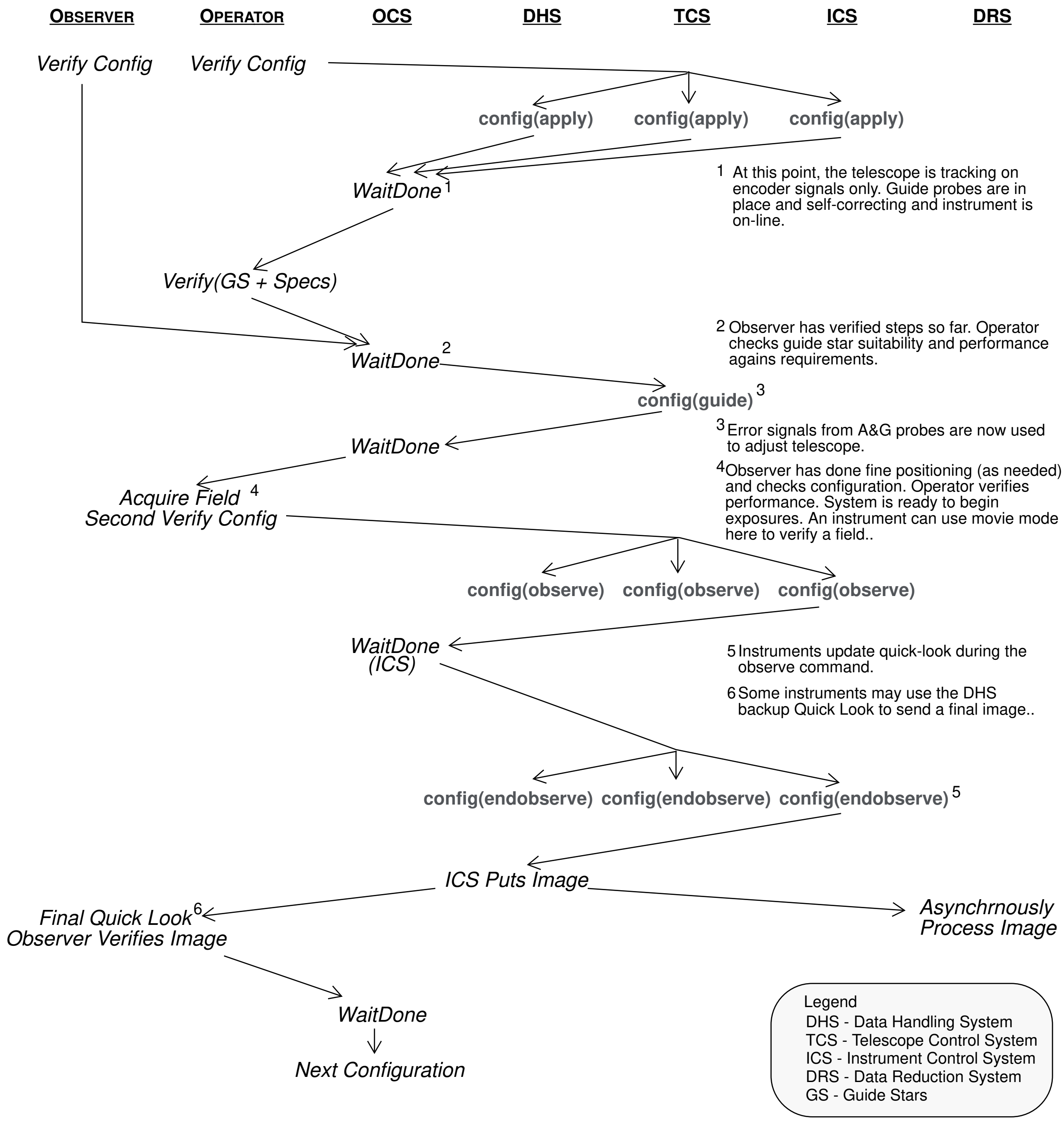


TABLE 10 - 3 **Description of interaction steps in Figure 10 - 12**

Step	Step Command	Activity
1	sequencer sends **config(apply)**	All subsystems are instructed to match the specified configuration. The telescope moves to the new target as do probes, etc.
1W	sequencer waits	The Sequencer waits until all subsystems are no longer busy and checks for any errors. The operator is alerted on error.
2	First Verify	Observer Verifies configuration to this point. Operator verifies guide object suitability before okay to guide.
2W	sequencer waits	Verification by operator and observer has taken place.
3	sequencer sends config(guide)	Error signals from A&G are now going to telescope.
3W	sequencer waits	Internal wait for guiding to commence.
4	Second Verify	Observer attempts to verify his target field. He may execute any system functions/operations to assist in verification. Some detectors may operate in movie-mode to assist field verification. The activities may result in a “modified” current configuration.
4W	sequencer waits	At some point the operator and observer have verified the field and adjusted the configuration. Once finished, the system compares current configuration with the adjusted configuration and allows the observer to “correct” the database configuration if needed.
5	sequencer sends **config(observe)**	The sequencer sends the “observe” message to all principal systems. The instrument begins collecting data!
5W	sequencer waits	All principal systems besides the ICS return immediately. The instrument controls systems return from observe indicates the end of the exposure.
6	sequencer sends **config(endobserve)**	All principal systems are notified that an exposure has completed. The OCS sends header data to the DHS at this time.
6W	sequencer waits	All instruments process **config(endobserve)**. The ICS puts the image data to the DHS which updates a DHS quick-look if needed. The DHS then sends the new image off to the DRS for asynchronous reduction.
7	Final Verify	The observer verifies his data and the system moves on to the next configuration.

Figure 10 - 12 presents another simplified Sequencer Recipe that emphasizes who does what during the execution of a sequence recipe. It shows the flow of sequencer

commands and user interaction steps that might occur in a recipe used to process a planned observing configuration. The columns show when activity occurs in a principal system (the Data Reduction System is actually within the Data Handling System) or when interactions with the operator and observer are required.

Table 10 - 3 is a summary of the recipe with descriptions of the activities that take place at each step. Steps 1W, 3W and 6W show internal rendezvous points and steps 2, 4, and 7 show verification stages requiring the operator and observer to verify the system configuration before proceeding.

Observer/Operator Verification. The verification points are the places that allow for interactivity in a recipe. During these times the operator and observer may execute other applications, scripts, or commands from consoles that may be required before observing can continue. These applications might be quite complicated involving the telescope and the Data Handling System Quick-Look tools based on PV-Wave and these tools may interact with more sophisticated data reduction systems.

After a verify step is completed, the expected configuration and the actual configuration are compared, and the observer has the option of updating the original configuration if needed.

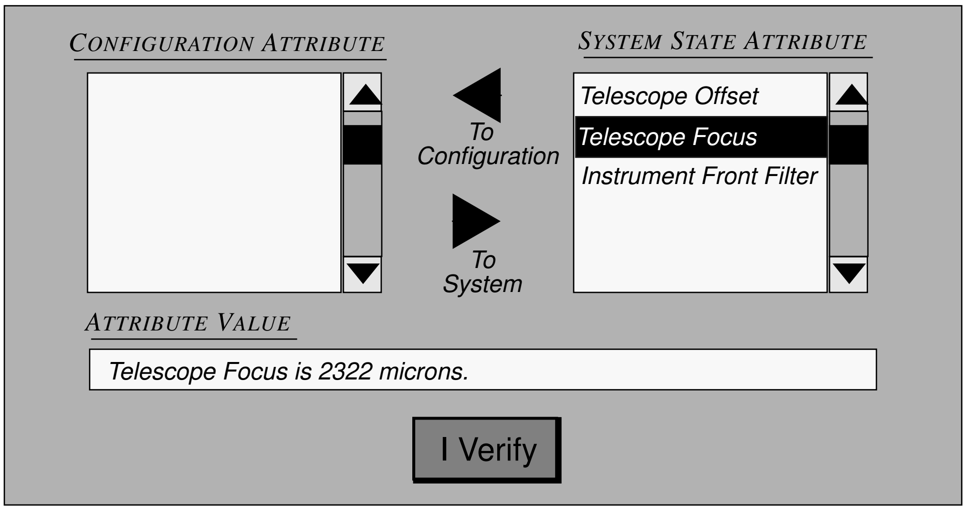
How will the observer and operator know that a recipe is at a point where verification activity should take place? The observer and operator will be using the Observing Tool to monitor the observations as they are executing. There are least two possibilities for communication with the observer, and both are based upon up-calls (calls from a lower level system *up* to a higher level system) from the Sequence Executor to the applications used by the observer and operator.

1. *Use of the Observing State*. The Observing Tool monitors the execution of the observer's observation in the Observing Database. At verification points, the Recipe notifies the observer that it is time to verify and waits until he checks "okay".
2. *Observing Database Interface*. The Observing Tool provides a public interface that allows other programs to control it. The Sequence Executor contacts the observer's OT through a public command requesting a verification (`config(verify)`). The OT verification sets a status field in the global configuration just like other principal systems.

Method two is attractive because it says the Observing Tool responds to Sequence Commands in the same way as other principal systems.

During verification there is a large possibility that the observer or operator will adjust the principal systems in some way that changes their configuration. Once a verification step has concluded, the Sequence Recipe compares the current master configuration in the Observing Database with the observation's Science Configuration. The observer and operator are then given an opportunity to indicate which attribute values are to be used in the configuration. The correct values are saved in the configuration and the principal systems are sent **config(apply)**. A prototype verification dialog is shown in Table 10 - 13.

FIGURE 10 - 13 **A prototype verification dialog**



The example prototype dialog appears when the verification stage is entered. During verification, the two configurations are repeatedly compared and attributes that are different in the master configuration appear in the right side list. Once changes are made, the observer and operator decide which values should be used in the configuration. If no action is taken, the updated master values on the right will be used in the configuration after verification. Selecting an attribute in a list shows its value in the attribute value box below. Selecting an attribute and clicking the *To Configuration* arrow moves the attribute name to the left list indicating that the Configuration attribute value should be used after verification. Clicking the *I verify* button indicates to the Sequence Executor Recipe that the user is ready to continue.

Recipe Rendezvous. Recipe rendezvous is easy to implement. The Recipe Library will provide primitives that allow the Recipe to loop until one or more principal systems reach a desired state. The example TCL recipe that follows shows an example of a rendezvous recipe construct.

10.2.4.10 HOW WOULD A RECIPE LOOK?

For visualization purposes only, Algorithm 10 - 2 implements a recipe like the one in Figure 10 - 12. It is written in TCL with a smattering of prototype TCL Sequence Executor commands added. Actual recipes would be more complex and would need to have well thought out "evasive actions." The acquisition of resources would be more complex since a required resource may not be available when the recipe begins executing.

ALGORITHM 10 - 2 A sample Sequencer Recipe

```
# Get the name of the observing database entry for the
# configuration. The entry is passed as an argument to
# the script.
set config_name arg1

# Find out all subsystems in a configuration
set subs [subsin $config_name]
# Find out instruments in configuration
set insts [instsin $config_name]

# Acquire resources
foreach subsystem $subs {
   acquire_resource $subsystem $config_name
}

# Send each subsystem a config_ready message and wait
foreach subsystem $subs {
   config_apply $subsystem $config_name
}

# Wait for completion
# health is set to ok if all ended normally
set health [waitfor $subs]
if {$health != "ok"} {
  # Alert operator and clean up
  return $health;
}

# Now the operator and observer must verify so wait
set observer_verify [verify "observer"]
set operator_verify [verify "operator"]
if {$observor_verify != "ok" || $operator_verify != "ok"} {
  # Something was wrong so take evasive action
```

```
  # This may involve an abort, or restarting the config
  evasive_action $config_name
  return not_ok
}

# Turn the guiders on if they are present in the config
# Send each subsystem a config_guide message and don't wait
foreach subsystem $subs {
  config_guide $subsystem $config_name
}

# Wait for completion
# health is set to ok if all ended normally
set health [waitfor $subs]
if {$health != "ok"} {
  # Alert operator and clean up
  return $health;
}

# Now the operator and observer must verify field
# verification so wait
set observor_verify [verify "observer"]
set operator_verify [verify "operator"]
if {$observor_verify != "ok" || $operator_verify != "ok"} {
  # Something was wrong so take evasive action
  # This may involve an abort, or restarting the config
  evasive_action $config_name
  return not_ok
}

# Send each subsystem a config_observe message and don't wait
foreach subsystem $subs {
  config_observe $subsystem $config_name
}

# Wait for completion
# health is set to ok if all commands were received
set health [waitfor $subs]
if {$health != "ok"} {
  # Alert operator and clean up
  return $health;
}
```

```
# Wait for observation to end
set health [waitobs $insts]
if {$health != "ok"} {
  # Alert operator and clean up
  return $health;
}

# Send each subsystem a config_endobserve message
# and don't wait
foreach subsystem $subs {
  config_endobserve $subsystem $config_name
}

# Wait for completion
# health is set to ok if all commands were received
set health [waitfor $subs]
if {$health != "ok"} {
  # Alert operator and clean up
  return $health;
}

# Now the observer must verify final image verification so wait
set observor_verify [verify "observer"]
if {$observor_verify != "ok"} {
  # Something was wrong so take evasive action
  # This may involve an abort, or restarting the config
  evasive_action $config_name
  return not_ok
}

# Free resources
foreach subsystem $subs {
    free_resource $subsystem $config_name
}
```

This is an example only and is meant to show how a high-level sequencer language could communicate with the principal systems and how TCL could be used to implement Sequencer Commands. The Sequence Library allows the observatory staff to compose Sequencer Recipes from parts of other recipes that are known to work.

The recipe of Algorithm 10 - 2 is the recipe of a simple, single frame configuration. Complex observations such as a mosaic or a wobbling scenarios, may actual execute a special "observe script" that is included with the configuration data. The observe script would be executed in the recipe where the config_observe is executed in Algorithm 10 - 2. Algorithm 10 - 3 is an example of a mosaic observe script.

ALGORITHM 10 - 3 A More Complex Observe Script

```
proc observe con{config_name} {
    [show tcs*primar:altitudetracking
    set x [show config_name:mosaic:offset:x]
    set y [show config_name:mosaic:offset:y]
    set xnum [show config_name:mosaic:offset:xnum]
    set ynum [show config_name:mosaic:offset:ynum]

    # Assume 0.0 is the always upper left corner of mosaic
    offset 0.0 0.0

    for {set i 0} {i<$ynum} {incr ynum} {
        for {set i 0} {i<$xnum} {incr xnum} {
            # Send config_observe message and don’t wait
            foreach subsystem $subs {
            config_observe $subsystem $config_name
            }

            # Wait for completion
            # health is set to ok if all commands were received
            set health [waitfor $subs]
            if {$health != “ok”} {
                # Alert operator and clean up
                return $health;
            }

            # Wait for observation to end
            set health [waitobs $insts]
            if {$health != “ok”} {
                # Alert operator and clean up
                return $health;
            }

            # Send each subsystem a config_endobserve message
            # and don’t wait
            foreach subsystem $subs {
                config_endobserve $subsystem $config_name
            }
```

```
            # Wait for completion
            set health [waitfor $subs]
            if {$health != "ok"} {
                # Alert operator and clean up
                return $health;
            }

            # offset the x direction
            offset 0.0 $x
        }
        offset $y -$xnum*$x
    }
}
```

The script may not be perfect TCL, but it shows how more complex observe scripts can be build out of the sequence recipe library primitives.

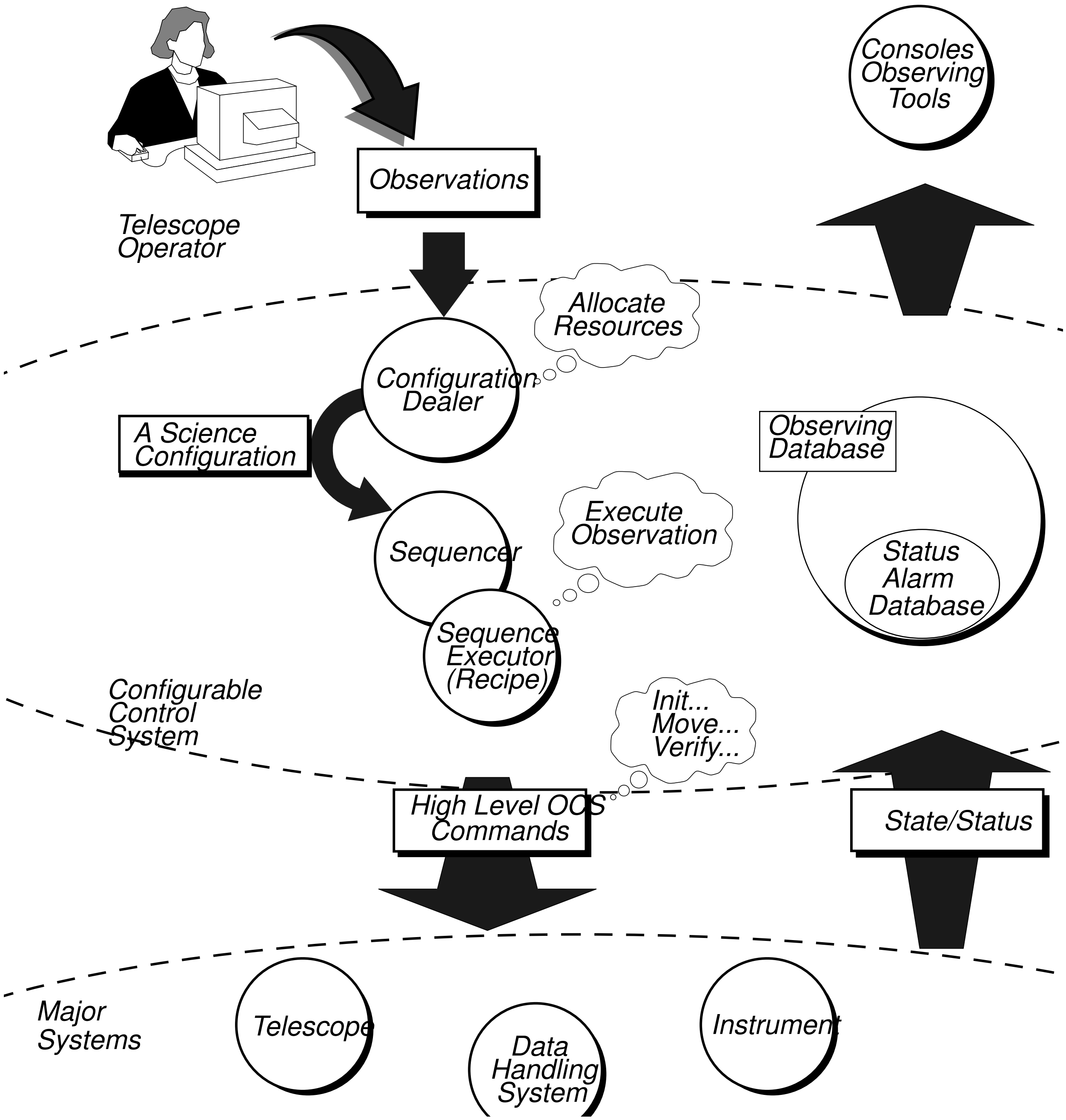
Basing the recipes on the TCL scripting language and the development of recipe libraries should result in relatively simple recipes that are flexible and can be easily modified by staff when required.

Even though the performance of TCL-based applications is good (as shown by the abundance of Tk programs - all of which execute entirely in TCL), the recipe library commands will be written in a compiled language. Not only will this guarantee recipe performance, it should simplify the TCL recipe scripts.

10.2.4.11 CONFIGURABLE CONTROL SYSTEM SUMMARY

The Configurable Control System is the part of the Observatory Control System that executes observations. This function has been presented in detail in order to be confident that the software is capable of executing observations and supporting the required observing modes. Figure 10 - 14 is a summary of the information flow in the CCS. Status and alarms are covered in an upcoming section.

FIGURE 10 - 14 **Summary of the information flow in the Configurable Control System**



10.2.5 The Observing Database

The Observing Database (ODB) is referenced throughout the high-level and low-level OCS design documents. The ODB role is one of support for the OCS applications and other principal software systems; it has no Visible User Interface. It is a major part of the software infrastructure of the Observatory Control System. The following are the most obvious places where the ODB has appeared in the OCS design.

Executing processes. The observations and their science configurations exist in the ODB once they are scheduled to execute by the operator. At the time they are scheduled the information describing the observation and configurations moves from the external database to the ODB.

This data movement requires that the ODB create a database record to represent the observation. The Observation structure is not a fixed size. Observations can have any number of configurations and the configurations can have any number of attributes.

Global system state. The principal systems all maintain the values of a large number of their public attributes (process variables in an EPICS system) in the global system state. The global system state is part of the Observing Database.

The dynamic headers feature of the OCS VUI (See “Run-time Dynamic Headers” on page 10 - 59) requires that the upper level system be able to include new attributes in the Observing Database.

Observing Tool. The Observing Tool can create configurations during the execution of observations. The observer can change attribute values any time before a configuration is executed.

VUI Consoles. The Observing Database must provide functionality that allows the updating of OCS consoles to reflect the values of the attributes they control and display.

This functionality should not be based on polling of the ODB by the consoles. Instead, the ODB allows consoles to register an interest in receiving updated attribute values when the attribute values change. The consoles can indicate conditions that describe under what conditions the updates should occur.

Console Commands. The consoles and the DHS data analysis tools must have the capability of issuing commands to the principal systems. The commands will exist in the Observing Database during the time they are extant.

A design for the part of the Observing Database that contains the global system state has been specified because it is part of the interface between the principal systems, but this document does not indicate a software package that should be used for the remaining parts of the Observing Database. Rather, it specifies a set of requirements that the OCS designers feel must be present in any new software or other software that can be used for the Observing Database. These requirements are based on the previous list of ODB uses and the operations on Science Programs, Observations, and Science Configurations on page 10 - 8.

10.2.5.1 REQUIREMENTS FOR ODB PACKAGE

1. Dynamic creation and initialization of database. The ODB will be empty when Gemini Control System starts up.
2. Dynamic creation and destruction of complex hierarchical database records.
3. Shallow and deep copy of entire records or portions of a record’s hierarchy.
4. Dynamic addition of hierarchical items to already existing hierarchical database records.
5. Dynamic addition of attributes to database records.
6. Ability to change the values of attributes at run-time.
7. Operations that allow hierarchical components to be compared are required.
8. The database should allow distributed access.
9. Applications using the database must be able to register a function that will be called when defined database conditions are met.
10. The database package must provide an interprocess communication functionality that allows applications to access the database features.
11. A programmer interface to all these required database features.

10.2.5.2 SIMILAR DATABASE SYSTEMS

There are at least three already-existing software systems that use attributes/values in a way that is similar to the discussions in this document. It will be up to the OCS Work Package group to indicate an implementation for the ODB.

X11 Window System. The representation presented for attributes and values presented in this document is based on the representation of widget resources in the XLIB library of the X11 Window System. The XLIB library supporting resources is publicly available and is maintained and improved by the X Consortium. There are a number of features related to attribute naming (such as wild-carding names) in XLIB that would also be useful for the OCS application. They have not been required in this design but would be useful.

The RTAP Environment. Hewlett-Packard’s RTAP/Plus is a real-time, workstation-based, commercially-available, database environment for industrial control. The real-time RTAP database is hierarchical and supports the operations of the previous section. The RTAP Application Programmer Interface provides access to the real-time database through a supplied interprocess communication library.

The EPICS System. EPICS, the software environment used in the real-time parts of the Gemini Control System, also provides a database system. The EPICS database is flat, static, and all attributes are public and global. The length limitation on attribute names also presents a difficulty. Currently, the database software is only available on VxWorks systems. At this time, it seems difficult to implement the component operations in the EPICS system since the EPICS database is static and loaded when the EPICS system is initialized.

10.2.6 Status and Alarms in the OCS

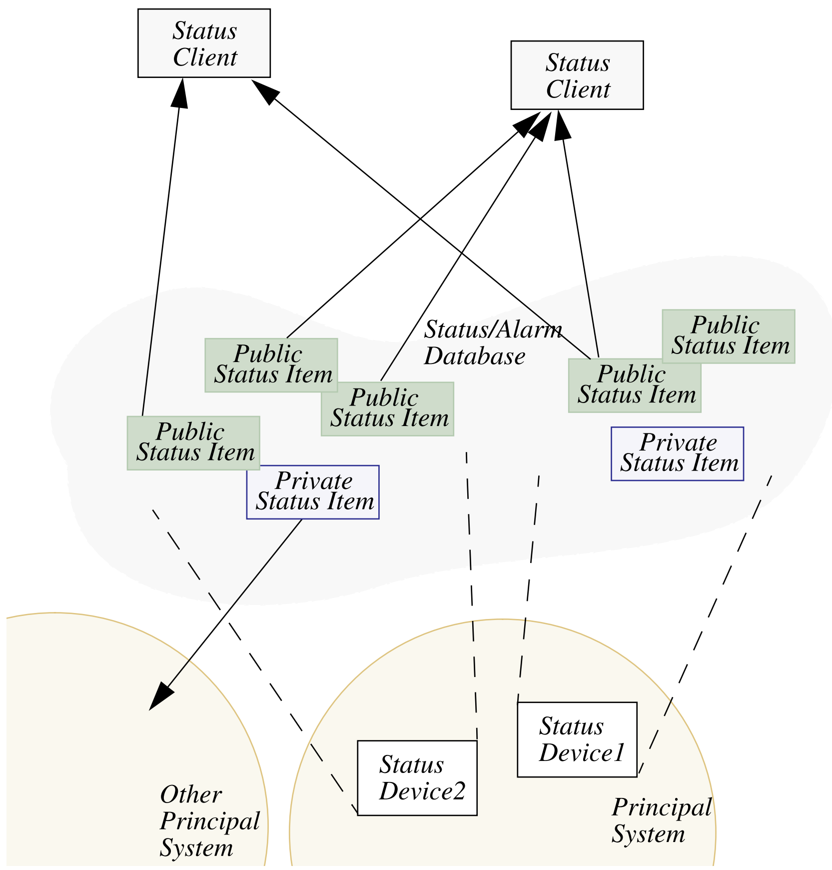
In “Alarms, Errors, and Status Reporting” in Chapter 5, the high-level approach to errors, alarms, and status was presented. This section describes the design and implementation of alarms, status, and health in the OCS. For more information, see ICD2—Systems Status and Alarm Interfaces [25].

10.2.6.1 STATUS ALARM TERMS

A *status device* is a piece of hardware or software that has state or information that is of interest to *status clients,* other parts of the software system that wish to monitor status. A *status item* is one piece of information provided by a status device, the *owner* of the status item. Status items are read-only; they can not be modified by any system entity other than the owner of the status item. A status item is part of the status device’s *public status interface* if it is available to any status client in the control system. A status device’s *private status* is the set of status items that are available only to a subset of possible status clients. For example, the TCS may have status items that are only readable by its subsystems.

Figure 10 - 15 shows the relationship between the terms and entities just defined. Each of the two status devices in the principal system owns a set of public and private status items indicated by the dotted boundary emerging from the status device. Each of the status devices keeps its set of status items up to date. Public status items can be monitored by any program in the system. Private status items can only monitored by systems allowed by the owner.

FIGURE 10 - 15 **Relationship between items and definitions in section 10.2.6.1.**



10.2.6.2 STATUS BEHAVIOR

The following items define the important aspects of the behavior of the GSC status system.

- Any status item that is part of a device's public interface is available at all times to any status client through a common status software interface.
- It is the responsibility of the owner of a status item to keep its public and private status items up to date; always reflecting the correct state of the status device.

- Status clients can read the value of status items at any time, but the primary way clients keep abreast of the values of status items is through a *monitor mechanism.* Status clients register interest in a status item and are notified when that item is changed by its owner.

10.2.6.3 ALARM BEHAVIOR

Alarms are built upon the functionality of the status system. An alarm is a notification of an unusual, abnormal, or failure condition in a status device. The act of notifying status clients of abnormal conditions is called *raising an alarm*.

The following items define the important aspects of the behavior of the GCS alarm system.

- Developers of alarm devices should make sure that alarms are raised as near in time to the system event that generates the abnormal condition as possible.
- The owner of a status item determines when the status item is in an alarm state and what the severity of the alarm is.
- The status monitor functionality of the previous section provides the means for alerting users of alarm conditions.
- The alarm mechanism should also provide notification of a failure of the alarm system. For instance, it should be possible to determine the difference between an alarm in the status device and a loss of all information from the status device as in a power failure.

10.2.6.4 HEALTH BEHAVIOR

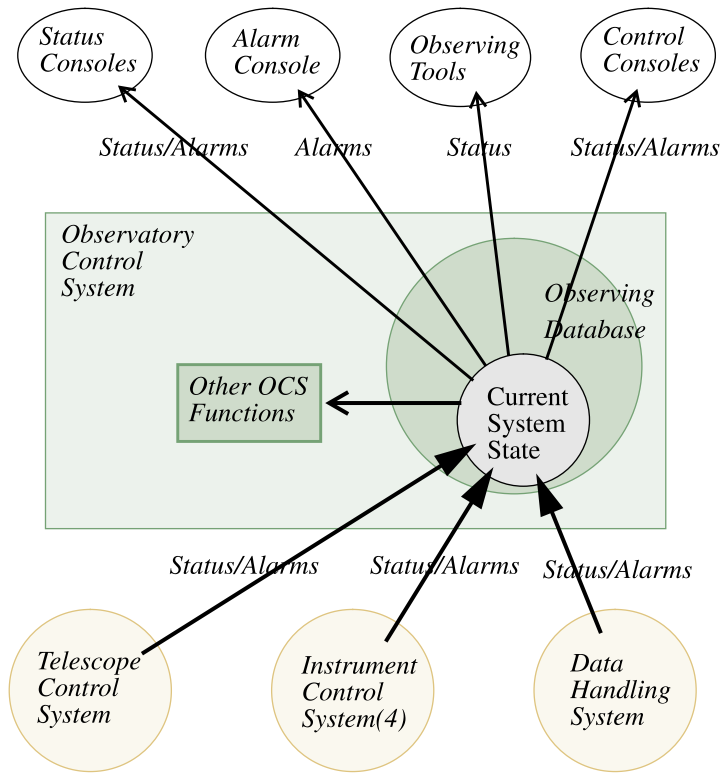
The GCS views the health information as a pre-defined kind of alarm that all systems must provide. Health is a global indication of the overall operational status of a particular status device or group of status devices. The health status device is recursively defined in terms of the health of all the status devices that are part of the parent status device.

Health can have three values (good, warning, and bad) and what determines whether the system can continue operations in the warning state shall be determined by the Work Package groups.

10.2.6.5 THE OCS ALARM/STATUS ENVIRONMENT

Figure 10 - 16 shows the relationship between the principal systems and the Observatory Control System and the flow of status information in the system.

FIGURE 10 - 16 The Flow of Status Information in the Gemini Control System

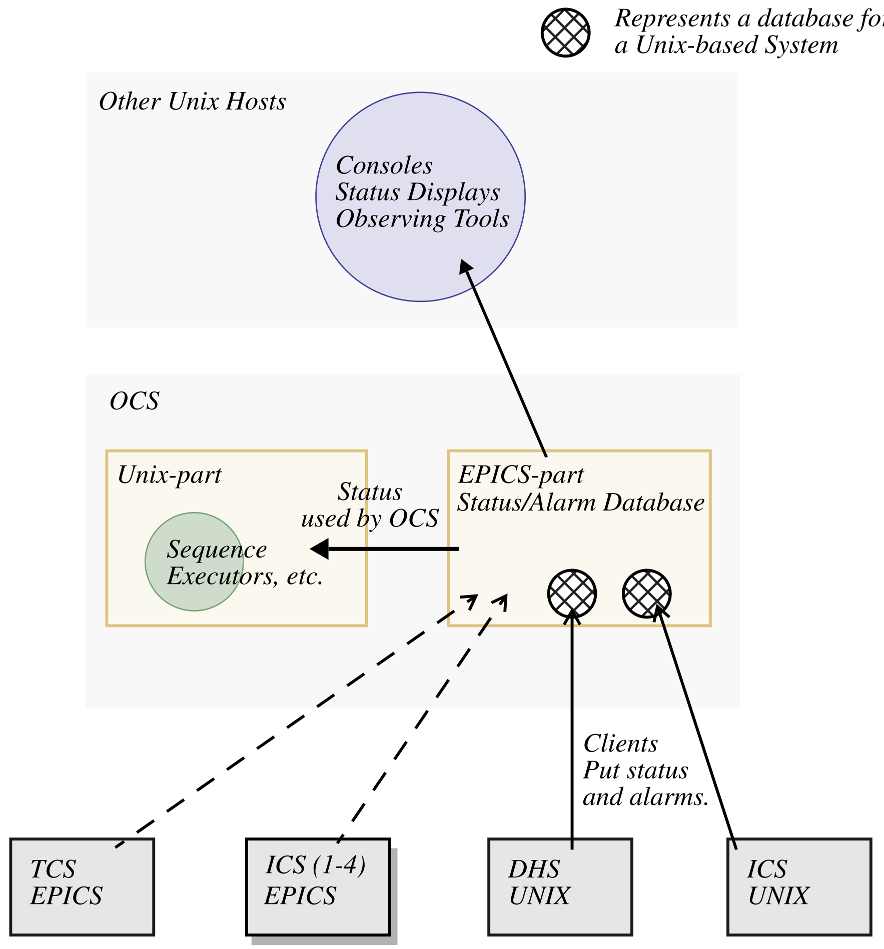


Part of the Observing Database is the current system state that is the union of all public status items of all status devices in the principal systems. Status information is used by the OCS to update the various kinds of interface screens, to properly control the execution of observations, and to update the Science Programs as their observations execute. Alarms and health also flow up from the principal systems through the OCS to the observers and operator screens.

The principal system alarm/status system must span the multi-system GCS environment –the DHS and OCS are Unix-based and the TCS and ICS(s) are VxWorks/EPICS-based.

The GCS status/alarm functionality is based on that provided by the EPICS database. To provide non-EPICS systems with the alarm/status capability, the OCS will provide an EPICS system dedicated to status and alarms (EPICS-based principal systems can also use it if they choose). The **Current System State** in Figure 10 - 16 is implemented as one or more EPICS databases in the OCS Status/Alarm Database (SAD). The implementation of Figure 10 - 16 is shown in Figure 10 - 17.

FIGURE 10 - 17 Distribution of OCS Functions by Hardware System



The entire OCS Observing Database is not based on an EPICS database—only the part of the ODB that provides the current system state and alarm/status monitoring. The separation of commands and the status/alarm system in the GCS design has made this possible.

There is no requirement for a host-level Channel Access server. Non-EPICS principal systems are clients of the OCS SAD and can put values into their status items. EPICS systems may also decide to centralize their status items in the SAD, but this is not a requirement at this time.

The Functional Specification for Release Four of Channel Access [13] indicates that the functionality of the OCS SAD can eventually move to from a VxWorks system to a UNIX host. This move is a long-term goal since it would reduce hardware costs and cause no impact or change to the behavior or implementation of the status/alarm system.

10.2.6.6 THE IMPLEMENTATION OF STATUS IN THE OCS/EPICS SAD

Records are used to represent status items in an EPICS database. A status device may have many status items in its EPICS database. EPICS status item records contain a number of fields that can be written into by their owner and monitored for change by status client applications.

The behavior of status as specified in “Alarms, Errors, and Status Reporting” in Chapter 5 is provided by the core EPICS system. Status clients will read and write status item fields using Channel Access. Status clients can monitor the status item field values using Channel Access monitors.

10.2.6.7 THE IMPLEMENTATION OF ALARMS IN THE OCS/EPICS SAD

The alarm functionality provided by EPICS provides the functionality of alarms as specified in “Alarms, Errors, and Status Reporting” in Chapter 5. The EPICS alarm system consists of IOC support and the user/operator alarm handler application (covered in a later section). All of the standard EPICS record types contain alarm related fields.

The determination of which alarm severity is appropriate in a situation should be based on the usage of alarms as specified in the Gemini SDD, not the EPICS documentation. Status clients use the Channel Access event facility to monitor the alarm fields of a status item. Alarms are delivered whenever the status device changes the alarm severity/status fields.

10.2.6.8 THE IMPLEMENTATION OF HEALTH IN THE OCS/EPICS SAD

Health is a predefined alarm status item indicating the general operational state of a status device. The determination of a systems health must be a function of its set of status items. There are two ways a principal system can implement health in the OCS/EPICS SAD and both can be used within a single principal system.

Health Status Item. A health status item has alarm status that represents the operational health of the principal system or a subset of the principal system. It must follow the recursive nature of health and reflect the operational state of all the parts of the system it depends on.

Health as a Group. The EPICS alarm manager program [14] [15] implements the concept of alarm groups that will be propagated into any alarm manager program that might be used in the Gemini Control System. An alarm group presents an alarm for a user-determined set of related status items or other groups. Health can be implemented as a group alarm and the recursive nature of health can be implemented by building a tree of health groups. This approach maps well to the health concept but requires groups to develop alarm handler configuration files.

10.2.6.9 THE REPRESENTATION OF STATUS IN EPICS

A standard EPICS Status Item Record (SIR) that must be used by non-EPICS systems and should be used by EPICS principal systems to provide status in the GCS. FOr more information on the details of the SIR and the alarm/status/health interface, see [25].

10.2.7 Fine Grain Control in the OCS

The activities of the OCS as defined up to this point are based upon configurations that completely describe the principal systems and the desired science activities during planned observing. The Sequencer Executor runs a recipe that controls the other principal systems with high-level Sequence Commands.

This level of control is not enough in all cases. Sometimes interactive activities are necessary that are of a finer grain than the high-level Sequencer Commands. For instance, it must be possible during verification sequencer states to allow the observer at the telescope to check and adjust the global state. There are a number of astronomer activities which might take place including:

Target Field Verification. The field may not be visible or the observer may have less than ideal coordinates. He may wish to offset to a nearby known position, or he may want to “peak up” on the position before beginning the exposure.

Quality Control. The observer may wish to optimize the configuration by adjusting some parameters in the configuration based on a sample exposure and some Quick-Look data analysis.

Focusing. The observer may feel that some attributes of the telescope that are outside of his configuration might need to be adjusted. For instance, the telescope might need to be focused. A focus routine might be needed to pick the correct telescope focus.

Console Control. The interactive control consoles also use fine-grained control to modify the hardware in the telescope and instruments. The consoles must issue system changes at any time, not just during a verification time.

Observe Recipe Sequences. For complex scenarios such as chopping and nodding or mosaics, the Sequence Recipe modifies the baseline configuration during execution. The modifications are affected by fine-grain commands such as telescope offsets.

In summary, there is a need for fine grain control at *all* times in the OCS. The consoles and PV-Wave Quick-Look tools require this functionality to be part of the OCS.

10.2.7.1 COMMAND USAGE IN THE OCS

The OCS provides a scripting language to provide fine grain control of the principal software systems that can be used in the applications just listed. The scripting language will be TCL-based—all the features of TCL are present with additional commands provided to allow fine grained control. There must also be a programming language interface to allow applications access to the control functionality.

This section does not define a complete command set for the low-level system control or a programmer API. It does show how an approach to commands within the OCS design should work. It is the job of the OCS Work Package group to refine the high-level command interface. The low-level implementation of commands is the subject of ICD1 [24]. The following are features that must be associated with the high-level command functionality.

- Commands will be entities in the OCS. This means that as long as a command is outstanding, the OCS will have a knowledge of it. A *handle* will be returned from every command that can be used by other status commands.
- All commands (just like Sequence Commands) will have command completion information: the command busy attribute, the command error attribute, and the command error string attribute.
- Users will be able to register callback functions that will be executed by the OCS when the status of a command changes.

- Supporting fine grain commands should not require an entirely new OCS system for control.

10.2.7.2 A COMMAND IMPLEMENTATION

In the GCS, all commands are implemented in the same way as sequence commands. A brief discussion of commands was given in “Sequence Commands” on page 10 - 20 and that information is used here. In the following, the destination of a command is called the *command target*.

Control commands are based on the attribute/values database model. Command language software commands build *mini-configuration* describing one or more changes that should take place in one or more command targets. A console panel might change multiple items in a target system, but a script line might only change one target item at a time. Figure 10 - 18 shows a mini-configuration that sends a single command to two command targets: the TCS primary airbag and the optical ccd instrument.

FIGURE 10 - 18 A Mini-configuration showing commands to two subsystems

c2321:occd:frontfilter	red
c2321:occd:frontfilter:busy	
c2321:occd:frontfilter:status	
c2321:occd:frontfilter:error	
c2321:occd:frontfilter:errorstring	
c2321:tcs:primary:airbag:altitudetrack	on
c2321:tcs:primary:airbag:altitudetrack:busy	
c2321:tcs:primary:airbag:altitudetrack:status	
c2321:tcs:primary:airbag:altitudetrack:error	
c2321:tcs:primary:airbag:altitudetrack:errorstring	

Mini-configuration sets optical CCD filter and turns on primary airbag altitude tracking in the tcs

Command mini-configurations exist in the Observing Database for their entire lives and have unique odometer values just like observations and science programs (in this case c2321). Also included in the mini-configuration are the error/status attributes. They are empty for a command but are filled in by the command once the command is accepted.

Mini-configurations are the same as Sequencer Commands and are passed to the principal systems using the OCS commands of Table 10 - 1 on page 10 - 22.

The final step to implementing commands is to provide the high-level commands themselves. OCS commands are not procedure oriented, but are based on the attributes of a system as with the Keck Keyword Layer[1] and the XView Programming Toolkit[2]. In this model, the number of commands is few. The attributes provide the interface and the arguments for the commands.

After the Keck Library, there are two commands: *modify* and *show*. The arguments are attributes and their values. Algorithm 10 - 4 shows an example of TCL-based commands that produce the mini-configuration of Figure 10 - 18.

ALGORITHM 10 - 4 A TCL example of commands

```
set control1 [modify occd*frontfilter red
                        tcs*primary:altitudetrack on]
waitfor $control1
puts stdout “The front filter is: [show occd*frontfilter]”
puts stdout “The primary altitude tracking is: \
    [show tcs*primar:altitudetracking]”
```

The modify command builds a mini-configuration in the Observing Database and sends a config(apply) along with the mini-configuration. The modify command returns a handle that is used with the built-in waitfor command that causes the execution of the script to wait until all the *busy* attributes in the mini-configuration are *done*. The values are then printed out with the show command. The asterisk in the attribute name is a method adopted from XLIB resources to match variable length name components.

The show command does not go directly to the command target. It merely gets the value of the attribute from the global state in the Observing Database. Principal systems must continually update the ODB with their current values.

If a show command is executed for an attribute that is not in the ODB, it will register an interest in the attribute with the principal system—it will grow the ODB global state.

1. Conrad, A. R. and Lupton, W. F. “The Keck Keyword Layer.” Astronomical Data Analysis Software and Systems II. ASP Conference Series, Vol. 52, 1993. Hanisch, Brissenden, Barnes. eds.
2. Heller, Dan, “XView Programming Manual.” O’Reilly & Associates. 1993.

10.2.7.3 OCS COMMAND SUMMARY

This completes the presentation of high-level scripting commands in the OCS. A design for commands was given that traces commands from the scripting level down to the systems that will be commanded and back up. The design for fine-grain commands is the same as the design for planned observing.

10.3 OCS SOFTWARE INTERFACES

The "High-Level System Interfaces" on page 9 - 1 and the ICDs [26]-[30] describe the software interfaces between the principal systems and subsystems. This section briefly discusses three of the OCS software interfaces that have been referenced several times in the OCS design chapters. The Interface Control Documents are the complete, in-depth definition of the GCS interfaces.

10.3.1 OCS To External Database

This external interface is used by the OCS to access Science Program data and other permanently stored data such as object catalogs. The OCS applications will access the external database through a request/reply client/server model. The OCS applications form and send Structured Query Language (SQL) requests and then await responses from the external database.

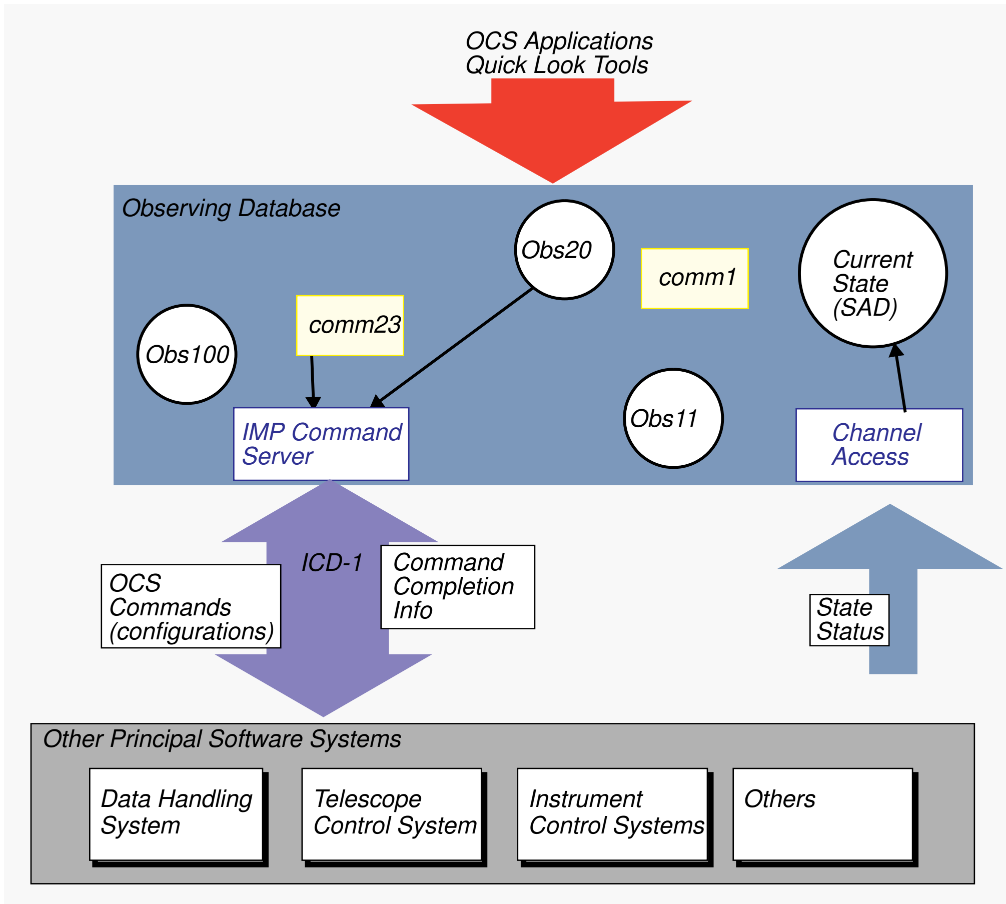
The OCS is required to process the Science Program data from the external database, create records in the Observing Database, and initialize those new records based on external database data.

10.3.2 OCS and Other Principal Software Systems

The OCS and principal software systems are all based on databases. The OCS communicates with the other systems by sending OCS commands and attribute configurations to the target principal systems. It is up to the systems themselves to decide how to execute the actions described by the configuration information. The dataflow between the OCS and the other principal systems is shown in Figure 10 - 19.

The communication interface for commands is based on the Interprocess Message Passing System (IMP) [4] to allow for a similar command interface between all principal systems. The status/alarm interface is based on the EPICS functionality and database.

FIGURE 10 - 19 **OCS Command and Status Communications (ICD-1 and ICD-2)**



10.3.3 OCS Internal Applications

The software interface that will be used between the various OCS components is To Be Determined and depends greatly on the implementation of the Observing Database. The IMP/SDS facility ([3] and [4]) used by the Anglo-Australian Observatory's DRAMA system [1], and used for communication between the Principal Systems in the Gemini Control System, is a good candidate. See ICD 1 [24] and ICD 3 [26] for a description of how IMP/SDS is used to transmit commands and bulk data between Gemini systems.

There are a few commercial interprocess communications software "standards" that either now or in the very near future will be coming available that could be used in the OCS. Candidates include: Sun's Tooltalk, an RPC-based system; and the ICE, Inter-Client Exchange Protocol, which is part of the X11R6 release. There are other more advanced approaches: Sun's Distributed Objects Everywhere based on Object Management Group's CORBA promises distributed application interoperability even when applications run on machines of different brands; and NeXT's OpenStep Portable Distributed Objects will also soon be available on Sun's Solaris operating system and supports CORBA.

10.4 VISIBLE USER INTERFACE DETAILS

The "High-Level Observatory Control System Concepts" chapter described the Visible User Interface presented by the Observatory Control System. The concepts used to support the different modes of observing were also presented. This section presents some additional VUI issues and details.

10.4.1 Application Dataflow

This section shows how the applications of the VUI fit into the detailed design presented in this chapter. In addition, some VUI functionality that could not be presented in the "High-Level" chapter is shown here.

10.4.1.1 THE OBSERVING TOOL

The Observing Tool is used off-line to construct Science Programs, and it is also used during the time an observation is executed to allow a user to monitor the progress of and interact with the executing observation. Figure 10 - 20 shows the configuration of the Gemini Control System while an observer is monitoring an observation.

FIGURE 10 - 20 **OCS when a user is monitoring an observation**

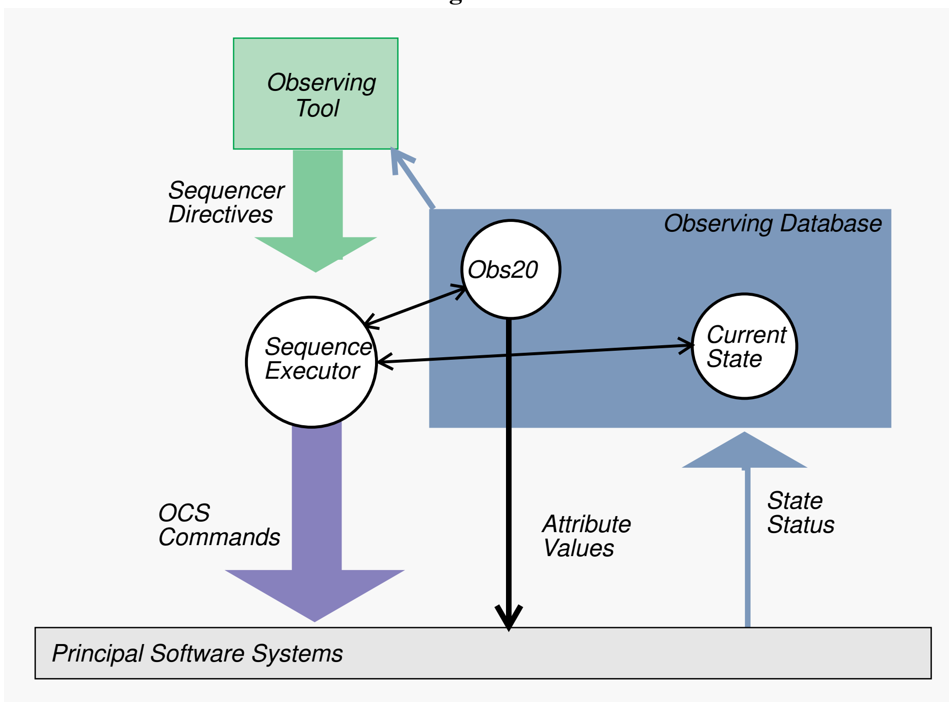
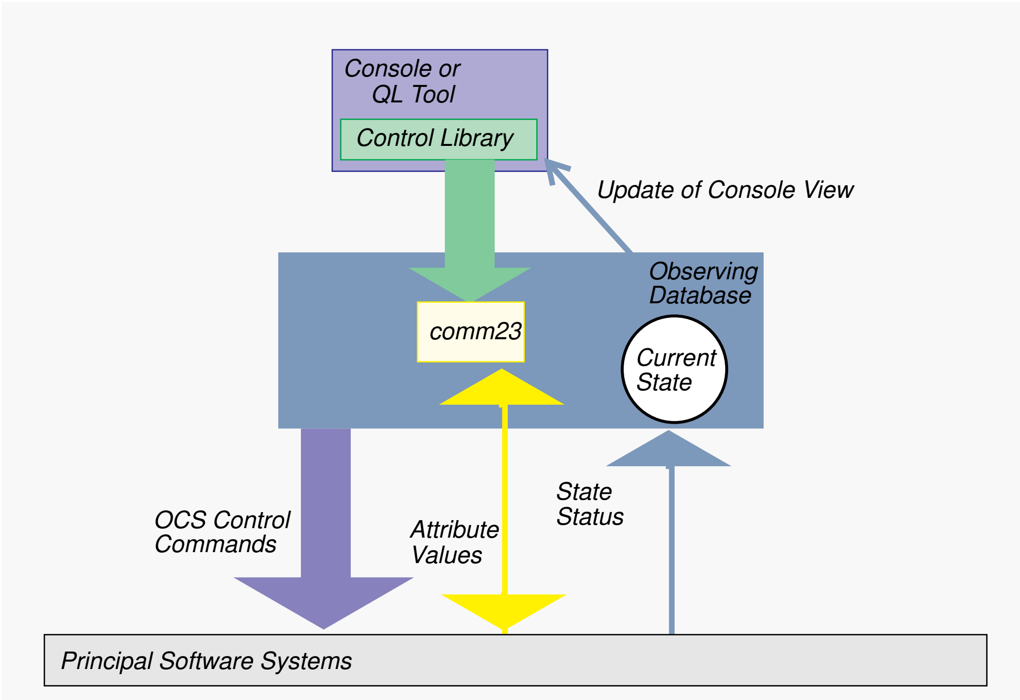


Figure 10 - 20 shows an Observing Tool using Sequencer Directives to control the Sequencer Executor that is executing its observation, Obs20. OCS Sequencer Commands flow from the SE to the principal software systems using command interface access that combines OCS commands and configuration attribute information. The principal systems act on the Sequencer Commands and update in the global current state in the ODB Status Alarm Database. The OT and the SE can be notified of changes in information in the ODB.

FIGURE 10 - 21 Console/Quick-Look Tool OCS dataflow



10.4.1.2 INTERACTIVE CONSOLES

Interactive consoles, Data Handling System Quick-Look Tools, and other tools require the command functionality described in "Fine Grain Control in the OCS" on page 10 - 49. Figure 10 - 21 shows the OCS system support for these interactive tools.

The figure shows a command mini-configuration in the Observing Database. This is a command configuration that was created by the console at the top of the figure. Each commanding application includes the *OCS Control Library* that provides support for creating and monitoring command configurations in the ODB. Issuing a command creates a configuration and results in an OCS Control Command to one or more of the principal software systems. The principal software systems then take whatever steps are necessary to bring the system into conformance with the command configuration.

The software systems continually update their status attributes. The ODB keeps the consoles and other tools updated through callbacks triggered on changes to the current global state.

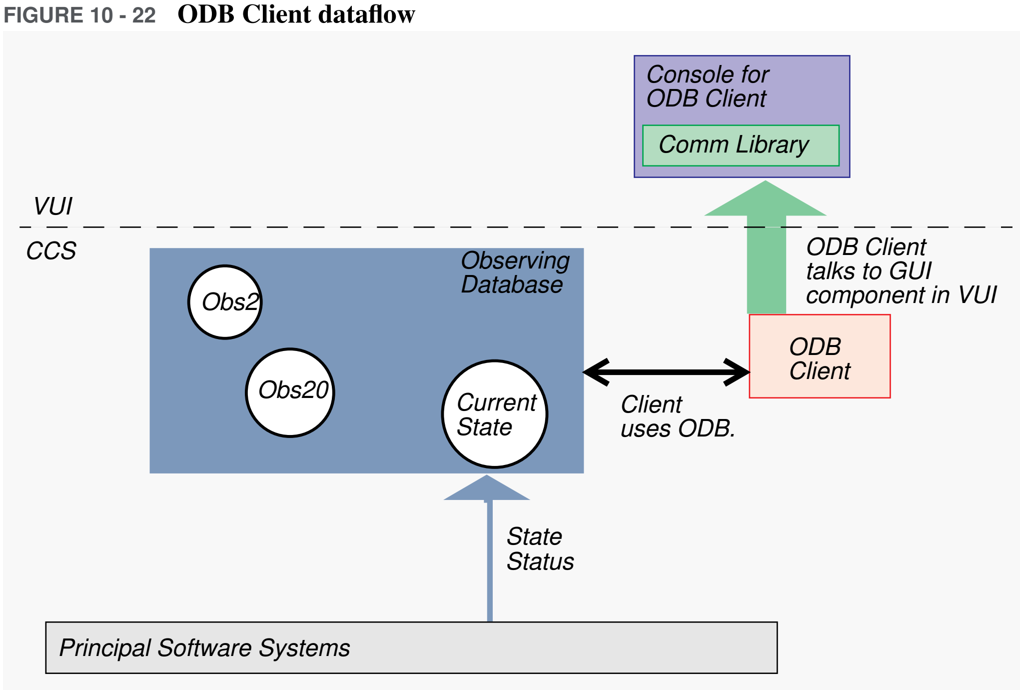
10.4.1.3 CLIENTS OF THE ODB

A client of the Observing Database is an application that resides in the Configurable Control System and uses the functionality of the Observing Database to provide OCS functionality. Consoles, while not in the Configurable Control System, behave like clients of the Observing Database.

Figure 10 - 22 shows how many clients of the ODB interact with the other parts of the OCS. First, notice that the clients of the ODB do *not* interact with the principal software systems. All the data they use to provide functionality comes through the information in the Observing Database (and its Status Alarm Database): observations, commands, and the global state of the GCS.

ODB clients will often have a GUI component that provides an operator interface for the ODB client and this implies communication between the CCS level ODB client and the GUI. One example is the error/alarm interface. (See “Alarms, Errors, and Status Reporting” on page 5 - 40.) Software interfaces between these OCS entities will be determined by the OCS Work Package group.

FIGURE 10 - 22 **ODB Client dataflow**



10.4.2 The Operator's Interface

Many of the operator tools are consoles as discussed in the "High-Level Observatory Control System Concepts" chapter. The design of the CCS requires that some additional functionality be added to the operator's VUI.

10.4.2.1 CONFIGURATION DEALER INTERACTIONS

The Configuration Dealer must be represented in the Visible Interface because operators must interact with the Configuration Dealer. The set of operations required at this time consist of the following:

- *Insert Observation to CD* - Observations must be inserted interactively into the Configuration Dealer by operators and by executing plans. Typically, selecting an observation adds the observation to the rear of the queue, but inserting observations into other positions could be necessary. Real-time decisions of an observer may require configurations be inserted at the front of the queue; for instance, an observer may find that an "auto-peak" configuration should be executed before the configuration that was to execute next.
- *Delete Observation from CD* - It may be necessary to remove an observation from the Configuration Dealer. Deleting an observation removes all references to the observation in the resource queues. It will not be possible to delete an executing observation.
- *Commands* - It is necessary to have operations like "pause after the next observation", "restart the Configuration Dealer." The commands might be special configurations (such as a configuration that pauses until the operator says okay) or commands that directly modify the actions of the Configuration Dealer itself.
- *View Options* - The operator may want to look at the contents of the Configuration Dealer according to appropriate criteria such as: show observations for instrument X, show observations requiring the telescope (or no telescope), or show executing observations. He/she needs to be able to examine the status of executing observations. Are they running or waiting for something?

10.4.2.2 RUN-TIME DYNAMIC HEADERS

Based on previous systems, the OCS designers feel it would be a step forward to allow observers to determine what information will be in the headers associated with their data files. Often, headers do not provide the correct information item an astronomer needs and his observing session and data quality suffers.

Therefore, the GCS design specifies that observers should be allowed to determine what is in their headers if they wish, and we are providing for this functionality at this stage of the OCS design.

The feature can be easily added since the OCS is based on a database system. Observers can view all the available attributes that a system will export and indicate through an OT interface which values they wish to have in their headers. Each public system attribute will provide enough information about itself to allow the Data Handling System to construct the data header from the configuration in the ODB.

This feature is simpler with an OCS Command. This command is a member of the OCS Status command group shown in Table 10 - 4. To support this class of functionality, a high-level OCS program requires the ability to browse the attributes of a principal system. This feature is very must like browsing a file system with the Unix commands *cd* and *ls*. A user can "cd" to a subsystem and "ls" the attributes at that level.

The **status(all)** command instructs a system to return a list of all its attribute names with enough detail such that the attribute can be uniquely identified by a user of the attribute. For example, an attribute in the CCD called *filter* is not enough if there are two filters. The CCD system must return *a.filter* and *b.filter*, not *filter*, *filter*.

OCS applications can use this information to show observers all the attributes a system has to offer. They can then decide which ones they wish to save with their data. This is an optional feature and default lists of attributes will always be available; astronomers will not be required to choose their header attributes.

TABLE 10 - 4 **OCS Status commands**

OCS Command command	Command Arguments	Definition
status(all)		This command instructs a principal system to return a list of all the public attributes that are exported to the OCS by the system. The format of the list is TBD but will be text-based.
status(one)	an attribute name unique in a system	This command instructs a principal system to return all the attributes that can be set as part of the description of another attribute.

Status items within the Status/Alarm Database provide information that allows a high-level OCS application to gather and present information to the user for any piece of information.

10.4.2.3 SEQUENCER TOOL INTERACTIONS WITH SCIENCE PROGRAMS

The operator uses a Sequencer Tool to allow him to compose plans, and to select future observations interactively. This tool has not been fully considered at this time and no prototype screens are available. The following are design comments based on the discussion in "The Sequencing System" on page 5 - 55.

When doing real-time selection of "next" observations (rather than planned observations), the group of *first configurations* from all the observations in the Observation Pool are the configurations that are of interest to the operator because one of them must be the "next" configuration. The operator's interface to the scheduling system allows him/her to quickly analyze the requirements of the first configurations in observations. The design of the science program document and its components helps to manage the complexity of the operator's job by limiting the number of possible "next" configurations.

The operator views the *first configurations* and selects one or more observations from the observation pool for execution. The chosen observations are entered into the Configuration Dealer (CD). The CD is a part of the Configurable Control System that manages the execution of observations.

It should be noted that the browsing design the Observing Tool works equally well for a Scheduling Tool since both programs serve very similar functions. Operators need to browse Science Programs, Observations, but they also need to make intelligent scheduling decisions. One approach is to augment the Observing Tool with an intelligent "agent"[1] that would constantly review the Science Programs selected for execution by the operator and recommend "next" observations based on scheduling policies and environmental data in the ODB.

It is important to note that the Scheduling Tool is loosely coupled to the rest of the OCS. Its one output is a "next" science configuration. Therefore, this tool is not a critical element and "next" observations could be picked by inspection from science programs in an Observing Tool.

10.5 OBSERVATORY CONTROL SYSTEM DEVELOPMENT PLAN

The Observatory Control System is a complicated system (like the other principal systems) and its development must be structured to guarantee deliverables at times that are suitable and in sync with the development of the other principal systems and with the telescopes themselves. The following shows a reasonable ordering for the development of the OCS software but does not assign times or other costs.

1. Laurel, Brenda. "Computers as Theater", Addison-Wesley Publishing Company, 1991, page 46.

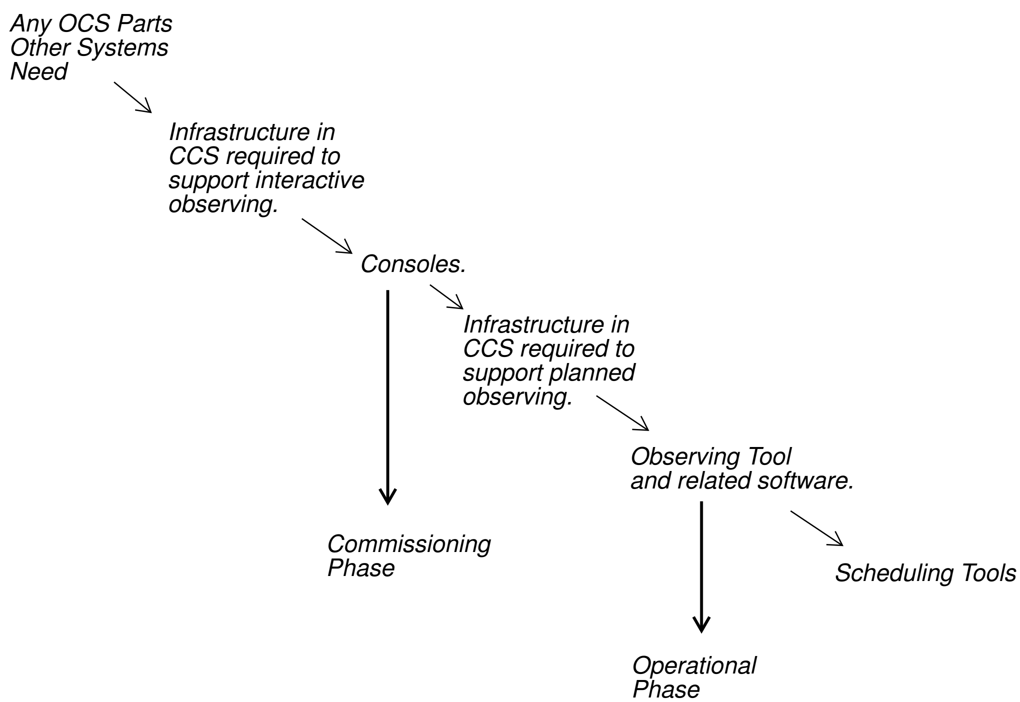
The development of the OCS should be phased to provide for at least the following:

- Any part of the OCS software that other major systems depend on or require should be developed early on.
- The development phasing should match the operational phases of the telescope project and the observing modes that will be used during those phases as specified in "Operational Phases" in Chapter 3.

As specified there, the observing modes used during the operational phases are

- engineering/acceptance - engineering interfaces used for checkout.
- commissioning - primarily interactive observing with consoles.
- operational/maintenance - interactive observing along with Observing Tool interactive observing and planned observing.

FIGURE 10 - 23 Sequential Development of the OCS



Based on these few issues the development of the OCS should take place such that critical items are delivered in the order shown in Figure 10 - 23. This figure assumes no parallel development, which would no doubt be present. The methodology used by the development group is also ignored. Naturally, the design team would need to consider the entire project at each phase. The figure is meant to show that there is a way to phase delivery of OCS software to take advantage of the phasing of the entire project.

10.6 SUMMARY OF DETAILS OF THE OCS

This concludes the Critical Design Review description of the Observatory Control System. The detailed design has focused on the execution of observations in the Gemini Control System and on showing that the design of the OCS and its software components can supply the kinds of applications and functionality required to do quality, efficient observing with the Gemini telescopes while supporting the required observing modes.

11 Details of the Data Handling System

11.1 Introduction

A high level description of the Data Handling System (DHS) has been given in “High Level Data Handling Concepts” in Chapter 6. This chapter will go into the detail of its operation and its interfaces to other principal systems. Common interfaces are described separately in “Interface Control Documents” (ICDs) which are referred to where appropriate. In particular, the reader should consult ICD/3 [26] and ICD/4 [27] alongside this chapter.

11.2 Changes Since Preliminary Design Review

The following changes have been made to this chapter since the Preliminary Design Review (PDR) on 20 April 1994:

- The DHS will consist of separate event-driven components. Its operations will not be tied so strictly to the sequencer states of the Observatory Control System (OCS).
- The sequencer commands listed on page 11 - 4 have been changed. VERIFY and GUIDE no longer have any meaning for the DHS, and INIT has a different meaning.
- Communication of commands between the OCS and DHS, and bulk data between the Instrument Control System (ICS) and DHS will *not* be by EPICS channel access. These systems will communicate using IMP messages encoded using SDS[1], as described in ICD/1 [24] and ICD/3 [26].

1. IMP and SDS are part of the DRAMA system of the Anglo-Australian Observatory.

- Status and logging information will be communicated by channel access, as described in ICD/2 [25] and ICD/4 [27].

See also "Changes Since Preliminary Design Review" on page 6 - 1.

11.3 DATA HANDLING SYSTEM OVERVIEW

The Data Handling System is not a discrete entity like the other systems. The data handling work package provides various components which may be used by the other systems, so various parts of the DHS may be found embedded in other systems (as may be seen in the data flow diagrams — processes 1.3, 2.3.3, 2.5 and 2.6 are all part of the DHS work package[1]).

In the following descriptions the term "Central DHS" will be used to refer to that part of the DHS which is contained within process 4 of the data flow diagrams (i.e. the Data Handling Command Processor, Logging System, Data Transport Manager, Data Storage Database manager, DHS Quick Look System and Data Reduction Agent). In the interface descriptions the interfaces between all the DHS components and their outside world are described.

11.3.1 External Interfaces

The Central DHS has the following external interfaces:

- Central DHS to Observatory Control System
- Central DHS to Instrument Control System
- Central DHS to Telescope Control System
- Central DHS to Observers
- Central DHS to External Databases
- Central DHS to External Data Reduction System
- Central DHS to Permanent Store and Archive

11.3.2 Other important interfaces

These other important interfaces are the responsibility of the DHS work package:

- Synchronous Data Reduction System to Observatory Control System.

1. It is unfortunate that the CASE tool we are using cannot display diagrams in color so that these processes can be distinguished more easily.

- Quick Look Server to Detector Array Controller, Instrument Control System or Observatory Control System.
- Quick Look Server to observers.
- Data Storage Server to Detector Array Controller or Instrument Control System.
- Observatory Control System to External Databases.

11.3.3 Major components

The Data Handling System consists of the following major components. The numbers refer to the process IDs on the data flow diagrams shown in "Decomposition/Dependency Descriptions" in Appendix A1. Several numbers indicate that several copies of this same process appear on the data flow diagrams, performing different functions:

- Quick Look Display Server (1.3.2, 2.3.3, 2.6, 4.3.2)
- Synchronous Data Reduction System (1.3)
- Data Storage Server (2.5)
- Data Handling Command Processor (4.5)
- Quick Look System (4.3)
- Data Storage Database Manager (4.6)
- Logging System (4.2)
- Data Reduction Agent (4.1)
- Data Transport Manager (4.4)

11.3.4 Internal Data Stores

The Data Handling System has the following major internal data stores:

- Data Store
- Data Handling Database
- Data Reduction Queue

11.3.5 Major Internal Interfaces

The Central DHS has the following major internal interfaces:

- Data Handling Command Processor to Data Reduction Agent.
- Data Handling Command Processor to Logging System.
- Data Handling Command Processor to Quick Look System.

- Data Handling Command Processor to Data Transport Manager.
- Data Handling Command Processor to Data Storage Database Manager.
- Data Reduction Agent, Data Storage Database Manager and Data Transport Manager to Data Handling Database.

11.4 EXTERNAL INTERFACES

11.4.1 Central DHS to Observatory Control System

Communication between the Central DHS and Observatory Control System consists of:

- DHS Control Commands that are passed from the OCS to the Central DHS.
- Data Handling Configurations that are returned from the Central DHS to the OCS.
- Logging data that are received from the OCS.

11.4.1.1 DHS CONTROL COMMANDS PASSED FROM OCS TO CENTRAL DHS

The Central DHS receives the standard sequence recipe commands from the Observatory Control System. An overview of the commands themselves is given in “Primary Internal Interfaces” on page 9 - 6. The protocol used for delivering the commands and determining command completion is described in ICD/1 [24]. The items in angle brackets are parameters that should be delivered with the command. The Central DHS should respond to the commands in the following ways:

- **TEST**

  The Central DHS should test its readiness to receive and handle data. For example, it should check that all the necessary disks are on-line and have sufficient free space for a typical night’s observing. If the previous night’s data has not been saved to the permanent store it should warn the observer.
- **INIT**

  The Central DHS should perform any necessary steps to initialize itself. This command is usually executed on system start-up.
- **RESET**

  The Central DHS should reset itself into the state it had after start-up initialization. (The INIT and RESET commands may mean the same thing for the DHS).
- **CHECK <config>**

Check that a data handling configuration is sensible and return an appropriate command completion status. For example, the DHS might check there is sufficient disk space to store a file of the intended size.

- **APPLY <config>**

Configure the Central DHS with the given parameters. The Central DHS configures its major components:

--The Data Reduction System stores any new data reduction recipes and rules.

--The Data Storage Manager allocates disk space.

--The Quick Look System (if any) configures its display.

--The Data Transport System modifies the criteria it uses to decide which files to copy to permanent storage.

- **VERIFY &ENDVERIFY**

These commands are ignored.

- **GUIDE &ENDGUIDE**

These commands are ignored.

- **OBSERVE <Unique ID> <filename>**

This command is ignored by the Central DHS. Although it signals the start of an observation, data will not become available until “ENDOBSERVE”.

- **ENDOBSERVE <Unique ID> <filename> <FITS header>**

This command signals the end of an exposure. The ICS will have created the given new file using the Data Storage Server. The Central DHS registers the name of the new file in the data handling database and flags it as “raw”, “unreduced” and “unsaved” (See “The Data Handling Database” on page 11 - 44). It writes the FITS header provided by the OCS into the file. If necessary, the file is sent to the DHS Quick Look System for display. The file is then entered into the data reduction queue at standard priority (See “The Data Reduction Queue” on page 11 - 46).

- **PAUSE, CONTINUE & STOP**

These commands are ignored.

- **ABORT**

This is an emergency stop command. The Central DHS will probably ignore this, as it does not move any mechanisms.

- **PARK**

The Central DHS should save its configuration to disk so it may be safely shut down. Any data buffers should be flushed and open files should be closed. The current state of the data reduction queue should be saved and the queue paused. If the queue is not empty a background job may be submitted to complete the remaining jobs. The command should not complete until all activity in the Central DHS has finished.

11.4.1.2 DATA HANDLING CONFIGURATIONS

The CHECK and APPLY commands will come with a data handling configuration, which contains a collection of new parameters for the Central DHS to store away. The configurations are sent using the “attributes and values” protocol described in ICD/1 [24].

The data handling configuration consists of the following pieces of information:

- The current directory into which files are being written.
- The expected size of the data array within the those files.
- Modifications to the default data reduction recipes and rules.
- Configuration information for the Quick Look System.
- Configuration information for the Data Transport System

11.4.1.3 STATUS INFORMATION RETURNED FROM CENTRAL DHS TO OCS

The OCS monitors the status of the Central DHS using the method described in ICD/2 [25]. The Central DHS should make the following status information available:

- A flag indicating the success or failure of the last command executed.
- An error message describing the status of the last command.

11.4.1.4 LOGGING DATA RECEIVED FROM OCS

History logging data are passed from the OCS to the Central DHS using the method described in ICD/4 [27]. The OCS will record a history of the observational procedures used, together with comments from the observer and/or operator.

11.4.2 Central DHS to Instrument Control System

There are two main information pathways from the ICS to the Central DHS:

- Bulk data from the ICS to the Central DHS.
- Logging data from the ICS to the Central DHS.

11.4.2.1 BULK DATA FROM THE ICS TO THE CENTRAL DHS

There is no direct data pathway from the ICS to the Central DHS. All bulk data are passed through the filing system; the ICS storing the data to disk with the aid of the Data Storage Server, and the Central DHS reading the data from disk. The OCS tells the Central DHS the name of the file created at the ENDOBSERVE stage of the sequence recipe.

Bulk data are transferred from the ICS to the Data Storage Server using the method described in ICD/3 [26]. The format of the files created by the Data Storage Server is the responsibility of the DHS work package. However, SDS structures are recommended, as illustrated in the appendix to ICD/3.

11.4.2.2 LOGGING DATA FROM THE ICS TO THE CENTRAL DHS

History logging data are passed from the ICS to the Central DHS using the method described in ICD/4 [27].

Engineering logging data are dumped to disk using the EPICS AR tool. The DHS provides a utility for converting these files into the standard format used by the DHS. See ICD/4 for details.

11.4.3 Central DHS to Telescope Control System

11.4.3.1 LOGGING DATA FROM THE TCS TO THE CENTRAL DHS

History logging data are passed from the TCS to the Central DHS using the method described in ICD/4 [27].

Engineering logging data are dumped to disk using the EPICS AR tool. The DHS provides a utility for converting these files into the standard format used by the DHS. See ICD/4 for details.

11.4.4 Central DHS to Observers

An observer is able to monitor the progress of the data acquisition and reduction by means of the following displays:

- The Observing Tool.
- A data handling console screen.
- A data reduction console screen.
- Graphical displays from the data reduction system.

11.4.4.1 THE OBSERVING TOOL

The observer is able to monitor the Data Handling System through the Observing Tool, and is provided with the following information:

- The unique ID for the present observation.
- The name of the file that is to contain the raw data for the present observation.
- The header information for the present observation.

11.4.4.2 DATA HANDLING CONSOLE

This has already been described in "Data Handling Console" on page 6 - 13.

11.4.4.3 DATA REDUCTION CONSOLE

This has already been described in "Data Reduction Console" on page 6 - 22.

11.4.5 Central DHS to External Databases

The exact format of this interface is to be decided as part of the Data Handling work package. The baseline for this interface is the ANSI standard Structured Query Language, SQL.

The same is true for the interface between the Observatory Control System and the External Databases.

11.4.6 Central DHS to External Data Reduction System

The exact make-up of the external data reduction system (ADAM, IRAF, FIGARO etc.…) is beyond the control of the Gemini project, and is regarded as being outside the Gemini System, which is why it is shown as a terminator on the data flow diagrams. Only the interface between the Gemini System and the External Data Reduction System can be specified here. The details of this interface are to be worked out as part of the Data Handling work package (in consultation with the data reduction groups).

The External Data Reduction System receives the following from the DHS:

- Commands. Each command corresponds to one particular function of the external data reduction system.
- A different set of parameters for each command.
- A file of raw data in FITS format.

The External Data Reduction System should provide:

- A status to indicate success or failure of each command, together with a textual message if that is appropriate. The textual message may consist of more than one line of information.
- A file of reduced data in the FITS format.

The Data Handling System needs to convert data between its own internal format and FITS format. It also needs to convert its own internal sequence recipes into sequences of commands for the External Data Reduction System.

The External Data Reduction System provides a service for the Data Reduction Agent, and it should provide sufficient commands and facilities to allow all the Data Reduction Agent's tasks to be completed. It is assumed that any External Data Reduction System used with the Gemini Control System is capable of at least the following operations:

- Arithmetic on N-dimensional arrays of data.
- The ability to identify and ignore bad elements within an array during a data reduction operation and pass this bad element information to the result.
- The ability to classify data elements outside a given threshold value as "bad elements".
- The extraction of slices from N-dimensional arrays of data (e.g. the extraction of a 2-D image from a 3-D cube, or a 1-D spectrum or profile from a 2-D image).
- The conversion of an N-dimensional array of data to an N+1-dimensional array by duplicating data along one of the axes.
- The calculation of statistics (such as the minimum, maximum, mean and standard deviation) within a given sub-area of an N-dimensional array.
- The ability to smooth an N-dimensional array using a variety of filters.
- The ability to flip an N-dimensional array along one of the axes.
- The generation of artificial data, for example

  --The generation of a theoretical black-body spectrum

  --The generation of a 2-D image with a flat or sloping background
- The ability to locate the brightest N objects within a 2-D image and calculate their centroids. (The minimum requirement is the ability to determine the centroid of the brightest object in the field).
- The ability to locate the brightest N emission lines within a 1-D calibration arc spectrum, and the ability to generate a "pixel number to wavelength" conversion table when given the wavelength of those lines.

In addition, the following operations would be desirable:

- The propagation of error information contained in the data in the way described by Bevington (1969) [6].
- The ability to display the data using a variety of graphical methods. (If an External Data Reduction System is not capable of this PV-Wave/IDL is used).
- The ability to place a cursor on any of the above displays and obtain information about the display.
- The ability to rotate an N-dimensional array through any angle.
- The ability to resample an N-dimensional array in the following ways

  --To shift the data along one of the axes by a non-integral number of pixels

  --To linearize the units along one of the axes, converting the data to linear wavelength units for example

  --To convert one axis to logarithmic units, prior to cross-correlating.

  --To correct 2-D data for any "S" or barrel-shaped distortion.
- The ability to fit a model to observed data. For example

  --To fit a Gaussian profile to a defined spectral line

  --To find the black body curve that best-fits a spectrum

  --To fit a Gaussian profile to a given object in a 2-D image
- The ability to merge (possibly overlapping) N-dimensional arrays in the following ways

  --To join two 1-D spectra end to end.

  --To combine several 2-D images into a mosaic

  --To combine several 2-D spectra into a 3-D data cube
- The ability to cross-correlate two spectra together and obtain a shift. This is useful for comparing arc spectra and for determining radial velocities.

High-level functions in the external data reduction system should be used wherever possible (especially if someone has written a function especially for the particular type of instrument being used).

The DHS group are free to choose which particular external data reduction system is most suitable for use by their data reduction agent. It is not necessary for an observer to be able to choose between different external data reduction systems for the purpose of on-line reduction (although observers can use whatever available system they like for off-line reduction). The DHS should be designed so the observer does not need to know which underlying external data reduction system is being used.

11.4.7 Central DHS to Permanent Store and Archive

Permanent Store and Archive refers to three different data storage locations:

- The permanent storage devices (probably optical disks) at the Gemini base facility.
- The portable storage media (probably DAT tapes) given to each observer.
- The Gemini archive.

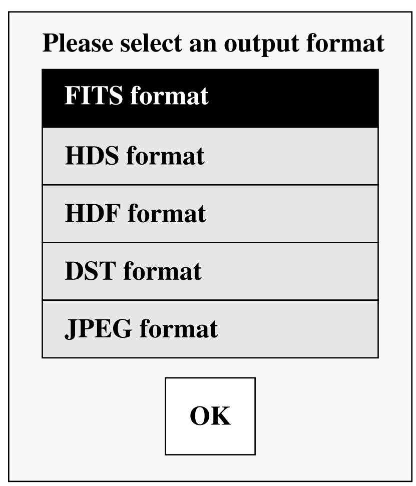
11.4.7.1 PERMANENT STORAGE AT THE BASE FACILITY

Data will be stored at the base facility in the same format it is transported to the Gemini archive (see below).

11.4.7.2 DATA GIVEN TO EACH OBSERVER

Data will be stored by default on the transportable media given to the observer in FITS format. However, if the data reduction groups provide utilities for converting FITS format data to their native format, then the observer can be given a choice of export format. An example dialogue box is show in Figure 11 - 1 on page 11 - 11. The default output format will always be FITS.

FIGURE 11 - 1 **An example dialogue box giving the observer a choice of output format**



11.4.7.3 THE GEMINI ARCHIVE

The Gemini archive is recorded in FITS format [46]. The baseline for this system is a duplicate of that currently in place between CFHT and Canadian Astronomical Data Center (CADC). In this model the FITS data are written to both an optical disk, in CADC disk format, and to a DAT or Exabyte tape, in FITS format, as the data are available. The FITS headers are copied to a separate file that is sent to CADC on a daily basis. These FITS headers are used to create database entries at CADC for the data files. Whenever the optical disk is full it is removed and sent to CADC — the DAT/Exabyte tape is removed and stored against loss of the optical disk during transport. Once the optical disk data are successfully incorporated into the CADC data base the DAT/Exabyte tapes are recycled or destroyed.

It is part of the data handling work package to provide all the software and hardware needed to implement the baseline system.

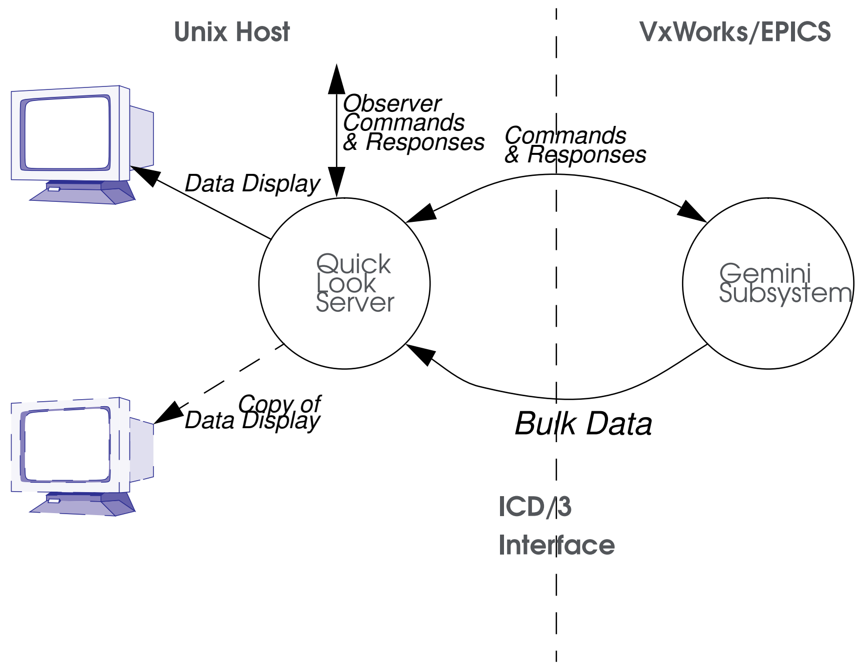
11.5 MAJOR COMPONENTS

The following descriptions can be compared to the data flow diagrams presented in the appendix.

11.5.1 Quick Look Display Server (1.3.2, 2.3.3, 2.6, 4.3.2)

The quick look display server is a process that may be used by other systems to display their data (see “Quick look display server” on page 6 - 6). Its interaction with the outside world is summarized in Figure 11 - 2 on page 11 - 13.

FIGURE 11 - 2 **The Interaction of a Quick Look Server with its Outside World**



The server accepts data in the format described in ICD/3 [26] and obeys the following commands:

- **QLCONFIG <qlid> <config>**

  Configure a display server. The configuration will contain the title, labels and axis limits for the display. The data structures in ICD/3 also allow for an optional “COMPASS” item which, if present, can be used to indicate the compass direction corresponding to the axes of the data.

- **QLLOAD <qlid> <operation> <scidata>**

  Download a calibration frame together with the mathematical operation to be used when applying the calibration frame.

- **QLDISPLAY <qlid> <quicklook>**

  Refresh display and display some new data.

- **QLUPDATE <qlid> <operation> <quicklook>**

  Update the current display with a data frame, combining it using the given mathematical operation.

- **QLCLOSE <qlid>**

Close a display server.

The detailed behavior of the Quick Look Display Server is response to these commands, and the structure of all the arguments, is given in ICD/3.

11.5.2 Synchronous Data Reduction System (1.3)

The Synchronous Data Reduction System (1.3) consists of the following processes:

- Synchronous Data Reduction Processor (1.3.1)
- Quick Look Display Server (1.3.2)

The Quick Look Display Server accepts data for display from the Synchronous Data Reduction Processor in the usual way, using the protocol described in ICD/3 [26]. The Synchronous Data Reduction Processor accepts commands from the OCS's Configurable Control System and returns information on the "best configuration" (for a focus run the "best configuration" would be the best focus — See "Synchronous data reduction server" on page 6 - 10 for details).

The Synchronous Data Reduction System is baselined as a PVWave/IDL system which executes IDL scripts (or ADAM/IRAF scripts if appropriate) supplied by the instrument developers. It accepts commands from the OCS using the protocol described in ICD/1 [24] and makes available status information using the protocol described in ICD/2 [25]. The following commands will be accepted:

- **SYCONFIG <syid> <config>**

  Configure the synchronous data reduction system. The configuration will contain the name of the IDL script to be applied to each data frame. It will also contain information on the type of fit to be performed to the data.
- **SYRESET <syid>**

  Clear all internal buffers.
- **SYUPDATE <syid> <filename> <given-values>**

  Update the internal buffers using the given file and calibration values. **<filename>** is the name of a file to be processed using the IDL script selected above. **<given-values>** is a list of one or more values describing the system configuration (e.g. a focus value) corresponding to the data in the file.

The synchronous data reduction processor will apply its IDL script to the file, displaying the result, and store the result in a table, together with the values supplied with the command. As an example, the focus run described in “Synchronous data reduction server” on page 6 - 10 should result in a table something like this:

File name	Line width	Focus setting
N123-456-789-01.sdf	5.782	1.2
N123-456-789-02.sdf	4.626	1.4
N123-456-789-03.sdf	3.745	1.6
N123-456-789-04.sdf	3.520	1.8
N123-456-789-05.sdf	3.489	2.0
N123-456-789-06.sdf	4.418	2.2

- **SYFIT <syid> <minmax>**

  Fit the value generated by the IDL script (line width in the above example) against the value or values supplied with each file (focus setting in the above example). The command will return **<return-values>** which will contain an estimate of the **<given-values>** corresponding to the maximum or the minimum of the property measured. In the above example the focus setting corresponding to the minimum line width will be returned.
- **SYCLOSE <syid>**

  Close the synchronous data reduction system.

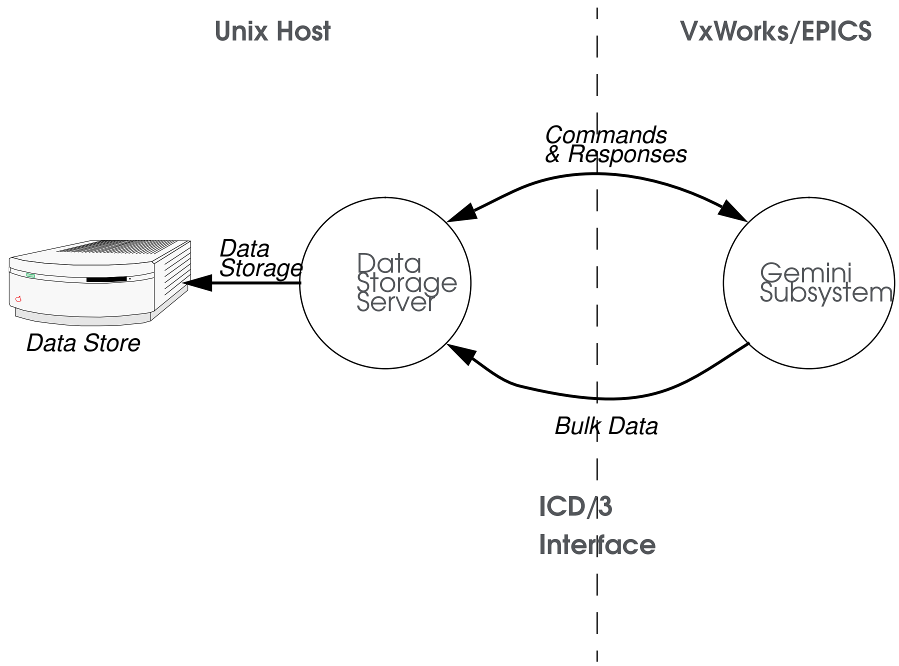
Typical uses of the Synchronous Data Reduction System are:

- Taking several observations with a spectrograph at different focus settings and determining the focus which minimizes the line width (as described previously).
- Taking several pairs of observations with a spectrograph at different focus settings, each member of a pair exposed through a different Hartmann shutter, and determining the focus which minimizes the line shift between pairs.
- Taking several observations with a spectrograph at different slit orientations and determining which orientation minimizes the shift between the same spectral line measured at the bottom and top of the slit.
- Taking several images with a camera at different focus settings and determining the focus which minimizes the Point Spread Function of an object in the frame. The reader is referred to [7] for a new algorithm for measuring the Point Spread Function of an image.
- Taking pairs of images using different Hartmann shutters at different focus settings and determining which focus setting minimizes the shift between the two sets of images.
- *etc.…*

11.5.3 Data Storage Server (2.5)

The quick look display server is a process that may be used by other systems to store their data to disk (see "Data storage server" on page 6 - 11). Its interaction with the outside world is summarized in Figure 11 - 3 on page 11 - 16.

FIGURE 11 - 3 **The Interaction of a Quick Look Server with its Outside World**



The server accepts data in the format described in ICD/3 [26] and obeys the following commands:

- **DSCONFIG <dsid>**

  Configure a Data Storage Server.

- **DSSTORE <dsid> <scidata>**

  Store some data.

- **DSUPDATE <dsid> <operation> <scidata>**

  Update the internal buffer with a data frame, combining it using the given mathematical operation.

- **DSFLUSH <dsid>**

Store the contents of the internal buffer.

- **DSCLOSE <dsid>**

Close a Data Storage Server.

The detailed behavior of the Data Storage Server is response to these commands, and the structure of all the arguments, is given in ICD/3.

11.5.4 Data Handling Command Processor (4.5)

The data handling command processor interprets the sequence recipe commands received from the OCS (described on page 11 - 4) and issues appropriate commands to the other DHS subsystems.

Whenever an observation is completed (and the ENDOBSERVE command is received) commands are entered into the *Data Reduction Queue* to tell the Data Reduction System to reduce the data. Instructions to change the data reduction configuration (for example changes to the data reduction recipes, or to the procedures used to find suitable calibration frames) are also entered as commands in this queue. Communicating with the data reduction system via a queue in this way allows the Data Handling Command Processor to get on with handing the next set of data. It does not need to wait for the data reduction to complete. In fact, the Data handling System can work quite happily in the absence of any data reduction if that becomes necessary.

The Data Storage Command Processor controls the Quick Look Display Manager and Data Transport Manager, telling them both when data are available.

11.5.5 Quick Look System (4.3)

The Quick Look System (4.3) consists of the following processes:

- Quick Look Client (4.3.1)
- Quick Look Server (4.3.2)

The Quick Look Server behaves exactly as described in "Quick Look Display Server (1.3.2, 2.3.3, 2.6, 4.3.2)" on page 11 - 12 and receives data from the client using the protocol of ICD/3 [26]. At the end of each observation (when the Data Handling Command Processor receives an ENDOBSERVE command from the OCS) the Quick Look Client can be instructed to read the data file from disk and submit it to the Quick Look Server for display.

The Quick Look System exists to cater for instruments which cannot supply a quick look display of their own. It can also be used to display the contents of any file at the request of the observer from the DHS console.

The commands exchanged between the Data Handling Command Processor and the Quick Look System are a matter for the DHS work package.

The quick look system is baselined on PV-Wave and is programmable via IDL.

11.5.6 Data Storage Database Manager (4.6)

This process responds to *Stored Data Info* commands from the Data Handling Command Processor. A command is sent each time the Data Handling Command Processor receives an ENDOBSERVE command from the OCS. *Stored Data Info* contains the name and the unique ID of the file which the ICS will have created during the observation, together with the FITS header. The Data Storage Database Manager registers the name of the file in the Data Handling Database, together with any other important information about the file (See "The Data Handling Database" on page 11 - 44 for details). It also combines the file with its FITS header information.

11.5.7 Logging System (4.2)

The logging system is a collection of programs and utilities for compiling various logs describing the use made of the system. See "Logging and problem tracking utilities" on page 6 - 15 for a description of the kinds of logging information collected. The logging system consists of the following processes:

- Logging System Manager (4.2.1)
- EPICS Archiver (4.2.2)
- History Logger (4.2.3)
- Make Observing Log (4.2.4)
- Combine Logging Information (4.2.5)

The exact design of the Logging System is a matter for the DHS work package, but the following description may be used as a baseline. See also ICD/4 [27].

11.5.7.1 LOGGING SYSTEM MANAGER (4.2.1)

This process accepts commands from the Data Handling Command Processor and manages the operation of the other processes.

11.5.7.2 EPICS ARCHIVER (4.2.2)

This process is the EPICS archiver tool "AR", described in [17]. In practise an EPICS archiver server "AR_cmd" will collect information from the EPICS boxes while observing. See ICD/4 for details.

11.5.7.3 HISTORY LOGGER (4.2.3)

This process is the "Logging Server" which other processes on the system may use to record messages in the history log. It can receive *Logging Requests & Data* from anywhere on the system and writes these into a *System History Record*. See ICD/4 for details.

11.5.7.4 MAKE OBSERVING LOG (4.2.4)

This process represents the utility used at the end of an observing session to collect together the information from the FITS headers of all the files created on the system (*File Headers*) and compile this into an *Observing Log*. See ICD/4 for details.

11.5.7.5 COMBINE LOGGING INFORMATION (4.2.5)

This process combines together the various kinds of logging information generated during an observing session (history logs, EPICS archiver logs and observing log) and writes them into a standard form suitable for saving in the Gemini archive.

This process needs to include a utility for converting the contents of the archiver file (*Saved EPICS Logging Data*), created by the EPICS archiver, into a standard form.

It is recommended that a commercial database (such as SYBASE) be used to store standard logging information.

11.5.8 Data Reduction Agent (4.1)

The Data Reduction Agent behaves as if it were a data reduction system. It executes the instructions entered into its queue, takes *Raw Science Data* from the *Data Store*, reduces it, and returns the *Reduced Science Data* back to the *Data Store*. The exact design of the Data Reduction Agent is a matter for the DHS work package, but the following description may be used as a baseline. The agent consists of the following processes:

- Data Reduction Command Processor (4.1.1)

  This processes the commands received from the data reduction queue and converts them into commands and parameters for the External Data Reduction System. It also manages the data reduction system's interaction with the Data Handling Database.

- Data Display Agent (4.1.2)

This uses the data display capabilities of the External Data Reduction System to display the reduced data at various steps during the data reduction procedure. This process may be substituted with a copy of the Quick Look Display System if the External Data Reduction System does not provide any display capabilities that are not already covered in the Quick Look Display.

- Data Converter (4.1.3)

  This process converts data from the internal format used by the Data Handling System to FITS format. The External Data Reduction System must be capable of reading and writing files in FITS format (with the aid of conversion utilities provided by the External Data Reduction System if necessary).

The observer may interact with the data reduction agent using the data reduction console, altering the appearance of the displays, modifying or adding commands to the data reduction queue or altering the recipes and rules used for data reduction (as described in “Data Reduction Console” on page 6 - 22). Modifications to the data reduction procedures (such as changes to the recipes and rules) are communicated via commands in the data reduction queue (to avoid any contention between the commands generated by the observer and Data Storage System). The observer may alter the position of these commands in the queue to ensure they are executed at the right time. Commands to alter the appearance of the data reduction displays take immediate effect.

There has been some discussion as to whether the Data Reduction Agent should contain one or two data reduction pipelines. If there are two pipelines one could deal with single frame data reduction while the second could deal with situations that require the assembly of a single frame from several previously reduced frame (such as the assembly of a mosaic from several IMAGE observations or the assembly of a MASTER SKY from several SKY observations). The approach favored by the Gemini project office is to begin with a single pipeline but allow the possibility of an upgrade to a second pipeline. The detailed design of the Data Reduction Agent is left to the DHS work package.

The Data Reduction Agent must be capable of running in background without the presence of the rest of the system. At the end of each night it may need to run independently to complete the remaining jobs in the data reduction queue after the rest of the system has been shut down.

11.5.8.1 DIRECT DATA REDUCTION SYSTEM COMMANDS

The data reduction agent accepts the following commands directly from the data handling command processor:

- **DRINIT <queueid>**

Define location of data reduction queue.

- **DRSTART**

  Start monitoring data reduction queue.

- **DRSTOP**

  Stop monitoring data reduction queue and shut down.

- **DTSTATUS**

  Tell me your current status

The data reduction agent responds to these commands immediately with a status indicating success or failure.

11.5.8.2 QUEUED DATA REDUCTION SYSTEM COMMANDS

Once operational, the data reduction system is driven by commands received on the data reduction queue. (See “The Data Reduction Queue” on page 11 - 46 for a description of the format of this queue). The following commands can be received through the data reduction queue:

- **DRCONFIG <config>**

  Change the data reduction configuration to that desired. This involves making enabling or disabling any steps in the recipes and rules, configuring the graphical displays and setting the various verification flags that control how frequently the data reduction system prompts the observer.

- **PROCESS <filename>**

  Reduce the data contained in **<filename>**. The system will go through the following steps:

1. Read the header of the data file to be reduced, examine the observation type (BIAS, FLAT etc.…) and determine which recipe should be used for its data reduction.
2. Check there is sufficient disk space to reduce the file. The system just checks that a reasonable minimum amount of disk space is available based on the amount of disk space used the last time an observation of this type was reduced.
3. If a “display raw data” flag is set, display the raw data to the observer.
4. If a verification flag is set, then verify with the observer that the data reduction recipe is ok. Allow the observer to modify that recipe if necessary.
5. Reduce the data frame according to its recipe, displaying any intermediate stages to the observer if requested in the recipe. See “Data Reduction Recipes” on page 11 - 25 for details of the data reduction recipes.
6. Display the reduced data to the observer.
7. Register the name of the newly created reduced file in the Data Handling Database.

8. Remove any intermediate data files if they are flagged as “auto-remove”. This step enables temporary intermediate frames to be removed. For example, a bias frame may be made of the median of several intermediate frames taken in quick succession, and only the end product is useful. See “Data Reduction Principles” on page 11 - 24.

   The data reduction agent *does not* acknowledge these commands, so the data handling command processor can continue with data acquisition at its own pace. Each time a queued command completes the success or failure of the command is reported directly to the observer on the data reduction console, and a flag is set against that item in the data reduction queue.

   Failed items in the queue are moved to a “holding area” where they wait for something to happen to them. There are two possible things that might happen:

- The observer can manually remove the failed items from the queue, or can resubmit them at a suitable time.
- The Data Reduction Agent can be programmed to retry these filed items automatically. It can resubmit failed items

  --at regular intervals (say every 10 minutes)

  --after every new observation

  By configuring the automatic retry of failed items an entry can be made to wait until all the suitable calibrations required to reduce it have been acquired. The observer must be able to switch this automatic retry off, to prevent unwanted jobs hogging the queue when the observer is trying to do something else.

11.5.9 Data Transport Manager (4.4)

The data transport manager is responsible for saving the data created during an observing session to permanent store, giving each observer a copy of the data belonging to them, and transporting the data to the Gemini archive, as described in “Data Transport and Archiving System” on page 6 - 24. The following description is a suggestion. The detailed design of the Data Transport Manager needs to be worked out as part of the DHS work package.

11.5.9.1 SUGGESTED DESIGN FOR DATA TRANSPORT MANAGER

The Data Transport Manager runs continuously during an observing session. It monitors the Data Handling Database and recognizes files which are available for copying to the permanent store. The database may be monitored by a polling mechanism or by using RDBMS triggers to recognize when files become available. The DHS work package will decide how this is to be done. The files are copied silently as a background job and, in normal circumstances, the observer and operator need not interact with the Data Transport Manager.

The Data Transport Manager accepts the following commands from the Data Handling Command Processor:

- **DTSTART**

  Start monitoring the Data Handling Database.

- **DTCONFIG <config>**

  Alter the configuration of the Data Transport Manager. This might include the frequency with which it checks the database and the kind of files it should save and archive (e.g. whether it should save just the raw files or the reduced file as well[1]).

- **DTSAVE (<list of files>)**

  Save the given files to the permanent store. If no files are given, all available files are saved.

- **DTCOPY <observer> (<list of files>)**

  Save the given files belonging to the specified observer to tape. If no files are given, all available files belonging to that observer are saved.

- **DTARCHIVE (<list of files>)**

  Copy the given files to the Gemini archive. If no files are given, all available files are archived.

- **DTSTOP**

  Stop monitoring the Data Handling Database.

11.5.9.2 DATA TRANSPORT UTILITIES

In addition to the above process, utilities should be provided to enable the observer to:

- Display the contents of a FITS format tape.
- Copy files manually to and from a FITS tape.
- Display the contents of the permanent store.
- Display information from the Gemini archive.
- Obtain data from the Gemini archive.

These utilities will probably be separate programs that run outside the Gemini Control System.

1. See “Policy for Archiving Reduced Data” on page 6 - 25.

11.6 DATA REDUCTION PRINCIPLES

The data reduction system reduces each *raw science data* frame using a **recipe** which varies according to the type of that data. Some steps in the data reduction recipe may involve the use of a calibration frame, such as a bias or flat-field. Whenever a calibration frame is needed, the Data Handling Database is searched for a suitable one of a given type. The suitability of a particular calibration frame is governed by a set of **rules** associated with each type.

The type of data which an instrument is capable of producing is defined by the instrument builders. They also need to define the default recipes and rules to be used for reducing the data generated by that instrument. There may need to be a different set of recipes for each of the working modes of an instrument (e.g. no bias, dark or sky subtraction is needed when operating in CHOP mode). There should be a standard set of names for common data types, and these are suggested in the following sections.

Note that data reduction recipes are not fixed. They may change as research into the best way of processing the various kinds of data progresses. The DHS should allow for this and make them easy for the *programmer* to modify. The observer should normally configure recipes simply by turning various steps on and off or altering parameters associated with each recipe step.

11.6.1 Data Reduction Data Types

Here are some standard data types:

TABLE 11 - 1 **Standard Data Types**

Data type	Description
MASK	A map of "bad pixels" within an array which should be ignored.[a] A MASK is prepared in advance rather than being generated by the instrument.
BIAS	An very brief exposure made with the detector blanked off, used to record the electronic bias.
DARK	An exposure made with the instrument detector blanked off, with the same exposure time as the frame it is used to calibrate. This is used to remove any dark current, and is the same thing as the "blanked-off" frame referred to in "Long Slit Spectroscopy of M82 at 10 Microns" in [40].
FLAT	A long exposure of a calibration lamp used to measure the variation in sensitivity across a detector's surface.

TABLE 11 - 1 Standard Data Types

Data type	Description
SKY-FLAT	A flat-field made by observing a patch of blank sky, used to calibrate out any uneven illumination of the detector caused by vignetting through the optical path not included in a standard flat-field. This is sometimes referred to as a “cold sky flat”.
DOME-FLAT	The same as a sky flat, except that the underside of the dome is observed.
ARC	An observation of a calibration arc lamp, used for wavelength calibration.
SKY	An observation of a patch of blank sky, used to subtract the sky background from an object frame.
IMAGE	An exposure of an astronomical object made using an imaging instrument.
SPECTRUM	An exposure of an astronomical object made using a spectrograph.
SPECTRAL-STANDARD	An observation of a spectral standard, used for removing the effects of varying atmospheric transparency from a spectrum.
FLUX-STANDARD	An observation of a flux standard, used for calibrating a spectrum into absolute flux units.
PHOTO-STANDARD	An observation of a photometric standard, used for calibrating a photometric observation into standard units (be they magnitude or flux units).
VELOCITY-STANDARD	A radial velocity standard, normally cross-correlated with the spectrum of the object whose radial velocity is to be determined.

a. Note that the frames generated by a detector controller should automatically have the detector’s bad pixels imprinted in their quality array. Using a MASK as a calibration frame during data reduction gives the observer a chance to apply an additional mask to the data (e.g. to mask off unwanted areas).

There may be others, depending on the requirements of particular instruments.

11.6.2 Data Reduction Recipes

The recipes used when reducing calibration frames of a particular type are configurable from the Data Reduction console. It is possible for there to be more than one related observation of the same type, where the data reduction system is expected to combine these together. If this is the case the final observation is flagged as a “master”, which changes the recipe. Related observations are identifiable because each set of related observations has a unique “group number”. Single observations of a particular type are flagged “master”. In this case the recipe does not find any related observations but it executes any optional steps reserved for the last observation in a group (such as the division of an OBJECT by a spectral standard). The default recipes vary according to the instrument used, and the mode in which that instrument is used, but they are usually as follows:

TABLE 11 - 2 Suggested Data Reduction Recipes

Data type	**Typical Recipe**
MASK	None. A MASK does not need to be reduced.

BIAS	Apply MASK.
	If this is a “master” BIAS then
	    Find all the related BIAS observations and take the median.
	End if

DARK	Apply MASK.
	Subtract BIAS.
	If this is a “master” DARK then
	    Find all the related DARK observations and take the median.
	End if

FLAT	Apply MASK.
	Subtract BIAS.
	Linearize the data using a defined look-up table.
	Subtract DARK.
	If this is a “master” FLAT then
	    Find all the related FLAT observations and take the mean.
	    Normalize using an algorithm which varies according to whether this is an imaging or spectroscopy flat field. The average value within the most important part of the flat-field should end up 1.0.
	End if

SKY-FLAT	Apply MASK
	Subtract BIAS.
	Linearize the data using a defined look-up table.
	Subtract DARK.
	Multiply or divide by FLAT.
	If this is a “master” SKY-FLAT then
	    Find all the related SKY-FLAT observations and take the median
	    Normalize using an algorithm which varies according to whether this is an imaging or spectroscopy flat field.
	End if

DOME-FLAT	The same as a SKY-FLAT.

TABLE 11 - 2 **Suggested Data Reduction Recipes**

Data type	**Typical Recipe**

ARC	Apply MASK
	Subtract BIAS.
	Linearize the data using a defined look-up table.
	Subtract DARK.
	Divide by FLAT.
	If this is a “master” ARC then
	    Find all the related ARC observations and cross correlate them.
	    Record the shifts between the different ARC observation.
	    Shift all the related ARCs to the same wavelength scale as this one.
	    Take the mean of all the ARCs.
	End if

SKY	Apply MASK
	Subtract BIAS.
	Linearize the data using a defined look-up table.
	Subtract DARK.
	Divide by FLAT.
	If this is a “master” ARC then
	    Find all the related SKY observations and sum them (can be configured to take the median instead).
	End if

IMAGE
(STARE mode)
	Apply MASK
	Subtract BIAS.
	Linearize the data using a defined look-up table.
	Subtract DARK.
	Divide by FLAT.
	Subtract SKY.
	If this is a “master” IMAGE then
	    Find all the related IMAGE observations and combine them, coadding all the observations, or combining them into a mosaic depending on the shifts between the image centers.
	End if

IMAGE
(CHOP mode)
	Apply MASK
	Divide by FLAT.
	If this is a “master” IMAGE then
	    Find all the related IMAGE observations and combine them, coadding all the observations, or combining them into a mosaic depending on the shifts between the image centers.
	End if

TABLE 11 - 2 **Suggested Data Reduction Recipes**

Data type	Typical Recipe

SPECTRUM
(STARE mode)
	Apply MASK
	Subtract BIAS.
	Linearize the data using a defined look-up table.
	Subtract DARK.
	Divide by FLAT.
	Subtract SKY.
	Calibrate into wavelength.
	If this is a "master" SPECTRUM then
	    Find all the related SPECTRUM observations and combine them, coadding all the observations, or combining them into a longer spectrum depending on the shifts between the wavelength ranges.
	    Divide by SPECTRAL-STANDARD.
	End if
	Extract into a 1-D spectrum using an optimal extraction algorithm.

SPECTRUM
(CHOP mode)
	Apply MASK
	Divide by FLAT.
	Calibrate into wavelength.
	If this is a "master" SPECTRUM then
	    Find all the related SPECTRUM observations and combine them, coadding all the observations, or combining them into a longer spectrum depending on the shifts between the wavelength ranges.
	    Divide by SPECTRAL-STANDARD.
	End if
	Extract into a 1-D spectrum using an optimal extraction algorithm.

SPECTRAL-STANDARD
(STARE mode)
	Apply MASK
	Subtract BIAS.
	Linearize the data using a defined look-up table.
	Subtract DARK.
	Divide by FLAT.
	Subtract SKY.
	Calibrate into wavelength.
	If this is a "master" SPECTRAL-STANDARD then
	    Find all the related SPECTRAL-STANDARD observations and sum them.
	    Remove all variations along the slit by summing the rows in the spectrum to generate a 1-D spectrum.
	    Generate an equivalent black-body spectrum, given the temperature of the star, and divide by it.
	    Normalize so the average value is 1.0.
	    Create a 2-D spectrum with the 1-D standard duplicated in every row.
	End if

TABLE 11 - 2 **Suggested Data Reduction Recipes**

Data type	**Typical Recipe**

SPECTRAL-STANDARD (CHOP mode)	Apply MASK
	Divide by FLAT.
	Calibrate into wavelength.
	If this is a “master” SPECTRAL-STANDARD then
	    Find all the related SPECTRAL-STANDARD observations and sum them.
	    Remove all variations along the slit by summing the rows in the spectrum to generate a 1-D spectrum.
	    Generate an equivalent black-body spectrum, given the temperature of the star, and divide by it.
	    Normalize so the average value is 1.0.
	    Create a 2-D spectrum with the 1-D standard duplicated in every row.
	End if

FLUX-STANDARD	Same as SPECTRAL-STANDARD?
PHOTO-STANDARD	Same as IMAGE?
VELOCITY-STANDARD	Same as SPECTRUM?

11.6.3 Rules for Selecting Calibrations

The rules used when selecting calibration frames of a particular type are configurable from the Data Handling System console. Here are some examples for the default rules describing which configuration parameters to match. The rules can be configured so that some of the tests indicated can be ignored if desired by the observer:

- **MASK**

  --Match array size.
- **BIAS**

  --Match detector name.

  --Match array size.
- **DARK**

  --Must be younger than <hhh> hours old.

  --Match detector name.

  --Match array size.

  --Match exposure time.
- **FLAT**

  --Must be younger than <hhh> hours old.

  --Match detector name.

--Match array size.

--Match filter.

--If spectroscopy then also:

...Match grating type, order and angle.

...Match slit type.

- **SKY-FLAT**

--Must be younger than <hhh> hours old.

--Match detector name.

--Match array size.

--Match exposure time on the sky.

--Match the units of the axes and the data.

--Match filter.

--If spectroscopy then also:

...Match grating type, order and angle.

...Match slit type.

- **DOME-FLAT**

The same as a SKY-FLAT.

- **ARC**

--Must be younger than <hhh> hours old.

--Match grating type, order and angle[1]

- **SKY**

--Must be younger than <hhh> hours old.

--Match exposure time on the sky.

--Match the units of the axes and the data.

--Match filter.

--Match the air mass of the observation (to within a tolerance)

--If spectroscopy then also:

...Match grating type, order and angle.

...Match slit type.

1. It is possible for the matching to be even stricter than this. If wavelength accuracy is crucial it is possible to ensure than the wavelength scale has not changed at all by checking a "wavelength calibration index", which is incremented each time the instrument moves its grating, slit, or detector. This can reveal whether the instrument has changed its configuration and then restored it.

- **OBJECT**

  --Must be younger than <hhh> hours old.

  --Match exposure time on the sky.

  --Match the units of the axes and the data.

  --Match filter.

  --Match RA and DEC of the observation (to within a tolerance).

  --Match the name of the object being observed.

  --If spectroscopy then also match:

  ...Match grating type, order and angle.

  ...Match slit type.

- **SPECTRAL-STANDARD**

  --Must be younger than <hhh> hours old.

  --Match exposure time on the sky

  --Match wavelength axis (must be identical).

  --Match filter.

  --Match air mass of the observation (to within a tolerance).

  --Match grating type, order and angle.

  --Match slit type.

- **FLUX-STANDARD**

  The same as SPECTRAL-STANDARD.

- **PHOTO-STANDARD**

  --Must be younger than <hhh> hours old.

  --Match exposure time on the sky

  --Match filter.

  --Match air mass of the observation (to within a tolerance).

- **VELOCITY-STANDARD**

  --Match the wavelength axis units.

11.7 FILE NAMING

It is essential that file names be unique on the system. The OCS will generate names for the raw files to be written by the ICS based on the date and the unique observation identifier. For example a file called

1998:08:14:N047-016-001B.sdf

would represent an raw BIAS observation (B) made on 14th August 1998 with the Gemini North telescope (N). It is the 1st integration (001) of the 16th observation (016) made for science program number 47 (047).

If the DHS generates intermediate files their names should be based on this unique identifier with different codes appended. For example

1998:08:14:N047-016-001BR.sdf

could be used to represent a reduced BIAS file based on the one above. The exact details of the names are for the DHS group to decide, as long as they are unique and subject to the following rules:

- A period (".") in a file name should only be used to identify the extension. It should only appear once within the name.
- Temporary files should be clearly identifiable so they may be tidied up when necessary. For example

  1998:08:14:N047-016-001B_TMP.sdf

  would be a good name for a temporary file.
- The following file extensions should be used:

Type of file	**Extension**
SDS format file.	.sds
FITS format file.	.fits
SYBASE database.	TBD
Hardware setup file.	TBD
HDS format file (ADAM)	.sdf
IRAF format file	*(ask a member of the IRAF group)*

11.8 COLLECTION OF HEADER INFORMATION

Whenever an observation is made, a FITS header [46] needs to be stored alongside the data indicating the status of the observatory during the observation. There are the following different classes of header information:

TABLE 11 - 3 **Various Classes of FITS Header Information.**

Class of header item.	Description	Examples
Default	Items which are always the same and virtually never change.	OBSERVAT — Name of observatory.
Constant	Items which are the same for an entire science program.	OBSERVER — Name of observer.
Static	Items which are the same for a whole observation.	RA, DEC — Observation co-ordinates. CAT-NAME — Name of object from catalogue. OBSTYPE — Type of observation.
Dynamic	Items which change with time and need to be sampled at a particular time, such as at the start or at the end of the exposure. Their sample time only need be accurate to a within a few seconds.	AMSTART — Air mass at start of exposure.
Time Critical	Items which change with time and need to be sampled at *exactly* the right moment, i.e. within a few milliseconds.	UTSTART — Universal time at start of exposure EXPTIME — Exact exposure time.

It is essential for the Data Handling System to store the correct header information with each file, and it is essential for that header information to be collected at the right time. There is also a requirement on the Gemini system is that the header information collected should be configurable. The Gemini system design addresses all these requirements as follows.

11.8.1 The Unique ID and Packet Description File Concept

FITS header information is collected in the following way:

1. When a new instrument is installed, the OCS is provided with a "Packet Description File" (PDF). The PDF concept used here is based on that used on the William Herchel Telescope, as described in [70]. This file contains a list of all the status information that the ICS can provide, with the subsets of that information to be included in the FITS header and to appear on the instrument console. See "Requirements for New Instruments" on page 7 - 11 for an example of such a file. The information given for each FITS item is:

--The name of the FITS keyword.

--The location within the ICS's EPICS database where the item can be found.

--The data type for the item (INTEGER, FLOAT, CHAR, LOGICAL).

--A FITS comment for the item.

Usually a PDF contains information on when that particular item should be collected (e.g. at the start or at the end of an observation). In the Gemini system *all* instrument FITS header items are collected at the end of the observation, as described below.

The TCS subsystems will also provide the OCS with PDFs describing the header and status items they are capable of providing. Unlike the PDF for an instrument, the TCS file should inform the OCS whether to collect an item at the beginning or the end of an observation (e.g. for items such as AMSTART and AMEND[1]). See "Collection of Header Information From the TCS" on page 13 - 19 for details.

2. Just before each observation the OCS generates a unique identifier[2] (ID) for that observation, constructed from the telescope identifier (north or south), the science program number, the science observation number and the science configuration number (See "Science Programs and Observations" on page 10 - 2). A unique file name is also constructed, based on this ID.
3. When an observation is to start, the OCS communicates the observation ID and file name to the ICS with the OBSERVE command, using the mechanism described in ICD/1 [24].

   The OCS collects TCS header information marked for collection at the beginning of observation at this time.
4. While the observation is in progress, the ICS gathers and stores the information at appropriate times. If, for example, a time stamp is required at the beginning and end of the observation, the ICS will store the time into two separate process variables, UTSTART and UTEND, at the exact moments it begins and ends the observation. When an observation completes the ICS uses a Data Storage Server to store the data and unique ID to disk, using the file name provided by the OCS. The mechanism it uses to do this is described in ICD/3. The ICS should have all the FITS header information ready *before* it signals the completion of the OBSERVE command.
5. When the ICS signals completion of the OBSERVE command to the OCS, the OCS collects the FITS header information from the ICS using the standard status gathering mechanism described in ICD/2 [25].

   The OCS also collects header information from the TCS which is marked for collection at the end of the observation.

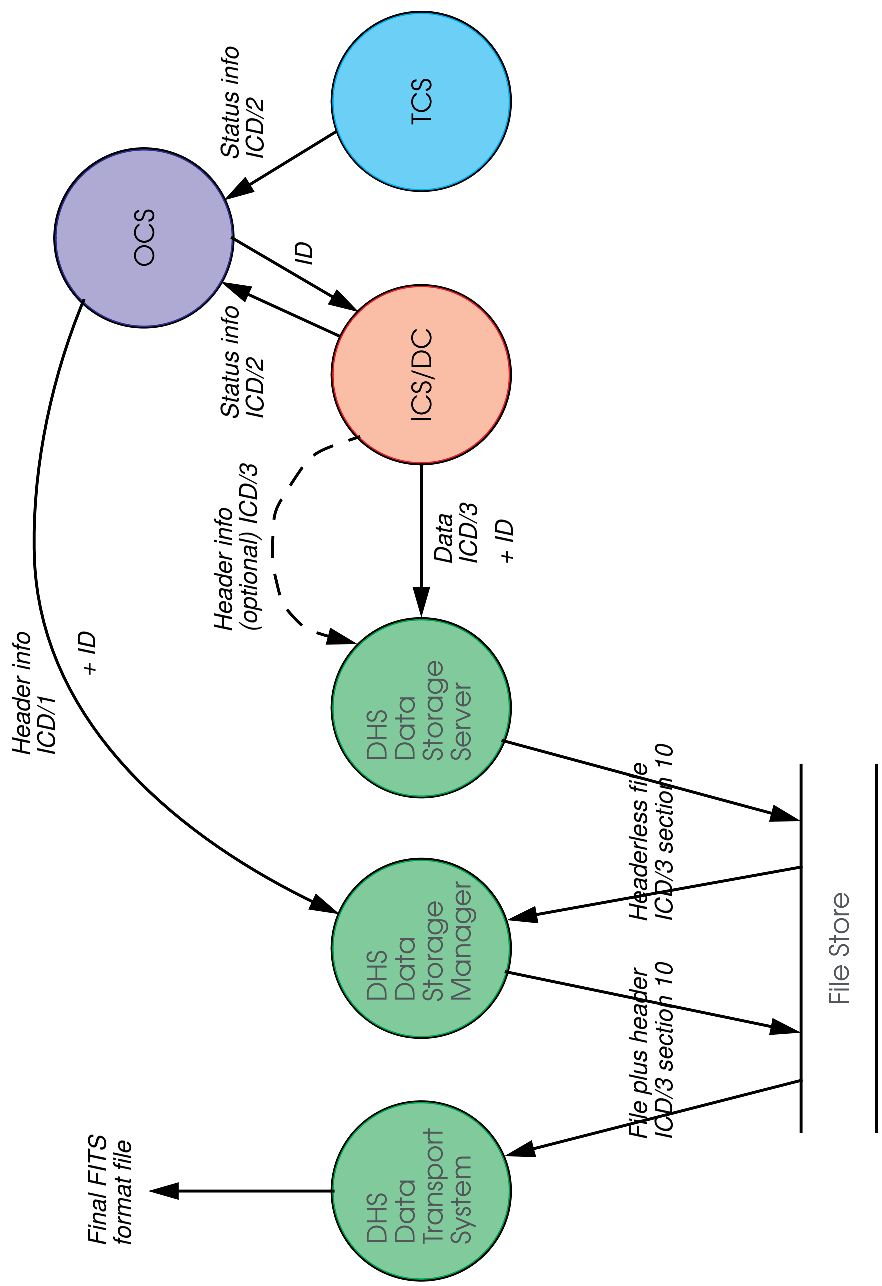
1. The air mass at the start and end of the observation.
2. Sometimes referred to as the "odometer number".

6. The OCS then transmits the observation ID, file name and FITS header information to the DHS with the ENDOBSERVE command, using the mechanism described in ICD/1.
7. On receipt of the ENDOBSERVE command, the DHS opens the file created by the ICS and checks the unique IDs are identical. If they are, the FITS header information provided by the OCS is converted into a FITS header and written to the file (probably using the data storage format recommended in ICD/3 [26]).

   Figure 11 - 4 on page 11 - 36 is a schematic diagram showing the flow of header information through the system and the assembly of that header information into the final FITS format file:

- The OCS collects header information from the TCS and ICS using the mechanism of ICD/2, adds some header information of its own and forwards that header to the DHS Data Storage [Database] Manager.
- The ICS sends its data to a DHS Data Storage Server, which writes that data to the file store as a headerless file in the SDS format described in ICD/3 section 10. The ICS may send a header with these data if there are items that cannot be collected by the OCS.
- The DHS Data Storage [Database] Manager matches together the header information and the headerless file and writes back a file plus header to the file store. This new file is not a FITS format file. It contains a FITS-like header but it is written in SDS format, as described in ICD/3 section 10.
- For simplicity the Data Reduction System is missing from the diagram. It will take the raw files with headers and write back reduced files with headers. The raw and reduced files will be in the same SDS format, but will contain something in their header identifying them as raw and reduced files. The Data Reduction System does not affect the collection of header information.
- The DHS Data Transport System takes each SDS format files with its FITS-like headers and writes a file in standard FITS format. Information contained in the SDS structures will be used to determine the BITPIX, NAXIS, NAXISn, BUNIT, BLABEL, BLANK, CTYPEn, CRPIXn, CRVALn, CDELTn and CLABELn FITS items (see Table 11 - 5, “Standard FITS keywords provided by the DHS,” on page 11 - 40).

FIGURE 11 - 4 **The Collection of Header Information**



The unique ID acts as a check to ensure that the correct header information is matched to each set of data. The PDF allows the FITS header information collected from a particular instrument to be configured. For example, there may be some status items which are not normally collected but whose collection an observer can switch on using a “FITS header collection” dialogue. The dialogue box can be used to modify the ✓ marks shown in the Table on page 7 - 12.

11.8.2 Storage of Awkward Header Information

The FITS header only allows scalar values to be stored, identified by an 8 character keyword and an optional 69 character comment. Some of the information to be stored along with a file may be difficult to fit into this format. Such unusual information may be stored in one of three ways:

- Simple lists of related parameters may be stored using keyword with an element number appended. For example

  --GSRA1, GSDEC1 = RA and DEC of guide star 1.

  --GSRA2, GSDEC2 = RA and DEC of guide star 2.

  --*etc.…*
- More complicated lists may be stored as a FITS “binary tables” extension containing the required information. Some examples are:

  --A list of the actuator forces applied to the primary mirror.

  --A history of the temperature of the detector during the exposure.

  All the examples here are engineering information that an observer would not normally be interested in. It may be more appropriate to log this kind of information separately with the EPICS archiver (See “Logging System (4.2)” on page 11 - 18 and also [17]).

  Note that some FITS readers may be unable to understand binary tables information.
- Complex information needs to be stored in another file and the name of that file included in the FITS header. Some examples are:

  --The FITS header of a reduced file will contain the name of the recipe used to reduce that file. The recipe itself (which is contained in a file or stored in a database) will contain all the data reduction steps used. The header should also contain parameters describing any modifications the observer has made to the recipe.

  --The FITS header of a reduced file will contain only the names of the calibration files used to reduce the observation. The headers of those calibration files will contain any further information about them.

  Note that, if files containing pointers to other files in their header are archived, those other files must be archived as well.

11.8.3 Standard FITS Keywords

Many observatories have defined standard FITS keywords to represent common header items. Having standard keywords makes it easier for FITS readers to make sense of header information. However, it is essential that a keyword only be used for its agreed purpose, to prevent FITS readers being confused. It is also essential that there are no name clashes with the reserved keywords listed in [46].

There follows some tables of initial standard keywords for use in Gemini FITS headers. These keywords are based on those described in [5], [50], [59] and [70], together with some information provided by Doug Tody of the NOAO. The columns in the tables are as follows:

Keyword. The ≤8 character FITS keyword representing the item.

Type. The type of the item (CHAR, INTEGER, LOGICAL or FLOAT).

Units, format or allowed values. The units if numeric, or the general format, a list of allowed values or an example if character.

Standard? This box is marked with a ✓✓ is the keyword is part of the FITS standard [46], or a ✓ if it is commonly recognized.

Mandatory? Keywords marked as ✓ *have* to be present in the header (the Gemini archive software will, expect them to be there). The others are optional and may depend on the observing mode or type of instrument used.

Description. is a textual description of the item, a shortened (≤69 characters) version of which should become the FITS comment describing the item.

The tables will need to be refined by discussions between the various groups and Gemini project office. Changes to the FITS keywords should only be made with the consent of the Gemini project office

The following items are provided by the OCS using information on the current observation contained in the observing database.

TABLE 11 - 4 **Standard FITS keywords provided by the OCS**

Keyword	Type	Units, format or allowed values	Standard?	Mandatory?	Description
GEMID	CHAR	'[N|S]ppp.ooo.ccc.iii'		✓	Unique observation ID.
ORIGIN	CHAR	(e.g. 'CADC').	✓✓	✓	The data handling institution who wrote the FITS file.
OBSERVAT[a]	CHAR	'Gemini Mauna Kea' 'Gemini Cerro Pachon'	✓	✓	The name of the observatory
LAT-OBS	FLOAT	degrees			Latitude of observatory.
LONG-OBS	FLOAT	degrees			Longitude of observatory.
TELESCOP	CHAR	'Gemini 8m'	✓✓	✓	The name of the telescope.
FSTATION	CHAR	'Cassegrain 1' … 'Cassegrain 5', 'Nasmyth', 'Prime'		✓	Focal station and port number.
INSTRUME	CHAR		✓✓	✓	Name of instrument.
DETECTOR	CHAR			✓	Name of detector.
OBSERVER	CHAR		✓✓	✓	Name of observer or observers.
PRINCIPA	CHAR			✓	Name of principal investigator.
PROPOSAL	CHAR			✓	PATT or TAC proposal reference.
DATE-OBS[b]	CHAR	'dd/mm/yy'	✓	✓	UT date of observation. (See EPOCH to determine in which century the observation was made).
OBJECT	CHAR		✓✓	✓	Title of observation (provided by the observer).
CAT-NAME[c]	CHAR			✓	The name of the object, exactly as it appears in a standard catalogue (i.e. not provided by the observer).
CAT-RA	CHAR	'hh:mm:ss.ss'		✓	RA of object exactly as it appears in a standard catalogue.
CAT-DEC	CHAR	'sdd:mm:ss.s'		✓	DEC of object exactly as it appears in a standard catalogue.

TABLE 11 - 4 **Standard FITS keywords provided by the OCS**

Keyword	Type	Units, format or allowed values	Standard?	Mandatory?	Description
CAT-EQUI	CHAR	(e.g. ‘J2000.0’ or ‘B1950.0’)			Equinox of catalogue co-ordinate system
OBSTYPE[d]	CHAR	‘BIAS’, ‘DARK’, ‘FLAT’, ‘SKY-FLAT’, etc.…	✓	✓	Type of observation.
COMMENT	CHAR		✓✓		Comments provided by the observer and/or operator.
Plus other items to be decided by the OCS and DHS groups and Gemini project office.…					

a. The OBSERVAT keyword is necessary because the FITS standard requires that ORIGIN be the name of the institution who actually wrote the file, and this may be different.

b. Note that some observatories use UTDATE = ‘yyyy:mm:dd’ because they have found the standard definition deficient, as it does not explicitly state the date should be UT, and it only allows 2 characters for the year number.’

c. Using CAT-NAME will greatly help when retrieving the data from the archive. OBJECT may contain a name familiar only to the observer.

d. IMAGETYP is sometimes used at other observatories but OBSTYPE seems to be more popular.

The following items are provided by the DHS by translating the data description contained in the SDS structure containing the data (see ICD/3[26]).

TABLE 11 - 5 **Standard FITS keywords provided by the DHS**

Keyword	Type	Units, format or allowed values	Standard?	Mandatory?	Description
SIMPLE	LOGICAL	Always TRUE.	✓✓	✓	Does FITS file conform to basic format?
BITPIX	INTEGER		✓✓	✓	Number of bits per pixel. Varies.
NAXIS	INTEGER		✓✓	✓	Number of axes in data array. Varies
NAXISn	INTEGER		✓✓	✓	Number of elements along axis n (n = 1 to NAXIS).

TABLE 11 - 5 **Standard FITS keywords provided by the DHS**

Keyword	Type	Units, format or allowed values	Standard?	Mandatory?	Description
BUNIT	CHAR	Use the units defined in the IAU style manual, [45].	✓✓		Units of the data.
BLABEL	CHAR				Label describing the quantity contained in the data (e.g. ‘intensity’).
BLANK	INTEGER		✓✓		A blank value used to represent bad pixel information. See [46] for details.
CTYPEn	CHAR	Use the units defined in the IAU style manual, [45].	✓✓	✓	Units of axis n (n = 1 to NAXIS).
CRPIXn	FLOAT	Always 1.	✓✓	✓	Array location of reference pixel for axis n (n = 1 to NAXIS).
CRVALn	FLOAT		✓✓	✓	Axis value at reference pixel for axis n (n = 1 to NAXIS).
CDELTn	FLOAT		✓✓	✓	Interval between pixel centers for axis n (n = 1 to NAXIS).
CLABELn	CHAR				A label describing the coordinate (e.g. ‘wave-length’) for axis n (n = 1 to NAXIS).
DISPAXIS	INTEGER				Axis number corresponding to dispersion direction (spectroscopy only).
DRRECIPE	CHAR			✓	Name of recipe used to reduce this file (‘NONE’ if the file does not contain reduced data).
DRRFLAGS	CHAR	‘T T F T T F T…’			Array of flags indicating which steps in the data reduction recipe were enabled.
DRNCALIB	INTEGER				Number of calibration observations used to reduce this file.
DRCTYPn	CHAR	(e.g. ‘BIAS’)			Observation type of calibration observation n (n = 1 to DRNCALIB).
DRCNAMn	CHAR				Name of calibration observation n (n = 1 to DRNCALIB).
Plus other items to be decided by the DHS group and Gemini project office.…					

The following keywords are provided by the TCS through the OCS.

TABLE 11 - 6 Standard FITS keywords provided by the TCS

Keyword	Type	Units, format or allowed values	Standard?	Mandatory?	Description
RA	CHAR	‘hh:mm:ss.ss’	✓	✓	RA of telescope[a]
DEC	CHAR	‘sdd:mm:ss.s’	✓	✓	DEC of telescope
EQUINOX[b]	FLOAT	civil years (e.g. 2000.0)	✓✓	✓	Equinox of telescope coordinates
RADECSYS	CHAR	(e.g. ‘FK4’, ‘FK5’)	✓	✓	The stellar reference frame used for the RA and DEC.
TIMESYS	CHAR	‘Besselian’ or ‘Julian’		✓	The time system used for the RA and DEC co-ordinates.
EPOCH[c]	CHAR	civil years (e.g. 1998.3)			The epoch of the observation.
TELFOCUS	FLOAT				Telescope focus (units to be determined)
AMSTART	FLOAT				Air mass at the start of the observation
AMEND	FLOAT				Air mass at the end of the observation
Plus other items to be decided by the TCS group and Gemini project office....					

a. The exact meaning of RA and DEC should be decided by the TCS group, bearing in mind the suggestions of the OGIP panel, [48].

b. Note that many observatories use a character item so they can combine the time system with the equinox in the form ‘J2000.0’ or ‘B1950.0’. However, the FITS standard [46] requires this item to be floating point.

c. Note that the FITS documentation [46] says that the use of EPOCH is deprecated because it can be confused with EQUINOX. However, it is included here because it helps to resolve any ambiguity in DATE-OBS. DATE-OBS is a human readable date and EPOCH a machine readable one.

The following keywords are provided by the ICS or DC through the OCS.

TABLE 11 - 7 **Standard FITS keywords provided by the ICS or DC**

Keyword	Type	Units, format or allowed values	Standard?	Mandatory?	Description
UTSTART	CHAR	‘hh:mm:ss.ss’	✓	✓	UT when exposure started
UTEND[a]	CHAR	‘hh:mm:ss.ss’	✓		UT when exposure ended
ELAPSED[b]	FLOAT	seconds		✓	Total elapsed time (UTEND-UTSTART) in seconds.
DARKTIME	FLOAT	seconds			Total time for which the detector was accumulating a dark count.
EXPTIME[c]	FLOAT	seconds		✓	Total integration time on the sky.
Plus other items which may vary from instrument to instrument. Instrument-specific items should have some code letters representing the instrument included in the keyword, to ensure the keywords are unique and can be seen to be related. See “Requirements for New Instruments” on page 7 - 11 for some examples. To be decided by the ICS and DC groups and the Gemini project office.…					

a. UTEND will be calculated from UTSTART and ELAPSED if not present.

b. Note that ELAPSED and DARKTIME can be different for infra-red observations where the detector is not integrating throughout the entire observation (e.g. when chopping). DARKTIME will be assumed the same as ELAPSED if not present.

c. EXPOSED is a popular alternative to EXPTIME.

11.9 MAJOR INTERNAL DATA STORES

11.9.1 The Data Store

A permanent data store. The baseline configuration for the *data store* is a set of networked disks attached to a central file server. It would be preferable to use a system such as the Epoch file server where the central file server consists of a hierarchy of on-line disk, on-line optical disk, on-line tape, and off-line media. Due to budgetary constraints it is necessary to install only the on-line disk at this point but to allow for the upgrade path to an “Epoch like” file system in the future.

The storage of the data on disk must take into account not only the requirements of the data acquisition system for rapid access but also the need for both ADAM and IRAF to access the data for data reduction. It is recommended that files be stored in an internal format based on SDS, but files should be exported to the External Data Reduction System and to the Gemini archive in standard FITS format. See “Central DHS to External Data Reduction System” on page 11 - 8 and “Central DHS to Permanent Store and Archive” on page 11 - 11.

It is up to the various data reduction groups (ADAM, IRAF, FIGARO etc.…) to ensure their software can read files in FITS format.

11.9.2 The Data Handling Database

The Data Handling and Data Reduction systems maintain a record of their current state the Data Handling Database. The state of the observatory and other items of interest are periodically logged to this database, which provides information for the day crew and statistics for administrative purposes (See “Logging System (4.2)” on page 11 - 18). A commercial database system is used, the current baseline being SYBASE.

Access to the contents of this on-line database is through established commercial software as is visualization of these data. Visualization, if not provided by the SYBASE product itself, is provided by the commercial product PV-Wave Table 11 - 8 shows the sort of functions the commercial database system should provide.

TABLE 11 - 8 Internal Database System Functionality

Component	Function
SYBASE Database Package	Create, access, update, maintain, etc... database
	Interactive database queries
	Programmatic database queries
	Backup/Restore database
	Data visualization (plotting, etc...)
Control software	Real-time database to/from SYBASE toolset
User interface specification	TBD

The database should contain the following information:

- A list of the files created on the system, together with the status of those files. The database needs to contain at least the following information for each file:

--Observation ID.

--File name.

--Science program ID.

--Principal investigator name.

--Observer name.

--Size of data array.

--A title for the observation.

--Date and time of observation.

--Observation type (BIAS, DARK, FLAT … See Table 11 - 1 on page 11 - 24).

--A pointer to the data reduction recipe to be applied to this file. (The pointer can be NULL, in which case either the user-defined recipe for this observations type, or the default recipe for this data type, will be applied).

--Flag indicating the observer's satisfaction with the observation (GOOD or BAD). (Calibration observations will only be used in a data reduction recipe if their satisfaction flag is GOOD).

--The reduction status of the data contained in the file (RAW or REDUCED). This is used to distinguish files containing raw and reduced data.

--If a raw file, the name of the corresponding reduced file. If a reduced file, the name of the raw file from which it was made.

--Has the file been reduced (YES/NO). This is used to signal when a raw file has been reduced.

--Has the file been saved to permanent store (YES/NO).

--Has the file been copied to the observer (YES/NO).

--Has the file been successfully archived (YES/NO).

--Plus all the instrument parameters which are commonly searched for during the application of a data reduction rule (e.g. the detector name, exposure time on the detector, filter, grating type, grating order, grating angle, slit type, etc.… — See "Rules for Selecting Calibrations" on page 11 - 29).

The last item in this list is mainly for efficiency. If the Data Reduction Agent is searching for a suitable calibration frame using a rule, and the quantity it is trying to match is not contained in this database, it will have to open and read all the file headers. The exact items to include in this database is a decision left to the DHS work package.

- A list of default data reduction recipes, as supplied by the instrument developers, one for each instrument and observation type. Also a list of default parameters associated with each recipe step.
- A list of the user-defined data reduction recipes, as defined by the observer using a particular instrument, once per instrument and observation type. These observer-defined recipes will override the default ones if they exist. Also a list of default parameters associated with each recipe step.

Note that the observer can only change a recipe by enabling or disabling the steps within it or by changing the parameters associated with a step (see Figure 6 - 7, “An Example Data Reduction Console,” on page 6 - 23 for en example).

Note also that if any file has a pointer to a particular recipe, that pointer overrides the default and the observer-defined recipes for that data type.

- Information on the configuration of the quick look system, the logging system, the data transport system and the data reduction displays. See the chapters describing these systems for information on what configuration information they need.

The database should contain all the information required by the Data Handling and Data Reduction Consoles. See Figure 6 - 6 on page 6 - 14 and Figure 6 - 7 on page 6 - 23.

11.9.3 The Data Reduction Queue

The data reduction queue is a first in first out queue which contains at least the following information for each entry:

- Date and time at which this entry was submitted.
- Priority of this entry.
- Status of this entry (pending, holding, executing etc.…).
- Information provided with the entry. This can be one of the following:

--A **CONFIG** command with a pointer to a new data reduction configuration.

--A **PROCESS** command with the name of a file to be reduced.

Entries will be entered into the queue by the Data Handling Command Processor at a standard priority. An observer can enter commands at a higher or lower priority, or can alter the priority of commands already in the queue. The queue is managed just like a batch queue on a computer system.

Each time the Data Reduction Agent becomes idle it extracts the next entry from the queue using the following algorithm:

- Select all the entries with the highest priority.
- Select the oldest entry from this list.

The queue should be able to support a “holding area” where failed jobs can be kept until the time comes to retry them. Jobs can be retried by selecting them manually with the Data Reduction Console, or the Data Reduction Agent can be programmed to retry them at regular intervals (See “Data Reduction Agent (4.1)” on page 11 - 19 for details).

The exact details of the Data Reduction Queue are to be decided by the DHS work package.

11.10 MAJOR INTERNAL INTERFACES

The major internal interfaces are to be designed by the DHS work package.

11.11 DATA HANDLING SYSTEM HARDWARE POLICY

The quantity and size of the computers purchased to support the subsystems contained in the Data Handling System depends on the design of the system and the budget. The major consideration, after budget, is failure recovery — the loss of a single computer shall not put the observatory off-line for longer than it takes to reconfigure the software/hardware. It is acceptable to have the hardware required for failure recovery specified as part of an upgrade path that is passed on to the operations staff.

The baseline is to have a single Sun to be used for all the subsystems as well as data processing. The upgrade path, dependent on budget, from this baseline is a multiprocessor Sun (such as the series 20).

The Data Storage hardware resides in the computer equipment room in the enclosure support building. The baseline is to have this hardware connected directly to the data processing Sun. The requirement to have an upgrade path to an Epoch (or equivalent) file server may require a change to this baseline. The goal is to have sufficient space to store 6 days of raw data.

The Data Transport hardware resides in the computer equipment room in the enclosure support building. The baseline is to have this hardware directly connected to the data processing Sun.

The Internal Database resides on a large (~2 Gbyte) disk attached to the data processing Sun.

The Gemini Archive hardware resides in the computer equipment room in the enclosure support building. The baseline is to have this hardware directly connected to the data processing Sun.

11.12 DELIVERABLES

The Data Handling Work Package will provide:

- A quick look display server, receiving data as described in ICD/3 [26].

- A quick look system, which is a packaged version of the quick look display server capable of reading data from files of Gemini data.
- A synchronous data reduction system (in consultation with the Observatory Control System group).
- A data storage server, receiving data as described in ICD/3 [26] and storing data to disk in an internal Gemini format (SDS format recommended).
- The Data Handling Command Processor and Data Storage Database Manager capable of obeying the sequencer commands from the Observatory Control System and managing the Data Handling Database.
- A logging system, which includes a history logging server receiving messages as described in ICD/4 [27].
- A data reduction agent (or agents). The agent or agents must be able to cater for the requirements of the high resolution wavefront sensor as well as the scientific instruments. See "High Resolution Wavefront Sensor" on page 12 - 36.

  The DHS group may choose the most suitable external data reduction system to use with their agent. More than one agent may be required because the one used for the high resolution wavefront sensor may need to have IDL-SH as an external data reduction system.
- A design for the internal data stores and interfaces of the Central DHS.
- Subroutine libraries for:

  --assembling and disassembling the data structures described in ICD/3.

  --accessing files of Gemini data.

  --accessing information from the External Databases.

  --assembling and disassembling the logging messages described in ICD/4.
- Utilities for:

  --displaying the contents of files of Gemini data.

  --converting Gemini data to and from FITS format.

  --converting EPICS archiver files to standard logging format (e.g. SYBASE)

  --copying Gemini data files to and from tape.

  --copying Gemini data files to and from the permanent store.

  --problem tracking (e.g. GNATS)

  --interrogating and obtaining data from the Gemini archive.

See the "Deliverables" sections in ICD/3 [26] and ICD/4 [27] for more details.

12 Details of the Instrument Control Systems

12.1 The Instrument and Detector Control Systems

The scientific instruments and detector array controllers currently planned for the Gemini telescopes are listed in Table 12 - 1 on page 12 - 1 (though they may not all be available at first light):

TABLE 12 - 1 **Current List of Gemini Instruments - DRAFT**

Instrument	Builder	Detector(s)	Detector Controller	Controlled Mechanisms
1-5 micron Imager	Hawaii	1024x1024 InSb	To be decided.	Filter Wheel (x2) Detector Focus AO Feed (?)
1-5 micron Spectrometer	USA	1024x1024 InSb	To be decided.	Filter Wheel (x2) Grating(s) Detector Focus AO Feed (?)
8-30 micron Imager	USA	256x256 BIB (?)	To be decided.	Filter Wheel (x2) Detector Focus
8-30 micron Spectrometer[a]	UK Observato-ries	256x256 BIB (?)	ALICE	Filter Wheel (x2) Slit Wheel Grating(s) Field Rotator Detector Focus

TABLE 12 - 1 Current List of Gemini Instruments - DRAFT

Instrument	Builder	Detector(s)	Detector Controller	Controlled Mechanisms
Multi-Object [Optical and UV] Spectrograph (x2)	UK/Canada	2048 x 4096 CCD (x2)	ARCON (x?)	Filters Slit Masks IF Fibers Gratings (?) Detector Focus AO Feed (?)
High Resolution Optical Spectrograph	UK Observatories/ UCL	2048 x 4096 CCD (x2)	ARCON (x?)	Filter Wheels Slit Cross Disperser Echelle Detector Focus AO Feed (?)
Acquisition and Guiding (x2)	UK Observatories	2048 x 2048 CCD 1024 x 1024 CCD 64 x 64 Frame Transfer (x2)?	ARCON + Fast Electronics?	Peripheral WFS (x2) Acquisition Camera Feed Science Fold AO Fold AtmDCs
High Resolution Wavefront Sensor[b]				High Resolution WFS
Adaptive Optics	Dominion Astrophysical Observatories	64 x 64 Frame Transfer (x2)?	Fast Electronics + Digital Signal Processors	Deformable Mirror DM Positioner AtmDC
Cassegrain Rotator	Project Office	None	None	Rotator Drive

a. This is being built now.

b. Part of Acquisition and Guiding.

The above list is provisional and may change as the Instrumentation Group assess the best use of the instrumentation budget.

The following table points to the relevant sections of this chapter concerned with each particular instrument:

TABLE 12 - 2 Cross references for Gemini Instruments

Instrument	Cross reference
1-5 micron Imager	See “Infra-red Imaging and Arrays” on page 12 - 33.
1-5 micron Spectrometer	See “Infra-red Spectroscopy” on page 12 - 35.

TABLE 12 - 2 **Cross references for Gemini Instruments**

Instrument	**Cross reference**
8-30 micron Imager	See “Infra-red Imaging and Arrays” on page 12 - 33.
8-30 micron Spectrometer	See “Infra-red Spectroscopy” on page 12 - 35.
Multi-Object [Optical and UV] Spectrograph (x2)	See “Multi-Object Optical and Ultra-Violet Spectrograph” on page 12 - 29.
High Resolution Optical Spectrograph	See “High Resolution Optical Spectrograph” on page 12 - 31.
Acquisition and Guiding (x2)	See “Details of the Acquisition & Guiding System” in Chapter 19, “The Field Acquisition Camera” on page 12 - 29 and “High Resolution Wave-front Sensor” on page 12 - 36
High Resolution Wavefront Sensor	
Adaptive Optics	See “Details of the Adaptive Optics Control System” in Chapter 20.
Cassegrain Rotator	See “Details of the Cassegrain Rotator Control System” in Chapter 15.

The following assumptions are made by this design document:

- There is a separate Instrument Control System (ICS) for each scientific instrument.
- There is a separate Detector Array Controller for each type of detector.
- When an instrument uses a particular type of detector it uses an instance of the appropriate Detector Array Controller as a subsystem.
- A Detector Array Controller is subservient to an Instrument Control System and accepts commands from it.
- When two instruments are installed, and they both use the same type of detector, then *two* copies of the appropriate Detector Array Controller are used. There is a separate Detector Array Controller for each detector array, and instruments never share a Detector Array Controller. See Figure 7 - 2, “The Relationship Between Instrument and Detector Control Systems,” on page 7 - 5.

This chapter will look at the Instrument Control System and Detector Array Controller separately, specifying the interface between them and their interfaces to the outside world.

12.2 THE INSTRUMENT CONTROL SYSTEM (ICS)

This section of the document will describe all parts of an instrument controller *except* the Detector Array Controller (which is described on page 12 - 19).

The scientific instrument controllers have the following external interfaces:

- Observatory Control System
- Data Handling System:
  --Central DHS
  --DHS Quick Look Server
  --DHS Data Storage Server
- Instrument setup information
- Instrument hardware
- Detector Array Controller
- Synchronization Bus
- Time Bus
- Event Bus
- Interlock System

The scientific instrument controllers have the following major internal subsystems. The numbers refer to the process IDs on the data flow diagrams shown in "Decomposition/Dependency Descriptions" in Appendix A1:

- Instrument sequencer (2.1)
- Optical components controller (2.2)
- Internal data processor (optional) (2.4)
- Data Storage Manager (ICS or DC but not both) (2.5)
- Quick Look Display Manager (optional) (2.6)
- Instrument Visualization Subsystem (c.f. the WHT's MIMIC display) (2.7)
- *Plus any other things required by specific instruments…*

The scientific instrument controllers have the following major internal data stores:

- Raw frame store
- Coadded frame store
- Instrument control database
- Instrument setup information store
- *Plus any other things required by specific instruments…*

The scientific instrument controllers have the following major internal interfaces:

- Digital to analogue converters (for signal control)
- Analogue to digital converters (for signal sensing)

- TTL digital I/O devices (for switching and sensing)
- Motor control electronics
- *Plus any other things required by specific instruments…*

12.3 ICS EXTERNAL INTERFACES

As the following are external interfaces between software packages they require interface control to make sure that the two packages involved can proceed in parallel. The details of these interfaces are contained in Interface Control Documents (ICDs), as described throughout the text.

In what follows we detail how the interface is used by the instrument control system.

12.3.1 Observatory Control System

Commands are sent from the OCS to the ICS either from sequencer commands issued during the execution of a sequence recipe, or from changes made to an instrument's console.

12.3.1.1 STANDARD COMMANDS

During execution of a sequence recipe the OCS issues a sequence of the following commands to the ICS using the protocol of ICD/1 [24]. See Table 10 - 2 on page 10 - 27 for a general list of these commands. All instrument control systems therefore need to respond to the following high level commands:

- **TEST**

  Test all the components of the instrument that are capable of being tested via software and report any failures. This command is used at least once at the beginning of each night to check that each instrument is functioning correctly before starting to observe.

  This command can be executed at any time, but it will reset the current configuration to *undefined.*

- **INIT**

  Initialize the instrument and read the instrument's hardware setup file. This command is used as soon as the instrument control system is loaded, or whenever the instrument's hardware configuration is changed (e.g. when a new filter wheel is installed).

  This command can be executed at any time, but it will set the current configuration to *undefined.*

- **RESET**

  Reset the instrument's configuration to the one it had at start-up. Do not re-read the hardware setup file.

- **CHECK**

  The instrument is provided with a target configuration, and it checks that it is capable of moving to that configuration. For example, does the specified filter exist in this instrument, or is the configuration beyond its capabilities? A status is returned to indicate the success or failure of this command.

  This command can be executed at any time and does not affect the current configuration of the instrument.

- **APPLY**

  The instrument moves to the requested configuration, positioning filter wheels, slit mechanism etc… as requested. The main shutter (if there is one) is kept closed. It is up to each particular Instrument Control System to decide how best to change configuration while protecting its sensitive components.

  This command will only execute successfully if a valid configuration is given. If successful, the new configuration becomes the current one.

- **OBSERVE**

  This command will open the main shutter[1] for the specified length of time. The configuration required to make the exposure should already have been adopted at the APPLY stage. If the instrument needs to take control of the chopper it does so when this command is issued and before the main shutter is opened. After the specified time the main shutter is closed automatically and the data frame stored to disk. If the instrument needs to do any internal data processing it does so before storing the data.

  This command can only be executed when the instrument has achieved a valid configuration (i.e. the configuration is not *undefined*). The command will move the instrument into an *exposing* state, where it will remain until the exposure completes.

- **PAUSE**

  On receipt of this command the main shutter is closed (if the instrument can do this), but the instrument maintains its current configuration and continues its monitoring and chopping operations. The countdown timer monitoring the length of the current exposure is paused. The observer is warned at regular intervals that the instrument is in a paused state.

  This command can only be executed in the *exposing* state. It is ignored in other states and a warning message issued. If the current state is *exposing* it is changed to *paused*.

1. By "main shutter" we mean whatever device is used by the instrument to control the length of the detector's exposure to light. Some instruments may rely on the Detector Array Controller to control the exposure and may not have a shutter.

- **CONTINUE**

  This command reverses the action of the PAUSE command. The shutter is reopened and the countdown timer resumed at the point it left off. The exposure continues as if the pause had never happened.

  This command can only be executed in the *exposing* or *paused* states. It is ignored in other states and a warning message issued. If the current state is *paused* it is changed back to *exposing*.

- **STOP**

  This command cancels an exposure. The main shutter should be closed. On some instruments it may be possible to salvage some of the data after a STOP, by issuing a FLUSH command.

  The command can only be issued from the *exposing* or *paused* states.

- **FLUSH**

  This command forces an instrument controller to salvage any data contained in its buffers, which may have been left there after a STOP command cancelled an exposure. The data should be stored to disk in the usual way.

- **ABORT**

  This is an emergency stop command, instructing the ICS to stop whatever it is doing immediately and protect itself. All motors are stopped, lamps switched off, and the detector protected by whatever means are at the disposal of the instrument (e.g. closing the shutter).

  The command can be executed at any time, and will set the current configuration to *undefined.* An APPLY command will be necessary before observing can be resumed, though it may be prudent to try a TEST command first.

- **PARK**

  The instrument adopts a safe configuration in which it can safely be switched off.

In addition to the above standard commands, an ICS may recognize commands which are specific to a particular instrument. The commands may have originated from an automatic sequence recipe or from the observer interacting at a console, but the ICS reacts in the same way independently of where a command has come from.

12.3.1.2 STANDARD COMMAND PARAMETERS

Details of the parameters passed with the above commands have yet to be decided. It is likely that the parameters will be as follows:

- TEST; no parameters.
- INIT; the name of the instrument setup file.
- RESET; no parameters.

- CHECK; an instrument configuration.
- APPLY; an instrument configuration.
- OBSERVE; observation ID and file name.
- PAUSE; no parameters.
- CONTINUE; no parameters.
- STOP; no parameters.
- FLUSH; observation ID and file name.
- ABORT; no parameters.
- PARK; no parameters?

12.3.1.3 CONFIGURATIONS PASSED FROM OCS TO ICS

The OCS supplies a target configuration (or mini-configuration, see below) for the ICS along with the CHECK and APPLY commands using the protocol described in ICD/1 [24].

The target configuration is copied into the ICS internal EPICS database. When the time comes to change configuration this target configuration is copied to a *demand configuration* in the internal EPICS database. The ICS then endeavors to make the *demand* and *actual* configurations the same. The knowledge of how to do this in the most efficient and safe way possible for a particular instrument is contained within the ICS.

12.3.1.4 OCS INSTRUMENT CONSOLES

Alterations to the configuration of an instrument can be made using the instrument's console in the OCS. The observer may alter one or more of the instrument's parameters within this console and request the instrument to adopt the new configuration by clicking on the "apply" button. See Figure 7 - 3 on page 7 - 10 for an example ICS console. When this happens the parameters which have changed are packaged up as a mini-configuration, and a standard APPLY command is sent to the ICS with this mini-configuration as a parameter. The changed parameters are copied to the *demand configuration* in the internal EPICS database. The ICS then endeavors to make the *demand* and *actual* configurations the same, as above.

The way in which non-conforming instruments receive sequencer commands is discussed in ICD/8 [32].

The console may also be used to send extra commands (e.g. "switch on verbose mode") which are not part of the standard sequencer command set.

12.3.2 Data Handling System

12.3.2.1 CENTRAL DHS

History logging information may be sent from any part of the ICS to the DHS logging system using the protocol described in ICD/4 [27].

12.3.2.2 DHS QUICK LOOK SERVER

The Data Handling System (DHS) provides the Instrument Control System (ICS) with a server for displaying quick look data. There may be zero or more quick look servers for each instrument. Data are transferred to the Quick Look Server using the protocol of ICD/3 [26].

12.3.2.3 DHS DATA STORAGE SERVER

The Data Handling System (DHS) provides the Instrument Control System (ICS) with a server for storing its bulk data to disk. There is a separate data storage server for each instrument. Data are transferred to the Data Storage Server using the protocol of ICD/3 [26].

Note that *either* the ICS *or* the DC will store data to disk, but not both. See Figure 6 - 5 on page 6 - 11.

The data storage server acts as a server to the ICS, but the ICS acts as a server to the OCS.

12.3.3 Instrument Setup Information

The instrument control subsystems need some knowledge of the layout and capabilities of the instrument. This *instrument setup information* is obtained by each ICS from the outside world. The information is specific just to a particular instrument, so it is up to each Instrument Control System to decide how and in what format to read this information. ASCII text files are recommended, as these may be viewed and changed easily. The instrument setup information is read when the ICS executes its INIT action and is stored internally by the sub-systems.

The instrument setup information may contain the following:

- A list of available filter combinations, together with their motor encoder positions
- A look-up table, or a formula, to convert grating angle and order into motor encoder position (if the instrument is a spectrograph)

- A look-up table or a formula, to convert slit height and width into motor encoder position (if the instrument is a spectrograph)
- The names and addresses of various sensor and control ports etc…
- If the instrument has an on-board processing capability, information on the layout of the processors
- Information on which configurations are legal and which are not, and information on how to change safely from one configuration to another
- *plus any other information describing the instrument…*

Details are left to the instrument developers.

12.3.4 Instrument Hardware

This interface depends on the instrument hardware used, and is the responsibility of the instrument development team. Standard Gemini hardware drivers (provided by the “Standard Control System” work package, see ICD/13 [37]) should be used where possible, and the instrument hardware interfaced to VME modules.

Details are left to the instrument developers.

12.3.5 Detector Array Controller

The ICS will command the Detector Array Controller using EPICS channel access., as described in ICD/7a [30].

The ICS may receive bulk data from the Detector Array Controller if it needs to carry out some instrument-specific processing on that data. If the ICS does not need access to the data it can be routed direct from the Detector Array Controller to the DHS data storage server.

Bulk data will be transferred from the Detector Array Controller to the ICS using a message transmitted using IMP and encoded using SDS, as described in ICD/3 [26].

If an instrument is “non-conforming” it is still possible to receive data from the Detector Array Controller using IMP as long as the instrument uses a VxWorks or unix host. See ICD/8 [32] for details.

12.3.6 Synchronization Bus

The synchronization bus is used to handle rapid exchange of digital information between Gemini systems. If the instrument has an on-board wavefront sensor it would be used to communicate the Zernike coefficients to the Secondary Control System, for example. See "Synchronization Bus" on page 12 - 22.

See ICD/5 [28] for details of the transfer of wavefront information and ICD/10 [34] for a description of the synchronization bus.

12.3.7 Time Bus

Time is distributed to all VME systems via a time bus with ±5 microsecond accuracy. Gemini supplied drivers will provide time via EPICS as Coordinated Universal Time (UTC) and will also slave the VxWorks clock to the time bus. This information may also be used for occultation observations and other esoteric uses of the instrument.

See ICD/9 [33] for a detailed description of the time bus.

12.3.8 Event Bus

The event bus is used (mainly) to distribute the phase of the chopper and position of the telescope in the nodding cycle. It is also used to distribute the demand signal to the secondary chopper or to the TCS. In general the chopper demand signal is generated by the TCS and sent to the instrument and the secondary. However there is a facility for the instrument to send the demand signal to the TCS/Secondary chopper. See Chapter 7, 'Interface to Chopping and Nodding'.

See ICD/11 [35] for a detailed description of the event bus.

12.3.9 Interlock System

The interlock system prevents an instrument from opening its main shutter at times when it may be unsafe, for example while the telescope is slewing or the cassegrain instrument rotator is in motion. It also prevents the Telescope Control System from slewing the telescope or the instrument rotator when an instrument has its main shutter or an access door open.

Interlock sensors are placed on the main shutter and all major access doors of each instrument. An instrument should be designed so it is not possible to work on it without tripping at least one of the interlock switches.

The baseline interface to the interlock system is 2 input TTL signals and 2 output TTL signals. The input TTL signals represent Set and $\overline{\text{Set}}$ while the outputs represent Request and $\overline{\text{Request}}$. Details are in Chapter 14, 'Details of the Mount Control System'.

See ICD/12 [36] for a detailed description of the interlock system.

12.4 MAJOR INTERNAL SUBSYSTEMS

12.4.1 Instrument Sequencer (2.1)

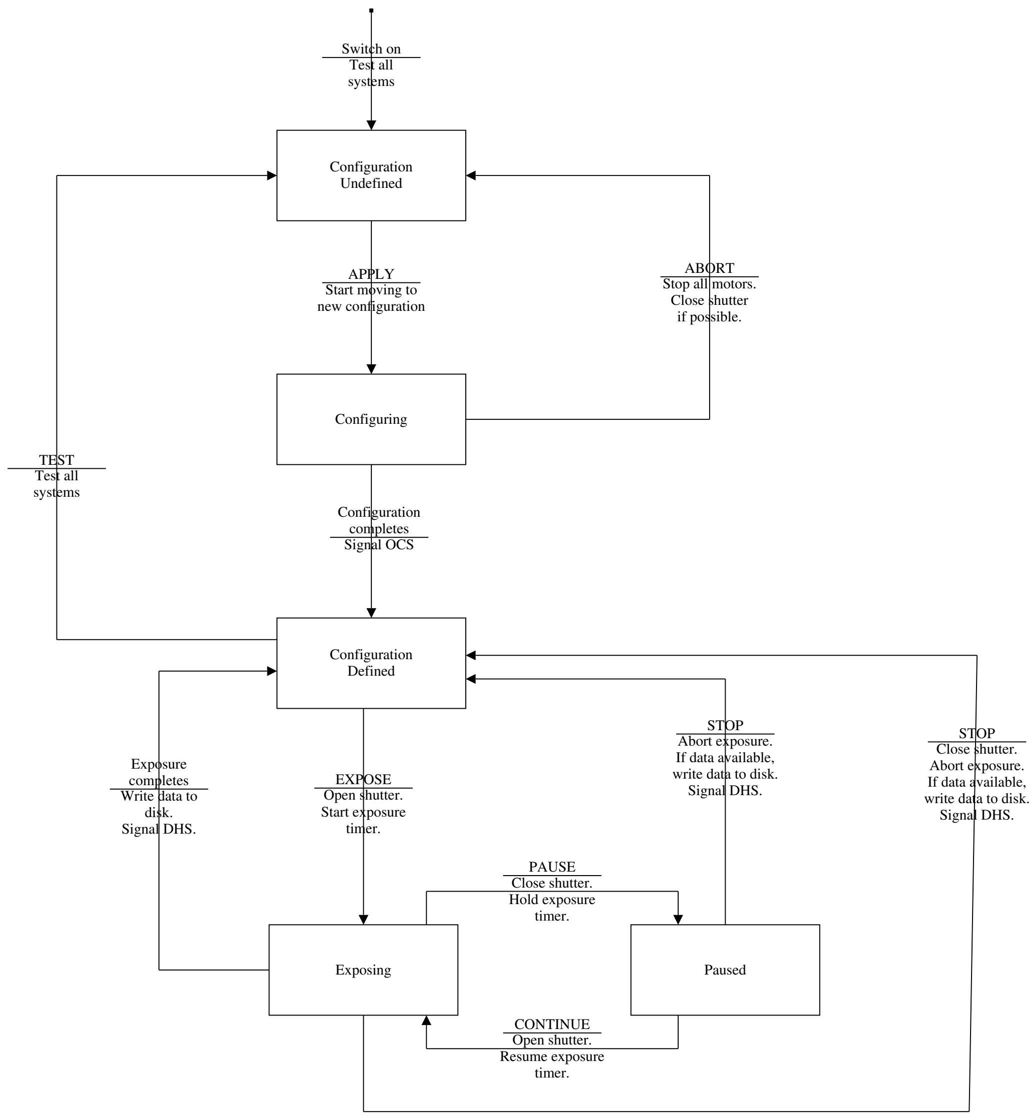
The instrument sequencer accepts commands from the OCS and controls the various ICS subsystems. It also controls the Detector Array Controller. The ICS is not strictly state-driven, but some of the OCS commands cause it to change state, as shown in Figure 12 - 1 on page 12 - 13. The upper case events shown on the diagram are the standard sequencer commands (APPLY, OBSERVE etc…). Lower case events are things that happen after a certain time lag (such as an exposure completing). Commands which are ignored or which do not change the state are not shown on the diagram. The ICS starts in the *undefined* state when first started up.

Commands are sent to the DHS data storage server and DHS quick look server using the protocol described in ICD/3 [26].

Commands are sent to the Detector Array Controller using the protocol described in ICD/7a [30].

Commands sent to the other ICS subsystems are a matter for each instrumentation group to decide.

FIGURE 12 - 1 Instrument Control System State Transition Diagram



12.4.2 Optical Components Controller (2.2)

The optical components controller has two main areas of work to do:

- Controlling the motors and mechanisms.
- Handling I/O signals exchanged with the instrument's electronics.

12.4.2.1 MOTOR CONTROL

The motor controller is responsible for moving the various mechanisms contained in the instrument. In response to a command from the instrument sequencer, it attempts to make the actual instrument configuration the same as the demanded configuration. The Motor Controller is intelligent enough to know how to maintain the instrument in a safe condition while changing configuration (for example it might be prudent to close a shutter or select a blank location on the slit wheel, say, before attempting to move a filter wheel). The Motor Controller is also capable of working out the most efficient path from one configuration to the next (e.g. the best direction to turn a wheel to get to a location).

The bottom layer of any motor controller should use either standard EPICS records [16] or those supplied by the "Standard Control System" work package (ICD/13 [37]).

12.4.2.2 I/O CONTROL

The I/O controller is responsible for the parts of the instrument not covered by the Detector Array Controller and motor controller. It is responsible for such actions as:

- Switching any calibration lamps on and off.
- Setting control voltages (such as the voltage to be applied to a calibration lamp, controlling its color and brightness).
- Sensing the position of switches (such as a main key switch) on the control panel of the instrument.
- Sensing the temperature at various locations within the instrument.
- Sensing and reporting error conditions.

The bottom layer of any I/O controller should use either standard EPICS records [16] or those supplied by the "Standard Control System" work package (ICD/13 [37]).

12.4.3 Internal Data Processor (2.4)

The Internal Data Processor, if this exists, is responsible for carrying out any pre-processing of the data before it is transferred to the DHS. Such pre-processing might include:

- Co-adding many short exposures together.
- Joining one or more exposures together.
- Subtracting beams A and B for chopped data to make one sky-subtracted frame.
- Changing the units of the data.
- Rotating or flipping the data.
- Resampling the data to make the slit vertical or take out distortions.
- “Windowing” capabilities of subarray areas.

Normally an instrument’s Internal Data Processor will carry out operations which are either specific to that particular instrument, leaving the more general data pre-processing to the Detector Array Controller. Their functions may overlap.

The ICS or DC should carry out any data pre-processing whose requirements are too quick for the DHS. (e.g. real-time “shift and add”).

12.4.4 Data Storage (2.5)

If an Instrument Control System needs to preprocess data using an Internal Data Processor (described above), it is then responsible for storing that data to disk. It should send the data to a DHS Data Storage Server using the protocol described in ICD/3 [26].

The Raw Frame Store or Coadded Frame Store will probably be used as a staging area to store data before it is stored to disk. See “Raw Frame Store” on page 12 - 16 and “Coadded Frame Store” on page 12 - 18.

If an Instrument Control System does not need to preprocess data then the Detector Array Controller becomes responsible for storing the data to disk, as described in “Data Storage (2.5)” on page 12 - 24.

12.4.5 Quick Look Display (2.6)

An Instrument Control System may optionally provide a quick look display facility. This can be used to show the latest frame obtained from the Detector Array Controller, or can show a data frame building up from co-added exposures (or both). The DHS Quick Look Server should be used, as described in ICD/3 [26].

Normally an instrument controller's Quick Look Display will include instrument-specific information (such as the number of frames being co-added internally), and the ICS may wish to process the data in an instrument-specific way before displaying it. The Detector Array Controller generating the data may well have a quick look display of its own (See "Quick Look Display (2.3.3)" on page 12 - 25).

12.4.6 Instrument Visualization Subsystem (2.7)

This is an optional subsystem which may be used to display the current state of the instrument graphically. Developers may use a combination of Tcl/Tk, PVWave and/or the EPICS GUI tools (such as the EPICS DM tool). PVWave is useful for graphical information (e.g. for showing the instrument's internal temperature as a function of time, or for showing the slit size, shape and orientation). The EPICS GUI tools are excellent for textual information and, with a little work, can be designed to show crude graphical information as well. A display showing the optical path through the instrument could be created, for example, with text items showing the filters in place and graphical objects showing the position of shutters and flip-in mirrors etc… A good example is the display produced for the WYFFOS instrument on the WHT.

The visualization display should be read-only, and should not be used to control an instrument. Instrument control should be done through the OCS instrument console.

12.5 MAJOR INTERNAL DATA STORES

12.5.1 Raw Frame Store

The raw frame store stores data obtained from the Detector Array Controller. It should be large enough to store at least one copy of the largest frame expected from the Detector Array Controller. In practise it would be prudent to implement this store as a ring buffer, so the Detector Array Controller can continue to take data if there are processing delays in the ICS. Data should be transferred to the store from the DC using the protocol of ICD/3 [26]. Table 12 - 3 gives a possible structure for the raw frame store or coadded frame store:

TABLE 12 - 3 **A possible data structure for the Raw Frame Store or Coadded Frame Store**

Slot No.	Awaiting data flag	Quick Look flag	Data Storage flag	Header pointer	Data Pointer
1	✓	✘	✘	32456	97678
2	✘	✘	✘	32454	97676
3	✘	✓	✓	32455	97677
⋮	⋮	⋮	⋮	⋮	⋮

The columns in this table are as follows:

Slot No. A sequential slot number in the ring buffer.

Awaiting data flag. A flag which is set when the slot in the buffer has been reserved for use and will receive the data from the exposure currently being made: ✓ = flag set; ✘ = flag not set.

Quick Look flag. A flag which indicates when a slot in the buffer needs to be displayed: ✓ = flag set; ✘ = flag not set. The flag remains set until the display completes.

Data Storage flag. A flag which indicates when a slot in the buffer needs to be stored to disk: ✓ = flag set; ✘ = flag not set. The flag remains set until the data is successfully stored.

Header pointer. A pointer to the header assembled by the instrument (if the instrument needs one — usually the OCS takes care of the header).

Data Pointer. A pointer to the data contained in that slot of the buffer. It is recommended that the data be stored in SDS format, to minimize the conversion required when the time comes to transmit the data as an SDS/IMP message (see ICD/3 [26], [3] and [4]).

An ICS could be designed so that the monitoring of this buffer and the acquisition of data run in separate parallel processes, and the storage and display of one observation can take place at the same time the next observation is exposing. The data storage and quick look processes are event-driven, in that they do something when data appears in a buffer rather than when the OCS sends a sequence recipe command. Details are left to the instrument developers.

Instruments which do not receive data from the Detector Array Controller will not need this store.

12.5.2 Coadded Frame Store

The coadded frame store is used when the instrument carries out any internal processing (such as coadding several short exposures together). This store will probably be used as a staging area where data are stored before being sent to the Quick Look Server and/or Data Storage Server using the protocol of ICD/3 [26].

The store could be built using a ring buffer, as shown in Table 12 - 3 on page 12 - 17. Details are left to the instrument developers.

Instruments which do not receive data from the Detector Array Controller will not need this store.

12.5.3 Instrument Control Database

This stores the target, demand and actual configurations of the instrument. For most instruments this will be an EPICS database.

Details are left to the instrument developers.

12.5.4 Instrument Setup Information Store

This is the internal store to which the *instrument setup information* is downloaded to when the Instrument Control System is first started up and the INIT command executed.

Details are left to the instrument developers.

12.6 MAJOR INTERNAL INTERFACES

It is recommended that the interface between the Instrument Sequencer and Optical Components Controller should obey the protocol described in ICD/7a, making it the same as that between the Instrument Sequencer and Detector Array Controller.

The other internal interfaces will be defined by the instrumentation groups.

12.7 THE DETECTOR ARRAY CONTROLLER

The detector array controllers have the following external interfaces:

- Instrument Control System
- Data Handling System:
  --Central DHS
  --DHS Quick Look Server
  --DHS Data Storage Server
- Detector setup information
- Detector hardware
- Fast data output (optional)
- Synchronization Bus
- Time Bus
- Event Bus
- Interlock System

The detector array controllers have the following major internal subsystems. The numbers refer to the process IDs on the data flow diagrams shown in “Decomposition/Dependency Descriptions” in Appendix A1:

- Detector Array Sequencer (2.3.1)
- Detector Array Preprocessor (2.3.2)
- Data Storage Manager (ICS or DC but not both) (2.5[1])
- Quick Look Display (optional) (2.3.3)
- *Plus anything else required by a particular detector controller.…*

The detector array controllers have the following major internal data stores:

- Detector Array Waveform Store
- Raw Frame Store
- Video Display Store (optional)
- Detector Setup Information Store

1. Note that in the data flow diagrams the data is shown being forwarded through the ICS’s “Internal Data Processor” on the way to the Data Storage Server. This data flow covers the cases of data storage from the ICS or DC since “Internal Data Processor” could be a null process.

- *Plus anything else required by a particular detector controller.…*

The detector array controllers have the following major internal interfaces:

- Detector controller/detector reader synchronization signals.
- *Plus anything else required by a particular detector controller.…*

12.8 DETECTOR ARRAY CONTROLLER EXTERNAL INTERFACES

12.8.1 Instrument Control System

The ICS will command the Detector Array Controller using EPICS channel access, as described in ICD/7a [30].

Bulk data will be transferred from the Detector Array Controller to the ICS using a message transmitted using IMP and encoded using SDS. as described in ICD/3 [26].

12.8.2 Data Handling System

12.8.2.1 CENTRAL DHS

History logging information may be sent from any part of the DC to the DHS logging system using the protocol described in ICD/4 [27].

12.8.2.2 DHS QUICK LOOK SERVER

The Data Handling System (DHS) provides the Detector Array Controller (DC) with a server for displaying quick look data. There may be zero or more quick look servers for each instrument. Data are transferred to the Quick Look Server using the protocol of ICD/3 [26].

12.8.2.3 DHS DATA STORAGE SERVER

The Data Handling System (DHS) provides the Detector Array Controller (DC) with a server for storing its bulk data to disk. There is a separate data storage server for each instrument. Data are transferred to the Data Storage Server using the protocol of ICD/3 [26].

Note that *either* the ICS *or* the DC will store data to disk, but not both. See Figure 6 - 5 on page 6 - 11.

12.8.3 Detector Setup Information

This information is part of the *instrument setup information*, and contains information on the capabilities of the detector. The information is specific just to a particular detector, so it is up to each Detector Array Controller to decide how and in what format to read this information. ASCII text files are recommended, as these may be viewed and changed easily. The detector setup information is read when the ICS receives the INIT command, and the information is stored internally by the DACS.

The detector setup information may contain the following:

- The size, name and type of the detector array
- The waveforms used to control the detector array
- A map of the bad pixels on the detector array
- The ideal operating temperature of the detector array
- Information on the sensitivity of the detector array to factors such as background light.
- The names and addresses of various sensor and control ports and (if available) any video display memory
- If the detector controller has an on-board processing capability, information on the layout of the processors
- *plus any other information describing the detector…*

Details are left to the detector controller developers.

12.8.4 Detector Hardware

This interface depends on the detector hardware used, and is the responsibility of the Detector Array Controller development team. Standard Gemini hardware drivers (provided by the “Standard Control System” work package, ICD/13 [37]) should be used where possible, and the detector hardware should interface to VME modules.

12.8.5 Fast Data Output

In some situations, when observing an occultation for example, the data storage server may not be capable of storing the data at a sufficiently high rate. Some Detector Array Controllers may provide a capability for storing these data in an internal format on a large local disk, allowing this to be taken this away for some special-purpose processing. The provision of a fast data output, if this is necessary, is the responsibility of each Detector Array Controller builder. The Data Handling System can only deal with data stored by the DHS data storage server.

However, it is a requirement that all Gemini data be archived, so the fast data dumped in this mode must eventually be transmitted to the DHS Data Storage Server for storage and archive. This transmission can be done off-line at some convenient moment (for example when another instrument is using the telescope beam).

12.8.6 Synchronization Bus

The synchronization bus exists for deterministic data transfer that is not appropriate for transmitting via a network. In the case of the scientific instruments it is used to transfer wavefront information to/from (in general) the array controllers.

12.8.6.1 WAVEFRONT INFO TO INSTRUMENT

Some applications want to shift and add the data frames in order to effectively remove the tip/tilt component of wind shake and atmospheric turbulence. Although this can be done internal to the array controller (by doing image centroiding on a bright object) it is also possible, in the Gemini system, to have the tip/tilt signal come from outside the instrument — for instance from a wave front sensor in the A&G unit.

There are a number of possible sources of tip/tilt (and higher modes) information within the Gemini system. All of these sources are contained in *software slots* in the synchronization bus. The synchro bus is implemented on reflective memory contained on VME cards interconnected with fibers — similar to shared memory.

In the Gemini implementation of this there are EPICS database values which are kept updated from the synchro bus. So to the instrument they appear as EPICS database values attached to a peripheral which occasionally updates the database value.

In general, the Observatory Control System instructs the instrument as to which wave front sensor *slot* it should use.

12.8.6.2 WAVEFRONT INFO FROM INSTRUMENT

There may be some instruments which can generate wavefront info (for instance from image centroiding on a bright object). In this case it is possible for the instrument to write to a wavefront sensor *slot* in the EPICS database. These particular database values are connected so that they write to the reflective memory. In this way an instrument could control the tip/tilt secondary.

Also see ICD/5 [28] and ICD/11 [35].

12.8.7 Time Bus

Time is distributed to all VME systems via a time bus with +/- 5 microsecond accuracy. See “Time Bus” on page 12 - 11 for details.

Also see ICD/9 [33].

12.8.8 Event Bus

The event bus is used (mainly) to distribute the phase of the chopper and position of the telescope in the nodding cycle. See “Event Bus” on page 12 - 11 for details.

Also see ICD/11 [35].

12.8.9 Interlock System

Interlock sensors should be used to prevent the Telescope Control System from slewing the telescope or the instrument rotator when a detector controller has an access door open. A detector controller should be designed so it is not possible to work on it without tripping at least one of the interlock switches.

See “Interlock System” on page 12 - 11 for details. Also see ICD/12 [36].

12.9 MAJOR INTERNAL SUBSYSTEMS

12.9.1 Detector Array Sequencer (2.3.1)

The Detector Array Sequencer responds to high level commands from the instrument sequencer, which are sent using the protocol of ICD/7a [30], and sends the appropriate voltage waveforms to the array to make it reset, integrate, read or whatever at the specified times. The voltage waveforms are obtained from the detector setup information. The Detector Array Sequencer is responsible for ensuring accurate timing between the detector array operations.

The sequencer should accept the same sequencer commands as those accepted by the ICS and described in “Standard Commands” on page 12 - 5. The commands have the same meanings.

12.9.2 Detector Array Preprocessor (2.3.2)

The Detector Array Preprocessor is responsible for reading a data frame from the array, transferring it to memory and signalling the availability of that array to the instrument's detector array interface. It does not command the array to read out, as this is the job of the Detector Array Sequencer. If the detector reads out non-destructively, then it is the responsibility of the Detector Array Preprocessor to combine together the results from the one or more samples read from the detector during the integration[1].

The Detector Array Preprocessor might also carry out the following processing operations on the data:

- Co-adding many short exposures together.
- Sharpening the image, using "shift and add" for example.
- Linearizing the response of the detector.
- Taking out complex electronic signatures.
- Changing the units of the data.
- Shifting, rotating and reformatting the data from multiple arrays/amplifiers into a single frame.

The Detector Array Preprocessor should not make any assumptions about the type of instrument being used.

12.9.3 Data Storage (2.5)

If the Instrument Control System commanding this Detector Controller does not need to preprocess data using its Internal Data Processor, then the Detector Array Controller is responsible for storing the data to disk. It should send the data to a DHS Data Storage Server using the protocol described in ICD/3 [26]. See "Data Storage Server (2.5)" on page 11 - 16.

The Raw Frame Store will probably be used as a staging area to store data before it is stored to disk. See "Raw Frame Store" on page 12 - 25.

If an Instrument Control System does need to preprocess data then the Detector Array Controller should forward the data to the ICS using the protocol described in ICD/3 [26]. Note that data is exported using the ICD/3 protocol regardless of its destination.

1. For example, when an infra-red detector is read non-destructively it is the responsibility of the Detector Array Preprocessor to calculate the slope of the signal in volts per second.

12.9.4 Quick Look Display (2.3.3)

A Detector Array Controller can optionally provide a quick look display facility. This can be used to show the latest frame obtained from the detector, or can show a data frame building up from co-added exposures (or both). A stand-alone real-time display can be created using direct video signals, or a more portable display can be created using a the DHS Quick Look Server. Data should be sent to the Quick Look Server using the protocol of ICD/3 [26]. See "Quick Look Display Server (1.3.2, 2.3.3, 2.6, 4.3.2)" on page 11 - 12. Note that a data array may be sent in small chunks, which can be used to show the image building up as the detector reads out (useful for detectors that read out slowly).

The Detector Array Controller's quick look display can include detector-specific information (such as the portion of the detector being read out). The instrument controller may well have an additional Quick Look Display of its own (as described in "Quick Look Display (2.6)" on page 12 - 15). Because these displays are optional, any duplicated or irrelevant displays may be turned off.

12.10 MAJOR INTERNAL DATA STORES

12.10.1 Raw Frame Store

The raw frame store stores data to be transmitted to the ICS. It should be large enough to store at least one copy of the largest frame expected from the detector array. In practise it would be prudent to implement this store as a ring buffer, so the Detector Array Controller can continue to take data if there are processing delays in the ICS.

12.10.2 Detector Array Waveform Store

The Detector Array Waveform Store contains the voltage waveforms needed to sequence the array. Some controllers may store this information in EPROM and others may download it from disk to RAM. The most efficient situation is to have all the waveforms required for an observing session ready and waiting in this store, though this may not always be possible.

12.10.3 Video Display Store

An optional store used, when a real-time video display is provided, to hold one or more complete images for display. In some cases this could be the same as the Raw Frame Store.

12.10.4 Detector Setup Information Store

This is the internal store to which the *detector setup information* is downloaded to when the Instrument Control System is first started up.

12.11 MAJOR INTERNAL INTERFACES

This internal interfaces will be defined by the detector groups.

12.12 CONSTRAINTS ON DETECTOR ARRAY CONTROLLERS

The difference between “conforming” and “non-conforming” instruments is described in “Varieties of Scientific Instrument” on page 7 - 7. Because a Detector Array Controller is subservient to an Instrument Control System it must be capable of being commanded by a “conforming” ICS, and it must therefore be “conforming” itself. A “conforming” Detector Array Controller must be based on EPICS and must be capable of being commanded entirely using EPICS channel access.

The one exception to this is where a “non-conforming” controller is delivered as part of a “non-conforming” instrument, and the two are used together.

12.13 INFORMATION FROM THE INSTRUMENT WORKING GROUP REPORTS

The “Instrument Working Group Reports Revision 1” [42] contains a description of the scientific requirements for the various facility instruments which are being considered for Gemini. The requirements which these instruments place on the Gemini Control System are considered and summarized below.

Note that the instrument specifications are constantly changing as the instrumentation group consider how best to meet the scientific requirements with the instrumentation budget. This section contains a snapshot for the specifications as they stood in October 1993, with some additional information supplied recently by the instrumentation group. The goal here is to ensure that the ICS software will be able to deal with any of the envisioned instruments.

12.13.1 Telescope Beam Requirements

Many of the Gemini instruments assume there are facilities for controlling the telescope beam. The facilities required are described below.

12.13.1.1 OPTICAL BAFFLES

The telescope needs to be baffled while observing at optical and ultra-violet wavelengths but not at infra-red wavelengths. There will be two kinds of baffles:

- a large baffle, which is fitted and removed by the day crew and not deployable with software. This is used to switch the telescope between optical and infra-red mode. This baffle is controlled by the Acquisition and Guidance subsystem (See A&G, “A&G Baffle Sensing Subsystem” on page 19 - 21).
- a small baffle attached to the secondary mirror which may be deployed by software. This may be used for acquiring a field at optical wavelengths before making an infra-red observation. The small baffle is controlled by the Secondary Control Subsystem (SCS, Chapter 18, 'Baffle Control System').

The requirements on the Instrument Control System are:

- The configuration for the ICS needs to contain information on whether the instrument needs the telescope beam baffled. The OCS can then request the TCS to ensure (via the A&G and SCS) that the correct baffling is in place.

12.13.1.2 FIELD CURVATURE CORRECTOR

Wide field optical instruments need a corrector to remove field curvature but narrow field instruments do not. There is a deployable field corrector controlled by the A&G unit (See “A&G Field Curvature Correction Subsystem” on page 19 - 21).

The requirements on the Instrument Control System are:

- The configuration for the ICS needs to contain information on whether the instrument needs the field curvature corrector. The OCS can then request the TCS to ensure (via the A&G) that it is deployed as required.

12.13.1.3 ATMOSPHERIC DISPERSION CORRECTOR

This is required for broad-band high-spatial-resolution optical instruments and it is *essential* for the multi-object spectrograph. It is controlled by the A&G unit (“A&G Atmospheric Dispersion Compensation Subsystem” on page 19 - 21) and could be combined with the field curvature corrector.

The requirements on the Instrument Control System are:

- The configuration for the ICS needs to contain information on whether the instrument needs the atmospheric dispersion corrector. The OCS can then request the TCS to ensure (via the A&G) that it is deployed as required.

12.13.1.4 CALIBRATION SOURCES

Some instruments need calibration sources, such as arc lamps and flat-field sources. HROS may require an optical reference beam for flexure correction, and some polarimeters require a rotating wave plate or a wire grid for calibration. These calibration sources are provided by the A&G unit. See “A&G Calibration Subsystem” on page 19 - 21 for details.

12.13.2 The Field Acquisition Camera

The current model for the field acquisition camera detector is described in Table 12 - 4.

TABLE 12 - 4 **Recommendations for the field acquisition camera detector.**

Property	Recommendation
Detector	Thinned CCD
Individual chip size	2048 x 2048 pixels
Field of view	2 arcmin x 2 arcmin
Read-out time	< 0.5 seconds
Implied maximum data rate	> 8.4 million pixels per second
Binning factor	probably 4 x 4
Resulting size	512 x 512 pixels
Resulting continuous data rate	probably > 525,000 pixels per second

12.13.3 Multi-Object Optical and Ultra-Violet Spectrograph

The current model for the MOS instruments is shown in Table 12 - 5.

TABLE 12 - 5 **Requirements for the Multiobject Spectrographs**

Property	Requirement
Field Size	That which could be filled with 4096x4096 CCD (>5.5 arcmin)
Image Scale	0.08 - 0.10 arcsec/pixel
Slit Sampling	2.5 times the image pixel size
Imaging mode	sufficient to support MOS mask production
Spectral Resolution	up to 10,000
Integral Field Mode	sub-apertures with diameters 2.5 times the pixel scale, 8 arcsec FOV
Slits	capability to manufacture precision slit masks

In addition, the MOS have the following optional/desirable features:

- extension of wavelength coverage

--to UV atmospheric cut-off and to IR 1.6 micron for Gemini South

--to IR, 1.6 micron and to UV atmospheric cut-off for Gemini North

- a high spatial resolution integral field mode to support adaptive optics at >700nm.

  This instrument is designed to produce many simultaneous spectra by imaging the sky onto a slit mask. Each slit in the mask is connected to the spectrograph by an optical fibre. The instrument may also use a close-packed array of optical fibres, an array of lenslets, or a single, variable-width, precision long slit in place of the slit mask. Each observation with the instrument will require the following stages:

1. Imaging of the field of view.
2. Location of the objects of interest in the field, and the calculation of their astrometry to a relative accuracy of 0.01-0.02 arcseconds).
3. Manufacture of the slit mask.
4. Loading and deployment of the slit mask (by means of a rapid interchange device).
5. Spectroscopy with the instrument using the slit mask.

It is expected that all of the above stages will take place during one night. The instrument will operate in one of the following modes (in each of which it can be used for imaging or spectroscopy):

- High Resolution mode with 0.04 arcseconds per pixel and a 2 arcminute field of view.
- Wide field optical mode, with refractive optics and a 7 arcminute circular field.
- Wide field ultra-violet mode, with reflective optics and a 6 x 2 arcminute field.

  In practise the last two modes may be mutually exclusive and require separate instruments.

The software requirements generated by this instruments are:

- The OCS must be able to schedule imaging observations with the instrument in advance of spectroscopy observations, leaving enough time for the slit mask to be manufactured.
- The on-line data reduction system should be capable of measuring the positions of the objects in the field accurately and despatching this information to the slit mask cutter.

  It must be possible to make other observations while this is taking place.
- There has to be a mechanism for informing the ICS for this instrument when the slit masks have been loaded, and in which locations they may be found. The ICS will control the rapid interchange device.
- The data reduction recipe used for this instrument will vary with the mode of operation.

- The data reduction system should at least be capable of showing the observer the combined 2-D spectrum. It should also be able to extract crudely and display some of the individual spectra.

  It is assumed that detailed field distortion correction and extraction of the individual spectra will be carried out off-line at the Observer's home institution.
- The Cassegrain Rotator System must take into account any cable-wrap requirements imposed by this instrument, depending on where the spectrograph is situated and where the optical fibres go.

12.13.4 High Resolution Optical Spectrograph

The exact nature of this instrument is to be decided. Its salient requirements are given in Table 12 - 6

TABLE 12 - 6 **Requirements for High Resolution Optical Spectrograph**

Property	Requirement
Focal Station	Cassegrain focus
Throughput	>10% at R=50,000 and 500nm, goal is 15%
Resolution	In range of 30,000 to 80,000, with >120,000 as a 2nd priority
Highest resolution mode	Implemented in a lab in the telescope pier
Fiber Feed	To CFHT Telescope Coude Spectrograph on Mauna Kea

- It will operate either at the Cassegrain focus or from a spectrograph laboratory fed by optical fibres.
- It will have two resolution modes, but a change between these will be a daytime operation.
- It needs a rotatable slit mechanism interchangeable with an image slicer.
- The instrument needs a high degree of stability, which will be especially difficult if it is operated at the Cassegrain focus.
- If the instrument is located at the Cassegrain focus, the flexure in its optics *might* be taken out using an on-board optical alignment monitor, which would monitor a reference beam and continuously adjust one of the components (probably the collimator). The reference beam would need to be provided externally to the instrument.

  The optical adjustment may be made intermittently by interrupting the science beam every now and then.
- The instrument may need to be calibrated off the telescope on the dome floor or in the spectrograph laboratory.

- The instrument has special user interface requirements as follows:

  --The user interface should include access to the acquisition/identification and guiding images, instrument status and control, and the observing log.

  --The interface should allow a quick look at incoming data to assess its quality and signal to noise, and to identify problems (fringing, detector failure, poor focus etc…) immediately.

  --The interface should provide a means of transferring data onto the permanent storage medium.

  --A simulated display of the echellogram, with interesting spectral lines marked, together with an overlay showing the detector coverage of the echellogram, would make a good interface. The observer should be able to position detector window until the desired wavelengths are covered.

  --The user interface should be supplied to the observer before a run so it is possible to learn how to use it.

- The instrument generates data frames containing spectra made of several overlapping orders.

The requirements placed on the control software are:

- If the HROS is situated at the Cassegrain focus, the OCS must be capable of keeping the telescope still while HROS is taking calibration observations. It must also prevent the telescope slewing between a calibration observation and astronomical observation with HROS.
- If the HROS is fed by optical fibres, the Cassegrain Rotator System must take into account any cable-wrap restrictions imposed by these fibres.
- The ICS for HROS will be responsible for controlling the rotatable slit mechanism and image slicer.
- The ICS will also be responsible for controlling the instrument's optical alignment monitor, if one is used. However, the OCS must know when to switch the external optical reference beam on and off.

NOTE: If the reference beam is switched intermittently during an exposure the OCS will not be able to command the switching quickly enough. In this situation the ICS will need to communicate directly with the TCS.

It is recommended that a standard set of calibration observations be defined for HROS. The data reduction system will need a recipe for reducing HROS spectra.

- The data reduction system needs tools for combining overlapping orders, and providing "consistent and reasonable continuum placement". It is suggested that this be available at the telescope for evaluation.

The on-line data reduction system needs to be capable of combining overlapping orders in a crude way. The external data reduction tools need to be available to allow the Observer to reduce the data properly by hand.

12.13.5 Infra-red Imaging and Arrays

The recommendations for the detector to be used with the 1-5 micron infra red imager are listed in Table 12 - 7.

TABLE 12 - 7 **Recommendations for the 1-5 micron infra-red imager detector.**

Property	Recommendation
Detector	InSb array
Individual chip size	1024 x 1024 pixels, 27 micron pixels
Pixel size	0.02 - 0.1 arcsec
Field of view	20.5 x 20.5 - 102 x 102 arcsec
Read-out time	Not given. Assume < 100ms to allow 10Hz chopping
Implied maximum data rate	> 10.5 million pixels per second

The instrument will have the following features:

- Two filter wheels of 20-30 slots for filters, grisms, and polarizers
- A cold focal plane wheel with various slits and stops
- A low resolution grism
- A pupil imaging capability for commissioning and engineering
- Ability to operate from both side- and upward-looking ISS ports
- Closed cycle cooling
- Image quality >90% encircled energy within a single detector pixel during a one-hour observation

In addition, the instrument design has the following goals:

- Capability to support a 2x2 mosaic, 1024x1024 arrays
- Incorporating a spectroscopic capability
- Incorporating a third filter whell
- Incorporating a coronographic mask

It is expected that the detector controller and imager will be separate instruments.

This instrument has the following software implications:

- The ICS for the infra-red imager will control the filter wheels, slit/stop wheel and grism.
- The Quick Look Display can be used to show the pupil imaging.
- Although not mentioned in the working group report, rapid choreography is required between the telescope and instrument, because infra-red observations often use chopping and nodding individually or simultaneously. The maximum chop frequency (mentioned in the Secondary Control System specification) is 10 Hz and the maximum nod frequency 0.2 Hz.
- Infra-red observations generally require a rapid read-out. It is assumed the ICS will have an on-board processor which will coadd the individual exposures. When chopping there will be a choice of 2 kinds of outputs:

  --A single coadded A-B subtracted frame.

  --Two separate coadded frames for beam A and beam B.

  The data handling system should be capable of handling both types of frames.
- There may be a requirement for the data reduction system to locate and extract low resolution spectra from the grism images.

In addition to the 1-5 micron imager there may be a 5-30 micron instrument, but this is to be decided. Such an instrument would have similar software requirements, but would require much more rapid readout. It is assumed its ICS will have more powerful on-board processors to cope with the increase in data rate. The requirements for the 5-30 micron instrument are shown in Table 12 - 8.

TABLE 12 - 8 **Mid IR Imager Requirements**

Property	Requirement
Plate Scale	~0.13 arcsec/pixel
Field of View	As set by array
Instrument Background	<1% effective emissivity
Wavelength Range	5 to 25 micron with extension to 30 micron as a goal
Filters	one cold filter wheel with 20-30 slots for filters, etc.
Operating Modes	Both side- and upward-looking
Cooling	Closed Cycle
Tip-Tile Correction	Achieved by sensing in the peripheral guide field

12.13.6 Infra-red Spectroscopy

There may be two or three infra-red spectrometers operating on Gemini. The capabilities required are:

- A near infra-red (1-5 micron) spectrometer with a 1024 x 1024 pixel (InSb?) detector array. The requirements for this instrument are given in Table 12 - 9

TABLE 12 - 9 **Requirements for near infra-red spectrometer**

Property	Requirements
Wavelength Range	0.9 - 5.5 microns
Detector Format	1024x1024
Spectral Resolutions	the lowest dispersion mode must allow each atmospheric window to be covered across the 1024 array (R~2000) and an intermediate dispersion mode must be provided that allows the observations of key astrophysical lines between the atmospheric OH lines (R>=8000)
Pixel Scale	should be capable of exploiting 0.1 - 0.2 arcsecond slits with sampling ~0.05 arcsec/pixel
Slit Length	>= 50 arcsec

This may split into two instruments with different resolving capabilities.

- A mid infra-red spectrometer with a 256 x 256 pixel (AsSi or GaSi?) detector array.Features of these spectrometers will be
- Pixel scales of 0.05 to 0.15 arcsec per pixel should be selectable in the near infra-red spectrometer while the instrument is cold.
- There should be an imaging capability for target acquisition, probably using a flip-in mirror.
- There should be a polarizing prism for spectropolarimetry.
- The near infra-red spectrometer may have a multi-object capability using either:

  --about 20 moveable slitlets imaged onto the detector; or

  --a bundle of optical fibres channeling light to a single long slit.
- Accurate guiding is required to keep an image on the narrow slit. The instrument will use a dichroic mirror to direct the optical portion of the incoming beam into its own on-board wavefront sensor.
- The infra-red spectrometers will have their own on-board calibration units.

The software implications are as follows:

- The ICS for the infra-red spectrometer should be responsible for controlling all its internal components, including the flip-in mirror, the grating, polarizing prism, dichroic mirror and slit assemblies.

  It may be desirable for the on-board calibration unit to have its own separate controller.
- The output from the on-board wavefront sensor will need to be channelled to the Adaptive Optics controller (AO) in the TCS, or to the A&G controller if the AO is not mounted.
- The multi-object modes imply one of the following software constraints:

  --If 20 moveable slitlets are used, some software has to convert the location of the objects to be observed into positions for the slits. The algorithm to do this has to be supplied by the instrument builder, but may also require knowledge of the optical path through the telescope. The ICS should control the slitlets, but there needs to be a mechanism for peaking up the signal involving the TCS and DHS.

  --If optical fibres are used the software requirement becomes similar to the optical and ultra-violet multi-object spectrograph. If a mask is used to position the optical fibres there will be a much longer delay between creating the mask and observing, because the mask will have to be cooled.
- Just as with the infra-red imagers, it is assumed any high data rates are handling by processors within the ICS, which will coadd exposures before passing them on to the DHS.
- The Data Reduction system will need to be able to handle the multi-object spectra.
- The Gemini System must be able to handle the observing scenarios proposed by S.M.Pompea in Appendix D of the infra-red Spectroscopy working group report. One example here is the use of dome flats to generate the linearization coefficients. This implies that one set of observations (e.g. dome flats) may be used for several different purposes (e.g. use as a flat field and as a linearization calibration), and the DHS must be able to handle that possibility.

12.13.7 High Resolution Wavefront Sensor

The high resolution wavefront sensor is part of the Acquisition and Guidance (A&G) unit (see "Details of the Acquisition & Guiding System" in Chapter 19), but its requirements are sufficiently different from the rest of the A&G unit that it will be discussed here as a separate instrument.

The high resolution wavefront sensor consists of a 20 x 20 element Shack-Hartmann lenslet array and a detector. It is deployed in the centre of the science field and so precludes observing while in use. The purpose of the high resolution wavefront sensor is to recalibrate the primary mirror model by making a series of wavefront measurements at different elevations and generating a primary mirror support look up table. The look up table is stored in a file which can be downloaded to the M1 (Primary Mirror) Control System (see “Details of the M1 Control System” in Chapter 16).

The data processing procedure for each high resolution wavefront sensor exposure is as follows:

1. A wavefront sensor exposure is taken. An exposure time of 20-60 seconds is used to eliminate the effects of seeing.

   The exposure contains an array of up to 20 x 20 images of a single star taken through a lenslet array. Unwanted star images will have been removed with an iris.
2. The image is divided by a flat-field (optional).
3. The image is thresholded and the centroids of all the spots determined.
4. The spot centroids are compared with those measured earlier from a standard lamp exposure and the shifts in position calculated. Care needs to be taken in matching up the spots to ensure they are not shifted by an integral number of rows or columns (blocking off one of the lenslets can help here).

   It is not yet clear how often a new standard lamp exposure would be taken.

   It must be possible to specify the standard lamp exposure explicitly.
5. The centroid shifts are converted into wavefront deviations, which are then converted into an M1 force offset distribution (using a polynomial or spline interpolation). Zernike polynomials cannot be used here because of the high resolution of the wavefront information, and because it would be difficult to represent the effect of a single force actuator sticking up with a Zernike.
6. The force offsets are stored, together with the elevation.
7. Repeat steps 1 to 6 at different elevations.
8. Finally a look-up table of force offsets against elevation is generated.

The details of the above procedure are to be decided by the A&G group.

The procedure will be used extensively during commissioning of the telescope and will be repeated typically once or twice per night.

The high resolution wavefront sensor will operate like an instrument. It may display the images taken using the Quick Look Server and it should store its data to disk using the Data Storage Server. The Data Handling System will be responsible for processing the data generated by the high resolution wavefront sensor using a data reduction agent and the recipes and rules technique described in “On-line Data Reduction System” on page 6 - 16. The question of the data reduction package to be used is yet to be decided, following an evaluation of IDL-SH. The data processing work will be divided between the A&G and DHS groups as follows:

- The A&G group will define the recipes and produce the data reduction package needed to process the high resolution WFS data (e.g. IDL scripts plus any necessary special-purpose data processing subroutines).
- The DHS group will provide the data reduction agent which sequences the above data reduction package.

Processing the high resolution WFS data automatically through the on-line data reduction system will only work if the telescope force offsets are close to their nominal set-up. If the force offsets are incorrect it is possible for the spot centroids to be mis-matched by a whole number of rows or columns (e.g. a spot in column 1 of one measurement being matched to a spot in column 2 of the next). To ensure this does not happen, it is necessary to process the high resolution WFS data off-line and match the spots by eye. This needs to be repeated as often as necessary to keep the telescope force offsets close enough for the daily on-line processing. Such off-line processing is entirely the responsibility of the A&G group.

12.14 REQUIREMENTS ON SPECIFIC DETECTOR ARRAY CONTROLLERS

The Gemini project is considering using two different detector array controllers (which currently exist) and may need a third to handle the A&G, the calibration sensor and AO. The following sections describe how these controllers may interface to the Gemini system, and what changes would be needed to the existing controllers to implement the interface.

12.14.1 Arcon

The current Arcon consists of a set of transputers interfaced to an Sbus board resident in a Sun workstation.

In the Gemini system the Arcon would be controlled via a VME interface to an EPICS system. This will require selection of appropriate VME hardware, creation of an EPICS driver, and interfacing of the required Arcon system functionality to this environment.

The data acquisition part of Arcon can still use the transputers and Sun, provided the data are transmitted to the ICS or Data Storage Server using the protocol of ICD/3 [26].

12.14.2 ALICE

ALICE consists of a set of T805 transputers on TRAM modules interfaced to a Sun workstation via TCP/IP over Ethernet.

In the Gemini system ALICE would be controlled via a VME interface to an EPICS driver. This will require the replacement of the TRAM modules with VME modules (perhaps incorporating the T9000 transputer), the creation of an EPICS driver and the incorporation of the ALICE system functionality into this environment.

12.14.3 General requirements

It would be beneficial if all the transputer-based controllers used a common set of transputer VME modules, to minimize the number of spare boards required.

Any new EPICS drivers should be developed via the "Standard Control System" subcontract (See "What is Provided by Gemini" on page 7 - 20).

13 Details of the Telescope Control System

13.1 Function of The Telescope Control System

Precise control is critical to the pointing, tracking and imaging of modern telescopes. The Gemini TCS is based on that used on the Anglo-Australian Telescope, Keck Telescope, and the William Herschel Telescope (amongst others). All of these have met stringent performance targets and their conceptual framework is adequate for the yet more demanding requirements of the Gemini project.

Each Gemini Telescope is an 8-meter telescope with an alt-azimuth mount. It has two focal stations: Cassegrain and a high resolution lab located in the telescope pier. The Telescope Control System (TCS) is the software required to point the telescope at an astronomical object, to track and focus it accurately, to make any offsets necessary, and to maintain both the optical alignment and optical figure. The TCS is also responsible for controlling the wind flushing of the enclosure and the temperature of the primary mirror surface.

13.2 Telescope Control System Overview

13.2.1 External Interfaces

The TCS has the following external interfaces:

- Observatory Control System
- the TCS Console (provided through the OCS)

- the OCS Screens System
- Hardware Setup Information
- Data Handling System
- TCS Subsystems

13.2.2 External Bus Connections

The TCS is connected to the following buses.

- Synchronization Bus
- Interlock System
- Time Bus
- Event Bus

13.2.3 Internal Subsystems

The TCS is composed of the following major internal subsystems:

- TCS State Machine, which is composed of
  --TCS State Processor
  --TCS Command Processor
  --TCS Screen Handler
- Pointing Engines
  --Telescope Pointing
  --A&G Wave Front Sensors (WFS) #1 and #2 Pointing, calibration WFS Pointing
  --Adaptive Optics WFS Pointing
  --Scientific Instrument(s) On Board WFS Pointing
- Optics Control Engine
- Wind/Thermal Management Engine
- TCS Handset

13.2.4 Internal Data Stores

The TCS has the following major internal data stores:

- TCS target/demand/actual configuration
- TCS screen information
- Setup and Initialization info

- Restore info (needs careful filtering if from just before last power fail)
- TBD

13.3 INTERFACE TO OBSERVATORY CONTROL SYSTEM

The TCS must provide a clean interface to the OCS. This interface is based on a *database* model. That is, the OCS issues directives to the TCS by changing attributes in the TCS' database and issuing a TCS *event flag* appropriately. Further, the OCS obtains status information from the TCS by *registering interest* in specific TCS database entries. This means that the OCS and TCS designers must agree on the *functionality* of all communications, but not the *syntax or parameters*, as the underlying database access methods are common to both systems. The TCS database embodies the interface of the TCS to external systems.

ICD 1 [24] describes the command interface between the OCS and the TCS and ICD 2 [25] describes the status/alarm interface.

The basic model is that the OCS maintains target values for the TCS and the TCS maintains a set of demand values for its subsystems. It is the TCS's job to keep the actual values as close as possible to the demand values.

13.3.1 OCS Sequencer

In addition to *primitive control directives* discussed below, the TCS must respond appropriately to the following directives from the OCS:

- **TEST**

  Test all the components of the telescope that are capable of being tested via software and report any failures. This command is used at least once at the beginning of each night to check that the telescope is functioning correctly before starting to observe.

  This command can be executed from any state except *configuring, exposing* and *paused*, and it moves the TCS into the *idle* state (which is the state the TCS has when first switched on).

- **CHECK**

  The telescope is provided with a target configuration, and it should check that it is capable of moving to that configuration. For example, is the specified target above the horizon, or is the configuration beyond its capabilities? A status is returned to indicate the success or failure of this command. The CHECK command should not affect the present configuration of the telescope.

  This command can be executed from any state except *configuring, exposing* and *paused*, and moves the TCS into the *initialized* state.

- **APPLY**

The telescope should move to the requested configuration, positioning mount, secondary etc... as requested. The OCS target values become TCS demand values.

This command can only be executed while the TCS is in the *initialized* state (and it is the responsibility of the TCS to check this is so). The command moves the TCS into the *ready* state. The TCS does not indicate to the OCS that it is in the *ready* state until all subsystems are *in position*.

In the *ready* state the telescope and its subsystems are adjusted to predicted positions. This is commonly referred to as open loop tracking.

- **VERIFY**

  Allow the operator to look at the telescope's console and verify that the correct configuration has been achieved. The operator should be able to interact with the telescope using the console and make any necessary adjustments to the settings. These adjustments should be reflected in the "current configuration" information stored in the Telescope Control System's internal database.

  This command can be executed at any time and does not change the current state of the TCS. It is up to the operator to ensure that sensible adjustments are made.

- **ENDVERIFY**

  End the verify phase.

- **GUIDE**

  This command can only be executed when the TCS is in the *ready* state (and the TCS must check this). The command moves the TCS into the *guiding* state. The TCS does not indicate to the OCS that it is in the *guiding* state until all subsystems have indicated that they are *in position*. The *in position* values for *guiding* are smaller than for *ready.*

  In the *guiding* state the TCS and its subsystems are adjusted to measured positions based on wave front sensor information from the telescope focal plane. This is commonly referred to as closed loop tracking - due to the number of different mechanisms in the TCS we distinguish between:

  --closed loop tracking - maintaining the telescope line of sight by correcting the mount

  --tip/tilt guiding - maintaining the telescope line of sight by correcting the secondary

  --fast focus - maintaining the telescope focus by correcting the secondary

  --active optics - maintaining the delivered wave front by correcting the primary mirror figure

  --adaptive optics - maintaining the delivered wave front by correcting a small deformable mirror

Monitor the current configuration and report any changes or any alarms. If at all possible, the TCS should execute this command all the time, but if this is not possible, the monitoring is switched on when the sequencer recipe reaches the "guide" stage. It may be that the TCS has different *monitor* capabilities depending on the current state.

- **ENDGUIDE**

  Stop guiding. The TCS moves from the *guiding* state to the *ready* state.

- **OBSERVE**

  Check that the current performance is within desired specifications (for instance the observer might wish to wait for the primary mirror figure to go through several cycles of adjustment before starting the exposure).

  This command can only be executed while the TCS is in the *guiding* state. It moves the TCS temporarily into the *exposing* state. During this state the TCS continues to monitor the current configurations as during the *guiding* state. When the exposure completes the TCS moves back into the *guiding* state.

- **PAUSE**

  This command can only be executed in the *exposing* state, and sends the TCS into the *paused* state. The command exists because there may be some operations (TBD that the TCS can only carry out when the current exposure is paused.

- **CONTINUE**

  This command can only be executed in the *paused* state, and sends the TCS back into the *exposing* state.

- **STOP**

  This command indicates that an exposure has been cancelled. It can be executed in the *paused* or *exposing* states and moves the TCS back into the *guiding* state.

- **ABORT**

  This is an emergency stop command. Its meaning is independent of the current state of the TCS. The TCS stops at its current position and puts the brakes on - it ramps down the drives before applying the brakes.

  This command can be executed from any state, and sends the TCS into an *aborted* state. An INIT command is then necessary before observing can be resumed, though it may be prudent to try another TEST command first.

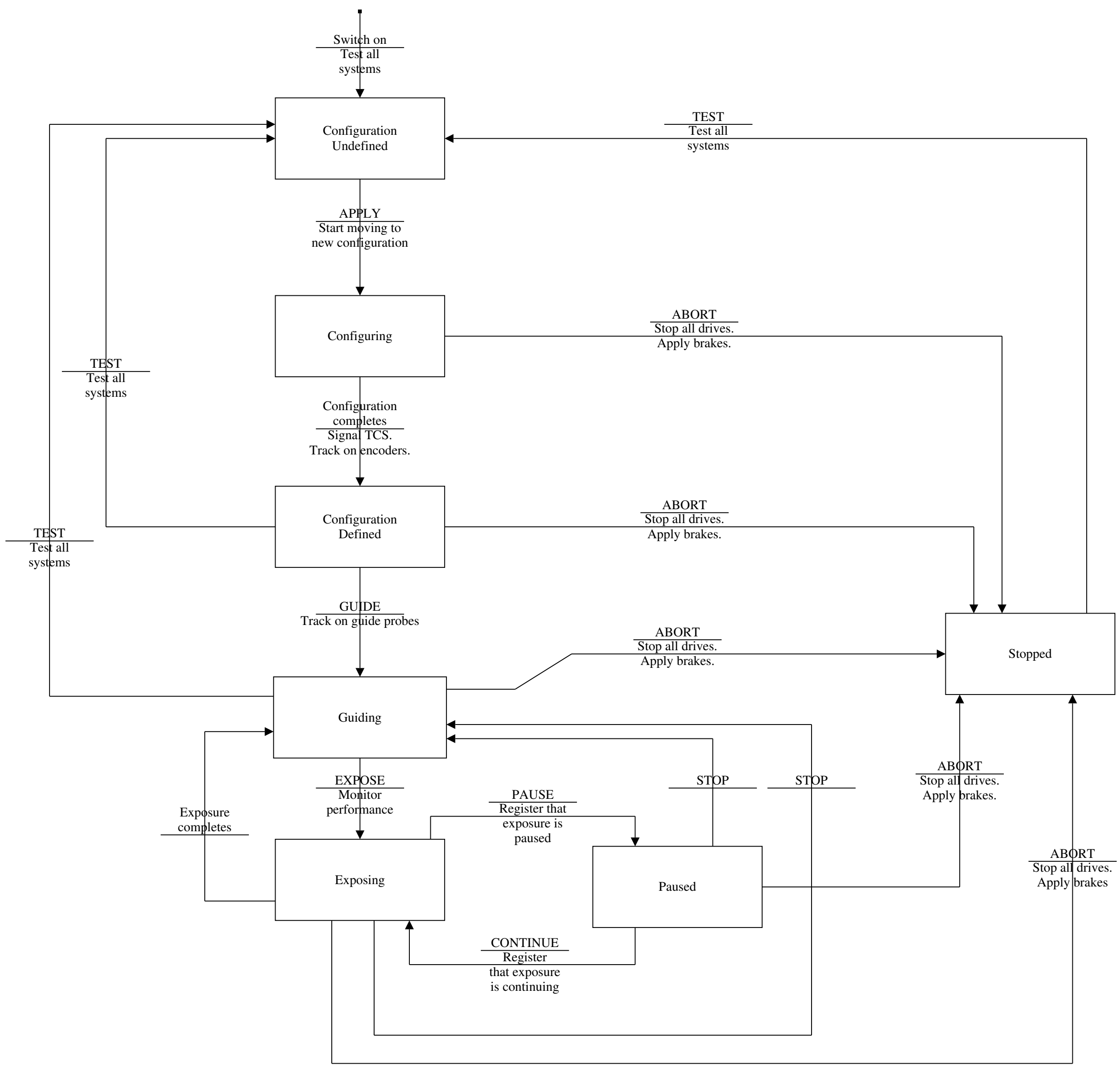
- **PARK**

  The telescope adopt a safe configuration and parks itself at the zenith; pins in, brakes on.

  This command can be issued in any state except *configuring, exposing* and *paused.* The TCS state is changed to *idle.*

There is an EPICS database entry for OCS sequencer commands and an EPICS database entry representing the current state of the TCS. When the OCS updates the sequencer command entry the TCS, if it is valid transition, changes from its current state to that required by the sequencer command. If the TCS requires information to make this transition it retrieves this information from the OCS observing database.

FIGURE 13 - 1 Telescope Control System State Transition Diagram



13.4 INTERFACE TO THE TCS CONSOLE

It is possible for external systems (such as the OCS) to obtain fine-grain control over the TCS through a similar database interface. There are directives that allow the setting and checking of individual parameters within the TCS database. These primitive directives allow the OCS to monitor status information within the TCS, construct custom control screens, etc.

13.4.1 TCS Console Screens

FIGURE 13 - 2 **TCS control console**

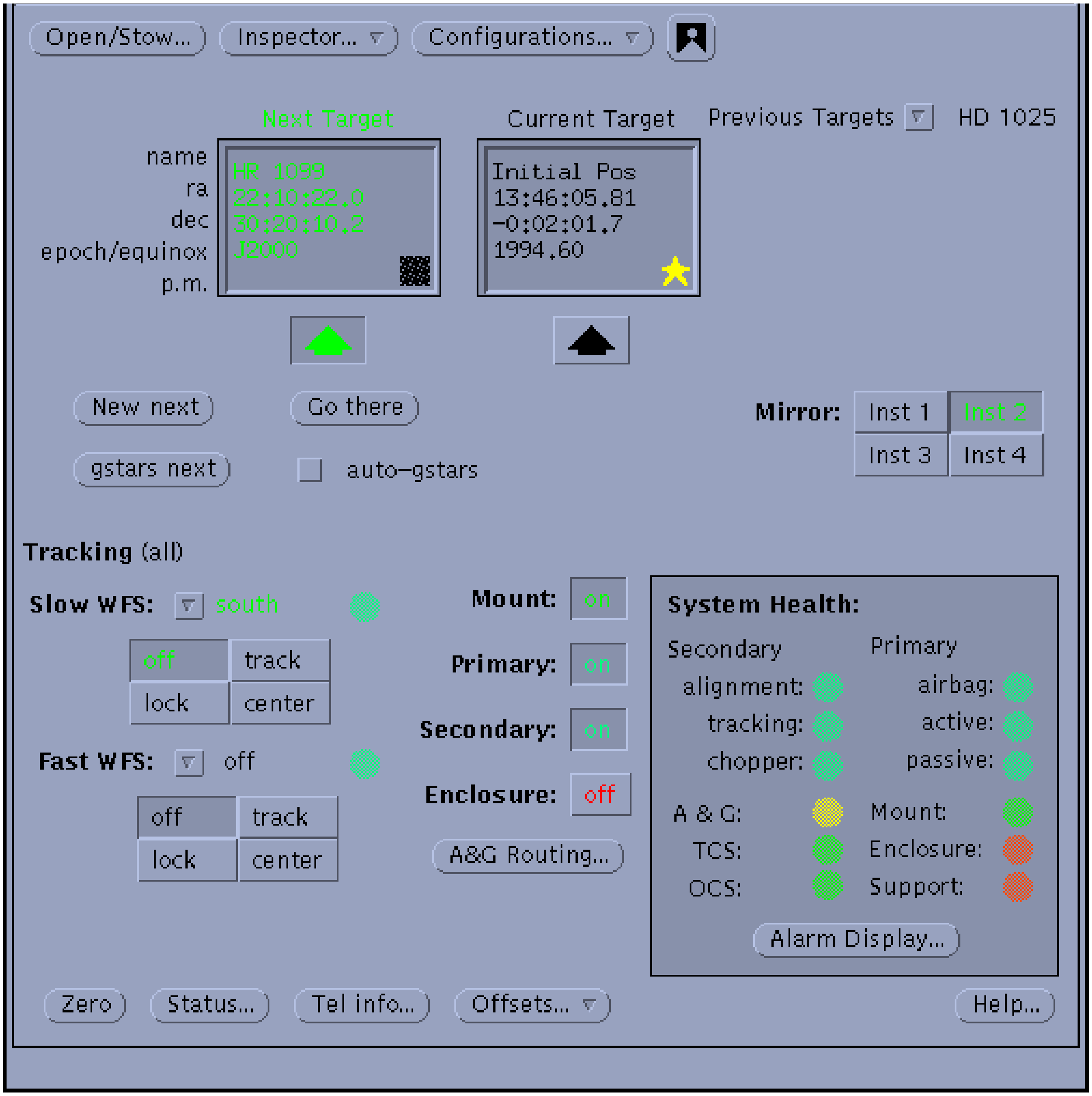


FIGURE 13 - 3 **TCS WFS routing console**

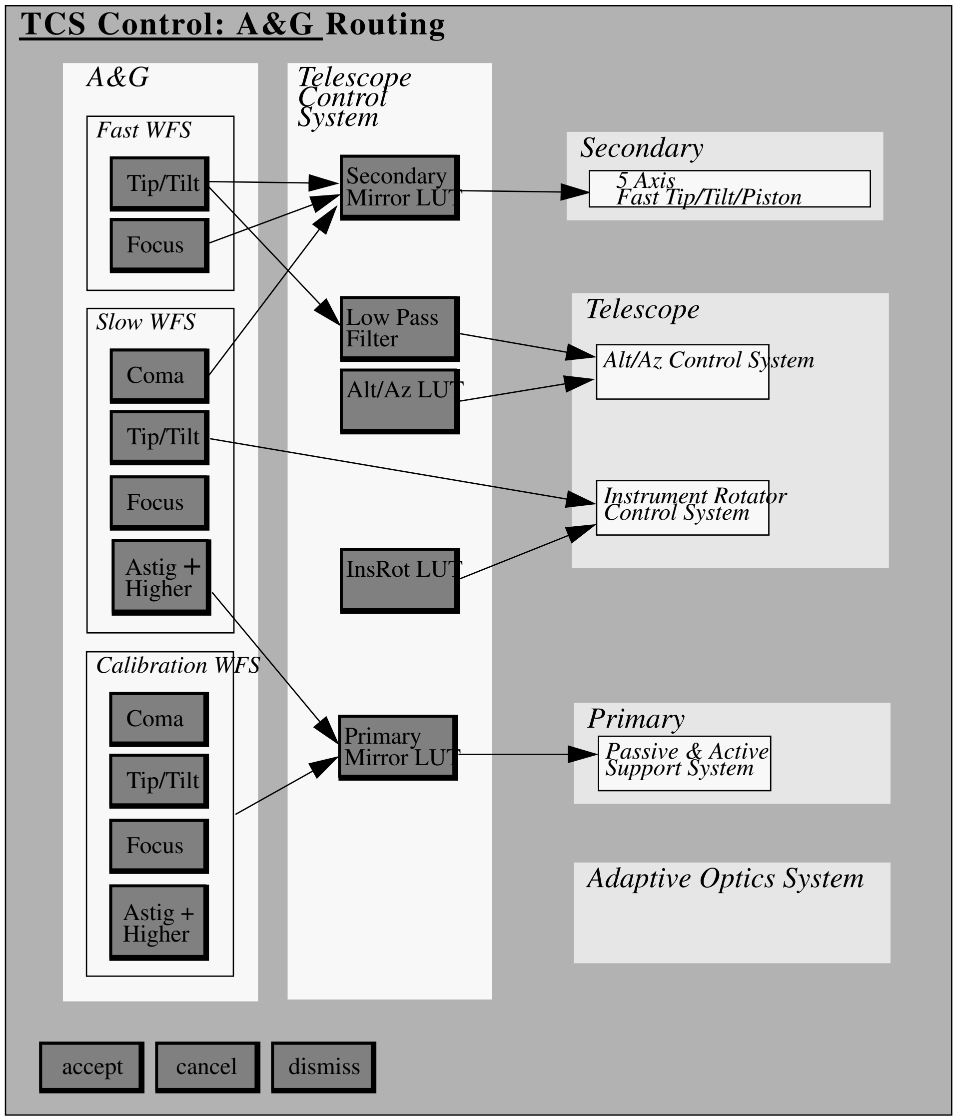


FIGURE 13 - 4 **Environmental monitor console (weather)**

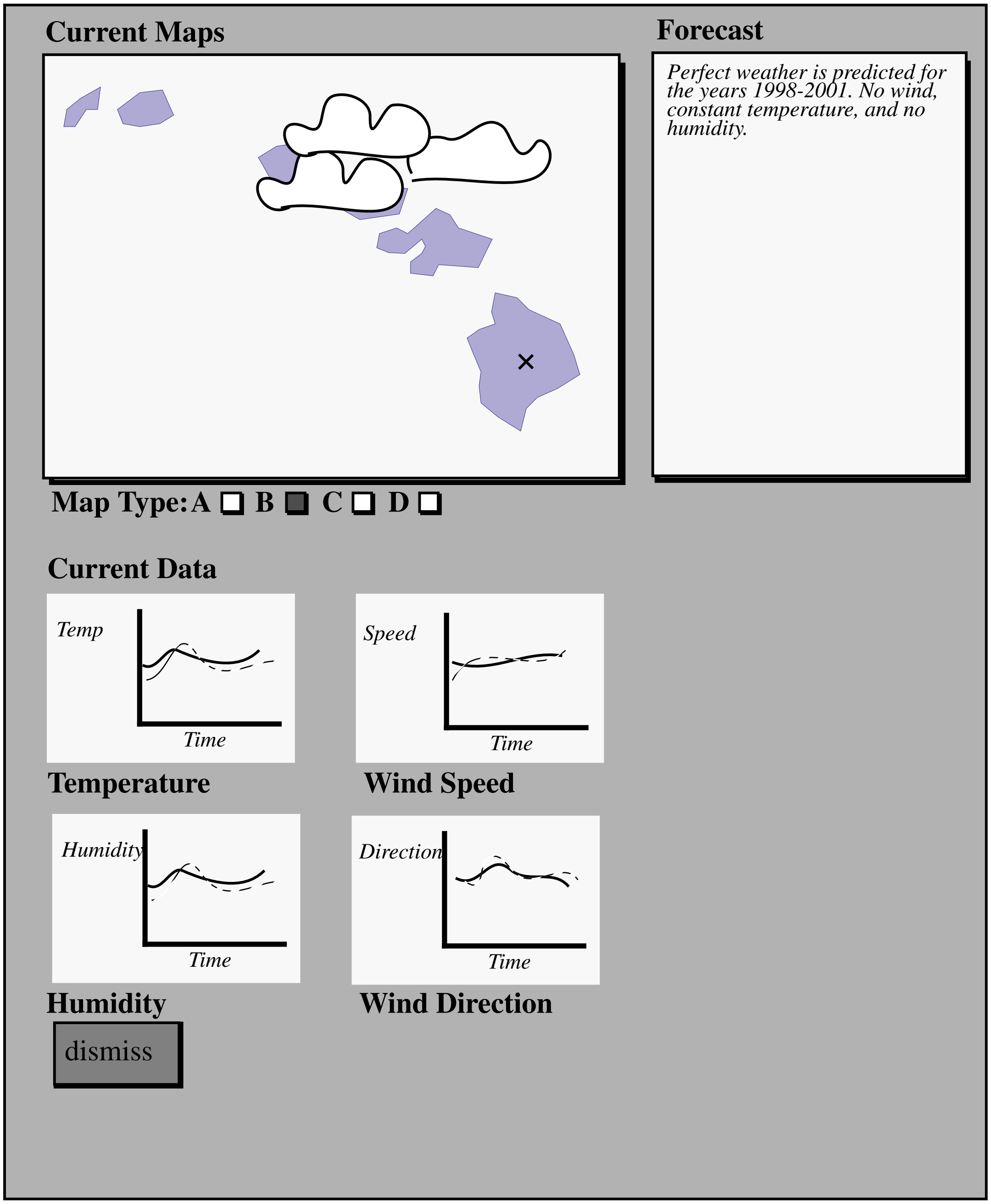
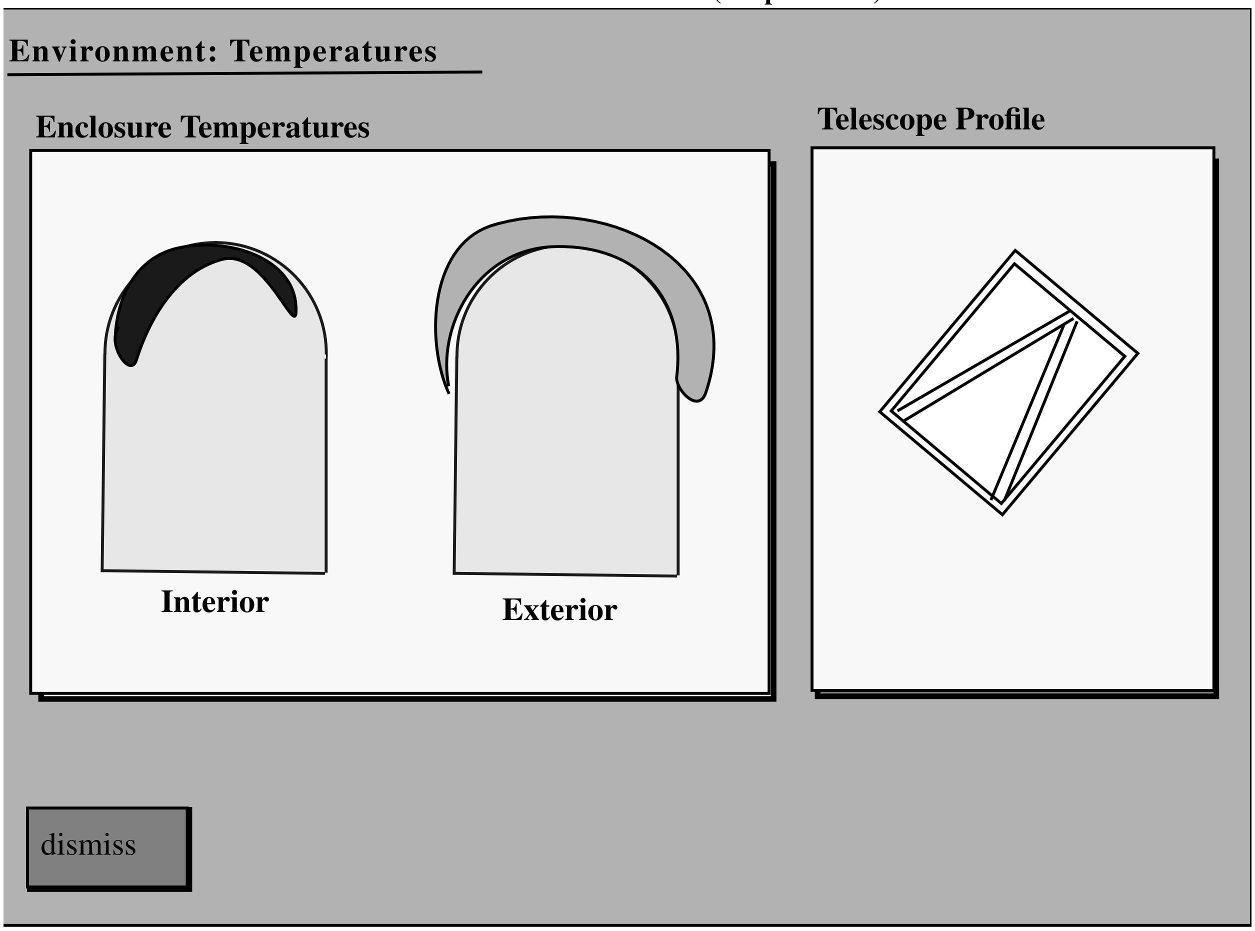


FIGURE 13 - 5 **Environmental monitor console (temperatures)**



13.4.2 Fine Grained Control of the TCS

In order to provide control of individual mechanisms and subsystems there needs to be a group of commands which can be sent to the TCS independently of the OCS Sequencer commands.

The mechanism for sending these commands is that there are separate database entries for individual modifiable attributes in the TCS EPICS database. The OCS updates these attributes and issues a command entry causing the TCS perform the function.

There is no need to have a separate database entry per command. There may be a need to provide separate database entries for individual subsystems.

The initial attribute set and the means of extending this attribute set are provided for the CDR.

13.5 INTERFACE TO OCS SCREENS SYSTEM

The TCS provides the following *items* to the different *screens*. These items are maintained in an EPICS database on the TCS. In order to prevent undue loading of the lower level subsystems it may be necessary to provide a separate *head only* EPICS system in parallel to the TCS in order to maintain these screens.

It is up to the higher level processes to register interest in or directly retrieve the screen items in this database and to handle displaying them.

13.5.1 Source and Telescope Information Screen Items

The following screen items (at least) are required.

- Time.
- Input data for the current demand.
- Apertures and offsets.
- Telescope state
- Telescope position.

Note that the coordinate system may be changed. The available options are:

--INPUT (default)---the coordinate system used to input the source data. Any space motions have been removed, so the position refers to the *current* epoch. If proper motions, parallax or radial velocity were specified, then the position differs from the input position even in the absence of offsets.

--B1950 α, δ ---current epoch

--J2000 α, δ ---current epoch.

--APPARENT α, δ ---geocentric apparent coordinates for the current epoch.

--HA_TOPO---topocentric hour angle and declination.

- acquisition camera / WFS coordinates
- Limit information
- Rotator information.

- Miscellaneous.

  --Focus position (mm).

  --The focus offset used to compensate for expansion of the structure.

  --The focus offset to correct for additional optical elements (e.g. filters) in the beam.

  --Need to think about the implications of adjusting telescope focus on the A&G etc Wavefront sensors !!

  --Dome azimuth.

  --Air mass (relative to the zenith).

- Alarm warning. This indicates that one of the alarm screen shows an alarm condition.

13.5.2 Encoder Screen

The TCS does not provide any encoder screen items as there are no encoders in the TCS system.

13.5.3 Limit Screen

The efficient planning of a sequence of observations depends on the ability to calculate the period of time a target can be observed, starting from a given telescope position. This is complicated in the case of an altazimuth mount by the fact that the azimuth and rotator drives have finite ranges of travel. See section 13.13.6 on page 22.

13.5.4 Alarms Screen

- Weather alarms
- Control room temperature alarm. This is used to detect a failure of the air-conditioning system and to turn off the power supply to the computers if the control room overheats. Normal state CLEAR; alarm state SET.

13.5.5 Interlocks Screen

TBD

13.6 HARDWARE SETUP INFORMATION

The TCS must be able to retrieve setup default information for the subsystems it is responsible for. This information is stored between power cycles. There are, in general, two sets of setup information:

- default initialization setup

- current initialization setup

The difference is that the default setup is that which can be used to go back to a known state. The current setup information can be used to recover the previous setup — perhaps due to restarting during the night after a power failure.

13.7 DATA HANDLING SYSTEM

The TCS must be capable of logging information to the DHS. The baseline for all logging of TCS information is to use the mechanism described in ICD/4 [27]. Engineering information is logged using the EPICS archiving utility [17].

The following general items are capable of being logged, depending on the current logging level:

- commands
- warnings
- errors
- TBD

13.8 INTERFACE TO THE TCS SUBSYSTEMS

These interfaces are *internal interfaces* within the TCS. Where possible, they all adopt the same database interfacing method described above. This section presents the prototype for all of these interfaces and the types of information that must be available through these interfaces. A detailed description of the interface between the TCS and its subsystems may be found in ICD/7b [31]. The details of each individual interface can be found in the appropriate chapter concerning the subsystem.

The interfaces to TCS subsystems can be divided into the following types:

- drives
- encoders
- wavefronts

Each mechanism within a subsystem is described within the TCS by a record containing the following fields:

Demand Position. The desired position for the mechanism in its local coordinate system, either updated continuously or set to a constant value if a mechanism is to be moved to a given position and stopped.

Actual Position. The position of the mechanism in its local coordinate system (corrected for any errors peculiar to the encoding system). This should be equal to the demand position if the mechanism is tracking or stopped in position.

Command field. The field set by the command routines in order to instruct the mechanism to move, stop or zeroset.

Status field. The field which records whether the mechanism is: moving or stopped, following or not following, in position or not and zeroset or not.

Servo constants. Some mechanisms may require servo parameters to be adjusted and/or initially set depending on the type and range of motion desired.

Limits. Software limits on position (positive and negative), velocity and acceleration can be set. If the demand position is outside a software limit, then the mechanism is stopped and an alarm is triggered; excessive velocities and accelerations are clipped.

Stopping radius. This is the tolerance within which a mechanism is set by a command to move to position and stop.

In-position radius. The in-position radius defines the tolerance within which observing is possible for a moving mechanism. If the position error is less than this value, then the mechanism is said to be ‘tracking’.

‘**Following’.** means that the position of the mechanism is being updated continuously. For altitude, azimuth, field and dome rotation this is done during a sidereal track; the focus is adjusted in response to changes in temperature.

13.8.1 Encoders

The *actual position* field is updated based on the encoder value returned from the mechanism. In order to generate the *actual position* the subsystem may have to combine readings from several different encoders, or from several different readings of the same encoder, either spatially or temporally separated.

Multiple encoding systems may be used on some mechanisms. The purpose of the encoder models used by the mechanism is to produce estimates of the position of each mechanism which are independent of the details of the encoding hardware, so that values from different encoders can be combined or compared. In general the actual encoder model is hidden in the subsystem concerned and the TCS instructs the subsystem as to which encoder model to use.

The following types of encoders may be used:

Fiducial. These are (in general) Sony mag-sensors which are used to establish absolute fiducial marks on the axis in question.

Absolute. These are gear driven absolute encoders of relatively coarse resolution used primarily to set initial zero-points and thereafter as a check.

Incremental Tape. The azimuth and altitude axes are equipped with an inductive tape encoder with four reading heads. This has higher performance than the gear encoders.

Friction Incremental. The altitude and azimuth axes are also fitted with friction-driven encoders.

The raw values read from the encoders are processed through a model which has the following components (not all of which are used for every encoder):

1. Scale and zero-point.
2. A look-up table, which is derived from smoothed tracking data.
3. Analytical functions
4. A correction derived from transducers.

The zero-points of the incremental encoders may be determined in a number of different ways: by comparison with the absolute encoders; by driving the telescope past a reference position where a proximity detector produces a hardware signal to clear the counter or by driving the telescope to one of its hardware park positions, which are defined by independent microswitches.

13.8.2 Drives

A drive mechanism is described within the TCS by a record where:

- the Demand Position is the actual position of the mechanism; discrete units (such as filter position) or in radians

13.8.3 Wavefronts

A wavefront is described within the TCS by a record where:

- the Demand Position is a description of the target wavefront in terms of Zernike coefficients {Z1,Z2,...,Zn}.

13.8.4 Forces

A force request is described within the TCS by a record where:

- the Demand Position is a vector of the forces desired $\{F_1,F_2,...,F_N\}$

13.8.5 TCS Subsystem Commands / Status

13.8.5.1 COMMANDS

The command field of the subsystem records can contain:

- STOP; stop moving and apply brakes, or stop sensing
- MOVE; go to target configuration and stay there
- DRIFT; go to target configuration and drift with a constant velocity
- UNWRAP; take out cable wrap
- FOLLOW; continuously monitor target configuration and adjust current configuration to match it
- STOP FOLLOWING; stop monitoring the target configuration and keep current configuration constant
- GUIDE CONFIG
- GUIDE
- STOP GUIDING
- STANDALONE
- PARK; move to a defined safe configuration, stop there and apply brakes
- ENCODER CONFIG
- ZERO SET

13.8.5.2 STATUS

The status field of the subsystem records can contain:

- IN POSITION
- MOVING
- STOPPED
- FOLLOWING
- GUIDING
- LIMIT EXCEEDED

- TIMEOUT
- ENCODER INCONSISTENT

13.9 EXTERNAL BUS CONNECTIONS

13.9.1 Synchronization Bus

The TCS is connected to the synchronization bus in order to monitor the information being passed directly between subsystems and to calculate the corrections needed for all subsystems except for the secondary fast tip/tilt/piston.

See ICD/10 [34] for a detailed description of the synchronization bus.

13.9.2 Interlock System

The TCS must have the ability to monitor the status of the passive interlocks in order to lock out the software. The intent is to prevent the software from trying to move a device that is passively locked out - even though the TCS cannot move it, it is better not to try.

In addition the TCS must be able to respond to and request interlocks from different systems.

See ICD/12 [36] for a detailed description of the interlock system.

13.9.3 Time Bus

The TCS is connected to the time bus. This time bus connection sets the internal time of the TCS.

See ICD/9 [33] for a detailed description of the time bus.

13.9.4 Event Bus

The TCS is connected to the event bus. The primary purpose of this bus is to allow the required analogue synchronization of chopping and nodding.

See ICD/11 [35] for a detailed description of the event bus.

13.10 COLLECTION OF HEADER INFORMATION FROM THE TCS

Just as with an instrument (See “Requirements for New Instruments” on page 7 - 11), the TCS needs to provide the OCS with a “packet description file” (PDF) containing a list of parameters which it will provide as status information. Unlike an instrument, the TCS does not save header information during an exposure (because it may be busy doing something else for a different instrument at the time), and it is up to the OCS to sample the header parameters at appropriate moments. The PDF for the TCS should therefore indicate whether a header parameter is a constant, should be collected at the beginning of each exposure, the end of each exposure, or should be the mean value of the parameter during the exposure. The OCS will decide how to collect this information. The PDF should contain the sort of information contained in this table:

FITS keyword	Location	Type	FITS header?	Sample time	Telescope consoles?	FITS Comment
RA	TCS.RA	CHAR	✓	START	✓	Right ascension of telescope
RASTART	TCS.RA	CHAR		START		Right ascension of telescope at start of observation.
RAEND	TCS.RA	CHAR		END		Right ascension of telescope at end of observation.
DEC	TCS.DEC	CHAR	✓	START	✓	Declination of telescope
DECSTART	TCS.DEC	CHAR		START		Declination of telescope at start of observation.
DECEND	TCS.DEC	CHAR		END		Declination of telescope at end of observation.
EQUINOX	TCS.EQUINOX	FLOAT	✓	START	✓	Equinox of RA and DEC
RADECSYS	TCS.RADECSYS	CHAR	✓	START	✓	Stellar reference frame
TIMESYS	TCS.TIMESYS	CHAR	✓	START	✓	Time system
EPOCH	TCS.EPOCH	CHAR	✓	START	✓	The epoch of the observation.
TELFOCUS	TCS.FOCUS	FLOAT	✓	START	✓	Telescope focus.
	TCS.AIRMASS	FLOAT			✓	Current air mass.
AMSTART	TCS.AIRMASS	FLOAT	✓	START		Air mass at start of observation.
AMEND	TCS.AIRMASS	FLOAT	✓	END		Air mass at end of observation.
AZIMUTH	MNT.ALTITUDE	FLOAT		START	✓	Altitude of mount.
ALTITUDE	MNT.AZIMUTH	FLOAT		START	✓	Azimuth of mount.
DALT	ENC.ALTITUDE	FLOAT		START	✓	Altitude of enclosure.
DAZ	ENC.AZIMUTH	FLOAT		START	✓	Azimuth of enclosure.

FITS keyword	**Location**	**Type**	**FITS header?**	**Sample time**	**Telescope consoles?**	**FITS Comment**
M1TEMP1	M1.TEMP1	FLOAT		START	✓	Primary mirror temperature - sensor 1.
GSRA1	AG.GS1.RA	CHAR	✓	START	✓	Right ascension of guide star 1.
GSDEC1	AG.GS1.DEC	CHAR	✓	START	✓	Declination of guide star 1.
etc.…	etc.…	etc.…		etc.…		etc.…

The columns in this table are as follows:

FITS keyword. Is the FITS keyword to be used when this item is included in a FITS header. This column should only be blank if the item will *never* be included in a FITS header (for example the current airmass). All other items should have a FITS keyword standing by, even if their "FITS header?" column is blank. If possible there should be a sensible relation between the name of the EPICS record and the name of the FITS keyword, but this may not always be possible.

Location. The location in the TCS or TCS subsystem's EPICS database where the value may be found.

Type. The type of the value (CHAR, INTEGER, LOGICAL, FLOAT).

FITS header? A ✓ in this column indicates a mandatory item that should always be included in the FITS header.

Sample time. This column specifies when the value should be sampled for the FITS header:

--START; sample the value at the beginning of each exposure;

--END; sample the value at the end of each exposure.

A blank here means the item is not to be included in the FITS header. Note that the same item can appear in the FITS header more than once sampled at different times (e.g. AMSTART and AMEND). The TCS group should supply a list of the default items to appear in the header. The other items can be written into the header by configuring the OCS to read them.

Telescope consoles? A ✓ in this column indicates an item that the TCS group would like to see displayed on one of the telescope consoles within the Control Tool.

See also Table 11 - 6, "Standard FITS keywords provided by the TCS," on page 11 - 42.

The above table gives only examples. The exact details of the contents of this table are to be decided by the TCS and TCS subsystem groups.

13.11 TCS COORDINATE SYSTEMS

1. Azimuth (A) is measured from north through east in the plane of the horizon and elevation (E) or altitude is measured perpendicular to the horizon ('elevation' and 'altitude' are used interchangeably, following conventional usage). Zenith distance $z = \pi/2 - E$ is more often used in calculations, since it is a natural polar coordinate. Note that Wallace (1990) measures azimuth from south though east internally, in order to avoid using a left-handed system; externally the north through east convention is used.
2. Hour angle (h) is measured westwards in the plane of the equator from the meridian and declination (δ) is measured perpendicular to the equator, positive to the north.
3. Right ascension (α) is measured from the equinox eastward in the plane of the equator.
4. Longitude is measured from the equinox eastward in the plane of the ecliptic and latitude (ϕ) is measured perpendicular to the ecliptic, positive to the north.
5. Sky position angle (PA), θ, is measured anticlockwise from north.

13.12 BASIC GEOMETRY

[*to be supplied by R.Laing, RGO, for CDR]*

13.12.1 Equatorial to Altazimuth Conversion

13.12.2 Velocities and Accelerations

13.12.3 Coordinates in the Tangent Plane

- **Equatorial**
- **Altazimuth**
- **xy**

13.13 TCS POINTING CALCULATIONS

[*to be supplied by R.Laing (RGO) & P.Wallace (RAL) for CDR]*

13.13.1 General

The philosophy adopted for the pointing calculations in the TCS is based on the 'virtual telescope' (Straede & Wallace 1976), whose imperfections such as flexure and refraction are concealed from the user. All of the control is specified in terms of standard astronomical coordinate systems (for targets) or detector coordinates (for images).

13.13.2 Time

13.13.3 Astronomical Coordinate Systems

The TCS accepts input in one of three equatorial coordinate systems:

1. Geocentric apparent coordinates of the current date.
2. Mean coordinates in the post-IAU1976 (FK5) system.
3. Mean coordinates in the pre-IAU1976 (FK4) system.
4. JPL ephemeris format for solar system objects.

13.13.4 Offsets

[R.Laing to expand for PDR]

A number of methods are provided to allow offsetting. These are used to move the telescope so that the image of a *different* object appears at the *same* place on the detector and therefore cause the displayed right ascension and declination to change. The opposite approach (moving the image of the *same* object to a *different* place on the detector is described in the section on coordinates.

13.13.5 Refraction

13.13.6 Limits

In this section we consider the following constraints on continuous observation of a target:

- elevation limits
- azimuth cable-wrap limits
- instrument rotator cable-wrap limits
- zenith blind spot

- slow spots

13.13.6.1 BASIC FORMULAS

We start from the familiar analytical transformation formulas for position, velocity and acceleration in azimuth, elevation and parallactic angle. These are available elsewhere in the present document and can also be found in the technical reports from other telescope projects (Keck, WIYN, WHT etc.). For convenience, they are given below, in a vector-related form which lends itself to efficient computer implementation. For brevity, only those expressions likely to be needed during the production of Gemini software are given, and detailed derivations are omitted. Note that the analytical formulas are unsuitable for direct use in accurately pointing and tracking a telescope; the need to allow for refraction, mechanical imperfections and off-axis instrument location introduces complications, and it is easiest to obtain velocities and accelerations by direct numerical differentiation. The notation is as follows:

TABLE 13 - 1 Limits formula symbols

hour angle	h
declination	δ
azimuth	A
elevation	E
parallactic angle	q
latitude (geodetic)	ϕ

Notes:

- We require the function $\tan^{-1} a/b$ to return a result in the full $-\pi$ to $+\pi$ range, according to the signs of a and b, as implemented in the Fortran and C functions **ATAN2** and **atan2**. We additionally stipulate that when $a = b = 0$ the function takes an innocuous value, for example zero.
- The formulas ignore the question of whether "longitude" angles should be expressed in the range $[-\pi, +\pi]$ or $[0, 2\pi]$. However, for user-interface and documentation purposes it is suggested that right ascension is $0-24^h$, hour angle is $\pm 12^h$ (W +ve), azimuth is 0 - $360°$ (N= $0°$, E= $90°$) and geographical longitude is $\pm 180°$ with east or west specified in full.

Hour angle, declination to azimuth, elevation:

$$x\ (=\ \cos E \cos A\ ) = -\cos h \cos \delta \sin \phi + \sin \delta \cos \phi$$

$$y\ (=\ \cos E \sin A\ ) = -\sin h \cos \delta$$

$$z\ (=\ \sin E) = \cos h \cos \delta \cos \phi + \sin \delta \sin \phi$$

$$r\ (=\ \cos E) = \sqrt{x^2 + y^2}$$

$$A = \tan^{-1} \frac{y}{x}$$

$$E = \tan^{-1} \frac{z}{r}$$

Parallactic angle:

$$C\ (=\ \cos q \cos E) = \cos \delta \sin \phi - \cos h \sin \delta \cos \phi$$

$$S\ (=\ \sin q \cos E) = \cos \phi \sin h$$

$$q = \tan^{-1} \frac{S}{C}$$

Velocities and accelerations at constant $\dot{h}$ and constant δ (for example when tracking a star):

$$\dot{A} = \sin\phi - \frac{xz}{r^2}\cos\phi$$

$$\dot{E} = \frac{y}{r}\cos\phi$$

$$\dot{q} = -\frac{x}{r^2}\cos\phi$$

$$\ddot{A} = \frac{y}{r^2}\cos\phi\left(z\sin\phi - \frac{(2 - r^2)\,x}{r^2}\cos\phi \right)$$

$$\ddot{E} = \frac{x}{r}\cos\phi\left(\sin\phi - \frac{xz}{r^2}\cos\phi \right)$$

$$\ddot{q} = \frac{y}{r^2}\cos\phi\left(\sin\phi - 2\frac{xz}{r^2}\cos\phi \right)$$

A more compact formulation is:

$$\dot{q} = -\frac{x}{r^2}\cos\phi$$

$$\dot{A} = \sin\phi + z\dot{q}$$

$$\dot{E} = \frac{y}{r}\cos\phi$$

$$\ddot{A} = \frac{\dot{E}}{r}(z\sin\phi - (2 - r^2)\,\dot{q})$$

$$\ddot{E} = -r\dot{q}\dot{A}$$

$$\ddot{q} = \frac{\dot{E}}{r}(\sin\phi + 2z\dot{q})$$

The above expressions give the velocities and accelerations in natural units. For the case of tracking a star, to convert to deg s^{-1} , multiply velocities by:

$$\frac{2\pi}{86400} \times \frac{360}{2\pi} = \frac{1}{240}$$

and to convert accelerations to deg s^{-2} , multiply by:

$$\left(\frac{2\pi}{86400}\right)^2 \times \frac{360}{2\pi} \cong \frac{1}{3300236.9}$$

The transformation from azimuth, elevation to hour angle, declination is also worth noting here:

$$X\,(=\ \cos\delta\cos h)\ =\ -\cos A\cos E\sin\phi + \sin E\cos\phi$$

$$Y\,(=\ \cos\delta\sin h)\ =\ -\sin A\cos E$$

$$Z\,(=\ \sin\delta)\ =\ \cos A\cos E\cos\phi + \sin E\sin\phi$$

$$R\,(=\ \cos\delta)\ =\ \sqrt{X^2 + Y^2}$$

$$h\ =\ \tan^{-1}\frac{Y}{X}$$

$$\delta\ =\ \tan^{-1}\frac{Z}{R}$$

13.13.6.2 PREDICTING LIMIT TIMES

Neglecting refraction and other pointing effects, the time at which a given limit will be reached can be predicted by using the appropriate expression

$$h_\lambda\ = f\,(\phi, \delta, \lambda)$$

where δ is the declination of the target and λ is the limit concerned - for example the most negative azimuth permitted by a cable-wrap, or the maximum allowed azimuth acceleration. The time until reaching the limit, in SI seconds, is simply:

$$(h_\lambda - (\theta - \alpha)) \times \frac{86164.099}{2\pi}$$

where θ is the local sidereal time and α is the apparent right ascension of the target.

Expressions for h in terms of the other quantities can in a some cases be obtained by straightforward application of the basic formulas given earlier or by applying standard spherical trigonometry expressions. However, in the important cases of predicting h from, respectively, azimuth and parallactic angle, this is not possible. Instead, application of the four-parts rule to the spherical triangle pole-target-zenith produces an expression involving terms in both $\sin h$ and $\cos h$, of the form

$$P \sin h + Q \cos h = R$$

We can rewrite this as

$$S \sin (h + T) = R$$

where

$$S = \sqrt{P^2 + Q^2}$$

and

$$T = \tan^{-1} \frac{Q}{P}$$

If R > S there are no solutions. Otherwise there are potentially two solutions:

$$h_1 = \sin^{-1} \frac{R}{S} - T$$

and

$$h_2 = \pi - \left(\sin^{-1} \frac{R}{S} + T \right)$$

The validity of each of these solutions can then be established by tests based on sign.

Starting from the above expressions, the following alternative formulation can be obtained:

$$h_1 = \tan^{-1} \frac{PR - QW}{QR + PW}$$

and

$$h_2 = \tan^{-1} \frac{PR + QW}{QR - PW}$$

where $W = \sqrt{P^2 + Q^2 - R^2}$.

Which of these formulations is most convenient depends on circumstances. Note that unbounded functions such as **tan** and **cot** may have to be eliminated from the chosen formulas before computer implementation.

13.13.6.3 ELEVATION LIMITS

The hour angle h at which a star at declination δ reaches a given elevation E_L is as follows:

$$\gamma = \frac{\sin E_L - \sin\delta\sin\phi}{\cos\delta\cos\phi}$$

If $\gamma < -1$; the star never rises

If $\gamma > +1$; the star never sets

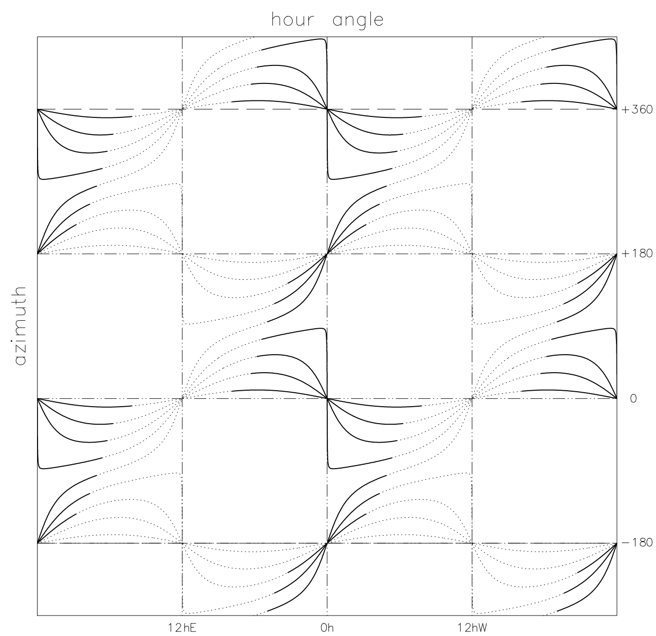
Otherwise, the star rises at $h = -\cos^{-1}\gamma$ and sets at $h = \cos^{-1}\gamma$

The operational limits in elevation for the Gemini telescopes are $15°$ and $90°$. (It is mechanically possible to reach $0°$ and $93°$.) To determine whether a Gemini telescope can see a given star, and if so when the star rises or sets, set $E_L = 15°$ in the above expression for γ.

13.13.6.4 AZIMUTH CABLE-WRAP LIMITS

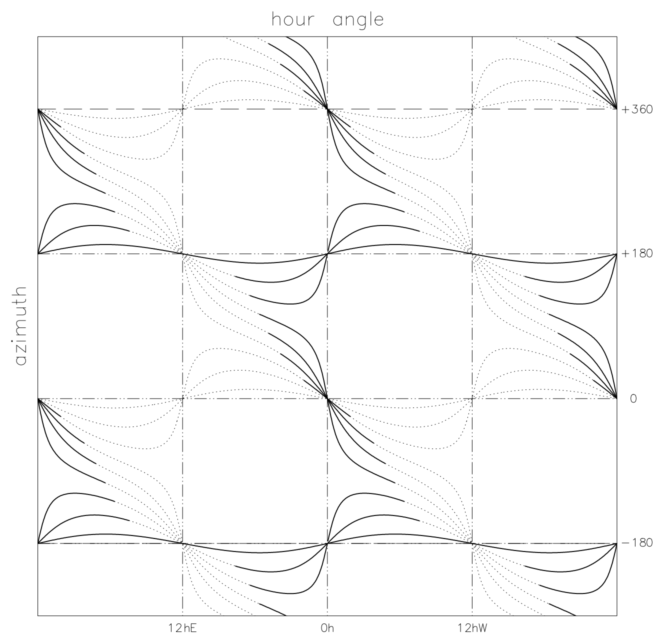
Figure 13 - 6 and Figure 13 - 7 show how azimuth changes while tracking stars at various declinations.

FIGURE 13 - 6 Azimuth plotted against hour angle for the northern Gemini telescope, for various declinations.



In Figure 13 - 6, the thick lines are where the star is above the 15° elevation limit**.** The small-amplitude wave passing through the plot center confirms that at declination +80° the azimuth varies very little because the star is close to the pole. The lines for +60° and +40° show increasingly wide cyclic motion. The 0°line shows monotonic motion; cyclic motion has returned by the –20° line and flattens out as the (unobservable) south pole is approached.

FIGURE 13 - 7 Azimuth plotted against hour angle for the southern Gemini telescope.



Note the following features of Figure 13 - 6 and Figure 13 - 7:

- Azimuth and hour angle always have different signs (when expressed in the range [$-\pi$; $+\pi$ that is).
- A given star may reach a specified azimuth twice per day, once per day, or never.
- The operational azimuth range for the Gemini telescopes, due to the azimuth cable-wrap, is $550°$, centered on $90°$ (due east) for both Gemini telescopes. (There are no corresponding limits on enclosure rotation.) The cable-wrap limits are marked on the plots.

To predict whether an observation will be curtailed by arrival at an azimuth limit A_L, we require expressions giving the hour angle(s) corresponding to a specified declination. The following procedure is based on the approach described earlier in the section *Predicting Limit Times*:

1. Express the supplied azimuth and declination in the range [$-\pi$, $+\pi$] and make infinitesimal adjustments as necessary to avoid numerical problems (rounding-errors, indeterminate signs, overflows) in critical places (e.g. $A = \pi$, $\delta = \phi$).
2. Evaluate:

$$a = \sin\delta\sin A_L\cos\phi$$

$$b = \cos\delta\sqrt{\cos^2 A_L + \sin^2 A_L \sin^2\phi}$$

3. If a > b there are no solutions - the azimuth limit cannot be encountered at the specified declination.
4. Otherwise evaluate:

$$t = \tan^{-1}\frac{\sin A_L \sin\phi}{-\cos A_L}$$

5. The two potential solutions are

$$h_1 = \sin^{-1}\frac{a}{b} - t$$

$$h_2 = \pi - \left(\sin^{-1}\frac{a}{b} + t\right)$$

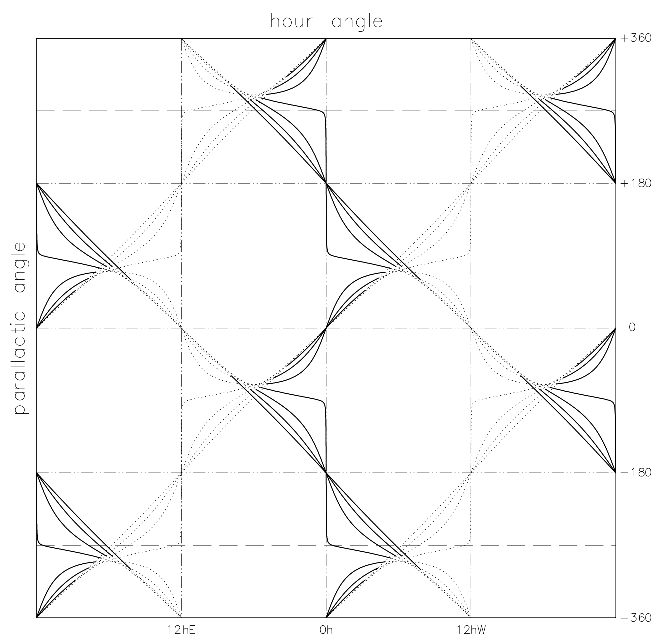
6. Accept solution h_n only if A_L and h_n have different signs.

Note that the nominal azimuth wrap limits on the Gemini telescopes are $+360°$ (due north) and $-180°$ (due south), and hence the corresponding hour angle is always simply 0^h or 12^h. The above formulas will nonetheless be required if there are small departures from the nominal limits, or if there are any pre-limit restrictions on speed, *etc.*

13.13.6.5 INSTRUMENT ROTATOR CABLE-WRAP LIMITS

Figure 13 - 8 and Figure 13 - 9 show, for each of the Gemini telescopes how parallactic angle changes while tracking stars at various declinations.

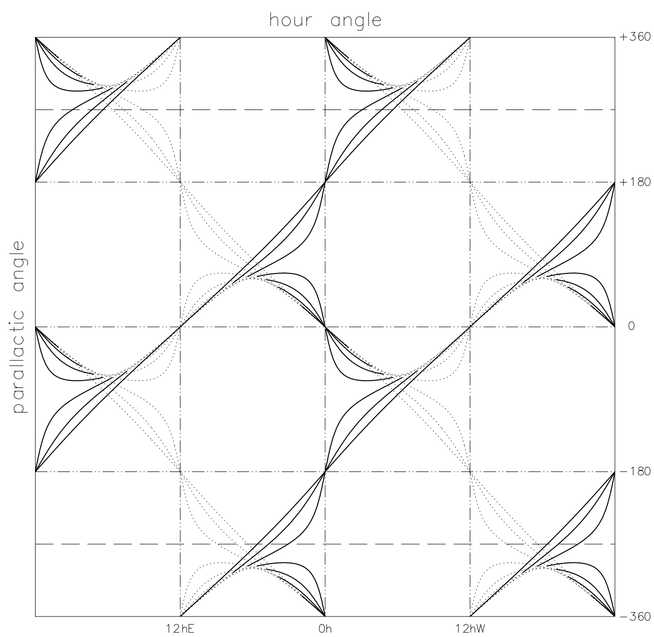
FIGURE 13 - 8 Parallactic angle plotted against hour angle for the northern Gemini telescope.



In Figure 13 - 8, the lines correspond to different declinations $20°$ apart. The diagonal lines which start at top left illustrate the continuously rotating parallactic angle for $+80°$, near the pole. The $+60°$, $+40°$ and $+20°$ lines are progressively less straight as the zenith declination is approached. The $0°$ line is within the region where the parallactic angle varies cyclically. Past the nadir, at $-20°$ and further south, the motion once again becomes monotonic. The thick lines are where the star is above the $15°$ elevation limit.

The cable-wrap limits for instrumental offset $k = 0$ are marked.

FIGURE 13 - 9 Parallactic angle plotted against hour angle for the southern Gemini telescope



Note the following:

- Parallactic angle and hour angle always have the same sign (when expressed in the range [$-\pi$, $+\pi$] that is).
- A given star may give rise to a specified parallactic angle twice per day, once per day, or never.

Note also that the sense of the rotator position angle ψ is opposite that of parallactic angle q : using the instrument rotator to compensate for field rotation requires (to first order) the following equation to be satisfied:

$$\psi + q = k$$

where k is an offset which depends on the design of the instrument and the wishes of the observer. The operational rotator position angle range for the Gemini telescopes, due to the cable-wrap, is $540°$. The center of the range corresponds to rotator position angle zero, for both Gemini telescopes. The cable-wrap limits for offset $k = 0$ are marked on the plot. For other k values the two lines will move up or down but maintain their $540°$ separation.

To predict whether an observation will be curtailed by arrival at a parallactic angle $q_L = k - \psi_L$, we require expressions giving the hour angle(s) corresponding to a specified declination. The following procedure is based on the approach described earlier in the section Predicting Limit Times:

1. Express the supplied parallactic angle and declination in the range $[-\pi, +\pi]$ and make infinitesimal adjustments as necessary to avoid numerical problems (rounding-errors, indeterminate signs, overflows) in critical places (e.g. $q = \pi$, $\delta = \phi$).
2. Evaluate

$$a = \sin\phi \sin q_L \cos\delta$$

$$b = \cos\phi \sqrt{\cos^2 q_L + \sin^2 q_L \sin^2\delta}$$

3. If a > b there are no solutions - the parallactic angle limit cannot be encountered at the specified declination.
4. Otherwise evaluate:

$$t = \tan^{-1}\frac{\sin q_L \sin\delta}{\cos q_L}$$

5. The two potential solutions are

$$h_1 = \sin^{-1}\frac{a}{b} - t$$

$$h_2 = \pi - \left(\sin^{-1}\frac{a}{b} + t\right)$$

6. Accept solution h_n only if q_L and h_n have the same sign.

13.13.6.6 ZENITH BLIND SPOT

With an altazimuth mounting, following stars across the zenith region requires arbitrarily large speeds and accelerations. Whatever the available performance, there will inevitably be an inaccessible, or at least unusable, region of sky. The size and shape of this “blind spot” depend on what speed and acceleration limits are in force.

In many existing equatorially-mounted telescopes there are clearly distinguished *slewing* and *tracking* states. Slewing from one target to another involves much faster motion than does tracking, even where “tracking” includes offsetting and scanning maneuvers. Moreover, there may well be quite different servo, drive train or encoding arrangements during slewing compared with tracking.

In a modern altazimuth design, the distinction is less obvious and may indeed be absent. In particular, if the full “slewing” speed is available when tracking stars close to the zenith the inaccessible region will be kept to a minimum. The natural approach is for the idealized virtual telescope not to have any “slewing” state, and to give the underlying software *carte blanche* to use whatever capabilities the mount has, in its efforts to follow the changing direction of the target.

In the Gemini design, however, there are other constraints apart from sheer azimuth speed. These involve image quality, so that for instance mount accelerations must be kept small if the active optics system is to keep the imaging performance within specification. Note, however, that this does not necessarily mean that the full capabilities of the mount cannot be used to follow a star, only that at some stage the system must flag that image quality can no longer be guaranteed. One simple way that this might be done is for the system to declare that it is in “slewing” state once the tracking speed and acceleration limits have been exceeded, and this is recommended later.

All the important zenith limitations arise from azimuth:

- Azimuth speeds and accelerations near the zenith become arbitrarily high.

- Elevation speeds and accelerations are too small near the zenith, or anywhere else in the sky, to matter. Elevation dynamics affect “offsetting” performance (e.g. settling at the end of a slew or a nod) and are not stretched by sidereal tracking.
- Like azimuth, instrument rotator speeds and accelerations become arbitrarily large near the zenith. However, in the zenith region the rotator simply undoes what the azimuth does, and so the same level of performance is demanded from rotator and azimuth. The great difference in mass and size between the two areas of the telescope mean that rotator speed and acceleration capabilities can safely be assumed to exceed those of azimuth. Moreover, acceleration limits imposed by the active optics system apply only to azimuth, not position angle.

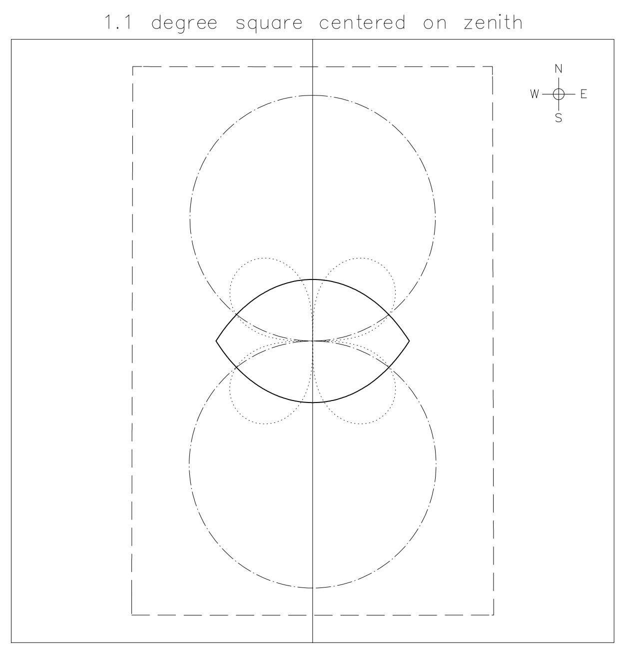
Three sorts of zenith blind-spot arise from on-axis sidereal tracking:

1. The region where azimuth speed is excessive.
2. The region where azimuth acceleration is excessive.
3. The region where maximum azimuth speed is required in order to produce a loss of acquisition which is symmetrical about the meridian.

In addition to these geometrically-derived regions there is the possibility of voluntary constraints affecting observation planning but not directly affecting observing itself. This is desirable because (i) the real blind-spots have complicated shapes which make observation planning awkward, and (ii) the true shape of the various regions may in any case be distorted by instrument offsets, by temporary variations from the nominal limits or by departures from purely sidereal tracking.

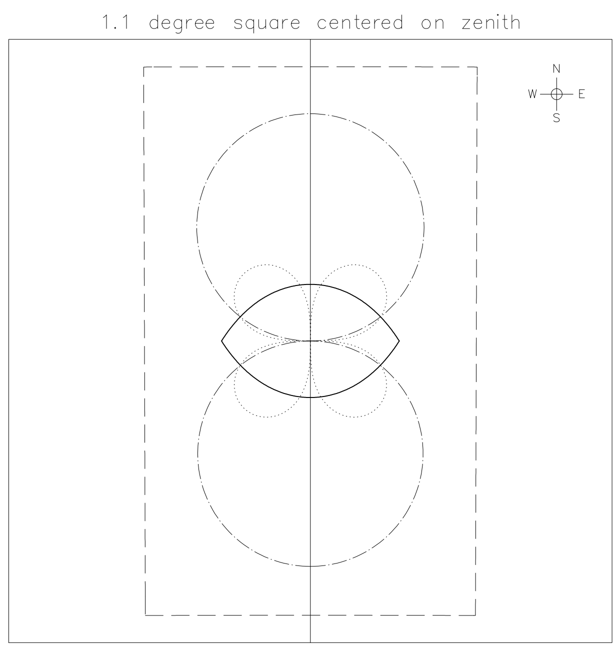
Figure 13 - 10 and Figure 13 - 11 show the various blind spots, for the two Gemini telescopes.

FIGURE 13 - 10 The zenith blind spots for the northern Gemini Telescope.



In Figure 13 - 10, the two circles are the region within which the maximum azimuth tracking rate would be exceeded. The four-lobed shape is the region within which the azimuth acceleration would exceed the maximum allowed during tracking. The lenticular region comes from the maximum azimuth slew rate. The rectangular region is the blind-spot for observation planning.

FIGURE 13 - 11 The zenith blind spots for the southern Gemini Telescope.



In Figure 13 - 11, the noticeable reduction in size is a consequence of the larger distance of the Chile telescope from the equator compared with the Hawaii telescope.

The analytical formulas used to generate these plots are available in the references; they are not repeated here as they are not required by the Gemini telescope control system, for reasons that will become clear. The different regions are as follows:

1. The figure-of-eight “keyhole” region is where the tracking azimuth speed limit of $0°:5$/s is exceeded. The “waist” occurs because stars which transit very close to the zenith require little azimuth movement until just before transit, when the full 180° is suddenly required.
2. The tracking acceleration limit of 0°.25/s/s produces the four-lobed pattern in the middle. Note that almost the whole of the pattern lies within the speed-limit blind-spot; a considerable reduction in the acceleration limit would be needed before significant enlargement of

the unobservable region would result. Conversely, if the tracking speed limit were to be increased to the slewing value, the “keyhole” dimensions would shrink by a factor of four and the four-lobed acceleration blind-spot would dominate.

3. The lenticular region in the center shows loss of acquisition at the slewing speed limit of $2°$ /s ignoring acceleration/deceleration delays. The cusps occur because the region is the intersection of two separate curves, one north and one south of the zenith. A sudden change to slew speed on entry to the region will cause the telescope to move ahead of the star; as the telescope crosses the meridian the star overtakes it; the telescope finally catches up with the star on reaching the far edge of the region. Of the various blind-spots plotted on the diagrams, this is the one that is unavoidable (at least for instruments mounted near the rotator axis) because it reflects the limits of performance of the drives. All of the other blind-spots are essentially voluntary, designed to guarantee image quality.
4. The “observation planning” zenith blind-spot is bounded by $\delta = \phi \pm 0.5°$ and $h = \pm 0.35°$.

The various blind-spots are perceptibly smaller for the Chile telescope than for the Hawaii telescope, because of the difference in latitude. The ratio is $(\cos\phi_s) / (\cos\phi_n)$, roughly 11:12.

Experience with existing altazimuth telescopes (notably the WHT) suggests that an elaborate treatment of the zenith region is unnecessary and may interfere with the wishes of the observer in individual cases. The following plan is simple and robust, and delivers a close approximation to optimum efficiency:

- Observations which may overlap the “planning” blind-spot should, by default, be automatically rescheduled to earlier or later times.
- When a target is approaching the planning blind-spot, there should be a display of the time of entry into the region, to 1 minute resolution. During passage through the region, the time of exit from the region should be displayed.
- Data-acquisition applications should have access to information about whether the current target will enter the region, or is already within the region, and the time estimates.
- Inside the region, tracking should proceed normally. The full performance of the mount should be available to follow the target, but whenever a tracking speed or acceleration limit is being exceeded the status “slewing” should be declared. In many cases, adequate tracking accuracy and optical quality will be maintained throughout the transit; it is up to the observer and the data-acquisition system whether to play safe by suspending the observation in good time or whether to watch for explicit loss of imaging before stopping. In cases where slew speed is insufficient to track the tar-

get, the approach just described will delay the required large azimuth movement and displace the inaccessible region to the west of the zenith. However, this will not significantly increase the lost time.

13.13.6.7 SLOW SPOTS

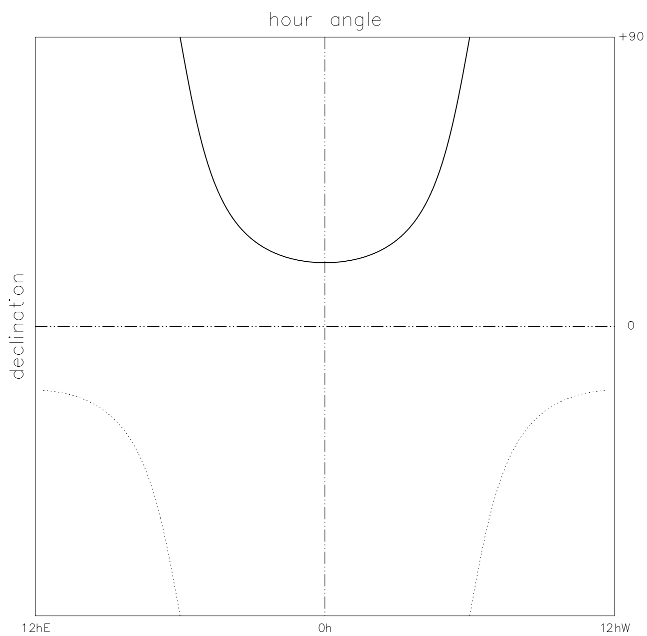
Compared with most servo applications, an unusual characteristic of telescope drives is the enormous dynamic ranges demanded in the control of both position and speed. Equatorial and altazimuth mountings have the same positional accuracy goals, namely a few percent of an arcsecond all over the visible sky, and both have to be capable of adequate slewing speeds, but their behavior in terms of tracking velocity is markedly different. In the equatorial design, the declination axis is stationary during sidereal tracking (except for small adjustments due to refraction etc.), while the polar axis rotates at a steady 15 arcseconds per second or thereabouts, always in the same direction. An altazimuth mounting, on the other hand, has to deliver continuously changing velocities in both azimuth and elevation. The high speed case poses few problems; only in azimuth are high tracking speeds called for, and only then near the zenith, where a large azimuth error produces only a small pointing error. However, there are potential control problems in those areas of sky where the tracking velocities change sign and where any stiction or hysteresis effects will therefore become apparent. Here, limit-cycling or a dead-zone may be unavoidable, exceeding normal tracking accuracy error-budgets. It is therefore necessary to identify where in the sky any zero-velocity effects will occur, so that performance can be tested and so that critical observations can be scheduled so as to avoid the danger areas.

In elevation, zero tracking velocity occurs at the poles and along the meridian; the hour angle for an elevation slow-spot is simply:

$$h = 0$$

The azimuth slow-spots are plotted in Figure 13 - 12 and Figure 13 - 13.

FIGURE 13 - 12 Azimuth slow spots for the northern Gemini telescope.



In Figure 13 - 12 the lines mark the places where azimuth velocity drops to zero, for the northern Gemini telescope. Only the northern cases (heavy lines) are above the elevation limit.

FIGURE 13 - 13 Azimuth slow spots for the southern Gemini telescope.

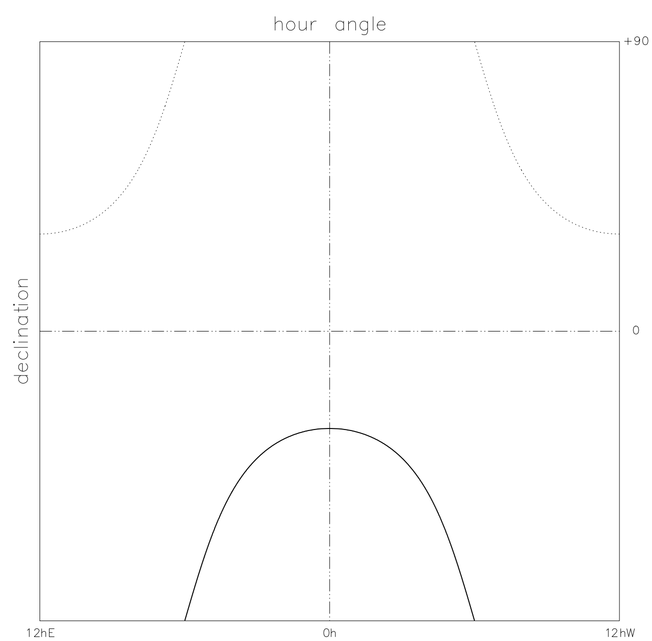


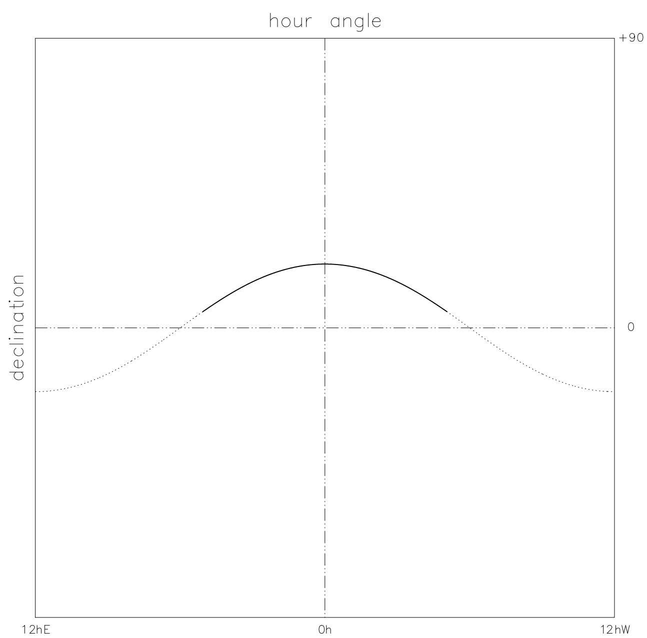
Figure 13 - 12 and Figure 13 - 13 are each a simple-cylindrical projection of the sky, i.e. with h and δ plotted as Cartesian coordinates; the lines show where, apart from at the poles, $\dot{A}$ is zero. The condition occurs when:

$$h = \pm\cos^{-1}\frac{\tan\phi}{\tan\delta}$$

for $(|\delta| \geq |\phi|)$.

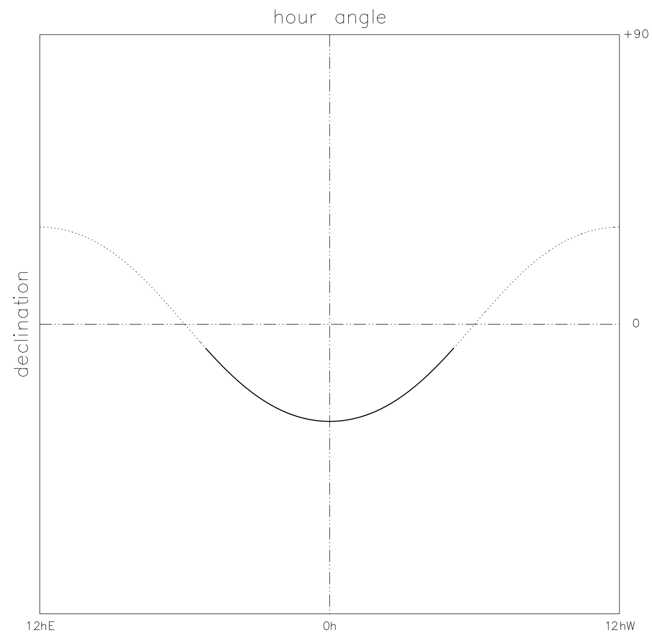
The parallactic angle slow-spots are plotted in Figure 13 - 14 and Figure 13 - 15. As for the $\dot{A} = 0$ plot (see above) this is a simple-cylindrical projection of the sky.

FIGURE 13 - 14 Parallactic slow-spots for the northern Gemini telescope



In Figure 13 - 14, the line marks the places where parallactic angle velocity drops to zero, for the northern Gemini telescope. The heavy line indicates the portion which occurs above the elevation limit.

FIGURE 13 - 15 Zero parallactic angle velocity for the southern Gemini telescope.



The lines show where $\dot{q}$ is zero. The condition occurs when:

$$h = \pm\cos^{-1}\frac{\tan\delta}{\tan\phi}$$

for $(|\delta| \le |\phi|)$.

Slow spots in parallactic angle are unlikely to pose any threat to image quality.

13.13.6.8 STAR VISIBILITY CRITERIA

The following table shows the declination ranges for different "rise/set" conditions, for the northern Gemini telescope, taking into account the elevation limit, the azimuth cable-wrap limits and the "observation planning" zenith blind-spot.

TABLE 13 - 14 Northern Telescope Declination Ranges

declination	condition
$-90°$	
	Star never rises above the $15°$ elevation limit.
$-55.17° = \phi + E - 90°$	
	Star rises at either the elevation limit or the $-180°$ azimuth cable-wrap limit, and sets at the elevation limit.
$+19.33° = \phi - 0.5°$	
	Star rises at either the elevation limit or the $-180°$ azimuth cable-wrap limit, crosses the zenith blind spot, and sets at the elevation limit..
$+19.83° = \phi$	
	Star rises at either the elevation limit or the $+360°$ azimuth cable-wrap limit, crosses the zenith blind spot, and sets at the elevation limit.
$+20.33° = \phi + 0.5°$	
	Star rises at either the elevatio limit or the $+360°$ azimuth cable-wrap limit and sets at the elevation limit.
$+85.17° = -\phi + E + 90°$	
	Star sets eithr at the $+360°$ azimuth cable-wrap limit or not at all.
$+90°$	

The equivalent table for the southern Gemini telescope is as follows.

TABLE 13 - 15 Southern Telescope Declination Ranges

declination	condition
$+90°$	
	Star never rises above the $15°$ elevation limit
$+44.80° = \phi - E + 90°$	
	Star rises at either the elevation limit or the $+360°$ azimuth cable-wrap limit, and sets at the elevation limit.
$-29.70° = \phi + 0.5°$	
	Star rises at either the elevation limit or the $+360°$ azimuth cable-wrap limit, crosses the zenith blind spot, and sets at the elevation limit.
$-30.20° = \phi$	
	Star rises at either the elevation limit or the $-180°$ azimuth cable-wrap limit, crosses the zenith blind splot, and sets at the elevation limit.
$-30.70° = \phi - 0.5°$	
	Star rises at either the elevation limit or the $-180°$ azimuth cable-wrap limit and sets at the elevation limit.
$-74.80° = -\phi - E - 90°$	
	Star sets either at the $-180°$ azimuth cable-wrap limit or not at all.
$-90°$	

Note that the Gemini nominal azimuth cable-wrap limits of $-180°$ and $+360°$ are degenrate cases. The general case, where the limits are at other, arbitrary, values, is more complicated, with additional critical declinations and conditions.

As well as the possibility of tracking into an azimuth or elevation limit, there is also the possibility of tracking into a rotator limit. It is not possible to say in advance if and when this can happen without knowing about the circumstances of the observation. For example, there is clearly no danger of encountering a rotator limit during an observation where the instrument-mount position angle is kept fixed, perhaps to keep a spectrograph slit vertical. Even when the rotator is in normal operation, i.e. compensating for the changing parallactic angle, the disposition of the cable-wrap limits depends on the orientation dictated by the instrument or chosen by the observer. However, some general statements can be made, as follows.

As can be seen from the graph of δ versus h for $\dot{q} = 0$, zero $\dot{q}$ can occur only when $|\delta| < |\phi|$, and only twice per (sidereal) day. Because zero $\dot{q}$ corresponds to a minimum or maximum in q, we conclude that within the above declination range the parallactic angle cycles back and forth. The amplitude of the motion depends on δ, and decreases from $90°$ for $\delta = \pm\phi$ (the zenith and nadir cases) to $\pm 90° - |\phi|$ at $\delta = 0$.

For declinations $|\delta| > |\phi|$, the parallactic angle changes monotonically: the picture rotates indefinitely. (Thus the rotator cable-wrap sets an ultimate limit of about 1.5 days on continuous observation of a single circumpolar source.) As can be seen from the graph of q versus h, $\dot{q} < 0$ for $\delta > \phi$ and $\dot{q} > 0$ for $\delta < -\phi$ (for both Gemini telescopes). Therefore an observation north of $\delta = \phi$ may encounter the positive rotator cable-wrap limit, and an observation south of $\delta = -\phi$ may encounter the negative limit.

Summarizing, the procedure for identifying the next limit during an observation is to use the expressions given earlier to obtain the hour angle corresponding to each limit (elevation, azimuth, parallactic angle, zenith blind-spot), if reachable at the current declination. The effective limit is the one closest to, but later than, the current hour angle.

13.16 THE GEMINI POINTING MODEL

13.16.1 INTRODUCTION

13.16.1.1 POINTING

We use the term “pointing” here to mean knowledge of, and control over, the celestial coordinates corresponding to a specified place in the focal plane. Given the celestial coordinates of a target, various parts of the telescope can be moved so as to image that target onto the desired part of the focal plane; alternatively, given a location in the focal plane, an estimate of the corresponding celestial position can be made.

The mathematical model which describes the relationship between the sky and the focal plane falls naturally into two parts:

1. The astrometric calculations which link the catalog position of the target and its “observed place”. These are well-understood and will not be considered further here.
2. A representation of the geometry of the telescope structure and of the various misalignments, flexures and irregularities that are present.

In most existing telescopes, Part 2 of the model is a “black box” which generates corrections to the mount position as a function of either the observed place or the mount position. The mechanical details - for example whether a given shift is caused by flexure in the tube or movement in the secondary mirror support system - are irrelevant as far as pointing is concerned, and setting the telescope involves wholesale reorientation of the telescope tube by means of the mounting. This classical approach is not adequate for the Gemini telescopes, where stringent image-quality requirements mean that independent control of the primary and secondary mirrors is available (as well as mount movement), and where control of image position must, via M2 tip/tilt, extend beyond the bandwidth of the telescope mount.

13.16.1.2 FRAMES OF REFERENCE

At some level, we need to recognize a single coordinate system in which to express the positions and orientations of all the telescope components. Unfortunately, unwanted motions such as those caused by flexure complicate the realization of such a frame. What we have instead are various *local* coordinate systems, individually well-behaved but not tied together well enough for Gemini’s stringent needs. We can, for instance, reasonably expect M2 to be stable within the coordinate system of its control and metrology arrangements, but we certainly cannot rely on this coordinate system remaining fixed with respect to the M1 coordinate system as the telescope moves about. Similar remarks apply to the coordinate systems of the Cassegrain turntable (or rotator, called the CR for short from now on), of the instrument itself, and of the telescope tube assembly in the vicinity of the elevation bearings.

Each of these local coordinate systems can be defined through a set of x,y,z axes, or *triad*. For the purposes of our discussion, we will regard the z-axis as being “along” the optical axis, so that movements along the z-axis affect, in particular, focus. In each case, the x-axis and y-axis (order is unimportant here) correspond to “up/down” and “left/right”. Because “up” is a fairly well-defined concept, especially on an altazimuth telescope, and because the unwanted movements are likely to be small, the precise orientation of the x and y axes is relatively unimportant (for, say, servoing a mirror back to the right orientation). But what *is* critical is the orientation of the z-axis, as any translations or tilts with respect to the z-axis will be strongly coupled to movements of the image in the focal plane.

13.16.2 The Gemini Pointing Philosophy

As we have seen, the pointing philosophy for a telescope not equipped with active optics is relatively straightforward. The mechanical components do the best job they can to keep the optics aligned, and the relationship between the optical ensemble and the telescope mount can be expressed in terms of conventional “flexure” models.

Gemini is more complicated. Its active optics mean that optical alignment and pointing can both be affected in multiple ways.

The Gemini pointing philosophy is to work with respect to a clearly specified *master reference frame*, and to have individual models for each of the subsystems which describe the varying positions and orientations of their local coordinate systems relative to the master reference frame. The master reference frame that has been selected has a z-axis which is normal to the back of the primary-mirror cell in the vicinity of the CR bearing. Because this choice affects the character of the various subsystem-models and the behavior of the telescope, the reasons for it are presented below.

13.16.2.1 Choosing a master reference frame

The different places within the telescope which could conceivably be chosen as the basis of a master reference frame can be arranged in a sequence which moves out from the instrument into the telescope structure:

1. Some line passing through the “instrument aperture”.
2. The intersection of the focal plane with the (instantaneous) CR axis.
3. A line which passes through the CR axis but which remains normal to the back of the primary mirror cell, on which the CR is mounted.
4. The secondary mirror optical axis.
5. The primary mirror optical axis.

Option 1 is the purist’s choice, the only one which guarantees optimum image quality in the place that matters - the instrument aperture. However, it is located on a component which can flex and move, and in any case many instruments sample an extended field.

Option 2 is less dependent on the details of the instrument and its mounting arrangements than option 1. In principle, image quality is similarly compromised, but for any reasonable assumptions about the sizes of instrument shifts and flexures the effect is negligible. This option is discussed in more detail later.

Option 3 is a refinement of Option 2; it eliminates the entire instrument support structure including the CR bearing but represents a further departure from the ideal of optimizing image quality at the instrument aperture. This option is discussed in more detail later.

Option 4 has few advantages given that the secondary mirror has 5-axis control. In a telescope which had only tip/tilt control of M2 it might be a feasible choice.

Option 5 has few advantages given that the primary mirror has 5-axis control. In a telescope which had only tip/tilt control of M1 it might be a feasible choice, and would be a strong contender in a design where M1 support was passive.

The issues that guide our choice include:

- The practicability of modeling the various subsystems. How do we devise the models in the first place? How do we calibrate them in service?
- The independence of the models. If the telescope and the models diverge, can we diagnose the reasons? If different instruments or multiple optical configurations cause changes, can these be confined to just one of the component models?
- Avoiding unnecessary control movements. Unless image quality is threatened, we wish to avoid (for example) large M2 5-axis corrections synchronized with mount adjustments that merely compensate for the resulting pointing errors.
- Avoiding the need to move M1 quickly.
- The need to keep control mechanisms near the middle of their operating ranges.
- Matching intuition as far as possible.

13.16.2.2 OPTION 2 - THE CR AXIS

The CR axis is a feasible choice. It has the advantages of being realizable in practice and bypassing imperfections in the instrument support structure, but has disadvantages as well.

The mechanical imperfections in the CR bearing are of significant size when compared directly with the imaging and pointing goals of the Gemini telescope design. As the elevation changes, and as the CR turns, unwanted tilts at the 5 arcsecond level are to be expected. Because the distance between the bearing and the focus is small compared with the effective focal length (1.8 meters compared with 128 meters at f/16), these CR "wobbles" have only a tiny effect on the position of the image in the focal plane, and a negligible effect on image quality. Intuition would therefore suggest that little or no action was called for.

However, if the CR axis were chosen as the basis of the master reference frame, M1 and M2 would be slaved to this line, and so a 5 arcsecond tilt in the CR would translate directly into a 5 arcsecond pointing error, correction which would in turn require a 5 arcsecond mount movement. We thus have unnecessary control actions and possibly rapid movements of M1; there is also the risk of excessive translation of M2.

The CR “wobbles” are with respect to the comparatively rigid base on which the CR bearing is mounted, namely the back of the primary-mirror cell. When such a wobble occurs, the primary and secondary mirrors have both to move in order to remain centered on and normal to the instantaneous axis of rotation of the CR. Thus the defects in the CR (whether mapped as a function of the telescope orientation, predicted from earlier CR movements, or measured directly by mechanical sensors) affect the desired positions and orientations of the primary and secondary mirrors. Therefore the pointing maps for the primary and secondary mirrors contain information about the CR as well as information belonging exclusively to the mirror concerned. This is untidy and confusing. The extreme case would be where the mechanical structure supporting the two mirrors turned out to be near-perfect, so that no corrections were needed to stabilize the mirrors within the telescope, and yet the CR bearing was prone to wobbles. Under these circumstances, elaborate pointing maps would be needed for both mirrors, the same map in each case. This complicates modeling, calibration and adaptation to changes in one subsystem.

We conclude that Option 2 is unacceptable.

13.16.2.3 OPTION 3 - THE “MEAN CR AXIS”

The disadvantages of Option 2 disappear if instead of the instantaneous “wobbling” CR axis, some better-behaved “mean” CR axis is chosen instead, and this is Option 3. This option uses, in effect, the back of the M1 cell to tie down the master reference frame. The much greater mechanical rigidity of this telescope component compared with the CR makes it a better choice in most respects.

Advantages:

- It corresponds with intuitive ideas about where “the telescope” is pointing. The back of the M1 cell is likely to be tied firmly enough to the telescope mounting and to the M2 support framework for the residual “tube flexure” model to be stable and to be dominated by well-behaved terms of low spatial-frequency.
- It frees the M1 and M2 pointing models from pollution by CR-bearing wobbles, as well as flexures in the instrument support structure.
- The stiffness of the M1 cell makes Option 3 relatively immune from self-cancelling control actions.

Disadvantages:

- The proposed "mean CR axis" is not physically realizable. This is not a serious problem. An analogy is the "mean" celestial pole; this isn't physically realizable, but it is a convenient abstraction and allows the complexities of the nutation effect to be ignored for many purposes.
- The image quality will be optimized at the intersection of the mean CR axis and the focal plane, not at the instrument aperture. However, as for Option 2, the maximum displacements of the instrument aperture relative to the mean CR axis, due to CR wobbles, flexures in the instrument support structure, and so on, will be much too small to produce significant deterioration in image quality.

We conclude that Option 3 is the best choice.

13.16.3 FORMAL STATEMENT OF THE GEMINI POINTING PHILOSOPHY

We define the Gemini "optical support structure master reference frame", or *OSS triad*, as follows:

- The z-axis is normal to that portion of the back of the primary-mirror cell onto which the Cassegrain rotator bearing is attached.
- The x-y plane intersects the z-axis at the nominal detector position when the telescope is vertical.
- At the zenith, the field rotates around the (x,y,z)-origin as the Cassegrain rotator is turned through 360 degrees, in a "best-fit" sense.

with these corollaries:

- It is the task of the mount to point the OSS z-axis in the required direction.
- It is the task of the active optics system to maintain the required optical alignment and focus with respect to the OSS triad.

It is important to understand here that the "required" direction and the "required" optical alignment are not necessarily the direction of the target and the alignment for optimum imagery respectively:

- The required OSS triad orientation will differ from the target direction if the specified "pointing origin" (for example an instrument's entrance aperture) does not lie on the rotator axis, or if temporary mirror misalignments make it necessary to compensate by adjusting the mount orientation.
- Perfect optical alignment may be sacrificed if secondary mirror tip/tilt is being used to correct wind-shake or servo errors, or is being used to implement rapid scanning.

Each of the controllable subsystems concerned - mount, M1, M2 - can displace the image in the focal plane, and it is clearly necessary for these not to fight one another.

The implications of the Gemini pointing philosophy will be detailed in subsequent sections. The approach will be to look at the pointing effects of motions, relative to the OSS z-axis, of the instrument and its support structure, the primary mirror, and the secondary mirror.

Notes:

- Units are radians and meters throughout except where expressly given.
- In narrative passages, the plain-English terms “CR bearing”, “CR axis”, “focal plane” and so on strictly mean “OSS triad z-axis”, “OSS triad x-y plane” etc as appropriate. Similarly, the “optical orientation of the telescope” means the direction of the OSS triad z-axis.
- No attempt is made to employ a particular sign convention for the pointing corrections as this will depend on the individual coordinate systems in which the various translations and tilts are measured. The objective here is to describe the character and absolute size of each effect. The direction for each effect is expressed in words.
- The simple relationships described here are for small displacements and tilts of the various components so that, for example, questions of focal plane projection geometry do not arise. Also, where the possibility arises of correcting the pointing by changing the mount orientation, it is assumed that the resulting small change in telescope attitude does not, to first order, affect the size of the misalignment concerned.
- Issues to do with focus are not considered here.

13.16.4 MOTIONS OF THE INSTRUMENT

Motions of the instrument with respect to the OSS triad can happen because of slop, flexure and other mechanical deficiencies of the CR bearing, and flexure in the instrument support arrangements. Small rotations of the instrument about the an axis parallel to that of the CR, and tip/tilt or piston movements of the instrument relative to the focal plane, do not cause appreciable pointing errors. However, a small translation of the instrument relative to the origin of the OSS triad will cause the following pointing error:

R = 0.0078*TI

where R is the size of the pointing error and TI is the translation distance. (The factor 0.0078 is simply the reciprocal of the telescope focal length in meters.)

Therefore, a translation of the instrument in the focal plane by 100 microns will cause a pointing error of 0.78 microradians or 0.16 arcseconds. For a fixed optical orientation of the telescope, a “downwards” slip of the instrument means the pointing direction moves upwards on the sky.

One particular way in which translation can occur between the instrument and the OSS triad z-axis is when the “true” axis of rotation of the Cassegrain turntable tilts with respect “mean” CR axis which is fixed with respect to the back of the primary-mirror cell. The 1.8 meter distance between the CR bearing and the focal plane acts as a “lever arm”, amplifying a movement of the bearing of (say) 5 arcseconds into a 44 micrometer translation of the instrument, corresponding to about 0.07 arcseconds.

Most instruments and support structures as currently specified include no control of the instrument position within the focal plane (the wavefront sensors are an exception). It is therefore important that as in any other telescope these devices are stiff and stable.

Given that we have decided to slave the telescope optics to the OSS triad, if the instrument moves relative to this frame, then mount movement will be required to restore correct pointing. If the instrument were to shift downwards 100 microns, the telescope would have to be moved down in elevation 0.16 arcseconds to bring the target back onto the instrument aperture.

13.16.5 MOTIONS OF THE PRIMARY

The motions of the primary mirror relative to the OSS triad which will cause pointing errors are:

- translation in X/Y relative to the OSS triad
- tip/tilt relative to the OSS triad z-axis

Rotation of the primary mirror about the CR axis and piston movements do not cause pointing errors.

13.16.5.1 TRANSLATION OF THE PRIMARY

Translation of the primary mirror relative to the OSS triad will cause the following pointing error:

R = 0.0078*TP

where TP is the amount of sideways movement.

Therefore, a translation of the primary mirror by 100 microns will cause a pointing error of 0.78 microradians or 0.16 arcseconds. The motion of the image in the focal plane and the direction of motion of the primary mirror are in the same direction. For a fixed optical orientation of the telescope, a "downwards" slip of the mirror causes the pointing direction to move downwards on the sky. As we have control of the translation of the primary mirror, this pointing error would normally be corrected by translating the mirror upwards by 100 microns. If for any reason control of primary mirror translation was unavailable, the pointing would have to be corrected by rotating in elevation to raise the telescope by 0.16 arcseconds.

13.16.5.2 Tip/Tilt of the Primary

Tilt of the primary mirror relative to the CR axis will cause the following pointing error:

$$R = 2*RP$$

where RP is the amount of tilt.

Therefore, with everything else fixed, a tilt of the primary mirror of 1 arcsecond will cause a pointing error of 2 arcseconds. For a given optical orientation of the telescope, a "downwards" tilt of the primary mirror (i.e. the elevation of the mirror axis decreases) causes the pointing direction to move downwards on the sky. As we have control of the tilt of the primary mirror this pointing error would normally be corrected by tilting the mirror upwards by 1 arcsecond. If for any reason control of primary mirror tilt was unavailable, the pointing would have to be corrected by rotating in elevation to raise the telescope by 2 arcseconds, or by some combination of mount movement with secondary mirror translation and tilt.

13.16.6 MOTIONS OF THE SECONDARY

The motions of the secondary mirror relative to the OSS triad which will cause pointing errors are:

- translation
- tip/tilt

Rotation of the secondary mirror about the OSS triad z-axis and piston movements do not cause pointing errors.

13.16.6.1 TRANSLATION OF THE SECONDARY

Translation of the secondary mirror relative to the OSS triad will cause the following pointing error:

R = 0.0616*TS

where TS is the size of the translation.

Therefore, a translation of the secondary mirror by 100 microns will cause a pointing error of 6.16 microradians or 1.27 arcseconds. For a fixed optical orientation of the telescope, a "downwards" translation of the secondary mirror causes the pointing direction to move upwards on the sky. As we have control of the translation of the secondary mirror, this pointing error would normally be corrected by moving the mirror upwards by 100 microns. If for any reason control of secondary mirror translation was unavailable, the pointing would have to be corrected by rotating in elevation to lower the telescope by 1.27 arcseconds, or some combination of this and other control movements.

13.16.6.2 TIP/TILT OF THE SECONDARY

Tilt of the secondary mirror relative to the OSS triad will cause the following pointing error:

R = 0.2584*RS

where RS is the amount of tilt.

Therefore, a tilt of the secondary mirror of 1 arcsecond will cause a pointing error of 0.2584 arcseconds. For a fixed optical orientation of the telescope, a "downwards" tilt of the secondary mirror (i.e. one where, with everything else in correct alignment, the axis of the secondary mirror falls below the center of the primary mirror) causes the pointing direction to move downwards on the sky. As we have control of the tilt of the secondary mirror, this pointing error would normally be corrected by tilting the mirror upwards by 1 arcsecond. If for any reason control of secondary mirror tilt was unavailable, the pointing would have to be corrected by rotating in elevation to raise the telescope by 0.2584 arcseconds or by some combination of this and other control movements.

13.16.7 MOUNT POINTING MODEL

The Gemini pointing philosophy can be regarded as a straightforward extension of the classical telescope. The z-axis of the OSS triad is the counterpart of the entire tube and optics of the classical telescope, and the arrangements for stabilizing the optical components relative to that axis constitute a "smart structure" of infinite stiffness.

If the telescope were perfect, the optical orientation would be given directly by the azimuth and elevation encoders and would be identical to the target's "observed place". However, various properties and deficiencies of the telescope will cause a difference between the actual optical orientation and that inferred from the encoder readings. This difference is what the mount pointing model estimates. The effects that such a model attempts to describe include various geometrical properties and inaccuracies, errors in encoders and their drive trains, and distortions and other irregularities in the azimuth and elevation bearings.

The form and size of the various pointing terms which affect the alignment (actual and desired) of the OSS triad z-axis can be determined in the classical way, through observations of stars of known position and the fitting of models which allow both geometrical and empirical terms.

The core of the Gemini pointing model (as for all altazimuth designs) will be the six geometrical terms:

- azimuth encoder zero point
- elevation encoder zero point
- non-perpendicularity between the elevation axis and the specified pointing origin
- non-perpendicularity between the azimuth and elevation axes
- azimuth axis tilt north-south
- azimuth axis tilt east-west

Further details of the likely models are unimportant at this stage, but it can be confidently stated that (i) theoretical terms like the above six, (ii) empirical terms and (iii) lookup tables will all play a part. It is, incidentally, likely that one consequence of the Gemini active optics system will be that the "tube flexure" terms will be much smaller than for passive telescope designs.

The Gemini control software will need to include automated procedures for performing pointing tests from the outset, but off-line analysis using existing tools such as TPOINT (reference Starlink User Note 100.9, P.T.Wallace, 1994) will be acceptable initially. Later it may prove desirable to have tailored pointing-analysis tools built into the Gemini software. In addition to comprehensive pointing tests, inbuilt facilities will be needed to perform rapid recalibration of low-order terms such as "collimation" errors and encoder zero points, using a small number of stars.

13.16.8 PRIMARY AND SECONDARY MIRROR POINTING MODELS

From finite-element-analysis and from tests during commissioning, pointing models for the primary and secondary mirrors will be developed. These will predict translation and tip/tilt, relative to the OSS triad, and also piston information for focus corrections. The input variables will include:

- Mount orientation - presumably elevation alone as azimuth dependence would be surprising.
- Temperature differentials in the telescope structure (perhaps).
- Sensor readings showing mechanical movements directly.

The form of the models has not at this stage been predicted. Simple trigonometrical or polynomial formulas will prove adequate, but also possible that lookup tables will be needed.

A possible strategy for developing models is as follows:

1. Mount an alignment telescope in place of the instrument and observe a reflecting target or a collimator placed in the center of the CR bearing. By observing the displacements and tilts at different combinations of telescope elevation and turntable position angle, devise a model to describe instrument translation as a function of elevation and position angle.
2. Move the reflecting target or collimator to the hole in the middle of M1 and map the 4-axis translation and tip/tilt corrections needed to maintain M1 alignment as the telescope moves in elevation (and perhaps azimuth).
3. Move the reflecting target or collimator to the hole in the middle of M2 and map the 4-axis translation and tip/tilt corrections needed to maintain M2 alignment as the telescope moves in elevation (and perhaps azimuth).
4. With the three models in place, carry out Shack-Hartmann tests to investigate variations in image quality. As well as allowing the required pattern of M1 support forces to be investigated, these tests may expose deficiencies in the M1 and M2 pointing maps. Depending on the statistical significance of the analyses, the adjustments to these maps should ideally be constrained to be "pointing-neutral".
5. Ordinary pointing tests can then be carried out to establish the mount pointing model.
6. Wavefront checks carried out in service will yield information about focus and both image position (Zernike coefficients 1 and 2) and image quality (coefficients 3 and above). Image quality should be allowed to influence the M1 and M2 models only in a "pointing- neutral" manner, while image position should be allowed to affect only the mount model.

Very rapid changes to the mount model (resulting from windshake in particular) will normally have the effect of causing fast M2 tip/tilt activity, with the mount itself supplying the low-frequency de-trending corrections.

If predicted or measured M1, M2 and instrument pointing models are not available by the time of commissioning, dummy ones should be set up which assume that no mirror translations or tilts occur as the telescope is being used. The zero coefficients in these "null" models can then be adjusted in accordance with the flow of readings from the wavefront sensors, to improve alignment during the course of the observation.

The rates at which the primary and secondary mirror pointing models will be applied will be such that there will not be significant delays at the ends of slews before full image quality is restored. Assuming reasonable values for the sizes of the corrections, it seems highly unlikely that an easily-attainable update rate of 1 Hz will be too slow, and this rate is recommended.

13.16.9 POINTING CALCULATIONS

The telescope pointing model will need to be applied at the full 20 Hz rate of the TCS control loop. However, some terms will change so slowly as a function of telescope position that they could be recomputed at a much lower frequency if this were desirable. Other terms, in particular the "collimation" terms, are liable to be changed abruptly as part of normal telescope operation and have to be recomputed in full at the 20 Hz rate.

The design proposed in the mid-1980s for the Keck telescope (ref Keck pointing report, P.T.Wallace) included the construction of special 3 x 3 matrices, called "osculating transformation matrices" or OTMs. These are an interpolating device, derived by sampling the pointing corrections at several places around the target. Their use means that small telescope movements such as sidereal tracking or offsetting are automatically corrected for the differential of the pointing model without the need for a fresh recomputation at the full update rate. Experience with the OTM scheme on Keck, UKIRT and WHT shows that it produces adequate accuracy and saves significant amounts of computer time. However, with the faster computers used by Gemini it is not obvious that the OTM scheme is required, and there are grounds for dispensing with it. Recomputations of slowly-changing items such as the precession/nutation matrices or the refraction coefficients will in any case continue to be done relatively infrequently.

One aspect of the telescope pointing model that certainly will need recomputing at the full 20 Hz rate is the group of so-called "collimation" corrections. These describe essentially the discrepancy between the "pointing origin" (i.e. the place in the focal plane we wish the target image to fall upon) and the normal to the focal plane which passes through the elevation axis. The latter will be close to the Cassegrain turntable axis, but there will in general be some residual displacement and this can be absorbed into the offsets to the pointing origin. The vertical offset will be absorbed into the elevation encoder zero- point correction; the other (sideways) component is called the "collimation" error. One way in which the TCS will implement nodding is to allow the pointing origin to be changed instantaneously, which will cause the telescope to move rapidly so as to align the target to the new pointing origin.

Notes:

- For pointing origins far from the optical axis (i.e. the origin of the OSS triad), proper projection geometry will need to be employed. This may involve tangent-plane projection plus a power-law description of the pincushion/barrel distortion.
- The pointing-origin coordinates will be expressed in the x,y plane of the OSS triad. Transformation to elevation and collimation errors will take into account rotator position-angle.
- It is geometrically convenient to include azimuth/elevation nonperpendicularity and any residual "tube flexure" effects in the collimation calculations even though they change only slowly.

13.17 TCS Internal Subsystems

The following sections describe the major internal subsystems to the TCS and how they interact with each other.

13.17.1 TCS State Machine

The TCS state machine provides the interface between the TCS EPICS database and the rest of the system. It interprets the commands which are placed in the database by the OCS and moves the TCS from state to state.

The state machine is made up of the following subsystems:

- TCS State Processor - which handles the OCS sequencer commands
- TCS Command Processor - which handles the fine grained commands
- TCS Screen Handler - which calculates the screen items based on values from the underlying subsystems

13.17.2 TCS Pointing Engines

There are separate pointing engines for the telescope and all the different wavefront sensor probes.

Each pointing engine takes demand RA/DEC and outputs demand Az,El,Ψ. The telescope demand is used to drive the mount, instrument rotator and the enclosure position. The WFS demand are differenced with the telescope demand in order to calculate the position within the focal plane.

13.17.3 Optical Control Engine

The telescope demand (Az, El) is used to calculate the demand wavefront and position of the primary mirror (sometimes it may be better to make the enclosure follow a different target, for example the center of a scan pattern that the telescope is executing). Similarly the telescope demand (Az, El) is used to calculate the demand position of the secondary mirror.

Mirror figure sensing and the computation of current wavefront are accomplished in the TCS using a wavefront sensor in the A&G or, optionally, in the Science Instrument. The A&G system shall provide information about the current wavefront and is not responsible from generating primary mirror specific commands. The TCS looks at the current wavefront information, compare it to the demanded wavefront, and compute and applies the correction required. The wavefront data may come from a variety of external sources - all of the sources present data to the TCS and the TCS decides which data source to use and what algorithm to use. The TCS monitors coordinates (orientation) and rotates WFS data appropriately.

Operation is transparent to the observer after initialization. The operator selects the desired settings of the alignment, support, and thermal control systems at the start of an observing run on the basis of measurements. Default settings may be used from previous nights. The initial settings are varied based on lookup tables that incorporate both the altitude and azimuth of the mount as well as current structural temperatures and the current settings are changed based on the TCS receiving appropriate wavefront data from a sensor.

Pointing accuracies due to the position and figure of the primary are included in and consistent with the previously defined telescope pointing and tracking requirements. Actuator motions during tracking are coordinated to avoid image motion exceeding the error budget.

13.17.3.1 SOURCE OF WAVEFRONT DATA

In active operation an external wavefront sensor such as a Shack-Hartmann supplies information to the TCS. This information is in a self defined format that is independent of the source of the data.

In adaptive operation the majority of the wavefront corrections are done by the deformable mirror in the Adaptive Optics system. However the active operation data that are the DC and low temporal frequency wavefront corrections may be passed on the TCS in the same self defined format as above. The intent is that the PCS neither knows nor cares where the wavefront information comes from.

13.17.3.2 Initial Calibration of System

The calibration WFS in the A&G unit is used to obtain reference Zernike measurements at different telescope positions (mostly in elevation). These reference Zernike are then used to command the primary mirror in open loop mode.

13.17.3.3 Use of Wavefront Data during Operations

While the A&G system is delivering low frequency wavefront information to the TCS (active optics loop closed) the TCS does the following calculation:

- use open loop model from initial calibration to predict demand surface
- use in offsets determined from last use of calibration WFS
- use information from A&G WFS as an additional correction
- rotates data appropriately to apply to systems for correction (primary, secondary position, etc.)

The TCS adds up these components, expresses them as a demand surface expressed as Zernike and delivers them to the primary mirror control system.

13.17.4 Wind/Thermal Management Engine

There are two main functions to be handled here. The predicted start of night temperature is used to precondition the primary mirror as well as set the cooling temperature for the enclosure - during the night these are continuously adjusted. The current wind conditions are used to determine the amount of enclosure flushing but adjusting the wind gates, fans, and louvers.

13.17.5 TCS Handset

The decision has been taken not to provide a separate electrical handset which bypasses the TCS and directly inputs signals to the drives in order to do real time handset control. The main drivers for this decision were:

- the cost of providing such a control
- the increased complexity in the control system to allow an input at such a low level

- the lack of any real need to provide such a control electrically

The decision as to where to locate the handset in the software is a compromise. Similar to the TCS consoles, which exist in the OCS and communicate with the TCS through the OCS observing database, a case can be made for connecting the handset to the OCS. However the OCS is not a real time system - so there is no way to guarantee deterministic performance. We have connected the TCS handset to the TCS.

13.17.6 Handset Modes

There are a number handset modes, each of which can be selected using appropriate radio button on the handset screen. The functions divide naturally into three groups:

- ALT_AZ, RA_DEC and X_Y the demand position in the input coordinate system and differ only in the directions of the increments. The increments displayed are therefore the accumulated values from all three modes in the coordinate system of the current mode. The tracking position on the information display also changes.
- OFFSET and APOFF are used to set up positional and aperture offsets interactively. The tracking coordinates change in the OFFSET case but not in the APOFF case.
- FOCUS and ROTATOR move individual mechanisms.

14 DETAILS OF THE MOUNT CONTROL SYSTEM

14.1 FUNCTION OF THE MOUNT SYSTEM

The mount control system (MCS) provides the basic ability to slew and track the telescope. It also interfaces to a number of secondary systems that are required to provide services. The MCS is intended as an engineering interface to the mount and its subsystems — it is intended that the telescope could be run from here for initial setup and engineering work. It is not intended as an interface where the telescope would meet specification. It is essential to the success and longevity of the Gemini project that the details of these devices are hidden from the rest of the system. For instance, it should be possible to feed a continuous stream of demanded positions, velocities, and accelerations to the servo system without knowing whether it can actually make use of this information.

14.2 MOUNT CONTROL OVERVIEW

14.2.1 External Interfaces

The Mount Control System (MCS) has the following external interfaces:

- Telescope Control System
- the MCS Control Console(s) provided by the Observatory Control System
- the Observatory Control Screens system
- the MCS Engineering display
- the mount hardware

- the MCS initialization files

14.2.2 External Bus Connections

The MCS is connected to the following external buses:

- Interlock System
- Time Bus
- Control LAN

14.2.3 Internal Subsystems

The Mount Control System is composed of the following major subsystems:

- Servos Subsystem
- Drives Subsystem
- Encoders Subsystem
- Interlock Subsystem
- Protection Subsystem
- Counterweights
- Service Wrap-ups
- Monitoring & Metrology Subsystem
- Electrical Systems Interface
- In Situ M1 Cleaning Interface
- M1 Dry Gas Flush System Interface [TBD]
- Mirror Covers
- Top-End Ring
- Telescope Thermal Control Subsystem

The Mount Control System is not responsible for control of the Mount Brakes or Hydrostatic Bearings as these are part of a separate work package (Interlock, Brakes, and Bearings Control System). For further details please reference ICD 12 (Interlocks) and SDD Chapter 22 (Details of the Interlock, Brakes, and Bearings Control Ssytem).

14.2.4 Internal Interfaces

The MCS has the following major internal interfaces:

- Interaction between EPICS database and EPICS sequencer 'state machine' program(s).

14.2.5 Internal Data Stores

The MCS has the following major internal data stores:

- EPICS Database

14.2.6 Computer Hardware

There is a VME crate dedicated to mount control located on the mount base. This VME crate may have multiple CPU's in order to handle all of the functionality required. The VME crate control CPU (MVME-167 M68040) will not be directly responsible for closing the servo loop between the azimuth and elevation drive motors and their respective encoders. This function will be provided by the Gemini standard DC Servo controller board (DeltaTau PMAC-VME) which is based on the Motorola DSP56001.

There is a two headed Sun Workstation permanently connected to the mount control VME crate via ethernet in order to control the system from the rotating enclosure floor/telescope mount. A light-tight cover will be provided for this console as not to impact the required enclosure darkness during observing.

FIGURE 14 - 1 **Mount Control System Electronic Interfaces**

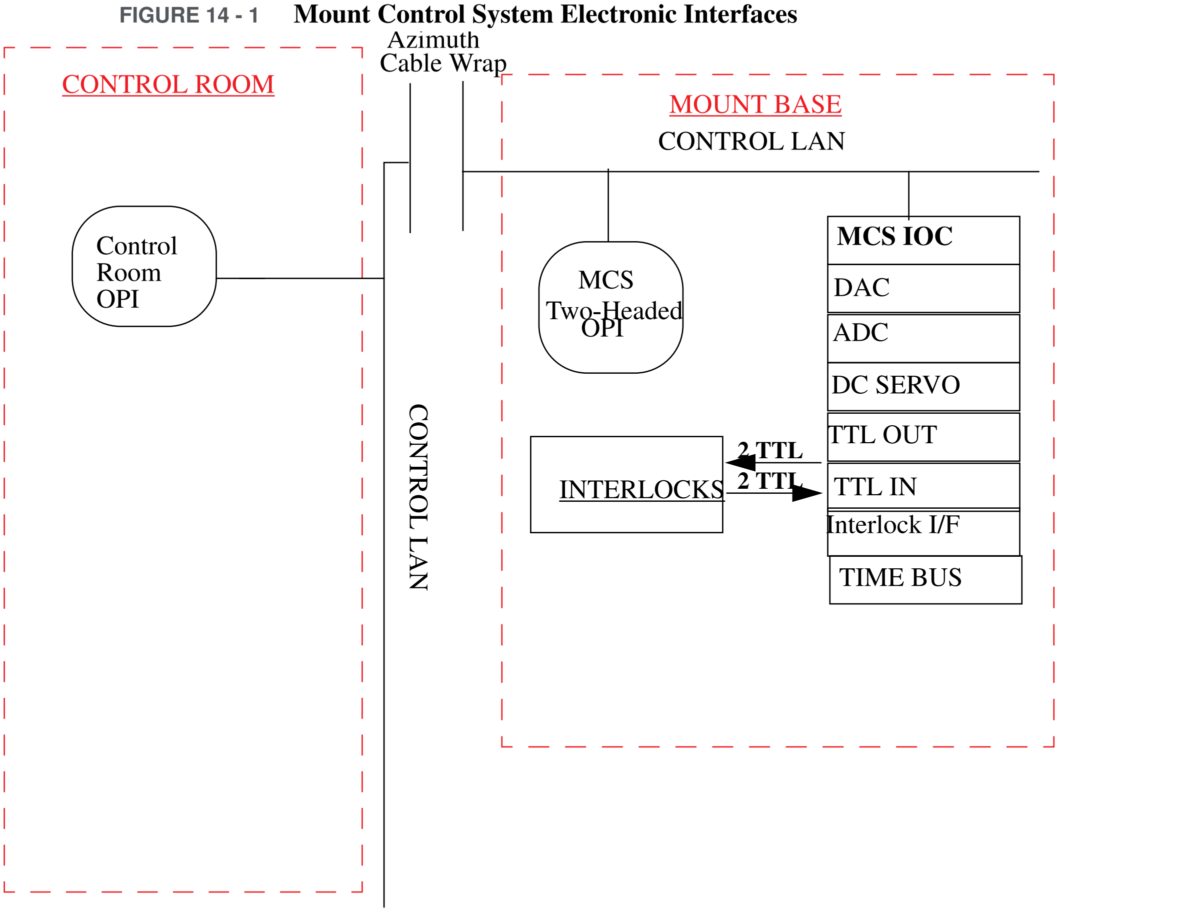
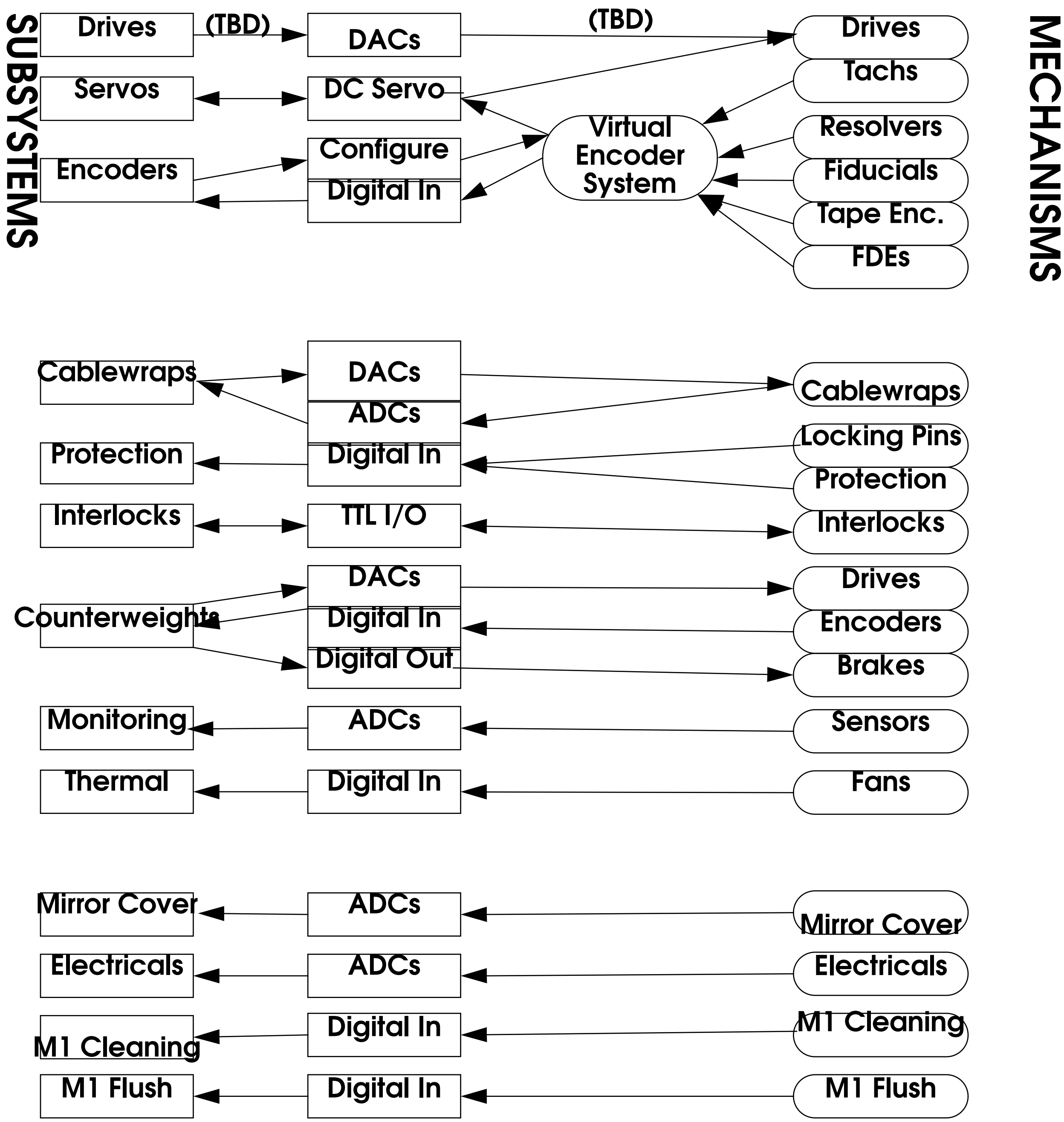


FIGURE 14 - 2 **Mount Control System Subsystems**



14.2.7 Software Philosophy

The main objective is that the mount control system should be capable of running just given a stream of position, velocity, and acceleration information from the Telescope Control System — all of the intelligence is in the TCS.

14.2.8 Referenced Documents

14.2.8.1 GEMINI 8M TELESCOPE CRITICAL DESIGN REVIEW (RPT-TE-G0018)

The following sections of this document are referred to in this chapter.

--6.3.2 (a,b,g) [Drives]

--6.3.3 (a) [Encoders]

--6.3.3.1 (a-f) [Tape Encoders with Fiducials]

--6.3.3.3 (a-c) [Friction Driven Encoders]

--6.3.5 (a-d) [Over-travel Stops]

--6.3.6 (a-e) [OSS Counter Balance]

--6.3.7 (a-d) [Cable Transfers, Runs, and Disconnects]

--13.3 (a-g) [Interlocks]

--13.3.{1-13}(a-c) [Details of Specific Interlocks]

14.2.8.2 GEMINI 8M TELESCOPE INTERFACE CONTROL DOCUMENTS

The following interface control documents are referred to in this chapter.

--Telescope Hard Stop Angular Limits, Decelerations (ICD-G-0004)

--Telescope Maximum Velocity,Accel.,Min. Decel. Requirements (ICD-G-0005)

14.2.8.3 HYDROSTATIC BEARING SYSTEM FABRICATION SPECIFICATIONS (SPE-TE-G0042)

14.2.9 Referenced Drawings

14.2.9.1 GEMINI 8M TELESCOPE CRITICAL DESIGN REVIEW (#87-GP- SERIES)

--0501-0001 [Azimuth Drive Assembly]
--0502-0001 [Elevation Drive Assembly]
--0603-0003 [Azimuth Tape Encoder Assembly]
--0603-0004 [Azimuth Proximity Sensor Assembly]
--0604-0003 [Elevation Tape Encoder Assembly]
--0604-0004 [Elevation Proximity Sensor Assembly]
--0702-0001 [Azimuth Over-Travel Stops]
--0704-0001 [Elevation Over-Travel Stop Assembly]
--0800-0001 [Elevation Counter Balance Assembly]
--1000-0002 [Elevation Cable Wrap Assembly]
--1000-0005 [Azimuth Cable Wrap and Twister Assembly]

14.3 EXTERNAL INTERFACES

14.3.1 Interface to Telescope Control System

14.3.1.1 REQUIREMENTS

The mount control system accepts azimuth and elevation parameters {A,A',A''} {E,E',E''} for the target position, velocity, and acceleration of the two mount axes. These parameters may be sent once or they may be continuously delivered at up to 20 Hz.

The nodding performance requirements are to complete a nod of up to 5" (in telescope, not sky, coordinates) within 1 second for nodding periods of 5 seconds and to complete a nod of 60" within 5 seconds for longer nodding periods.

The MCS knows nothing about nodding, but responds to changes in demand positions from the TCS. The remaining concern would be the mechanism by which the nod completion would be synchronized with the ICS systems. This is an interface issue between the TCS and ICS.

In addition to the targets the mount accepts the commands:

- STOP; halt and put the brakes on
- MOVE; go to the current target and stop

- FOLLOW; go to the current target and follow the target continuously
- STOP FOLLOWING; stop where you are
- PARK; move to a predetermined position and put telescope in park state
- ENCODER CONFIG; configure the encoders
- ZERO SET; zero different parts of the system
- DRIFT; go to the current target and drift with a specified constant velocity
- UNWRAP; take out the cable wrap
- LOG ON/OFF, enable and disable logging for diagnostic purposes

The mount returns the following values:

- Encoder readings
- Drive demand
- Servo errors
- Alarms
- Status (e.g. “in position”)

14.3.1.2 IMPLEMENTATION

The above data values are contained in an EPICS database that communicates with the TCS via channel access.

14.3.2 Interface to OCS Consoles

14.3.2.1 REQUIREMENT

The consoles provided by the OCS must have fine grained control capability of the mount.

The following commands are available:

- All Servo System functions
- All Protection System functions
- All Counterweights System functions
- All Service Wrap-ups System functions
- All Monitoring and Metrology System functions
- All Electrical System Interface System functions

- All In Situ M1 Cleaning Interface System functions
- All M1 Dry Gas Flush Interface System functions
- All Mirror Cover System functions
- All Top-End Ring System functions
- All Telescope Thermal Control System functions

The following commands will be accessed via the Servo System:

- All Drive functions

The following commands will be accessed via OCS screens.

- All Encoder System functions
- All Interlock System functions

14.3.2.2 IMPLEMENTATION

The TCS uses EPICS Channel Access to set the appropriate parameters in the MCS's database and then issues an event that causes the MCS to match the system with those parameters.

14.3.2.3 SECURITY

When the MCS is in Engineering Mode and being controlled by an operator stationed at the MCS OPI located at the mount base most other accesses will be disallowed by use of the channel access security mechanism available in EPICS R3.12 and above. For monitoring and logging purposes channel access reads from the MCS database of selected process variables will be allowed.

14.3.2.4 SAMPLE CONSOLES

The console shown below is a sample of what the OCS provides to the operator for commissioning and interactive observing.

FIGURE 14 - 3 **Mount Control Console**

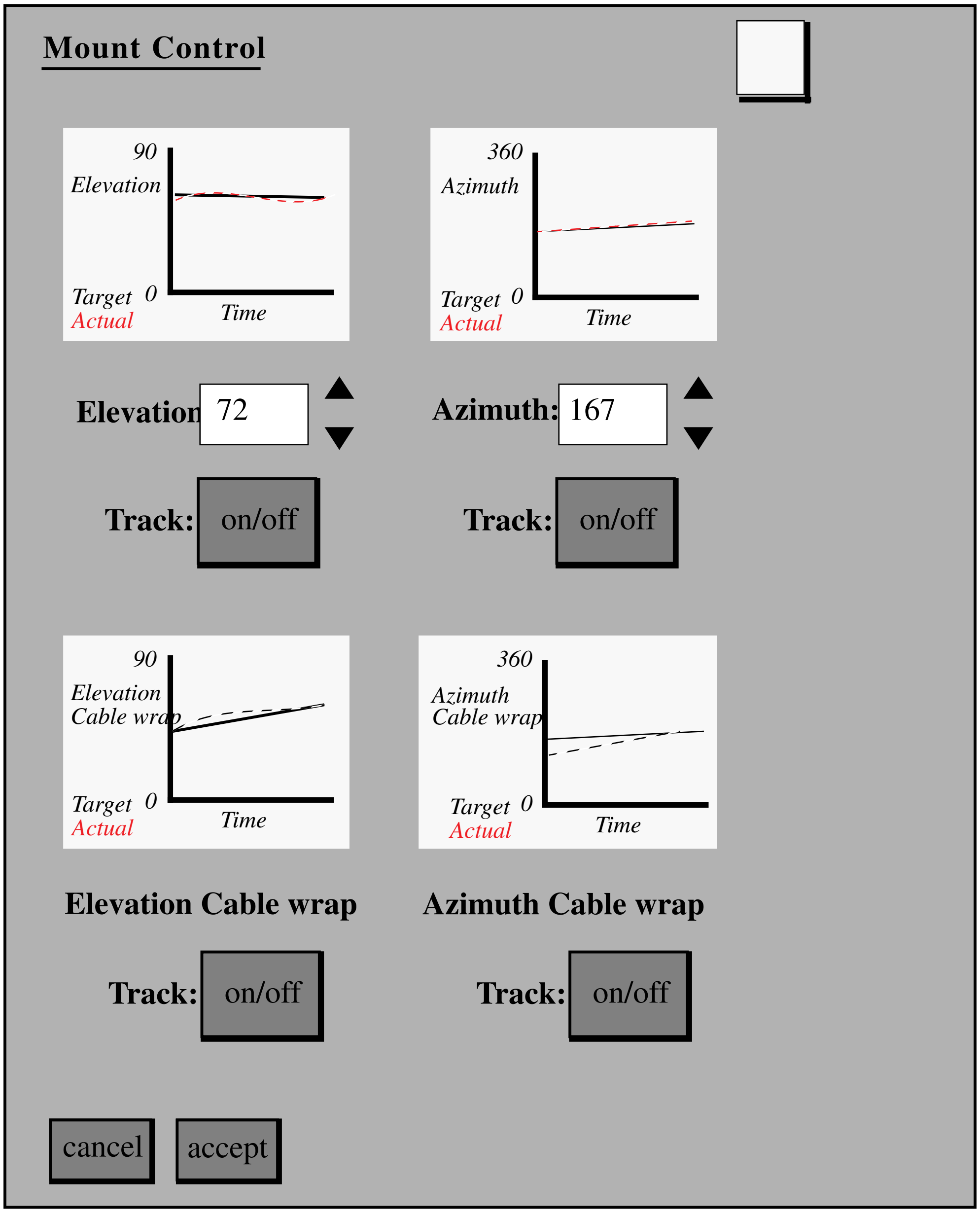
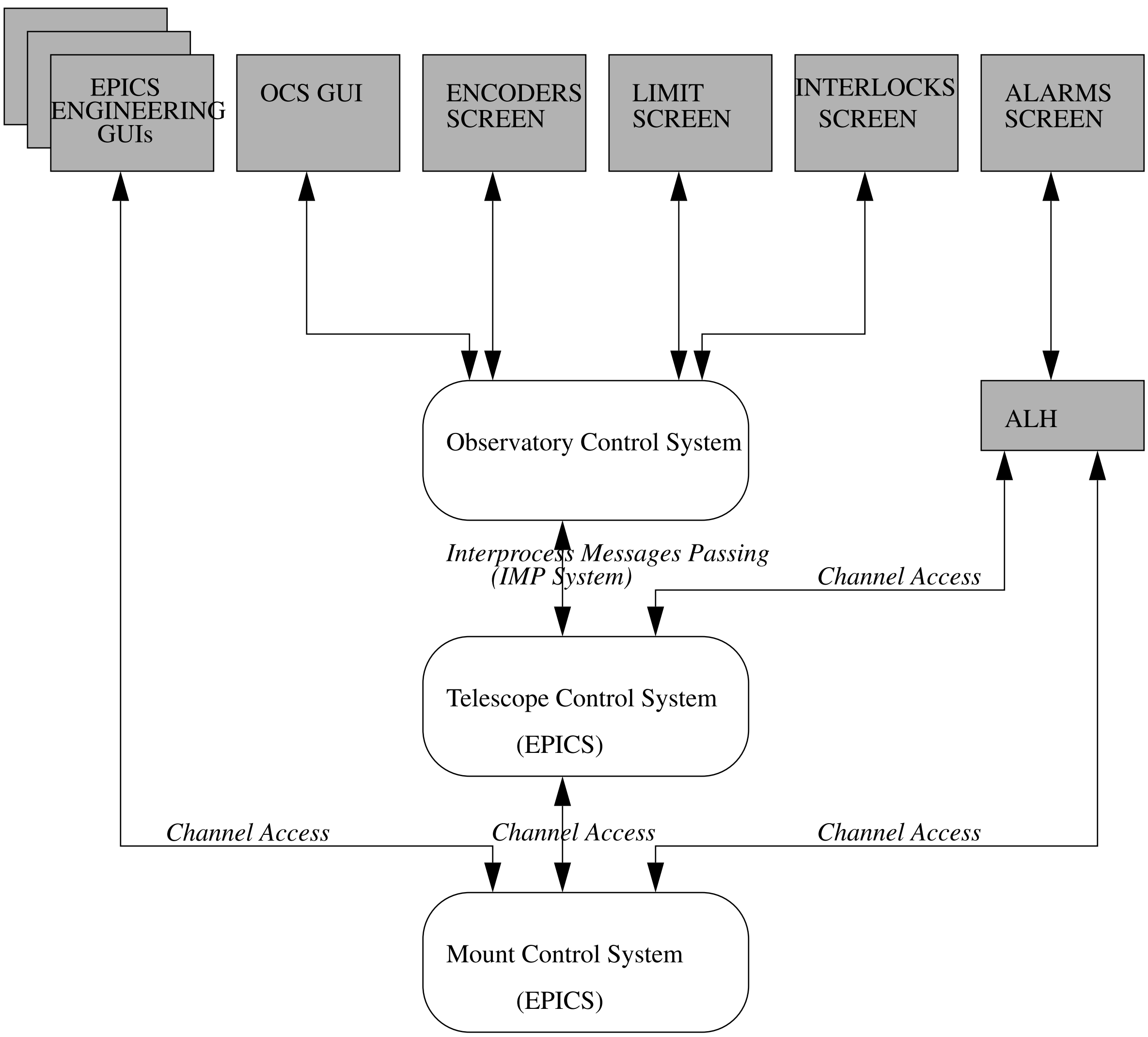


FIGURE 14 - 4 Mount Control System Software



14.3.3 Interface to OCS Screen System

The mount control system provide the following *items* to the different *screens*.

14.3.3.1 ENCODER SCREEN

- Azimuth

--Incremental FDE

--Drive unit resolvers. There are 8 of these.

--Incremental inductive tape encoder. There are 4 reading heads.

--Fiducials

--Tachometers

--Drive motor currents

- Elevation

--Incremental FDE

--Drive unit resolvers. There are 4 of these.

--Incremental inductive tape encoder. There are 2 reading heads.

--Fiducials

--Tachometers

--Drive motor currents

- Axial Counterweight Position

--Absolute encoders. There are 2 of these — one for each counterweight assembly.

- Cross-axial Counterweight Position

--Absolute encoders. There are 2 of these — one for each counterweight assembly.

- Azimuth Wrap

--Encoders

- Elevation Wrap

--Encoders

14.3.3.2 LIMIT SCREEN

--Azimuth limits.

These are *hardware* and *software* limits and should not be encountered in normal (computer-controlled) operation. The soft-limits are hit first, and cause the system to revert to engineering mode. The telescope can only be moved/pushed out of a damper limit by hand. The display should show CLEAR for all limits and in normal operation and SET (with the message in reverse video) if the limit has been hit.

The nominal limit positions are:

--DAMPER LIMIT

--HARD LIMIT

--SOFT LIMIT

--COMPUTER LIMIT

- Elevation limits.

  As for azimuth, except that the final limit refers to the top end exchange position, which can only be reached under engineering-mode control from the observing floor.

  The nominal limit positions are:

  --DAMPER LIMIT

  --HARD LIMIT

  --SOFT LIMIT

  --COMPUTER LIMIT

  --TOP END EXCHANGE LIMIT

- Azimuth Velocity limits
- Elevation Velocity limits
- Mirror Cover Status
- Top End Status
- Axial Counterweight limits (2).
  The nominal limit positions are:

  --+ LIMIT

  --- LIMIT

- Cross-axial Counterweight limits (2)
  The nominal limit positions are:

  --+ LIMIT

  --- LIMIT

14.3.3.3 ALARMS SCREEN

- Hydraulic and lubrication system.

  --Oil pad alarm. Indicates high or low pressure at one of the hydraulic support pads.

  --Elevation and azimuth oil filter alarms.

  --Oil temperature.

  --Oil pressure.

  --Oil level.

- Power.

--Mains alarm

--Power amplifier temperature alarm

--Power amplifier over-current alarm.

14.3.3.4 INTERLOCKS SCREEN

--Azimuth zone.

This gives the position of the topple bracket microswitches that are read on start-up to resolve the ambiguity in the azimuth encoder.

--Access/zenith park ties.

These are inserted to stop the telescope moving when it is out of balance; e.g. the mirror cell has been removed or during a top end change.

--Revolving floor latch.

This stops the telescope from moving in azimuth when it is at the correct position for the mirror to be removed

--Elevation Drive Interlocks #1-5
These will stop the telescope from moving in Elevation under certain conditions.

--Azimuth Drive Interlocks #1-3
These will stop the telescope from moving in Azimuth under certain conditions.

--Elevation Locking Pin Interlock
This will prevent removal of Elevation locking pin when primary mirror cell is removed.

--Top End Interlock
This will prevent removal of the top end when the Elevation locking pin is disengaged. This uses a keyed interlock: locking pin inserted enables removal of key; key inserted into the top-end enables top-end latches to be operated.

--Elevation Maintenance Interlock
This will stop the telescope from moving in Elevation when the Elevation Maintenance key is removed.

--Azimuth Maintenance Interlock
This will stop the telescope from moving in Azimuth when the Azimuth Maintenance key is removed.

--Emergency Stop Buttons
These will prevent the telescope from moving in Elevation and Azimuth and also prevent the enclosure from rotating when enabled.

14.3.4 Interface to Mount Subsystems

TABLE 14 - 1 General Functions

Function	Description	Range	Frequency	Type
Power	turn power on/off to entire unit	ON | OFF	< 0.1 Hz	digital switch
Health	monitor overall status of device	OK | BAD	1 Hz	digital i/p
Servo Parms	up/down load servo parameters	TBD	< 0.1 Hz	digital i/o
HeartBeat	incrementing counter	32 bits	20-50Hz	digital i/p
WatchDog	shutdown mount systems if host dies		20-50Hz	digital i/o

TABLE 14 - 2 Servo System Functions

Function	Description	Range	Frequency	Type
AZ Target	commanded azimuth angle	+/- 270 degrees	20-50Hz	DC Servo - output
AZ Velocity Target	commanded azimuth velocity	+/- 2.0 degrees/sec	20-50Hz	DC Servo - output
AZ Accel. Target	commanded azimuth acceleration	+/- 0.1 degrees/ sec^2	20-50Hz	DC Servo - output
AZ Actual	actual azimuth angle	+/- 270 degrees	20-50Hz	DC Servo - input
AZ Actual Velocity	actual azimuth velocity	+/- 2.0 degrees/sec	20-50Hz	DC Servo - input
AZ Actual Accel.	actual azimuth acceleration	+/- 0.1 degrees/ sec^2	20-50Hz	DC Servo - input
AZ IN POS	azimuth angle ‘in position’ radius	+/- TBD degrees	20-50Hz	Output
AZ Velocity IN POS	azimuth velocity ‘in position’ radius	+/- TBD degrees/sec	20-50Hz	Output
AZ Accel. IN POS	azimuth acceleration ‘in position’ radius	+/- TBD degrees/ sec^2	20-50Hz	Output
AZ Error	azimuth angle error	+/- TBD degrees	20-50Hz	DC Servo - input

TABLE 14 - 2 Servo System Functions

Function	Description	Range	Frequency	Type
AZ Velocity Error	azimuth velocity error	+/- TBD degrees/sec	20-50Hz	DC Servo - input
AZ Accel. Error	azimuth acceleration error	+/- TBD degrees/ sec^2	20-50Hz	DC Servo - input
El Target	commanded elevation angle	0-93 degrees	20-50Hz	DC Servo - output
El Velocity Target	commanded elevation velocity	+/- 0.75 degrees/sec	20-50Hz	DC Servo - output
El Accel. Target	commanded elevation acceleration	+/- 0.025 degrees/ sec^2	20-50Hz	DC Servo - output
El Actual	actual elevation angle	0-93 degrees	20-50Hz	DC Servo - input
El Actual Velocity	actual elevation velocity	+/- 0.75 degrees/sec	20-50Hz	DC Servo - input
El Actual Accel.	actual elevation acceleration	+/- 0.025 degrees/ sec^2	20-50Hz	DC Servo - input
El IN POS	elevation angle ‘in position’ radius	+/- TBD degrees	20-50Hz	Output
El Velocity IN POS	elevation velocity ‘in position’ radius	+/- TBD degrees/sec	20-50Hz	Output
El Accel. IN POS	elevation acceleration ‘in position’ radius	+/- TBD degrees/ sec^2	20-50Hz	Output
El Error	elevation angle error	TBD degrees	20-50Hz	DC Servo - input
El Velocity Error	elevation velocity error	+/- TBD degrees/sec	20-50Hz	DC Servo - input
El Accel. Error	elevation acceleration error	+/- TBD degrees/ sec^2	20-50Hz	DC Servo - input

TABLE 14 - 3 **Drive Functions**

Function	**Description**	**Range**	**Frequency**	**Type**
Azimuth Drive Motors (4x2)	command a drive rate	0-2 Degrees/ Sec.	20-50Hz	analog o/p
Azimuth Drive Motors (4x2)	monitor drive current	TBD Amps	20-50Hz	analog i/p
Azimuth Tachometers (4x2)	monitor tachometer readout	0-450 RPM	20-50Hz	digital i/p
Azimuth Resolvers (4x2)	monitor drive unit resolvers	TBD	20-50Hz	digital i/p
Azimuth Pre-load (4)	monitor hydraulically applied pre-load	TBD	20-50Hz	analog i/p
Elevation Drive Motors (2x2)	command a drive rate	0-2 Degrees/ Sec.	20-50Hz	analog o/p
Elevation Drive Motors (2x2)	monitor drive current	TBD Amps	20-50Hz	analog i/p
Elevation Tachometers (2x2)	monitor tachometer readout	0-450 RPM	20-50Hz	digital i/p
Elevation Resolvers (2x2)	TBD	TBD Degrees	20-50Hz	digital i/p
Elevation Pre-load (2)	monitor hydraulically applied pre-load	TBD	20-50Hz	analog i/p
Interlock Interface	Connection to interlock system. Needs to remove azimuth and elevation drive power if interlock is set.		1Hz	Interlock Interface

TABLE 14 - 4 Encoder System Functions

Function	Description	Range	Frequency	Type
Configure Azimuth Virtual Encoder	Instruct virtual encoder system how to combine raw encoder values into 32-bit quantity.	TBD	TBD	TBD
Azimuth Virtual Encoder	Encoder system mixes tape/fidicuals/FDE into a single virtual encoder.	32 Bit	20-50Hz	digital i/p
Azimuth Tape Encoders (4)	Inductosyn incremental tape encoder readout.	TBD Counts	20-50Hz	digital i/p
Azimuth Fiducials (36 or 72)	monitor reference signals	TBD	20-50Hz	digital i/p
Azimuth FDE	FDE incremental encoder readout	TBD counts	20-50Hz	digital i/p
Azimuth FDE Loadcell	FDE incremental encoder loadcell readout	TBD	20-50Hz	analog i/p
Configure Elevation Virtual Encoder	Instruct virtual encoder system how to combine raw encoder values into 32-bit quantity.	TBD	TBD	TBD
Elevation Virtual Encoder	Encoder system mixes tape/fidicuals/FDE into a single virtual encoder.	32 Bit	20-50Hz	digital i/p
Elevation Tape Encoders (2)	Inductosyn incremental tape encoder readout.	TBD Counts	20-50Hz	digital i/p
Elevation Fiducials (11 or 22)	monitor reference signals	TBD	20-50Hz	digital i/p
Elevation FDE	FDE incremental encoder readout	TBD counts	20-50Hz	digital i/p
Elevation FDE Loadcell	FDE incremental encoder loadcell readout	TBD	20-50Hz	analog i/p

TABLE 14 - 5 **Interlock System Functions**

Function	**Description**	**Range**	**Frequency**	**Type**
elevation Drive Interlock #1a	Low pressure on elevation hydrostatic bearings prevents power to drive motors.	SET | CLEAR	1Hz	Interlock Monitor
Elevation Drive Interlock #1b	Low oil pressure on elevation hydrostatic bearings prevents drive motor preloads.	SET | CLEAR	1Hz	Interlock Monitor
Elevation Drive Interlock #2	Elevation brakes engaged prevents power to drive motors. This is completely passive.	SET | CLEAR	1Hz	Interlock Monitor
Elevation Drive Interlock #3	Elevation locking pin engaged prevents power to drive motors.	SET | CLEAR	1Hz	Interlock Monitor
Elevation Drive Interlock #4	Elevation manual drive engaged prevents power to drive motors.	SET | CLEAR	1Hz	Interlock Monitor
Elevation Drive Interlock #5	Reaching elevation hard end of travel prevents power to drive motors.	SET | CLEAR	1Hz	Interlock Monitor
Azimuth Drive Interlock #1a	Low pressure on azimuth hydrostatic bearings prevents power to drive motors.	SET | CLEAR	1Hz	Interlock Monitor
Azimuth Drive Interlock #1b	Low oil pressure on azimuth hydrostatic bearings prevents drive motor preloads. This is completely passive.	SET | CLEAR	1Hz	Interlock Monitor
Azimuth Drive Interlock #2	Azimuth brakes engaged prevents power to drive motors	SET | CLEAR	1Hz	Interlock Monitor
Azimuth Drive Interlock #3	Reaching azimuth hard end of travel prevents power to drive motors.	SET | CLEAR	1Hz	Interlock Monitor
Elevation Locking Pin Interlock #1	Removal of Primary Mirror Cell prevents power to elevation locking pin.	SET | CLEAR	1Hz	Interlock Monitor
Top End Interlock #1	Disengagement of the elevation locking pin prevents power to top end latching mechanism.	SET | CLEAR	1Hz	Interlock Monitor
Elevation Maintenance Interlock #1	Removal of elevation maintenance key prevents power to drive motors and brakes.	SET | CLEAR	1Hz	Interlock Monitor

TABLE 14 - 5 Interlock System Functions

Function	Description	Range	Frequency	Type
Azimuth Maintenance Interlock #1	Removal of azimuth maintenance key prevents power to drive motors and brakes.	SET | CLEAR	1Hz	Interlock Monitor
Emergency Stop Buttons	Pressing any Emergency Stop button will remove power from all drives, engage all brakes, and sound audible and visual alarms.	SET | CLEAR	1Hz	Interlock Monitor

14.3.4.1

TABLE 14 - 6 Protection System Functions

Function	Description	Range	Frequency	Type
Azimuth Over-Travel Stop Condition(2)	Microswitches tied to each azimuth topple bracket indicate when travel limits (+/-270 degrees) have been reached.	SET | CLEAR	20-50Hz	digital input switch
Elevation Over-Travel Stop Condition (2)	Microswitches tied to each elevation strike bracket indicate when travel limits (15 (or 0) to 90 degrees) have been reached.	SET | CLEAR	20-50Hz	digital input switch
Elevation Sensor (?)	???Mercury switches???	TBD Degrees	20-50Hz	digital i/p
Azimuth Over-Velocity Limit	Limit is a function of azimuth.	TBD Degrees/ Sec	20-50Hz	digital i/p
Elevation Over-Velocity Limit	Limit is a function of elevation.	TBD Degrees/ Sec	20-50Hz	digital i/p
Azimuth Over-Velocity Condition	Has azimuth drive exceeded current velocity limit?	SET | CLEAR	20-50Hz	digital input switch
Elevation Over-Velocity Condition	Has elevation drive exceeded current velocity limit?	SET | CLEAR	20-50Hz	digital input switch

TABLE 14 - 6 **Protection System Functions**

Function	Description	Range	Frequency	Type
Azimuth Cable Twister Locations	Microswitches at upper and lower limits of rise indicate cable twister location.	SET | CLEAR	20-50Hz	digital input switch
Elevation Over Travel Stop Bracket Status	Microswitch indicates configuration of elevation over travel stop bracket (observing mode, maintenance mode).	SET | CLEAR	20-50hz	digital input switch
Elevation Locking Pin Status	Microswitch monitors status of elevation locking pin latching mechanism	SET | CLEAR	20-50Hz	digital input switch

TABLE 14 - 7 **Counterweights System Functions**

Function	Description	Range	Frequency	Type
Axial Balancer Target (2)	Demanded counterbalance weight position.	0-970 mm	1Hz	digital o/p
Axial Balancer Position (2)	Actual counterbalance weight position via encoder.	0-970 mm	1Hz	digital i/p
Axial Balance End-of-travel Condition(2x2)	End of travel (+/-) limit switch status.	SET | CLEAR	1Hz	digital input switch
Axial Balancer Brakes (2)	Counterweight braking system.	ON | OFF	1Hz	digital output switch
Cross-axial Balancer Target (2)	Demanded counterbalance weight position.	0-770 mm	1Hz	digital o/p
Cross-axial Balancer Position (2)	Actual counterbalance weight position via encoder.	0-770 mm	1Hz	digital i/p

TABLE 14 - 7 Counterweights System Functions

Function	Description	Range	Frequency	Type
Cross-axial Balance End-of-travel Condition (2x2)	End of travel (+/-) limit switch status.	SET | CLEAR	1HZ	digital input switch
Cross-axial Balancer Brakes (2)	Counterweight braking system.	ON | OFF	1Hz	digital o/p

TABLE 14 - 8 Service Wrap-ups System Functions

Function	Description	Range	Frequency	Type
Azimuth Cable Twister Control	Provides direction and velocity control for the azimuth cable twister drive motor.	TBD Amps	20-50Hz	analog o/p
Azimuth Cable Twister Encoder	Differential encoder readback.	TBD	20-50Hz	digital i/p
Azimuth Cable Twister Drive Current	Monitor cable twister motor drive current.	TBD Amps.	20-50Hz	analog i/p
Elevation Cable Wrap Control (2)	Provides direction and velocity control for the elevation cable wrap drive motors.	TBD Amps	20-50Hz	analog o/p
Elevation Cable Wrap Encoder	Differential encoder readback.	TBD	20-50Hz	digital i/p
Elevation Cable Wrap Drive Current	Monitor cable wrap motor drive current.	TBD Amps.	20-50Hz	analog i/p

TABLE 14 - 9 **Monitoring & Metrology Subsystem**

Function	Description	Range	Frequency	Type
Gyros	Gyroscope network.	TBD	1Hz	digital i/p
Temperature Sensors	Temperature sensor network.	TBD	1Hz	digital i/p
Strain Gauges	Strain Gauge network	TBD	1Hz	digital i/p
Accelerometers	Accelerometer network.	TBD	1Hz	digital i/p
Tilt meters	Tilt meter network.	TBD	1Hz	digital i/p
Displacement sensors	Displacement sensor network. Measures displacement of trunnion relative to columns.	TBD	1Hz	digital i/p

TABLE 14 - 10 **Electrical Systems Interface System Functions**

Function	Description	Range	Frequency	Type
UPS Voltage Level	Monitor UPS voltage level.	TBD V	1Hz	analog i/p
UPS Frequency content	Monitor UPS frequency content - online spectral analysis.	TBD	1Hz	array analog i/p
UPS Frequency Stability	Monitor UPS frequency stability of line. (60 Hz Mauna Kea, 50 Hz Cerro Pachon).	TBD	1Hz	analog i/p
Non-UPS Voltage Level	Monitor Non-UPS voltage level.	TBD V	1Hz	analog i/p
Non-UPS Frequency content	Monitor Non-UPS frequency content - online spectral analysis.	TBD	1Hz	array analog i/p
Non-UPS Frequency Stability	Monitor Non-UPS frequency stability of line. (60 Hz Mauna Kea, 50 Hz Cerro Pachon).	TBD	1Hz	analog i/p
Servo Amp Voltage Level	Monitor Servo Amp voltage level.	TBD V	1Hz	analog i/p

TABLE 14 - 10 Electrical Systems Interface System Functions

Function	Description	Range	Frequency	Type
Servo Amp Frequency content	Monitor Servo Amp frequency content - online spectral analysis.	TBD	1Hz	array analog i/p
Servo Amp Frequency Stability	Monitor Servo Amp frequency stability of line. (60 Hz Mauna Kea, 50 Hz Cerro Pachon).	TBD	1Hz	analog i/p

TABLE 14 - 11 In-Situ M1 Cleaning Interface System Functions

Function	Description	Range	Frequency	Type
Cleaning Interlock #1	Performing primary mirror laser cleaning disables all mount drives and engages all mount brakes. This could be the same as the normal interlock system. Required to move mount once to clean second half of mirror.	SET | CLEAR	<0.1Hz	Interlock Monitor
Status	The state of the mirror cleaning system will be monitored.	ON | OFF	<0.1Hz	digital input switch

TABLE 14 - 12 M1 Dry Gas Flush Interface System Functions [TBD]

Function	Description	Range	Frequency	Type
Flushing Interlock #1	Performing primary mirror dry gas flush disables all mount drives and engages all mount brakes. This could be the same as the normal interlock system. Is this interlock required?	SET | CLEAR	<0.1Hz	Interlock Monitor
Status	The state of the dry gas flushing system will be monitored.	ON | OFF	<0.1Hz	digital input switch

TABLE 14 - 13 Mirror Cover Functions

Function	Description	Range	Frequency	Type
Mirror Cover Position	Monitor mirror cover position.	OPEN | CLOSED	<0.1Hz	digital i/p

TABLE 14 - 14 **Top-End Ring Functions**

Function	Description	Range	Frequency	Type
Top-End Ring Status (4)	Monitor top-end ring position.	ON | OFF	<0.1Hz	digital i/p

TABLE 14 - 15 **Telescope Thermal Control Subsystem**

Function	Description	Range	Frequency	Type
Mount Fan Status (TBD)	The fans on the mount base will be monitored.	ON | OFF	<0.1Hz	digital input switch
Azimuth Drive Fan Status (4)	The fans on the azimuth drive assemblies will be monitored.	ON | OFF	<0.1Hz	digital input switch
Elevation Drive Fan Status (4)	The fans on the elevation drive assemblies will be monitored.	ON | OFF	<0.1Hz	digital input switch

14.4 EXTERNAL BUS CONNECTIONS

14.4.1 Synchronization Bus

The MCS is not connected to the Synchro bus as its timing needs, < 20 Hz, can be met by having the TCS process wave front information and pass it to the mount in its normal command stream.

14.4.2 Interlock System

The MCS must be capable of generating interlock requests as well as responding to them. This is probably implemented as input and output TTL signals.

14.4.3 Time Bus

The time within the MCS VME crate is set to +/- 5 microseconds from the time bus.

14.5 MCS INTERNAL SUBSYSTEMS

14.5.1 Servo System

There are two servo systems needed to point and track the telescope line of sight — elevation and azimuth. The third axis, the cassegrain rotator, is needed to maintain field orientation and is the subject of a separate work package under the control of the Instrument Group. The elevation and azimuth servo systems accept a stream of position, velocity, and acceleration commands generated externally at a 20-50 Hz frequency. The servo systems may use the past several commands to extrapolate the current command at a higher frequency, 200 - 500 Hz, required by the electronics. The servo electronics, baselined to be a digital signal processor, are executing the servo algorithms. In addition to accepting these commands the servo subsystem responds to the generic instrument commands.

14.5.1.1 DC SERVO

The Gemini standard DC servo system is the DeltaTau PMAC-VME DC Servo controller that is based on the Motorola 56001 DSP. The primary interface to this DC Servo controller is via an EPICS DC_SERVO record as defined by the Gemini Standard Instrument Controller Work Package undertaken at RGO. The details are TBD...

14.5.2 Drives

In operation the telescope elevation and azimuth drives are controlled by the servo subsystem. It may also be possible to directly control the drives in a stand-alone mode by directly connecting a workstation or terminal. The intent is to use the capabilities provided by the manufacturer of the drives and not to create or add functionality to these systems. If there is no built in capability for direct control of the drives then none shall be provided.

[6.3.2 (a)] Both the azimuth and elevation axes employ friction-type drive systems to supply the required torques for all telescope slewing, tracking, and offsetting operations. To simplify design and manufacture, the two drive systems use identical motor housings and motors (i.e. identical rollers, bearings, motors, and housing design).

14.5.2.1 AZIMUTH DRIVE SYSTEM

[6.3.2 (b)] As shown in drawing #87-GP-0501-0001, the azimuth drive system consists of four drive motor units located underneath the mount base and driving against the outside diameter of the azimuth track journal. Each motor drive unit consists of two brushless DC motors, two tachometers, two resolvers, two cylindrical rollers, eight angular contact ball bearings, and a steel motor housing. A hydraulically applied pre-load is applied at each drive unit.

The motor drive part specifications are:

--Drive motors shall be brushless DC, type: Inland BMS-11801.

--Servo-amplifiers shall be type: Inland BLMI-325.

--Tachometers shall be: Inland TG-5723. (One per drive motor).

--Resolvers shall be: BEI p/n #25 incremental optical encoder. (One per drive motor)

The analog motor board receives a motor rate command (not a current command) from the digital filter. This rate command is through a 16-bit D/A converter, so the resolution is max_rate/2^16, which is around 5.1×10^{-5} rad/sec. The max_rate as measured on the motor shaft is ± 1.7 rad/sec which divided by 48 gives the maximum drive rate of ± 35 rad/sec (± 2 degrees/sec) on the mount axis.

The tachometer is an analog device and it is kept in an analog loop. It does have an analog noise of about 0.1% of signal (~10 bits). The maximum rate for this tachometer is 47 rad/sec = 450 RPM, which is quite large for our purposes.

The worst case runaway velocity for the azimuth axis is 4.3 degrees/sec.

The analog motor board uses encoders (resolvers) to measure the position of the motor shaft in order to smoothly commutate the motor. The resolution on these encoders are a vendor decision but the specification is that these encoders must be whatever is necessary to limit the motor torque variation to 2% (~6 bits).

There shall be a mechanism in place by which the actual motor drive current can be monitored by the control system. The details of this are TBD.

14.5.2.2 ELEVATION DRIVE SYSTEM

[6.3.2 (g)] As shown in drawing #87-GP-0502-0001, the elevation drive system consists of two drive units attached to the mount columns and driving against the outside radii of the two elevation disks. The design of the drive motor units (housing design, motors, bearings, etc...] is essentially identical to that of the azimuth drives.

The worst case runaway velocity for the elevation axis is 2.1 degrees/sec.

14.5.2.3 ENCODER SYSTEM

The encoders for the elevation and azimuth axes are a combination of mechanical switches, magnetic position sensors, tape and FDE incremental encoders, and resolvers. The intent of the encoder subsystem is to hide the details of the physical encoding scheme and provide a device independent virtual encoder to higher level systems. This virtual encoder is baselined to be a 32-bit encoder. It must be possible to directly connect a computer to the encoder subsystem in order to run diagnostics and tests.

In normal operation of the MCS servo loops, the 32-bit output from the encoder subsystem will provide the position information for the DSP-based servo system.

For the Gemini telescope azimuth axis, the required resolution is 0.005 arcseconds over -/+270 degree range which is to one part in 388,800,000 or 28.5 bits.

The encoder subsystem will be able to supply the MCS IOC with the following values:

--Calculated 32-bit virtual encoder value

--Mechanical switch status

--Magnetic position sensor status

--Incremental tape encoder values

--Incremental FDE values

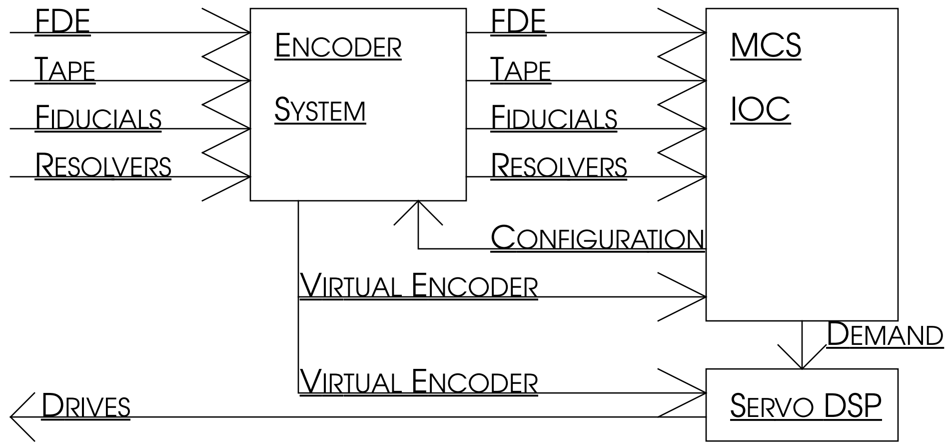
--Drive resolver values

The encoder subsystem will be able to accept the following commands from the MCS IOC:

--Configure

The actual configuration parameters are TBD. The requirement is that the encoder system must be able to dynamically alter its algorithm for computation for the virtual encoder value in response to the changing status of the hardware.

FIGURE 14 - 5 **Encoder Subsystem I/O Connections**



14.5.2.4 AZIMUTH TAPE ENCODERS

[6.3.3 (a)] Incremental Inductosyn tape encoders, with absolute fiducials, have been selected as the primary encoding system. Friction driven encoder will be provided as a back up encoder system. The encoders will use the outside diameter of the azimuth track journal, as a running surface, and as a mounting surface for the stationary portion of the encoder.

[6.3.3.1 (b)] As shown in drawing #87-GP-0603-0003 the Gemini azimuth axis tape encoder is mounted to the outside diameter (9600mm) of the azimuth track journal. Four read heads are arranged in diametrically opposed pairs which are, in turn, mounted orthogonal to each other (i.e. one read head located every 90 degrees). These four read heads are attached to the azimuth encoder brackets and are provided with rollers which maintain the required 0.25mm air-gap between the read head and the tape surface. Pre-amplifiers and A-to-D converters are attached under the base at each read head location.

14.5.2.5 ELEVATION TAPE ENCODERS

[6.3.3.1 (c)] As shown in drawing #87-GP-0604-0003 the elevation axis also employs an Inductosyn tape encoder system to provide rate and position information. A tape segment is mounted onto the outside diameter of each elevation disk. An air groove system similar to that of the azimuth tape encoder is used to ensure even tape tension. The read heads are mounted to the mount column cross tube. With only one read head present on each disk there may also be a requirement for a transducer.

14.5.2.6 AZIMUTH FIDUCIALS

[6.3.3.1 (d)] Proximity switches are used on both the azimuth and elevation axes to supply discrete, absolute position information to the control system. Non-contacting, Sony Magneswitches have been selected for providing this fiducial information. The repeatability of these switches are +/- 1 micron, which corresponds to an angular position accuracy of 0.043 arcseconds and 0.052 arcseconds at diameters of 9600mm (azimuth track) and 8000mm (elevation disk), respectively. These fiducials should also be located approximately in the center of a pitch.

[6.3.3.1 (e)] The azimuth axis employs 36 switches attached to the outside diameter of the azimuth track journal at ten degree increments. As shown in drawing #87-GP-0603-0004, provision for an additional 36 switches is included in the journal design and would allow a switch located at every five degrees. A single magnet assembly is located on one of the azimuth encoder brackets. This multiple-switch/single-magnet arrangement provides an individual signal to the control system for every switch location. The output of the fiducial system is used to continually update the control system's telescope position information.

14.5.2.7 ELEVATION FIDUCIALS

[6.3.3.1 (f)] The elevation axis employs 11 switches, shown in drawing #87-GP-0604-0004, attached to the outside edge of one of the elevation disks. These switches are located at ten degree increments. Provision for an additional 11 switches is included in the disk design and would allow a switch to be located at every five degrees, A single magnet assembly is attached to the mount column cross tube and allows for a discrete signal to the control system for every switch location.

14.5.2.8 RATIONAL FOR FRICTION DRIVEN ENCODERS

[6.3.3.3 (a)] Friction driven encoders (FDEs) will be used on both the azimuth and elevation axes (in addition to the tape encoder system) for the following reasons:

--FDEs provide a simple and inexpensive data check for the tape encoder.

--FDEs are relatively straightforward, reliable devices to install and operate. It should be noted, however, that FDEs are prone to slippage and hysteresis; because of this they are not intended to be used as the primary encoder system.

--FDEs are frequently used in operation on world-class telescopes with generally good results, while tape encoders have not.

14.5.2.9 AZIMUTH FRICTION DRIVEN ENCODERS

[6.3.3.3 (b)] The azimuth and elevation FDE systems use a ‘ground referenced’ design that was tested by Gemini and is now in use on the WIYN 3.5m telescope (a ground reference FDE mount is constrained in all degrees of freedom except radially). The Gemini FDE uses a Heidenhain ROD 800c incremental encoder with 100x interpolation. The roller diameter is approximately 100mm and has a spherical rolling surface. This encoder design incorporates a steering, or alignment mechanism with load cell to help minimize the effects of misalignment. As the encoder begins to drift due to a misalignment, the load cell is able to detect the corresponding build-up of restoring force. By iteratively measuring this axial force and adjusting the steering mechanism, an accurate fine adjustment of roller alignment can be made. The azimuth FDE is mounted on the azimuth encoder bracket and rides against the outside diameter of the azimuth rack journal. Depending on the choice of interpolation electronics, a theoretical angular resolution of approximately 0.004 to 0.015 arcseconds is possible. Non-repeatable errors, however, are perhaps more important and have been quantified by both extensive laboratory-type testing and measurement of on-telescope performance (WIYN telescope).

14.5.2.10 ELEVATION FRICTION DRIVEN ENCODERS

[6.3.3.3 (c)] The elevation FDE is mounted on the mount column cross tube and rides against the outside radius of one of the elevation disks. The theoretical angular resolution is approximately 0.0045 to 0.018 arcseconds. Non-repeatable errors should be of the same order of magnitude as for the azimuth FDE.

In the advent of the brakes being engaged via an interlock condition or other loss of power situation the brakes can be released manually via a hydraulic system. The exact details of this are TBD. The hook-up and control of the hydraulic system will be done at the actual brake location.

14.5.3 Interlocks

14.5.3.1 ELEVATION DRIVE MOTORS DRIVEN WHEN THE ELEVATION HYDROSTATIC BEARINGS ARE NOT FULLY PRESSURIZED

[13.3.1 (a)] **Hazard:** If the elevation drive motors are driven when the elevation hydrostatic bearings are not fully pressurized damage could be caused to the drive rollers, elevation disks, and/or hydrostatic bearing system.

[13.3.1 (b)] **Interlocks:** Two safety interlocks provided to prevent this occurring:

1. If the elevation hydrostatic bearings are not fully pressurized, the elevation drive motors shall be prevented from receiving electrical power.
2. If the elevation hydrostatic bearings do not receive fully pressurized oil, the elevation drive motor preloads shall not be provided.

[13.3.1 (c)] **Hardware Description:** Two hardware systems are provided to prevent this occurrence:

1. Two pressure transducers which provide electrical contact (continuity) when a minimum, preset pressure is attained, shall be located at the two elevation bearing oil manifolds located on top of each mount column. The transducers shall be wired in series; a low oil pressure condition which is sensed by either one or both of the transducers shall prevent electrical power from reaching both elevation drive motor assemblies.
2. A hydraulic preload cylinder shall be used on each elevation drive motor assembly to provide the normal force required for drive traction. This cylinder shall receive pressurized oil from the elevation bearing oil manifolds.

14.5.3.2 ELEVATION DRIVE MOTORS ACTIVATED WHEN ELEVATION BRAKES ARE ON (ENGAGED)

[13.3.2 (a)] **Hazard:** Elevation drive motors activated when elevation brakes are on (engaged). This condition could cause damage to drive rollers, elevation disks, and/or brake system.

[13.3.2 (b)] **Safety Interlocks**: If the elevation brakes are on (applied), the elevation drive motors shall be prevented from receiving electrical power.

[13.3.2 (c)] **Hardware Description:** Elevation brakes are ‘fail safe’; i.e. spring on (engaged), hydraulic powered off (disengaged). A pressure transducer, which provides electrical contact (continuity) when a minimum, preset oil pressure is attained, shall be fitted to the brake manifold. A low pressure condition which is sensed by the transducer shall prevent electrical power from reaching both elevation drive motor assemblies.

14.5.3.3 ELEVATION DRIVE MOTORS ACTIVATED WHEN ELEVATION LOCKING PIN ENGAGED

[13.3.3 (a)] **Hazard:** Elevation drive motors activated when elevation locking pins are engaged. This condition could cause damage to drive rollers, elevation disks, and/or locking pin system.

[13.3.3 (b)] **Safety Interlocks:** If the elevation locking pins are engaged, the elevation drive motors shall be prevented from receiving electrical power.

[13.3.3 (c)] **Hardware Description:** A electrical micro-switch shall be located on the elevation locking pin assembly. This switch shall provide electrical contact (continuity) to the elevation drive motor assemblies only if the locking pin assembly is disengaged (retracted).

14.5.3.4 ELEVATION DRIVE MOTORS ACTIVATED WHEN ELEVATION MANUAL DRIVE ENGAGED

[13.3.4 (a)] **Hazard:** Elevation drive motors activated when elevation manual drive system is engaged. This condition could cause injury to personnel working on the manual drive system. In addition this condition could cause damage to manual drive system and/or drive system.

[13.3.4 (b)] **Safety Interlocks:** If the elevation manual drive system is engaged, the elevation drive motors shall be prevented from receiving electrical power.

[13.3.4 (c)] **Hardware Description**: A electrical micro-switch shall be located on the elevation manual drive assembly. This switch shall provide electrical contact (continuity) to the elevation drive motor assemblies only if the manual drive assembly is disengaged (retracted).

14.5.3.5 ELEVATION END OF TRAVEL SENSED

[13.3.5 (a)] **Hazard:** Elevation drive motors activated when telescope tube has rotated to end-of-travel limits (zenith, horizon). This condition could cause damage to drive system and/or OSS assembly.

[13.3.5 (b)] **Safety Interlocks:** If an elevation 'hard' limit is sensed, electrical power to the elevation drive motors shall be removed.

[13.3.5 (c)] **Hardware Description:** A electrical micro-switch shall be located at each of the elevation 'hard' rotation limits.

14.5.3.6 AZIMUTH DRIVE MOTORS ACTIVATED IF AZIMUTH HYDROSTATIC BEARINGS ARE NOT FULLY PRESSURIZED

[13.3.6 (a)] **Hazard:** Azimuth drive motors activated if azimuth hydrostatic bearings are not fully pressurized. This condition could cause damage to drive rollers, azimuth journal, and/or hydrostatic bearing system.

[13.3.6 (b)] **Safety Interlocks:** Two interlocks are provided:

1. If the azimuth hydrostatic bearings are not fully pressurized, the azimuth drive motors shall be prevented from receiving electrical power.
2. If the azimuth hydrostatic bearings do not receive fully pressurized oil, the azimuth drive motor preloads shall not be provided.

[13.3.6 (c)] **Hardware Description:** Two pressure transducers which provide electrical contact (continuity) when a minimum, preset pressure is attained, shall be located at four of the azimuth bearing oil manifolds located underneath the mount base. The transducers shall be wired in series; a low oil pressure condition which is sensed by either one or both of the transducers shall prevent electrical power from reaching both azimuth drive motor assemblies.

A hydraulic preload cylinder shall be used on each azimuth drive motor assembly to provide the normal force required for drive traction. This cylinder shall receive pressurized oil from the azimuth bearing oil manifolds.

14.5.3.7 Azimuth Drive Motors Activated when Azimuth Brakes are on (applied)

[13.3.7 (a)] **Hazard:** Azimuth drive motors activated when azimuth brakes are on (applied). This condition could cause damage to drive rollers, drive rings, and/or brake system.

[13.3.7 (b)] **Safety Interlocks:** If the azimuth brakes are on (applied), the azimuth drive motors shall be prevented from receiving electrical power.

[13.3.7 (c)] **Hardware Description:** Azimuth brakes are ‘fail safe’; i.e. spring on (engaged), hydraulic powered off (disengaged). Two pressure transducers which provide electrical contact (continuity) when a minimum, preset oil pressure is attained, shall be fitted to the brake manifold. The transducers shall be wired in series; a low pressure condition which is sensed by either one or both of the transducers shall prevent electrical power from reaching both azimuth drive motor assemblies.

14.5.3.8 Azimuth Drive Motors Activated when Telescope Mount has Rotated to End-of-Travel Limits

[13.3.8 (a)] **Hazard:** This condition could cause damage to drive system and/or mount assembly.

[13.3.8 (b)] **Safety Interlocks:** If an azimuth ‘hard’ limit, an incorrect topple bracket position, or the azimuth cable wrap height microswitch is sensed, then electrical power to the azimuth drive motors shall be removed.

[13.3.8 (c)] **Hardware Description:** A electrical micro-switch shall be located at each of the azimuth ‘hard’ rotation limits.

14.5.3.9 ELEVATION LOCKING PIN RETRACTED IF PRIMARY MIRROR CELL IS REMOVED

[13.3.9 (a)] **Hazard:** Elevation locking pin retracted if primary mirror cell is removed. This condition would allow the unbalanced telescope OSS assembly to rotate inadvertently about the elevation axis. Injury to personnel working near the telescope would be possible. In addition, damage to the telescope, mirrors, and/or instrumentation would be possible.

[13.3.9 (b] **Safety Interlocks:** If the primary mirror cell is removed, the elevation locking pin shall be prevented from receiving electrical power, and therefore being retracted.

[13.3.9 (c)] **Hardware Description:** An electrical micro-switch shall be located on the elevation locking pin assembly. This switch shall provide electrical contact (continuity) to the primary mirror cell attachment assemblies only if the locking pin assembly is engaged. KR: Not correct, primary mirror attachment is mechanical?

14.5.3.10 TOP END REMOVED WITHOUT ELEVATION LOCKING PIN ENGAGED

[13.3.10 (a)] **Hazard:** This condition would allow the unbalanced telescope OSS assembly to rotate inadvertently about the elevation axis. Injury to personnel working near the telescope would be possible. In addition, damage to the telescope, mirrors, and/or instrumentation would be possible.

[13.3.10 (b)] **Safety Interlocks:** If the elevation locking pin is disengaged, the top end latching mechanism shall be prevented from receiving electrical power, and therefore being released.

[13.3.10 (c)] **Hardware Description**: This is implemented by a transferable castell key. Locking pin inserted enables removal of key; key inserted into the top-end enables top-end latches to be operated.

14.5.3.11 ELEVATION DRIVE MOTORS ACTIVATED AND/OR ELEVATION BRAKES DISENGAGED IF PERSONNEL ARE WORKING ON ELEVATION DRIVE MOTOR ASSEMBLIES

[13.3.11 (a)] **Hazard:** Elevation drive motors activated and/or elevation brakes disengaged if personnel are working on elevation drive motor assemblies, or near areas of the telescope that are subjected to 'pinches' if the OSS assembly were to rotate about the elevation axis. Injury to personnel working near the telescope would be possible. In addition damage to the telescope is possible.

[13.3.11 (b)] **Safety Interlocks**: If the elevation maintenance key is removed, the elevation drive motors shall be prevented from receiving electrical power.

[13.3.11 (c)] **Hardware Description:** A removable-type maintenance key shall be installed in the telescope local control panel located TBD. This key shall be electrically connected in series with the elevation drive motor electronics. Removal of the key shall electrically disconnect the drive motors.

14.5.3.12 AZIMUTH DRIVE MOTORS ACTIVATED AND/OR AZIMUTH BRAKES DISENGAGED IF PERSONNEL ARE WORKING ON AZIMUTH DRIVE MOTOR ASSEMBLIES

[13.3.12 (a)] **Hazard:** Azimuth drive motors activated and/or azimuth brakes disengaged if personnel are working on azimuth drive motor assemblies, or near areas of the telescope that are subjected to 'pinches' if the telescope mount were to rotate about the azimuth axis. Injury to personnel working near the telescope would be possible. In addition damage to the telescope is possible.

[13.3.12 (b)] **Safety Interlocks**: If the azimuth maintenance key is removed, the azimuth drive motors shall be prevented from receiving electrical power.

[13.3.12 (c)] **Hardware Description:** A removable-type maintenance key shall be installed in the telescope local control panel located TBD. This key shall be electrically connected in series with the azimuth drive motor electronics. Removal of the key shall electrically disconnect the drive motors.

14.5.3.13 INADVERTENT TELESCOPE ROTATION ABOUT AZIMUTH OR ELEVATION AXIS INJURING PERSONNEL AND/OR DAMAGING TELESCOPE, INSTRUMENTATION, ENCLOSURE

[13.3.13 (a)] **Hazard:** Inadvertent telescope rotation about azimuth or elevation axis injuring personnel and/or damaging telescope, instrumentation, enclosure.

[13.3.13 (b)] **Safety Interlocks:** Pressing any ‘Emergency Stop’ button shall:

1. Remove power from elevation drive assemblies.
2. Remove power from azimuth drive assemblies.
3. Engage (apply) elevation brakes.
4. Engage (apply) azimuth brakes.
5. Remove power from enclosure drive motors.
6. Remove power from enclosure shutter drive assemblies.
7. Engage (apply) enclosure brakes.
8. Sound audible and visual alarm inside enclosure and control room.

[13.3.13 (c)] **Hardware:** ‘Emergency Stop’ push-buttonss shall be provided in the control room, on the local control panel, in the Cassegrain instrumentation area, near both of elevation drive motor assemblies, near the azimuth cable twister, TBD....

Push-buttons shall remain activated until manually reset.

14.5.4 Protection Systems

A number of subsystems require active protection in order to guarantee safety. One of the most important of these is the over velocity protection system for the telescope. The system must actively check that the velocity of the different axes is within safe limits and take appropriate action if it is determined either that the system is not safe or that the status of the system is undetermined. The current baseline for the over-velocity system is to use mercury switches to determine the elevation of the telescope and to use a voltage derived from the tachometers in order to measure the velocity. The protection systems are monitored by the interlock system.

14.5.4.1 AZIMUTH OVER-TRAVEL STOP

[6.3.5 (a)] The azimuth over-travel stop assembly is designed to decelerate the telescope assembly to a stop if the azimuth rotation angle exceeds ± 270 degrees; this provides protection primarily to the cabling that crosses the azimuth axis on the azimuth cable twister. As shown in drawing #87-GP-0702-0001, the over-travel stop assembly uses two hydraulic shock absorbers attached to the underside of the mount assembly which contact ‘topple’ brackets connected to the telescope pier.

[6.3.5 (b)] At the end of azimuth travel (± 270 degrees of rotation), a hydraulic shock absorber contacts its ‘topple’ bracket and the mount assembly is brought to a stop. As the telescope rotates within the azimuth rotation limits, the shock absorber assemblies ‘topple’, or rotate their respective topple brackets, and the mount is allowed to continue rotation. Microswitches are included in each topple bracket assembly and sense the position of each bracket. These microswitches are tied to the safety interlock system; if the MCS receives a signal that a topple-bracket is in the incorrect position, the telescope is brought to a stop. Loads from impact are transferred, as shown, to the telescope pier through the load transfer beam. The azimuth over-travel stop assembly system is designed to provide a maximum deceleration of 3.0 degrees/second^2 from a velocity of ± 2.0 degrees/second. All over-travel stop components are designed with a minimum factor of safety of 4.

During an azimuth over-travel stop event even though the interlock system will remove power from the brakes and drives, the azimuth cable wrap shall remain powered up to facilitate manually bringing the azimuth out of limit.

The azimuth topple bracket positions are also used to resolve the azimuth position ambiguity at MCS IOC power-up. The valid topple bracket states are summarized in the following table.

TABLE 14 - 16 **Valid Topple Bracket States**

AZIMUTH ANGLE	OUTER BRACKET	INNER BRACKET
-270 < AZ < -90	Not Toppled	Toppled
-90 < AZ < 0	Not Toppled	Not Toppled
0 < AZ < 90	Not Toppled	Not Toppled
90 < AZ < 270	Toppled	Not Toppled

14.5.4.2 ELEVATION OVER-TRAVEL STOP

[6.3.5 (c)] The elevation over-travel stop assembly is designed to decelerate the OSS assembly to a stop if the elevation rotation angle approaches either 90 degrees (zenith) or 15 degrees (horizon observation limit) elevation angles. This provides protection for; (1) the OSS colliding with the enclosure elevator or other items within the enclosure; (2) over-rotation that could cause damage to the primary mirror and/or supports; and (3) over-rotation of the elevation cabling. As shown in drawing #87-GP-0704-0001, the over-travel stop assembly uses two hydraulic shock absorbers attached to the top of the mount columns which contact ‘strike’ brackets. For top-end exchanges, or other maintenance operations where the OSS is required to rotate to horizon (0 degrees elevation angle), the horizon strike brackets can be rotated to maintenance positions. Microswitches are included in this strike bracket and sense the current position of the strike bracket (the strike bracket is either in the observing position or the maintenance position). These microswitches are tied to the safety interlock system and will limit rotation of the telescope about the azimuth axis is the strike bracket is in the maintenance position. [6.3.5 (d)] The elevation over-travel stop assembly system is designed to provide a maximum deceleration of 0.77 degrees/second^2 from a velocity of +/- 0.75 degrees/second. All over-travel stop components are designed with a minimum factor of safety of 4.

14.5.4.3 AZIMUTH OVER-VELOCITY STOP

In the event of the mount exceeding the current azimuth axis velocity limit, the standard mount shut-down procedure will be followed: the power to the drives and the brakes will be removed which will cause the drive motors to stop and the brakes to engage.

The azimuth axis velocity limit is a function of the current mount azimuth angle. The velocity is obtained from the drive unit tachometers. The velocity limits are: +/- 2.0 degrees/seconds except in the regions near the azimuth end-of-travel stops in which case the velocity limits are TBD. This should operate in the absence of the MCS IOC and should be implemented in a PLC system.

14.5.4.4 ELEVATION OVER-VELOCITY STOP

In the event of the mount exceeding the current elevation axis velocity limit, the standard mount shut-down procedure will be followed: the power to the drives and the brakes will be removed which will cause the drive motors to stop and the brakes to engage.

The elevation axis velocity limit is a function of the current mount elevation angle. The velocity is obtained from the drive unit tachometers. The velocity limits are +/- 0.75 degrees/seconds except in the regions near the elevation end-of-travel limits where they are TBD. This should operate in the absence of the MCS IOC and should be implemented in a PLC system.

14.5.5 Counterweights System

Although the telescope is designed to be well balanced in any of the upper end and instrument cluster configurations it is anticipated that (possibly small) trim weights are required to balance the telescope. These weights are remotely controlled and remotely monitored. It is possible to remove the power from the weight motors and the weights will remain in their last commanded position. This could be accomplished either by the use of electrically actuated brakes and/or mechanical design of drive system. It is anticipated that the position of the weights is maintained in a LUT that is used to automatically set them in the correct position based on the current telescope configuration. In order to trim the telescope it is necessary to provide some metric of the extent of its out of balance condition. The suggested metric is the asymmetries in the azimuth and elevation axis drive unit drive currents.

14.5.5.1 OSS COUNTER BALANCE

[6.3.6 (a)] The individual components of the OSS assembly (center section, main trusses, top-ends, etc...) are all designed so that the telescope is nominally balanced about the elevation rotation axis. However, due to manufacturing tolerance and other factors, it is expected that the actual center of mass of the OSS assembly will not lie directly on the elevation axis. Therefore, at the time of initial assembly of the telescope, a minimal amount of non-structural static weight plates may be required to 'rough' balance the OSS assembly. Any fine-balancing that is required, as well as balance changes due to top-end exchanges will be accomplished with a separate, automated-type counter-balance system which is described below.

[6.3.6 (b)] As shown in drawing #87-GP-0800-0001, the OSS counter-balance is designed to provide an automated balance of the OSS after exchanging top-ends or performing other operations on the telescope that create minor out-of-balance conditions. As shown, this system is comprised of two axial and two cross-axial balance devices mounted to the outside face of the center section assembly. ('Axial' is defined here as along the axis of the OSS and 'cross-axial' is perpendicular to the OSS axis and to the elevation rotation axis). The counterbalance units are located on the same side of the OSS as the elevation disks.

[6.3.6 (c)] The balance devices consist of commercial 'off-the-shelf' type linear translation systems which move, or translate, a set of steel weight plates. Each balance device is designed to operate with the telescope in any orientation and can be operated independently of the others. Each balance device includes preloaded linear bearings and tracks, lead screw, and motor units. The devices also incorporate absolute-type encoders and end-of-travel limit switches which gives mass position information to the control system. In addition, fail-safe type brakes effectively lock the weight plates in position when the telescope is in operation.

[6.3.6 (d)] In the event of a power failure the weights will not move under any gravity orientation. When stationary the units will not dissipate any heat.

[6.3.6 (e)] As shown in the drawing, each of the two axial balance devices are designed to translate 385kg over a total range of travel of 970mm, providing a total moment capacity of 3650 N-m/device. Each of the two cross-axial balance devices translate 135kg over a total range of travel of 770mm, providing a total moment capacity of approximately 1000 N-m/device. Update cross-axial 45 degrees? The mass position resolution of each unit is approximately 0.5mm.

14.5.6 Service Wrap-ups System

In order to provide power, data services, networking, cryogenics, and other services to the cassegrain focus it is necessary to provide a means of passing the services through the azimuth, elevation, and cassegrain bearings. The cassegrain cable twister is a separate work package and is under the control of the Gemini Instrument Group. In order to minimize the disturbance torques introduced into the motions of the different axes it is advisable to drive the service separately from the telescope axis. This does not remove the effects of reaction torques. It is intended to separately servo the service wrap-ups and to drive them as a follower to the axis they are connected to.

14.5.6.1 CABLE TRANSFERS, RUNS, AND DISCONNECTS

[6.3.7 (a)] A large number of electrical power and signal cables, hydraulic lines, pneumatic lines, water-glycol lines, helium and nitrogen lines, and other miscellaneous cables are required to pass from the stationary pier onto the rotating mount assembly (+/-270 degrees of rotation). A similarly large number of cables and lines are required to pass across the elevation rotation interface between the mount columns and the OSS (0-93 degrees of rotation). Due to pointing consideration, the torques necessary for these cable transfers must as small, and as linear, as possible (i.e. less than 100 N-m/axis). For this reason, motorized cable transfer devices are used on the azimuth and elevation axes, and are referred to as the ‘azimuth cable twister’ and the two ‘elevation cable wraps’, respectively.

14.5.6.2 AZIMUTH CABLE TWISTER

[6.3.7 (b)] As shown in drawing #87-GP-1000-0005, the azimuth cable twister is designed to carry the hydraulic oil supply line, sixty to eighty 20mm diameter cable bundles, and eight bellows-type gas line across the azimuth rotation interface. As shown, the hydraulic line and the cable bundles are suspended from the bottom of the mount base and pass through a motorized ‘twister’. This twister provides nearly all of the torque required to rotate (and lift) the cable bundles as the telescopes rotates about azimuth. As shown, the cables hang downward through the upper and lower access platforms and are kept separated by annular spacer rings. At the bottom of the lower access platform, the cables are splayed-out into a guided cable clamp which rises and falls as the cables are twisted, The bellows-type gas lines are contained in an annular tray which is also driven by the motorized twister; these type of gas lines cannot be ‘twisted’ and the annular tray arrangement is designed to only impart bending loads to the bellows.

[6.3.7 (d)] The azimuth cable wrap had been designed to allow a 0.5m optical beam or an A.O. laser beam to pass from under the cable wrap into the mount base.

14.5.6.3 ELEVATION CABLE WRAPS

[6.3.7 (c)] Once across the azimuth cable twister, the cables either terminate at their respective mount-level locations (e.g. azimuth drive motor assemblies) or they continue up the mount columns to the two elevation cable wraps. As shown in drawing #87-GP-1000-0002, these wraps transfer the various cable bundles from the top of the mount columns onto the rotating center section. A motorized unit similar to that of the azimuth cable twister provides for the torque required to rotate (and lift) the cable bundles. Total allowable motion is from 0 to 93 degrees of elevation angle.

14.5.7 Monitoring and Metrology System

In addition to the standard encoders used to determine the current position of the telescope various different types of sensors may be implemented.This includes networks of temperature sensors, strain gauges, and accelerometers distributed along the mount structure. In order to reduce wind shake of the telescope it may be necessary to use gyros. These gyros would be connected to the pier, the mount, the tube, and possibly the optics in order to calculate their current position. Due to the inevitable drift of these devices it is still necessary to have a natural guide star. The intent is to decrease the sampling requirement on the A&G fast guiding sensor (currently 200 Hz) by providing an alternate sensor. If, as we suspect, most of the dynamic deflection of the telescope in the presence of wind is due to bulk motion of the telescope and pier on the soil, then a gyro strapped to the pier should be able to sense a large portion of this error signal. There will be a trade study required to decide on a generic front-end for interface into this multitude of sensors.

14.5.8 Electrical Systems Interface System

In order to meet the 2% down time requirement it is necessary to monitor the status of critical electrical systems. The intent is to prevent problems by a system of periodic monitoring and preventative maintenance. In this system critical electrical systems could be monitored for voltage level, high frequency content, and (for AC systems) conformance with frequency stability specifications. In order to be effective such a system must establish a standard means of interfacing to any electrical system. This standard could be used by fabricators in order to make their systems compatible.

14.5.9 In-Situ M1 Cleaning Interface System

The primary mirror cleaning system is baselined to be a laser system that scans across the primary while it is pointing at the horizon. There is no requirement to actually control the cleaning system other than to provide for interlocking all the systems needed to provide safe operation and to monitor the status of the operation.

14.5.10 M1 Dry Gas Flush System Interface (TBD)

It may be necessary to flush the area above the mount by dry gas in order to prevent dewing or icing on the mirror's surface. This system is initiated by the telescope operator manually, via a push button. It is only necessary to monitor the status of the system and to provide the appropriate interlocks. There are severe safety issues with respect to this system.

14.5.11 Mirror Covers

The status of the mirror covers must be monitored. It is TBD if this should be a simple open/closed status based on microswitches present in the closed position or if this should an analog value that represents the mirror cover position.

14.5.12 Top-End Ring

The status of the four top-end ring latching mechanisms must be monitored. It this should be a simple open/closed status based on microswitches present in the latching mechanism. This will interface into the interlock system. An interlock will be triggered when any of the latching mechanisms indicates an unlatched condition.

The operation of the top-end ring latching mechanisms and the elevation locking pins will be only via local manual control.

14.5.13 Telescope Thermal Control System

A series of fans shall be used to cool the mount and the drive assemblies. These fans shall be operated manually but their on/off status shall be monitored by the MCS. A TBD number of fans shall be placed at the mount base. Each azimuth and elevation drive unit consisting of a pair of drive motors shall also have its own cooling fan.

The fan status interface will be simple digital I/O directly into the MCS IOC.

15 Details of the Cassegrain Rotator Control System

15.1 Function of the Cassegrain Rotator

The cassegrain rotator control system provides the basic ability to slew and track the cassegrain rotator. The cassegrain rotator is used in an alt/az telescope to correct for the rotation of the field of view that occurs as a source is tracked across the sky. The cassegrain rotator system also provides for redirection of the beam into the science instruments and into and out of the adaptive optics unit. It also interfaces to a number of secondary systems that are required to provide services.

15.2 Control System Overview

15.2.1 External Interfaces

The cassegrain rotator connects to the following external interfaces:

- Telescope Control System
- TCS Console (provided by OCS)
- OCS screens system
- instrument rotator hardware

15.2.2 External Buses

The cassegrain rotator connects to the following external buses:

- Synchronization Bus (TBD)
- Interlock system
- Time bus
- Event bus (TBD)
- Control LAN

15.2.3 Internal Subsystems

The cassegrain rotator control system consists of the following major internal sub-systems:

- Servos Subsystem
- Drives Subsystem
- Encoders Subsystem
- Interlock Subsystem
- Service Wrapup
- Electrical Systems Interface

Many of the physical systems associated with the cassegrain rotator environment such as guide probes and the beam redirection system are covered in the Acquisition and Guiding Control System.

15.2.4 Internal Interfaces

The cassegrain rotator has the following major internal interfaces:

-

15.2.5 Internal Data Stores

The cassegrain rotator has the following major internal data stores:

-

15.2.6 Computer Hardware

There is a VME crate located on the outside of the telescope center section that is dedicated to control of the cassegrain rotator, instruments, adaptive optics, and some of the A&G systems (see page 19 - 3). This VME crate may have multiple CPU's in order to handle all of the functionality required. The VME crate is not necessarily responsible for closing the servo loop between drive motors and the encoders. It is a design choice whether to implement the servo on a VME crate CPU or to use a dedicated board which communicates with a VME CPU. Thus the VME crate may have an intermediate computer between it and the hardware controlling the cassegrain rotator.

It is possible to connect a workstation directly to the VME crate via ethernet in order to control the system in a stand-alone mode. This stand-alone user interface is provided through EPICS.

15.2.7 Software Philosophy

TBD

15.3 EXTERNAL INTERFACES

15.3.1 Interface to Telescope Control System

15.3.1.1 REQUIREMENTS

The cassegrain rotator control system accepts the parallactic parameters {P, P',P''} for the target position, velocity, and acceleration of the rotator. These parameters may be sent once or they may be continuously delivered at up to 20 Hz.

In addition to the targets the cassegrain rotator accepts the commands:

- STOP; halt and put the brakes on
- MOVE; go to the current target and stop
- FOLLOW; go to the current target and follow the target continuously
- STOP FOLLOWING; stop where you are
- GUIDE CONFIG; configure the wave front sensor(s) to guide on
- GUIDE; start guiding using configured wave front sensors
- GUIDE OFF; stop guiding and go back to following
- PARK; move to a predetermined position and put rotator in park state
- ENCODER CONFIG; configure the encoders
- ZERO SET; zero different parts of the system
- DRIFT; got to the current target and drift with a specified constant velocity
- UNWRAP; take out cable wrap

The cassegrain rotator returns the following values:

- Servo errors
- Status (e.g. "in position")

15.3.1.2 IMPLEMENTATION

The above data values are contained in an EPICS database that communicates with the TCS via channel access.

15.3.2 Interface to the Cassegrain Control System Console

15.3.2.1 REQUIREMENT

The consoles provided by the OCS must have fine grained control capability of the secondary.

The following commands are available:

- TBD

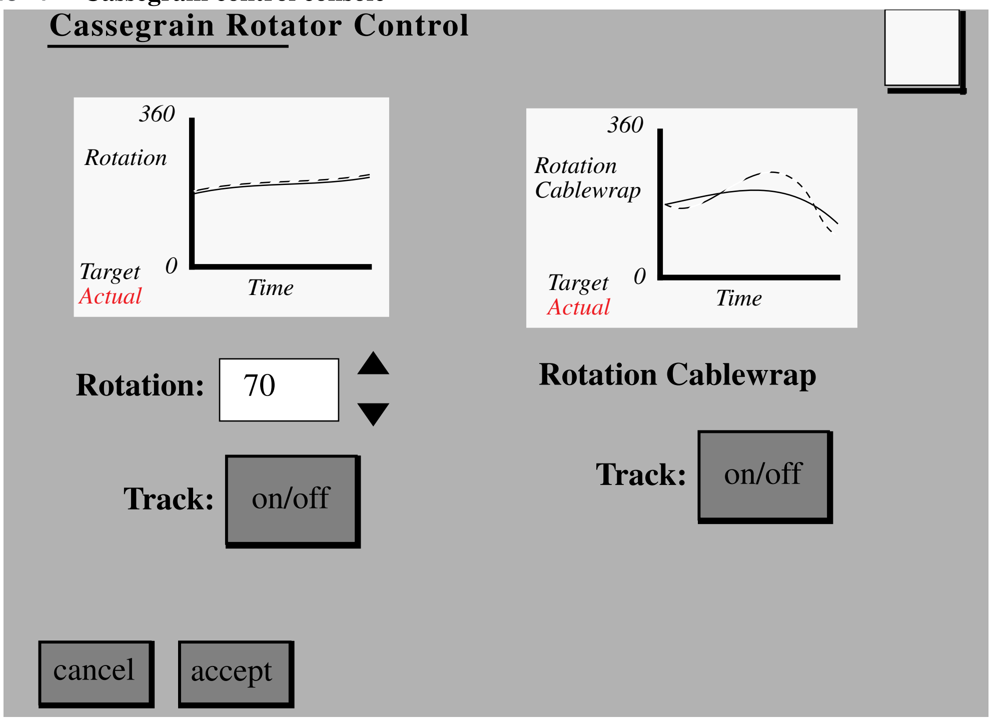
15.3.2.2 IMPLEMENTATION

The TCS uses EPICS Channel Access to set the appropriate parameters in the Cassegrain rotator's database and then issues an event that causes the Cassegrain rotator to match the system with those parameters.

15.3.2.3 SAMPLE CONSOLES

The consoles shown below are samples of what the OCS provides to the operator for commissioning and interactive observing.

FIGURE 15 - 1 **Cassegrain control console**



15.4 INTERFACE TO OCS SCREEN SYSTEMS

The cassegrain rotator system provides the following *items* to the different *screens*.

15.4.1 Encoder Screen

- Cassegrain rotator (units are degrees:minutes:seconds).

--Absolute

--Incremental tape

--demand

15.4.2 Limit Screen

- Cassegrain rotator limits.

- DAMPER LIMIT - none
  --HARD LIMIT
  --SOFT LIMIT
  --COMPUTER LIMIT

15.4.3 Alarms Screen

TBD

15.4.4 Interlocks Screen

TBD

15.4.5 Interface to Cassegrain Rotator Subsystems

15.4.5.1 SERVO SYSTEM

The CRCS monitors and controls the servo system. The following functions are required:

- Position control- Control
- Velocity control - Control
- Acceleration control - Control

15.4.5.2 DRIVE SYSTEM

The CRCS monitors and controls the drive system. The following functions are required:

- Direct operation - Control

The intent is to use the capabilities provided by the manufacturer of the drives and not to create or add functionality to these systems.

15.4.5.3 SERVICE WRAPUP

No functionality is required as the service wrapup is mechanically linked to the cassegrain rotator.

15.4.5.4 ENCODER SYSTEM

The CRCS monitors the encoder system. The following functions are required:

- 32-bit virtual encoder interface - Monitor

15.4.5.5 INTERLOCK SYSTEM

The CRCS monitors the interlock system. The following functions are required:

- Interlock status- Monitor
- Manual interlock set (Panic buttons, etc.) - Control
- Interlock PLC status - Monitor

15.4.5.6 ELECTRICAL SYSTEMS INTERFACE SYSTEM

The following functions are required:

- Voltage Monitor
- Frequency content Monitor
- Frequency stability Monitor

15.5 EXTERNAL BUS CONNECTIONS

15.5.1 Synchronization Bus

Given to low speed needs for wavefront information for updating the rotator speed and position it is not clear that a connection to this bus is justified.

15.5.2 Interlock System

The CRCS must be capable of generating interlock requests as well as responding to them.

15.5.3 Time Bus

The time within the CRCS VME crate is set to +- 5 microseconds from the time bus.

15.5.4 Event Bus

TBD

15.6 CASSEGRAIN ROTATOR INTERNAL SUBSYSTEMS

15.6.1 Servos

There is one servo system needed to position and track the cassegrain rotator angle of field rotation.

The cassegrain rotator servo system accepts a stream of position, velocity, and acceleration commands generated externally at a 20-50 Hz frequency. The servo system uses the past several commands to extrapolate the current command at a higher frequency, 200 - 500 Hz, required by the electronics. The servo electronics, baselined to be a digital signal processor, are executing the servo algorithms.

In addition to accepting these commands the servo subsystem responds to the generic instrument commands.

15.6.2 Drives

In operation the cassegrain rotator drives are controlled by the servo subsystem. It may also be possible to directly control the drives in a stand-alone mode by directly connecting a workstation or terminal.

The intent is to use the capabilities provided by the manufacturer of the drives and not to create or add functionality to these systems. If there is no built in capability for direct control of the drives then none shall be provided.

15.6.3 Encoders

The encoders for the cassegrain rotator are a combination of mechanical switches, magnetic position sensors, incremental encoders, and absolute encoders. The intent of the encoder subsystem is to hide the details of the physical encoding scheme and provide a device independent virtual encoder to higher level systems. This virtual encoder is baselined to be a 32-bit encoder. It must be possible to directly connect a computer to the encoder subsystem in order to run diagnostics and tests.

15.6.4 Interlock System

The interlock system monitors the status of a large number of devices and, based on their current status, either enable or disable specific devices in the observatory. This system is intended to operate in a double safe mode - by this we mean that it is not sufficient to only detect the presence of a condition that causes an interlock, it also necessary to sense the absence of this condition.

The current baseline for this system is to use a programmable logic controller (PLC) to monitor interlock signals and to initiate actions based on these interlocks. It is necessary to define a standard interface, if possible, to which systems desiring interlocks can connect.

It is intended that, where possible/practical, all interlock systems be self actuating and that only their status is monitored by the PLC system. The philosophy is that the primary system that is desired to be interlocked performs this function independent of the PLC system. If there are secondary interlocks to be triggered from the primary system then these, in general, operate through the PLC system.

15.6.5 Service Wrapups

In order to provide power, data services, networking, cryogenics, and other services to the cassegrain focus it is necessary to provide a means of passing the services through the cassegrain bearing. It is intended to have the service wrap physically linked to the instrument rotator. For this to function correctly, the mechanical limits to the service wrap rotation must be somewhat greater than those for the rotator itself.

Note that some instruments, such as the High Resolution Spectrograph (HROS), may require an optical fibre feed from the Cassegrain focus to a fixed laboratory.

It is a design decision whether or not to drive the service wraps separately from the instrument rotator itself.

15.6.6 Electrical Systems Interface

In order to meet the 2% down time requirement it is necessary to monitor the status of critical electrical systems. The intent is to prevent problems by a system of periodic monitoring and preventative maintenance. In this system critical electrical systems could be monitored for voltage level, high frequency content, and (for AC systems) conformance with frequency stability specifications.

In order to be effective such a system must establish a standard means of interfacing to any electrical system. This standard could be used by fabricators in order to make their systems compatible.

16 Details of the M1 Control System

16.1 Function of the M1 Control System

16.1.1 M1 Support System

The Gemini Science Requirements has set an image requirement of 0.1 arcseconds. The engineering interpretation of this requirement is that the entire telescope/enclosure system may only increase the 50% encircled energy diameter by 0.1 arcseconds at 2.2 microns while operating in an external wind speed of 11 meters per second.

The primary mirror subassembly is composed of a large, thin, meniscus mirror that is contained in a mirror cell attached to the telescope. In order to meet the imaging requirements the surface of this mirror must be maintained to the correct figure to within a tolerance of several 10's of nanometers RMS. It is the objective of this work package to provide the control system for the primary mirror subassembly.

This tolerance must be maintained in the presence of

- a changing gravity vector - the mirror is tilted from zenith pointing to horizon pointing
- mechanical flexures of the mirror cell - the cell bends due to changing gravity vector
- thermal flexure of the mirror cell - the cell warps due to thermal gradients, both spatial and temporal
- wind buffeting of the mirror - the mirror is exposed to both spatial and temporal variations in wind pressure.

- support system errors - the axial and lateral supports locally bend the mirror an observable amount.

The systems used to maintain this tolerance are a series of axial and lateral support systems and active optics systems and an axial air pressure support system There are 120 axial supports, 60 lateral supports, and 24 Xposition definers. The 120 axial supports are arranged to produce minimal surface error at zenith position. They are distributed in five concentric rings with number of supports in each ring (from the center of the mirror outward) being 12, 18, 24, 30, and 36. The supports are equally spaced on each ring.

The mirror weight at zenith is carried 80% by the air pressure support system with the other 20% being carried by the hydraulic wiffle tree which incorporates a pneumatic fine tuning system. The active system supplies a zero net force to the mirror. As the mirror tips towards the horizon the air system decreases its pressure so as to maintain the hydraulic wiffle tree at a constant pressure. The air bag pressure is zero at 12 degrees altitude. The lateral support system is designed to float the mirror in the cell.

Each axial support contains an ‘active’ pneumatic control element and a ‘passive’ hydraulic control element. Each lateral support contains only a ‘passive’ hydraulic control element. The axial supports carry 20% of the mirror weight at zenith and apply forces that are perpendicular to the curved mirror surface.

16.1.2 M1 Thermal Control System

In an astronomical telescope heat transfer between the primary mirror and the surrounding air can result in non-uniform air density and therefore non-uniform refractive index of the air above the mirror, producing the effect called mirror seeing. Mirror seeing blurs the images produced by the telescope. In this overview we describe a thermal control approach that we believe would reduce mirror seeing effects on an 8-meter diameter, 20-cm thick meniscus primary mirror to levels consistent with the Gemini Telescopes error budget.

The aim is drop the mirror temperature by ~3 degrees C during 6 hours of daytime cooling. The target temperature is the minimum temperature from the night before. The radiation plate will be used to cool or heat the mirror during observation. The temperature of the primary mirror is monitored by a network of thermal sensors.

16.2 PRIMARY SUPPORT CONTROL OVERVIEW

16.2.1 External Interfaces

The Primary Control System (PCS) has the following external interfaces:

- Telescope Control System - see *ICD 7b - TCS Subsystem Interfaces*.
- the PCS Control Console(s) - provided by the OCS
- the OCS screens system
- the primary mirror support hardware

The external interfaces to the PCS are shown in Figure 16 - 1 on page 16 - 5

16.2.2 External Bus Connections

The PCS is connected to the following external buses:

- Interlock system - see *ICD 12 - Interlock System*
- Time Bus - see *ICD 9 - EPICS Time Bus Driver*
- Control LAN

16.2.3 Internal Subsystems

The PCS is composed of the following major internal subsystems:

- Calibration and Test
- Active Optics
- Passive Support
- Air Pressure Support
- Safety System
- Thermal Control

16.2.4 Internal Interfaces

The PCS has the following major internal interfaces:

-

16.2.5 Internal Data Stores

The PCS has the following major internal data stores:

- PCS Real Time EPICS database
- active actuator influence function matrices
- actuator force lookup table data (as a function of altitude angle and temperature)

16.2.6 Computer Hardware

The primary mirror control system is located on the outside of the telescope center section in a temperature controlled enclosure designed to minimize the heat leakage into the telescope environment.

The hardware developed for the PCS is expected to follow the standards of the Gemini project. Please reference *ICD 13 - Standard Control System*.

16.2.7 Software Philosophy

The PCS system is based on the Standard Instrument Controller work package and reuses all of the EPICS software created by that work package.

It is anticipated that the PCS may need to create a custom EPICS driver for the interface between the VME crate and the cards located at or near the individual actuators. In this case the PCS work package team is expected to reuse the expertise of the SIC work package by contracting out the driver development to the SIC work package team.

The PCS is expected to provide a complete stand-alone engineering interface to the EPICS control system. All inputs and outputs must be available as EPICS database entries on the PCS. In this way higher level systems are guaranteed external access via channel access to all the functionality of the PCS.

In operation, the PCS is controlled from the TCS for all actions required. The OCS provides operator consoles for the PCS that communicate via the TCS.

The software developed for the PCS is expected to follow the standards of the Gemini project.

FIGURE 16 - 1 **M1 Control System Electronic Interfaces**

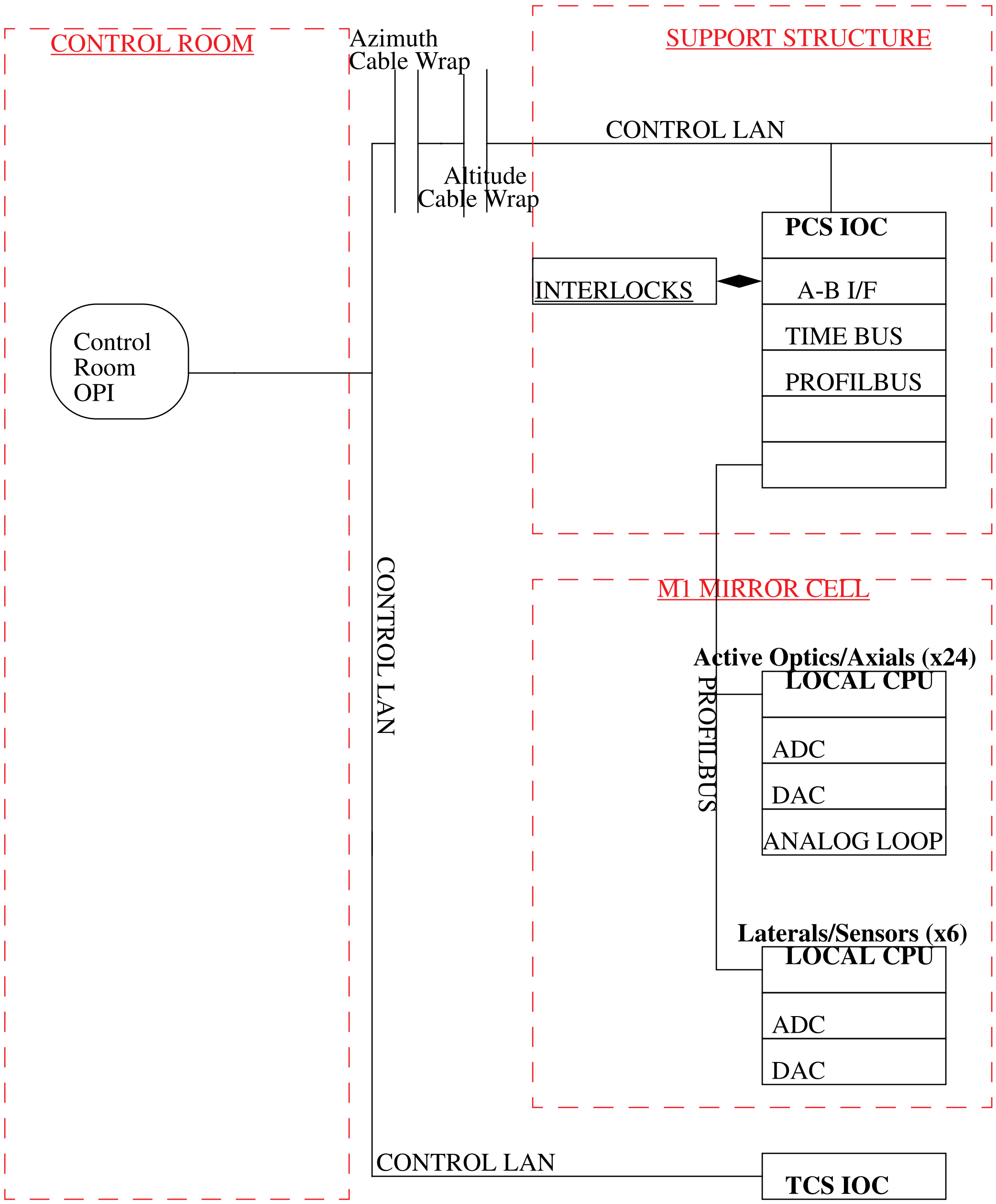


FIGURE 16 - 2 **M1 Control System Subsystems**

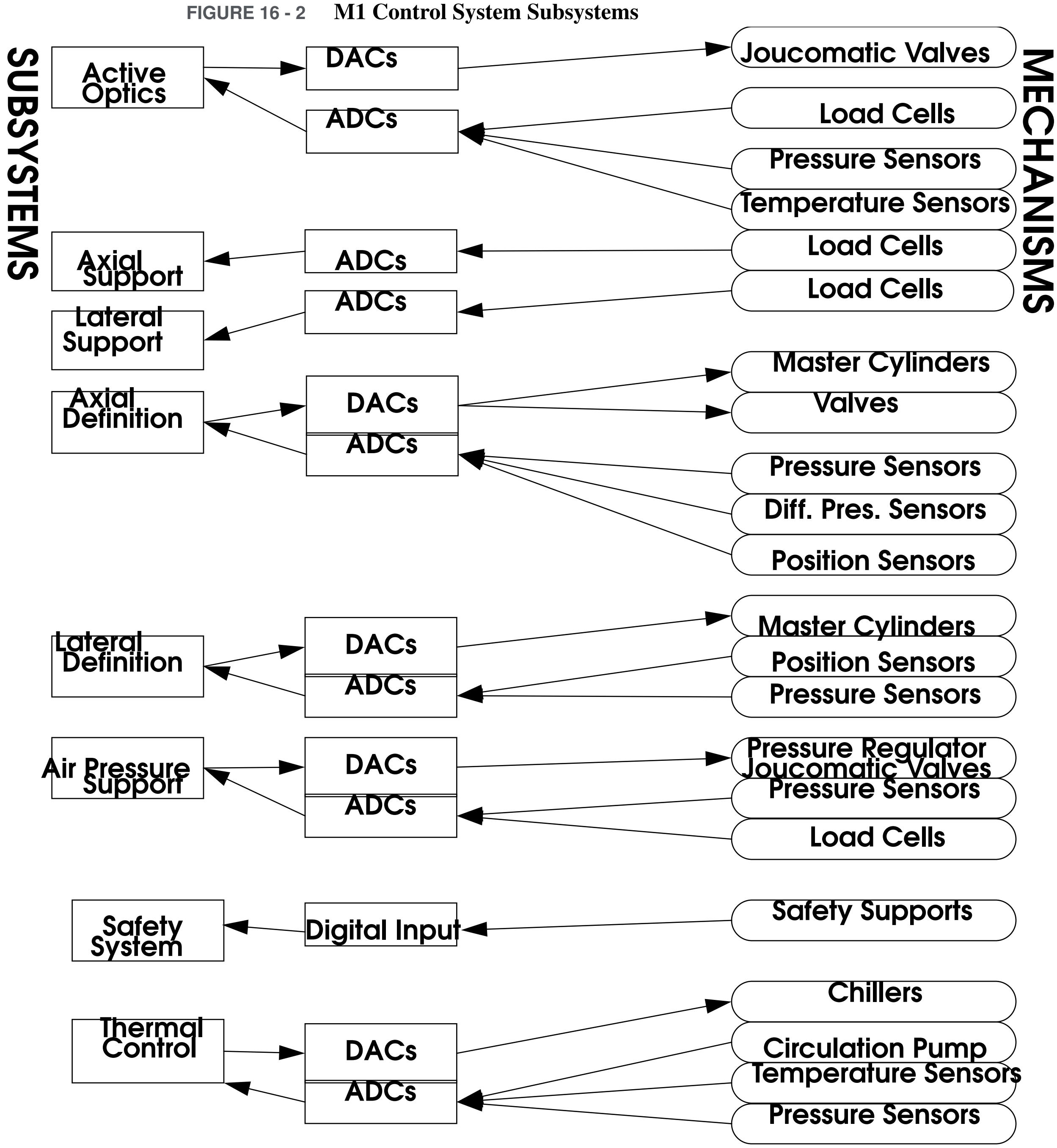
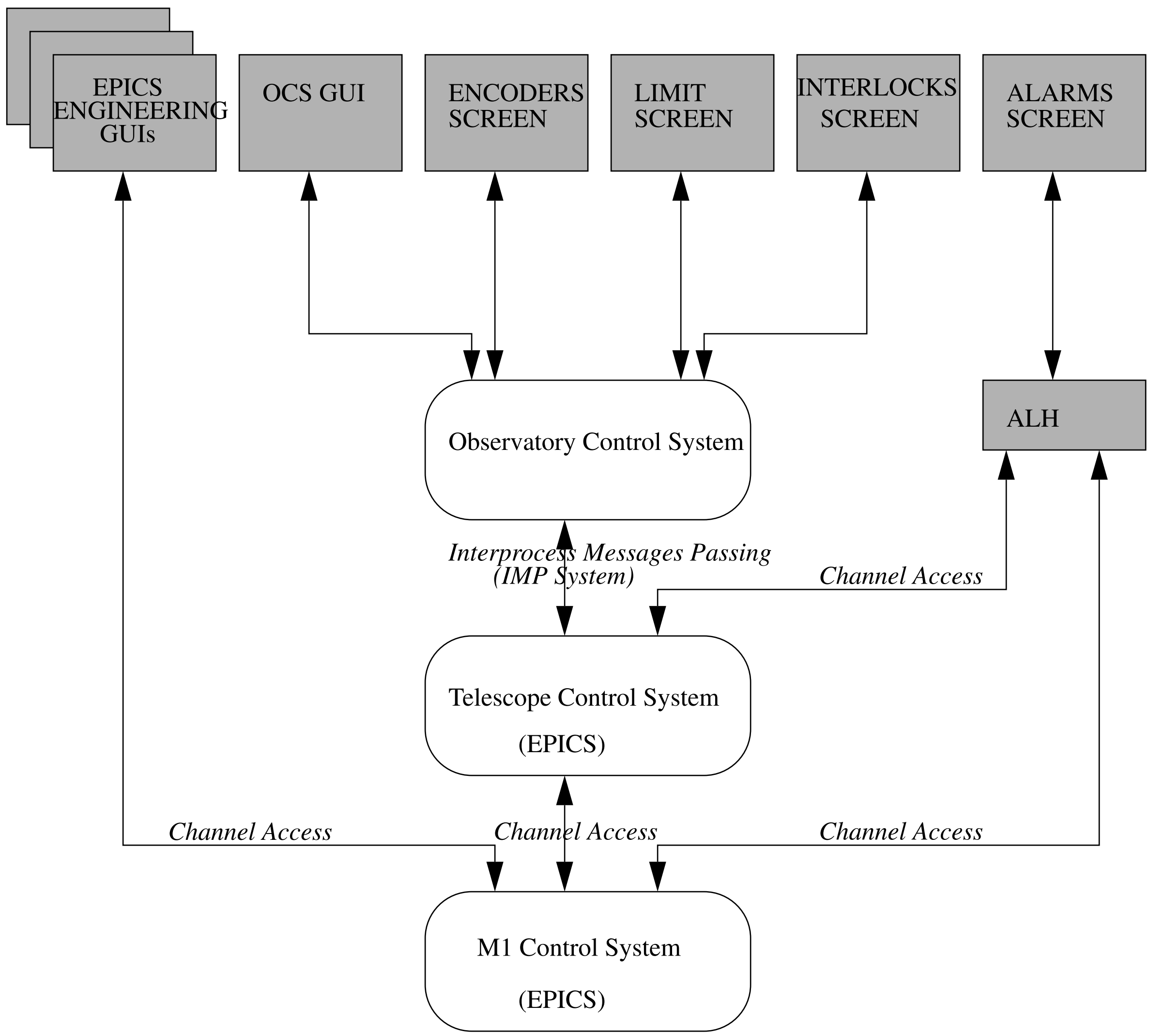


FIGURE 16 - 3 M1 Control System Software



16.3 EXTERNAL INTERFACES

16.3.1 Interface to Telescope Control System

16.3.1.1 REQUIREMENTS

The primary control system accepts the following parameters from the TCS:

- Altitude and Temperature when running in open loop. The PCS will use this info plus (say) its own internal lookup table to calculate the baseline set of forces to apply to the actuators.
- Zernike (or other orthogonal parameters describing the target surface of the primary) parameters $\{Z_1,Z_2,...,Z_N\}$ for the target figure of the primary mirror when running in closed loop. In open loop mode any Zernike parameters received will be interpreted as an offset correction to the target surface.
- Translations {X,Y,Z} for the target position of the primary mirror
- Tilts and rotation about the optical axis {Rx, Ry, Rz}, although the tilts could be encoded as the first two Zernike terms).

These parameters may be sent once or they may be continuously delivered at up to 20 Hz.

Instead of the above target the primary control system might be sent a vector of target actuator forces $\{F_1,F_2,...,F_{120}\}$.

In addition to the targets the primary accepts the commands:

- STOP; stop and hold the current set of actuator forces, translations
- MOVE SURFACE; go to the current target and stop
- MOVE ACTUATORS; go to the current target forces and stop
- MOVE HYDRAULICS; go to the current tilt, rotation and stop
- FOLLOW SURFACE; go to the current target sruface and follow the target continuously
- FOLLOW HYDRAULICS; go to the current target tilt, rotation and follow targets continuously
- STOP FOLLOWING (SURFACE|HYDRAULICS); stop where you are
- PARK; move to a predetermined target and put primary in park (safe) state

- Air Bag control commands - TBD
- TBD

The primary returns the following values:

- Encoder readings
- Target Actuator Forces
- Load cell readings
- Alarms
- Status (e.g. "in position")
- TBD

The primary thermal control system accepts temperature and temperature slope information {T,T'} for the target surface temperature of the primary. These targets are calculated by an algorithm in the TCS which attempts to minimize the deviations of the primary mirror surface from ambient during the night. These parameters may be sent once or they may be continuously delivered at up to 0.1 Hz.

In addition to the targets the primary thermal control system accepts the following commands:

- STOP; stop heating/cooling primary and let primary blank drift
- MOVE; go to the current target and hold it
- FOLLOW; go to the current target and follow the target continuously
- STOP FOLLOWING; stop where you are
- PARK; move to a predetermined temperature and temperature gradient
- SENSOR CONFIG; configure the temperature sensors
- ZERO SET; zero different parts of the system

The primary thermal system returns the following values:

- Sensor readings
- Drive demand
- Servo errors
- Alarms
- Status (e.g. "in position")

16.3.1.2 IMPLEMENTATION

The above data values are contained in an EPICS database that communicates with the TCS via channel access.

16.3.2 Interface to the OCS Console(s)

16.3.2.1 REQUIREMENT

The consoles provided by the OCS must have fine grained control capability of the primary support hardware.

The following commands are available:

- TBD

16.3.2.2 IMPLEMENTATION

The TCS uses EPICS Channel Access to set the appropriate parameters in the PCS's database and then issues an event that causes the PCS to match the system with those parameters.

16.3.2.3 SAMPLE CONSOLES

The consoles shown below are samples of what the OCS provides to the operator for commissioning and interactive observing.

FIGURE 16 - 4 **Primary mirror active control console**

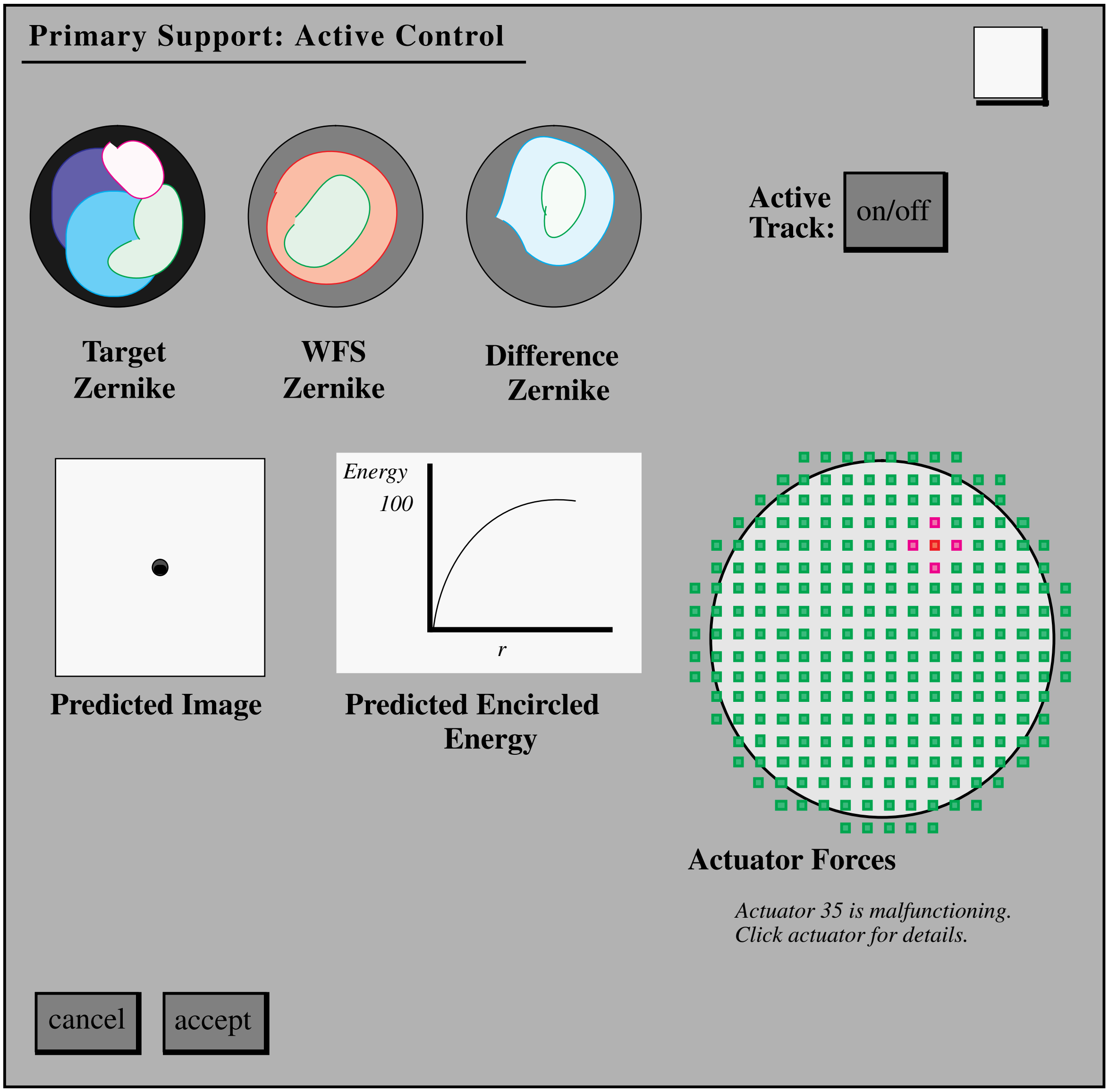


FIGURE 16 - 5 **Primary mirror passive support console**

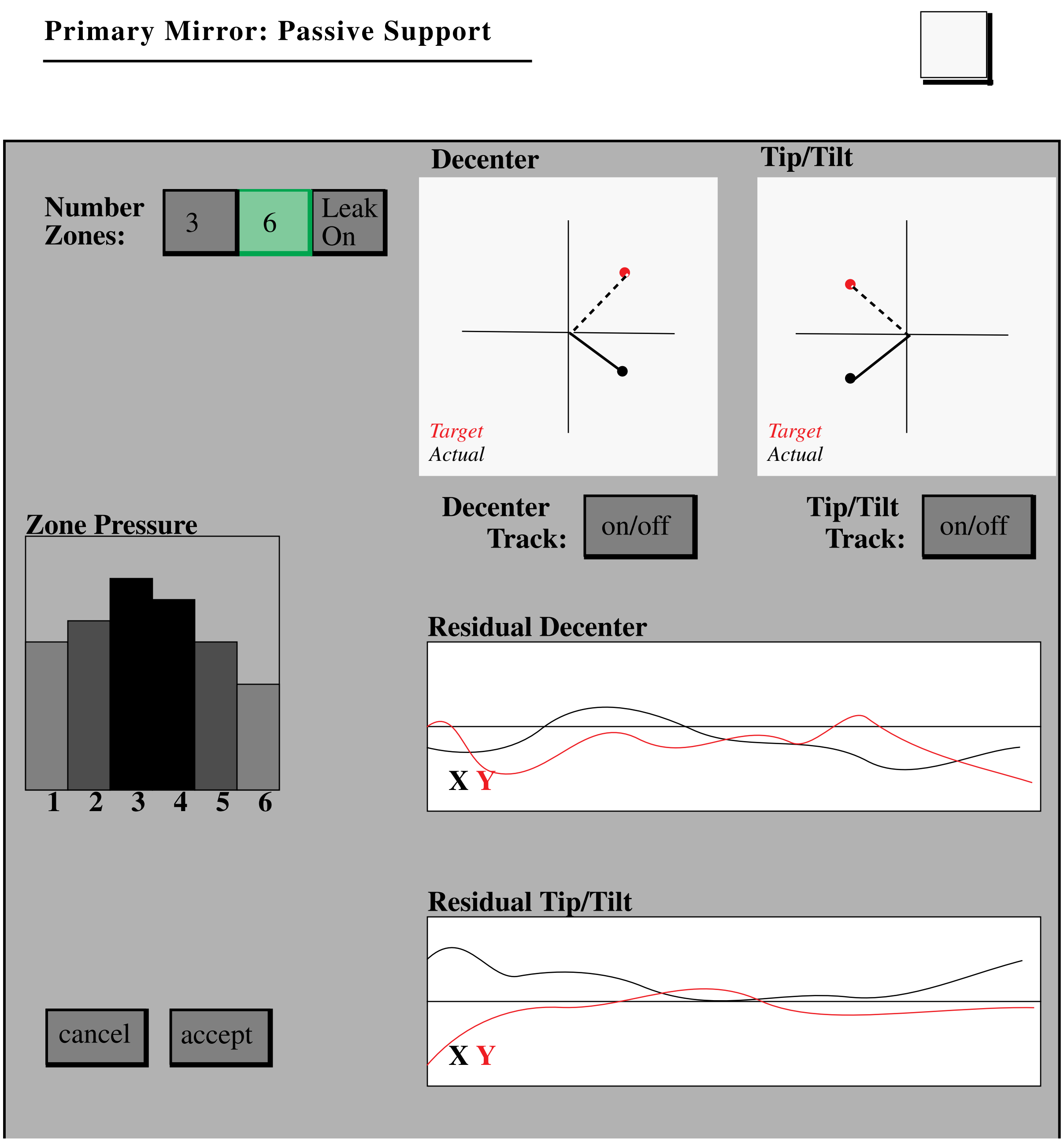
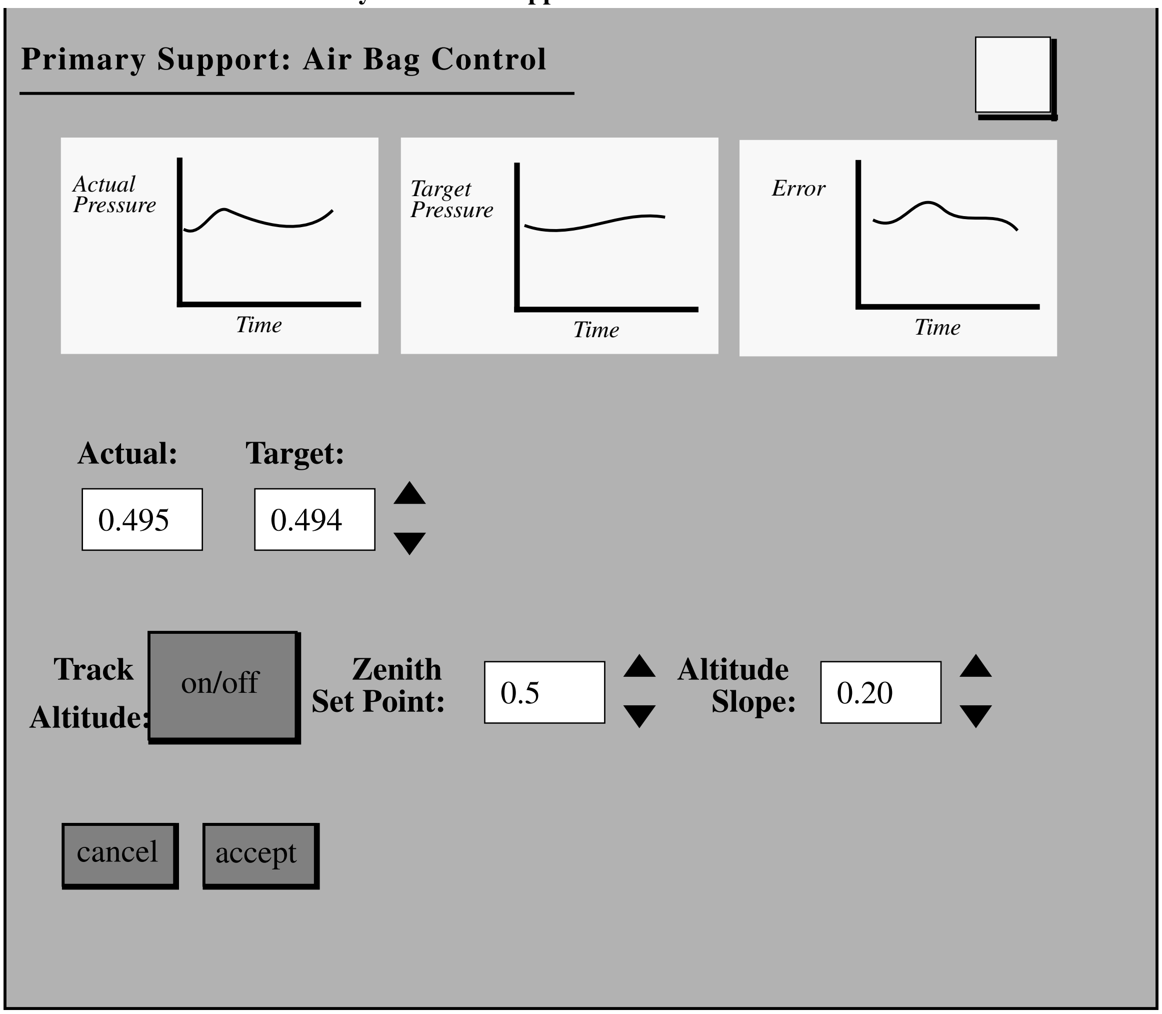


FIGURE 16 - 6 **Primary mirror air support console**



16.3.3 Interface to OCS Screen Systems

The following *screen items* are made available to the *screen* system.

16.3.3.1 ENCODER SCREEN

-

16.3.3.2 LIMIT SCREEN

- actuator bottomed out
- TBD

16.3.3.3 ALARMS SCREEN

- Primary mirror support.

  --Air pressure.

  --Load cell alarm

  --Mirror height
- TBD

16.3.3.4 INTERLOCKS SCREEN

--Safety System

--TBD

16.3.4 Interface to Primary Mirror Support Subsystems

The PCS interfaces to the primary support hardware with the following demarcation with the Mirror Support Work Package. All inputs and outputs to the Mirror Support Work Package are voltages. The PCS supplies a card that is compatible with the bus chosen to interface back to the PCS VME system with sufficient DACs and ADCs to control the Mirror Support functions. The form factor and connector of this card are chosen in collaboration. The cabling to and from this card are specified and purchased by the PCS but are installed by the Mirror Support WP.

It is a goal to have self contained actuators with individual controls. However, due to the large number of actuator sites it may prove more economical to have more than one actuator controlled by a single control site. The PC board chosen by the PCS is a commercially available board with no onboard programming capability. The goal here is to minimize the number of computers in the system.

If the number of control sites required by the Mirror Support WP exceeds the budget of the PCS for these items then the Mirror Support WP has to purchase the additional hardware from its budget.

TABLE 16 - 1 General Functions (for all M1 mechanisms)

Function	Description	Range	Frequency	Type
Power	turn power on/off to entire unit	ON | OFF	< 0.1 Hz	digital switch
Health	monitor overall status of device	OK | BAD	1 Hz	digital i/p
Servo Parms	up/down load servo parameters	TBD	< 0.1 Hz	digital i/o
HeartBeat	incrementing counter	32 bits	20-50Hz	digital i/p
WatchDog	shutdown M1 systems if host dies		20-50Hz	digital i/o

TABLE 16 - 2 M1 Active Optics

Function	Description	Range	Frequency	Type
Mode	Switch between open and closed loop mode.	OPEN/ CLOSED	TBD	state
Local Mode	Switch between local analog closure or local digital closure of lower load cell - joucomatic loop.	ANALOG/ DIGITAL	TBD	state
Load cell (120)	Monitor lower load cell	TBD	20Hz	analog i/p
Joucomatic Valve (120)	Set demand forces by controlling pressure regulator.	TBD	20Hz	analog o/p
Pressure sensors (120)	Monitor pressure sensor colocated with actuator.	TBD	20Hz	analog i/p
Temperature sensors (120)	Monitor temperature sensor colocated with actuator.	TBD	20Hz	analog i/p
WFS Data (A&G or AO)	Transform Zernike parameters into actuator demand forces. If in open-loop mode this are treated as offsets to the forces derived from the Look-up Table.	TBD	TBD (depends on WFS exposure time)	analog i/p
Look-up Table	Apply look-up table to altitude angle and temperature to produce demand forces.	TBD	20Hz	analog i/p

TABLE 16 - 3 **M1 Axial Support**

Function	Description	Range	Frequency	Type
Load cell (120)	Monitor upper load cell.	TBD	20Hz	analog i/p

TABLE 16 - 4 **M1 Lateral Support**

Function	Description	Range	Frequency	Type
Load cell (60)	Monitor load cell.	TBD	20Hz	analog i/p

TABLE 16 - 5 **M1 Axial Definition (Tz,Rx,Ry)**

Function	Description	Range	Frequency	Type
Position sensor (4)	Monitor position sensor.	TBD	1Hz	analog i/p
Master cylinder (7)	Control hydraulic master cylinders. Control loop closed by position sensors and differential pressure sensors (or WFS data).	TBD	0.1Hz	analog o/p
Pressure sensor (7)	Monitor pressure sensors colocated with master cylinders.	TBD	1Hz	analog i/p
Differential pressure sesnors (6)	Monitor differential pressure between zones.	TBD	1Hz	analog i/p
Valves (6)	Control valves between zones.	CLOSED| LEAK| OPEN	0.1Hz	state

TABLE 16 - 6 **M1 Lateral Definition (Ty,Rz)**

Function	Description	Range	Frequency	Type
Position sensor (1)	Monitor position sensor.	TBD	1Hz	analog i/p
Master cylinder (3)	Control hydraulic master cylinders. Control loop closed by position sensor.	TBD	0.1Hz	analog o/p
Pressure sensor (3)	Monitor pressure sensors colocated with master cylinders.	TBD	1Hz	analog i/p

TABLE 16 - 7 **M1 Lateral Definition (Tx)**

Function	Description	Range	Frequency	Type
Position sensor (1)	Monitor position sensor.	TBD	1Hz	analog i/p
Master cylinder (2)	Control hydraulic master cylinders. Control loop closed by position sensor.	TBD	0.1Hz	analog o/p
Pressure sensor (2)	Monitor pressure sensors colocated with master cylinders.	TBD	1Hz	analog i/p

TABLE 16 - 8 **M1 Air Pressure Support**

Function	Description	Range	Frequency	Type
Air pressure regulator(1)	Control air pressure regulator. Control loop closed by axial support load cell readout.	TBD	0.1Hz	analog o/p
Pressure sensor (1)	Monitor pressure sensor.	TBD	1Hz	analog i/p

Note: What happened to the elevation sensors? There must be some way for the Air Pressure Support subsystem to adjust the pressure when the TCS isn't running!

TABLE 16 - 9 **Safety Support [TBD]**

Function	Description	Range	Frequency	Type
Axial Support Status(18)	Monitor safety support status.	ON|OFF	20Hz	state
Laterral Support Status(18)	Monitor safety support status.	ON|OFF	20Hz	state

TABLE 16 - 10 **Thermal Control**

Function	Description	Range	Frequency	Type
Chiller Temperature Set Point	Control chiller set point for constant flow radiation plate system.	TBD	TBD	analog o/p
Chiller Temperature Slope	Control chiller set point linear rate of change for constant flow radiation plate system.	TBD	TBD	analog o/p
Circulation Pump Status (TBD)	Monitor circulation pump status.	ON|OFF	TBD	state
Temperature Sensor (30)	Monitor temperature.	TBD	TBD	analog i/p
Dewpoint Sensor (30)	Monitor dew point	TBD	TBD	analog i/p

16.3.4.1 TYPICAL USE OF PRIMARY SUPPORT

TBD

16.3.4.2 LIMITS AND SLEWING STRATEGY

It is possible to configure the active forces for the demand elevation (i.e. set the force distribution at the start of a slew). It is not clear that it is possible to do the same for the hydraulic system - the PCS must allow the following options for “slewing” the hydraulic system:

- no slew - demand hydraulic positions are set; PCS has no concept of slewing. The TCS can send a continous stream of demand hydraulic positions
- slew - demand and current elevation are monitored by PCS; for large telescope movements PCS can calculate what to do to reach demand hydraulic positions

16.4 External Bus Connections

16.4.1 Interlock System

The PCS must be capable of generating interlock requests as well as responding to them.

16.4.1.1 Requirements

The following interlocks are required:

- safety system
- TBD

16.4.1.2 Implementation

TBD

16.4.2 Time Bus

The time within the PCS VME crate must be set to an external time reference.

16.4.2.1 Requirements

16.4.2.2 Implementation

The PCS connects to the Gemini Time bus and uses the Gemini supplied EPICS implementation.

16.4.3 Event Bus

16.4.3.1 Requirements

Not clear that PCS needs to be connected to the event bus.

16.4.4 Control LAN

16.4.4.1 REQUIREMENTS

In order to send/receive values via EPICS channel access the PCS needs a TCP/IP connection.

16.4.4.2 IMPLEMENTATION

The PCS has a connection to the control LAN on the telescope.

16.5 PCS INTERNAL SUBSYSTEMS

16.5.1 Calibration and Test

The primary mirror support system shall be capable of being tested and calibrated when configured as a stand-alone system separate from the other observatory systems.

The principal operator interface shall be via a GUI based on a Sun Workstation. The interface into the real-time control system shall be via EPICS channel access.

There are tools available within the EPICS environment that may have the required functionality to implement this subsystem, most notably the CaWave interface between EPICS channel access and PV-Wave.

16.5.2 Active Optics

The primary mirror shall be supported axially by an active support system which shall be capable of maintaining the figure of the primary mirror to the required accuracy by correcting for high-order deformations in the mirror figure caused by changing gravity vector as well as dynamic effects due to thermal expansion stresses and wind loading.

The active system is implemented as actuators consisting of a pneumatic air cylinder with a load cell mounted on top. An actuator is astatic and adds or subtracts a force from the passive hydraulic support under which it is mounted. The air pressure in the active actuator is controlled by a pressure regulator that maintains constant pressure in response to a demanded pressure command. The force applied to the back of the mirror is commanded and may be maintained by closing the loop around the load cell. The computer algorithm must ensure that a zero net force and torque is transmitted to the mirror.

Although having the TCS send Zernike coefficients is cleaner and results in a more hierarchical design the problem is that, at high spatial resolution, the surface of the primary will not be well represented by Zernikes due to the large number of actuators (120). Thus it is necessary to have a look up table which generates a vector of 120 forces rather than a look up table which generates 8-12 Zernike coefficients - the table in either case is strongly dependent on altitude and weakly dependent (we hope) on temperature.

In operation there will be an open loop model which continually generates a vector of forces to apply to the mirror and a closed loop model which uses WFS information to genrate corrections to the forces.

This is implemented as follows:

- TCS will supply altitude and temperature to the PCS at a 20Hz rate as part of the open loop. The PCS may well use some of its own temperature sensors as well. The PCS will use this information plus its own internal lookup table to calculate the baseline set of forces to apply to the actuators.
- TCS will supply Zernike corrections to the PCS at 20 Hz (in addition to the open loop information) when running in closed loop with a WFS. The TCS will derotate the Zernikes to the primary frame of reference. The PCS will translate the Zernikes to corrections to the force matrix which it generates from its internal loop up table.
- The calibration WFS will be used in a TBD manner to generate corrections to the model inside the PCS which translates altitude/temperature to a vector of forces.

The primary mirror control system determines the desired set of output forces based on a set of desired Zernike coefficients representing the demand mirror figure. The PCS converts the Zernike representation of mirror figure to forces with the equation

$$O_i = \Sigma_j (M_{ij} * Z_j)$$

where O_i is the ith actuator force correction, M_{ij} is the influence matrix, and Z_j is the amplitude of the jth Zernike mode.

Although feasible there is no intent to close the active actuator loop by using the upper load cell to feedback into the PID controller. Rather the wavefront sensor and the lower load cell are used to close the loop.

16.5.2.1 SUPPORT UNIT LOCAL LOOP SPEED

The servo loop shall be capable of being closed between the actuator and the colocated lower load cell (sensor). This servo loop shall run at 10Hz. An average value for the load cell read back shall be calculated and stored. The number of samples present in the average is determined in collaboration with the primary mirror subassembly task.

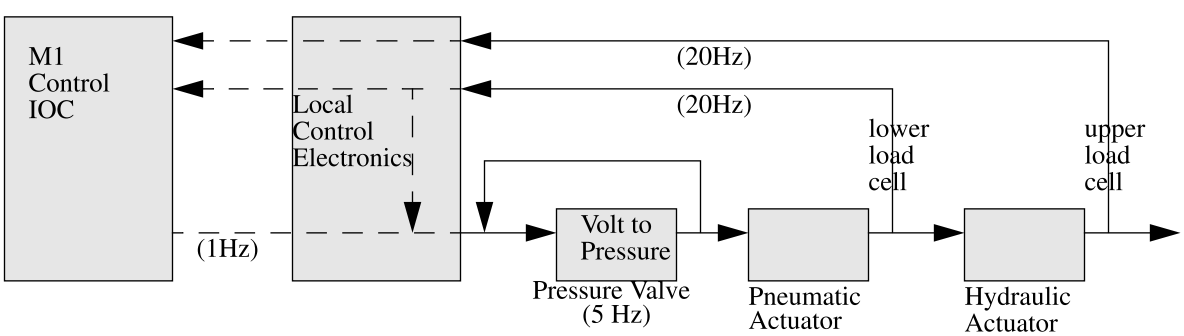
16.5.2.2 SUPPORT UNIT ACTIVE LOOP

At a rate determined by the WFS required exposure time the following actions shall be performed:

- The average lower (pneumatic component) load cell value shall be read from all of the support units and summed.
- The demand figure from the TCS shall be processed to obtain the required forces for each actuator.
- The new forces are sent to each of the support unit controllers.
- A clear average command is used to send commands to each of the support unit controllers.

This entire loop should run no slower than 0.2Hz.

FIGURE 16 - 7 **M1 Axial Support Control Logic**



16.5.3 Support and Definition

The primary mirror is mounted in a cell and faces upwards towards the zenith. This cell provides for axial and lateral support of the primary mirror in a changing gravity vector as the telescope tracks an object. The intent is that the optical axis of the primary shall be made to track the mechanical axis of the Cassegrain rotator at all times. Preliminary coarse alignment of the primary mirror shall be done at installation.

The primary mirror shall be supported axially and laterally by a passive support system which shall be capable of supporting the weight of the primary mirror cell in a changing gravity vector. This passive support system shall be capable of aligning the primary mirror along the six axes: x, y, z, tip/tilt, and rotation about the optical axis to maintain the optical alignment to the required accuracy. X and Y axis alignment and rotation about the optical axis shall be provided by the passive lateral supports while Z axis, tip and tilt alignment shall be provided by the passive axial supports.

The M1 Support and Definition System consists of the following five parts:

- Axial support
- Axial defining system for Tz,Rx, and Ry.
- Lateral support
- Lateral defining system for Ty,Rz.
- Lateral defining system for Tx.

The axial system is divided into either 3 or 6 zones depending on the amount of wind buffeting which must be resisted. The lateral system is divided into 2 zones. This division into zones is controlled by valves which subdivide the hydraulic actuators The overall height of a zone is controlled by a hydraulic cylinder which can add or subtract small amounts of fluid. The support pressures shall be changed due to the changing gravity vector and require an update rate of once every 10 seconds (TBD) as the telescope tracks.

The axial system in the 3 zone mode functions as a kinematic mount and can correct for rigid body piston (focus) and tip/tilt and is servoed off the axial position sensors.

When higher spatial frequency modes are present due to wind buffeting the axial support can be configured as a six zone system that is capable of correcting for astigmatism as well as focus and tip/tilt. In this mode it is servoed off of differential pressure sensors between each zone and by the wave front sensor data.

The zones may have controlled leaks between them to make them stiff at high frequency but compliant at low frequencies. The fluid volume in each passive zone is controlled based on feedback from position sensors and pressure sensors. The proper readings of these sensors are contained in look up tables as a function of telescope position and temperature.The frequency where the number of effective zones changed could be set by the flow constrictors. This would result in a mirror that resisted high frequency effects, such as wind buffeting, extremely well and yet appeared as a kinematic mount to low frequency effects, such as gravity and temperature.

An array of axial and lateral position sensors shall be used to provide the position of the primary mirror relative to the primary mirror cell. The location and numbers of the position sensors are TBD. The required readout rate is once per second.

The summation of the upper load cell readings for each zone can be used to monitor performance.

16.5.4 Air Pressure Support

The passive support system is augmented by the use of an Air Pressure Support System (APSS). The APSS bears 80% of the weight of the primary mirror cell at zenith and leaves the loading on the passive support system constant as the telescope tilts in zenith angle. The passive support system, the load cells, and the primary mirror must all survive failure of the APSS.

The air pressure support system employs the ‘air bag’ principle but is a fundamental departure from conventional air bag systems in that there is no physical bag. In this instance the ‘bag’ is formed by the rear face of the mirror and the upper face of the cell, with the gap between the two at the inner and outer diameters of the mirror being closed using a mechanical sealing device.

The chosen sealing system must effectively contain the pressurized air whilst allowing limited axial and transverse motion of the mirror.

The air pressure control is coarsely set by an independent, hardwired control system. The PCS just makes corrections to this value based on the summation of the upper load cell readings.

16.5.5 Safety Support (TBD)

There may exist both axial and lateral safety support systems. This system acts to guard against damage due to seismic events and to restrain the mirror (for instance) when the telescope is pointed at the horizon for top end exchanges.

The current safety support design baseline is a passive system, but it is required to use the axial support to push the mirror against the safety support during earthquakes and special handling (Zenith Angle = 90 degrees pointing and mirror recoating).

The intent is that the PCS simply monitors the state of the safety system. The safety system could be triggered by detection of large accelerations around the mirror cell (due to an earthquake) or it could be intentionally engaged by an operator turning a manual switch. There is no requirement for any computer control of this device.

16.5.6 Thermal Control

In order to subcool the primary mirror surface a cold plate may be used. This system is active during the day and possibly during the evening and consists of a cooled water-glycol mixture circulation system. The cooling control system could be used to set the temperature of the mirror surface to just below the predicted nighttime air temperature for that evening. The full details are TBD.

A network of thermal sensors are placed in 24 locations on and around the primary mirror and cell structure. These sensors are to be monitored at a TBD rate and are used to indicate the current temperature distribution on the surface of the primary mirror and in the air around the primary mirror.

A network of dewpoint sensors are placed in TBD locations on and around the primary mirror and cell structure. These sensors are monitored at a TBD rate and are used to indicate the current dewpoint distribution on the surface of the primary mirror and in the air around the primary mirror.

16.5.7 Interlock System

The following interlocks are all ‘passive’ meaning that the M1 support mechanisms are responsible for carrying out the remedies listed. The interlock system only monitors the status of these interlocks.

16.5.7.1 Air Pressure System Near Over Pressure Limit

Hazard: The sum of the 120 load cell forces is less than 10% of mirror weight.

Remedy: The following safety measures shall be taken:

- check air pressure sensor reading
- interlock air pressure regulator
- turn off air supply

16.5.7.2 AIR PRESSURE SYSTEM FAILURE

Hazard: The sum of 120 load cell forces is larger than 30% of the mirror weight. This will not harm the load cell system as it is designed to survive failure of the air pressure system.

Remedy: The following safety measures shall be taken:

- check air pressure sensor reading

16.5.7.3 LATERAL MIRROR DEFINING SYSTEM NEAR LIMIT

Hazard: The displacement at any lateral position sensor is larger than TBD mm.

Remedy: The following safety measures shall be taken:

- interlock master cylinder

16.5.7.4 AXIAL MIRROR DEFINING SYSTEM NEAR LIMIT

Hazard: The displacement at any axial position sensor is larger than TBD mm.

Remedy: The following safety measures shall be taken:

- interlock master cylinder

16.5.7.5 ACTIVE OPTICS SYSTEM LOADING NEAR LIMIT

Hazard: The actuator load cell reading, or the support load cell reading, is larger than 90% of the load cell loading limit.

Remedy: The following safety measures shall be taken:

- unload actuator force
- interlock actuator pressure regulator

16.5.7.6 LATERAL LOAD CELL LOADING NEAR LIMIT

Hazard: The lateral support load cell reading is larger than 90% of the load cell loading limit.

Remedy: The following measures shall be taken:

- rotate the telescope to zenith pointing

16.5.7.7 RADIATING PLATE SYSTEM

Hazard: The coolant line pressure drops.

Remedy: Turn off the circulation pump.

16.5.7.8 PERSONNEL WITHIN MIRROR CELL

Hazard: A person is inside the mirror cell.

Remedy: The following safety measures shall be taken.:

- interlock the telescope elevation drive system at zenith pointing
- interlock the telescope azimuth drive system

16.5.7.9 M1 CELL ASSEMBLY REMOVAL

Hazard: The M1 cell assembly is being removed for primary recoating.

Remedy: The following safety measures shall be taken:

- interlock the telescope elevation drive system at zenith pointing
- interlock the telescope azimuth drive system
- turn on the M1 cell axial support to push the mirror against safety support
- interlock all the M1 cell assembly active systems

16.5.7.10 SEISMIC HAZARDS

Hazard: The acceleration on the mirror is greater than 0.5g.

Remedy: The following measures shall be taken:

- turn on the M1 cell axial support to push the mirror against safety suppory
- rotate the telescope to zenith pointing

17 Details of the Primary (M1) Thermal Control System

17.1 Changes Since Preliminary Design Review

The M1 thermal system has been descoped to include only a cooling system. Due to this the functions were made part of the "Details of the M1 Control System" in Chapter 16 of this document in "M1 Thermal Control System" on page 16 - 2.

18 Details of the Secondary Control System

18.1 Function of the Secondary Control System (SCS)

The positions of the primary and secondary mirrors with respect to the mechanical rotation axis of the cassegrain rotator determine not only the position of the image centroid in the focal plane but also play a major role in determining the image profile at that position.

The dynamic component of the position of the secondary causes a temporal variation in not only the image centroid but also the image profile. The effect of integrating the science object on a detector is to broaden the image profile beyond that which would be observed in any one "snap shot" of the image — due to both the image centroid moving during the integration time but also the image profile changing during the same period.

A number of effects result in apparent motion of the secondary relative to the cassegrain rotator mechanical axis: gravity, temperature, and wind as well as errors in cassegrain rotator articulation.

Note that this applies to f/16 secondary - other configurations can be supported in a similar manner, though they may not use all of the features discribed here.

18.1.1 Gravity and Temperature Effects

Gravity and temperature changes cause repeatable and predictable effects resulting in decenter, defocus, and relative tip of the optics. Due to hysteresis, measurement error, sensor bandwidths, and structural complexities it is not possible to predict and remove 100%

of these effects based on measuring the current position and temperature of the structure involved. This results in some fraction of the power in these effects being classified as non-repeatable or unpredictable.

18.1.2 Wind Effects

Wind causes static and dynamic deflections of the secondary structure relative to the cassegrain rotator mechanical axis (its effect on the primary, while finite, is relatively small). The average wind speed results in a static deflection that, while in principle predictable, in practice is largely unpredictable due to the complex nature of the airflow within the enclosure. The dynamic component of the wind results in dynamic deflections about the static position. The most direct measurement of the dynamic effect is the focal plane position of a guide object. There are some effects, however, that result in similar motions in the focal plane. As an example, it is not possible to decouple the effect of decenter and relative tilt without further information. It may be necessary to directly measure the position of the secondary independently from the focal plane position of the image.

18.1.3 Sensors

A fast guider sensor in the Acquisition and Guiding (A&G) unit (or elsewhere) provides tip/tilt information at up to 200 Hz that can be used to maintain the image centroid in the focal plane. In practice this is used to correct the relative tip of the secondary — the problem that arises is that secondary decenter is being corrected by tilting the secondary and results in coma appearing in the wavefront.

The tip/tilt component of the atmosphere makes the image centroid appear to move in the focal plane of the telescope due to the overall tilt of the wavefront over the telescope aperture. The size of the individual atmospheric cells responsible for tilt decreases with wavelength and the aperture of the telescope effectively averages these. As one goes to shorter and shorter wavelengths the number of averaged cells increases to the point where the overall tilt is zero.

18.1.4 Atmospheric Effects

It is a requirement for the secondary system that it remove 90% of the tip/tilt power in the atmosphere. This requirement is valid during both 50% and 10% seeing conditions.

A complication arising from the atmospheric effects is that, if the object in the fast guider sensor is not close to the science object, their atmospheric tip/tilt components are decorrelated to some degree. This results in an additional broadening of the science image although there is a net gain over the guide field available — wind shake is always reduced.

18.1.5 Automatic Focusing

It is also a desire to provide some level of automatic focusing of the telescope while observing. It is believed that it is possible to acquire focus information at 15 Hz and this can be used to correct the focus of the telescope. The problem here is twofold; both secondary piston and primary figure can cause an apparent shift in focus and the atmosphere can cause a change in focus that are decorrelated at TBD distance from the guide object. The first problem results in an ambiguity as to what to do to correct the defocus measured — change the secondary piston or the primary figure. The second problem may result in a degradation of the science object if the guide object is not within the "isofocus" patch. Both of these effects will be addressed by future studies.

18.2 SECONDARY CONTROL OVERVIEW

18.2.1 External Interfaces

The Secondary Control System (SCS) has the following external interfaces:

- Telescope Control System (TCS)
- the SCS Control Console(s), provided by the Observatory Control System (OCS)
- the OCS screens system
- the M2 Tilt System
- the M2 Positioning System

18.2.2 External Bus Connections

The SCS is connected to the following external buses:

- Synchronization bus
- Interlock system
- Time Bus
- Event Bus

18.2.3 Internal Subsystems

The SCS is composed of the following major internal subsystems:

- 5-axis Articulation
- Mirror Support Monitoring
- Fast Focus
- Mirror Cover (in place switch/sensor only)
- Fast Guider
- Chopper
- Environmental Monitoring and Metrology
- Baffle control

18.2.4 Internal Interfaces

The SCS has the following major internal interfaces:

-

18.2.5 Internal Data Stores

The SCS has the following major internal data stores:

- SCS real time EPICS database

18.2.6 Computer Hardware

A VME crate containing all the interfaces necessary for the secondary control system is mounted on the exterior face of the telescope center section with a fiber optics connection to the upper end.

The VME crate is contained in an enclosure with a heat extraction system

18.2.7 Software Philosophy

The basic job of this software is to keep the static position of the secondary aligned to the cassegrain rotator mechanical axis, to correct for dynamic focal plane motion of the image centroid via tip/tilt of the secondary, and to run the chopper.

The philosophy is to embed as little intelligence as possible in the secondary control system. The SCS performs the actions commanded of it by the TCS. The TCS makes all decisions about the next desired position of the secondary and issue these commands. The Model for secondary positioning is embedded in the TCS.

The commands issued to the SCS are issued in the native coordinate system of the secondary system. This relieves the SCS from keeping track of its own absolute position.

The only intelligence embedded in the SCS is the capability to react to a stream of tip/tilt and defocus information supplied directly to reflective memory on the VME system. This is a bandwidth and time delay issue in that it is not possible for the TCS to acquire tip/tilt and defocus information, process it, transform it to the frame of reference of the secondary, and then issue commands to the SCS. In this case the TCS commands the SCS to accept data from a specific Wave Front Sensor (WFS), process it in a particular fashion, and then use this data to drive the secondary fast guider and fast focus systems.

18.3 External Interfaces

18.3.1 Interface to Telescope Control System

18.3.1.1 Requirements

The secondary control system accepts

- target translation, piston, and tip/tilt as $\{X,Y,Z,R_X,R_Y\}$.
- initial chop angle and angle velocity as $\{\Theta, \dot{\Theta}\}$

These parameters may be sent once or they may be continuously delivered at up to 20 Hz.

In addition to the targets the SCS accepts the commands:

- STOP; halt and hold the current position
- MOVE; go to the current target and stop
- FOLLOW; go to the current target and follow the target continuously
- STOP FOLLOWING; stop where you are
- GUIDE CONFIG; select WFS or combination of WFS for tip/tilt/focus information. If the secondary communicates directly with the WFS, then an additional interface supplied coordinate transform is required from the TCS so the SCS knows how to rotate the data into the secondary coordinate system.
- GUIDE; start correcting for tip/tilt/focus
- GUIDE STOP; go back to following
- SERVO CONFIG; configure servo system
- CHOP CONFIG; configure the chopper

- CHOP; start chopping
- CHOP STOP; go back to following or guiding
- SYNCH CONFIG; configure synchronization
- SYNCH; start synchronization
- SYNCH STOP; stop synchronization
- LEARN; adapt to input chopping waveform
- LEARN STOP; maintain adapted chopping
- PARK; move to a predetermined position and put secondary in a safe state
- ENCODER CONFIG; configure the encoders
- ZERO SET; zero different parts of the system

The SCS returns the following values to the TCS:

- Servo errors
- Status (e.g. "in position")

18.3.1.2 Implementation

The above data values are placed into an EPICS database by the TCS via channel access. The TCS then issues an event that causes the SCS to make the actual configuration conform to those values.

18.3.2 Interface to the SCS Console

18.3.2.1 Requirement

The consoles provided by the OCS must have fine grained control capability of the secondary.

The following commands are available:

- X TRANS; translate (decenter) the secondary along the x-axis
- X TIP; tip the secondary about the x (altitude) axis
- Y TRANS; translate (decenter) the secondary along the y-axis
- Y TIP; tip the secondary about the y-axis
- TBD

18.3.2.2 IMPLEMENTATION

The TCS uses EPICS Channel Access to set the appropriate parameters in the SCS's database and then issues an event that causes the SCS to match the system with those parameters.

18.3.2.3 SAMPLE CONSOLES

The consoles shown below are samples of those that the OCS provides to the operator for commissioning and interactive observing.

FIGURE 18 - 1 **Secondary alignment console**

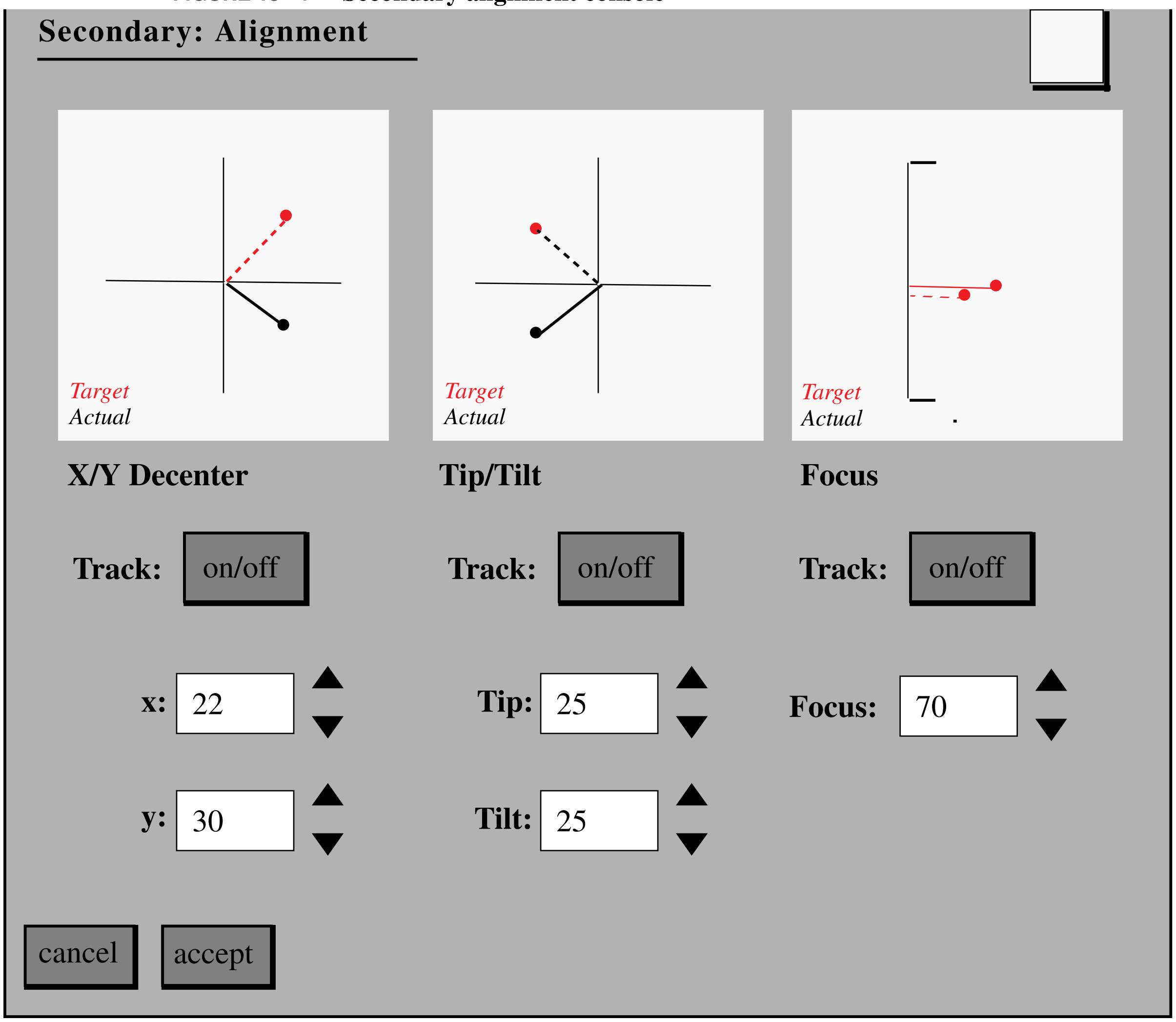


FIGURE 18 - 2 **Chopper control console**

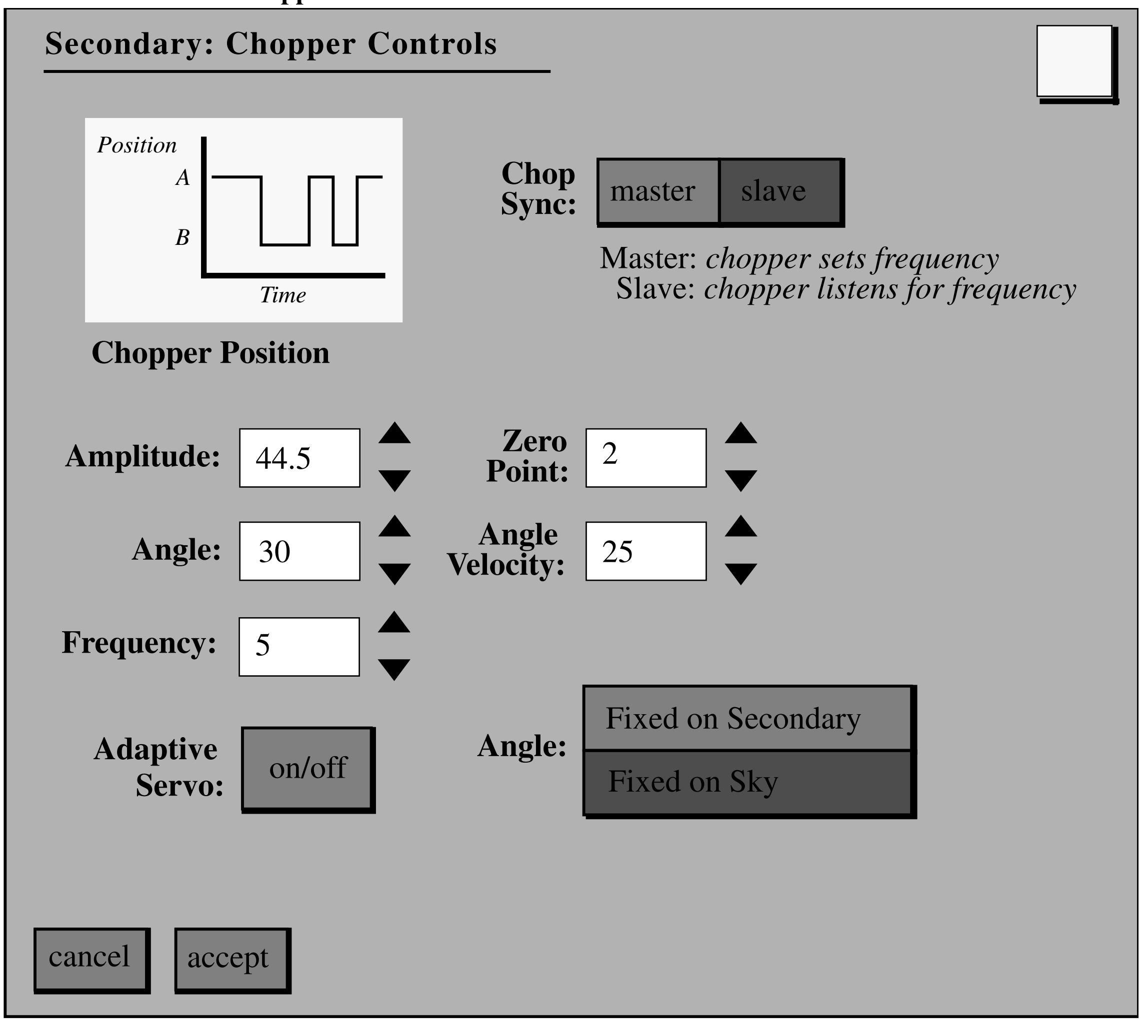
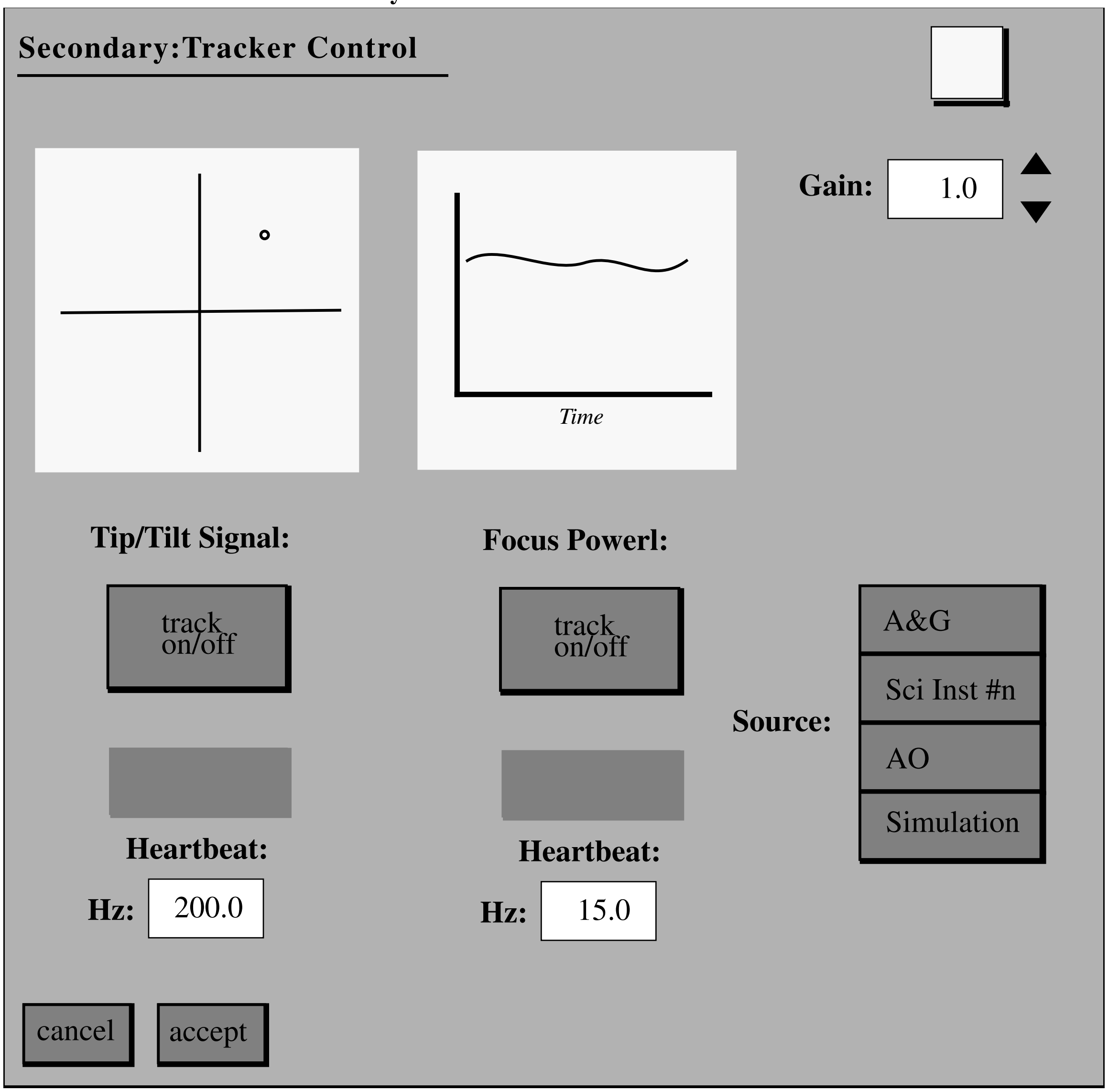


FIGURE 18 - 3 **Secondary tracker control console**



18.3.3 Interface to OCS Screens System

18.3.3.1 Encoder Screen

- Secondary mirror position transducers. These are used to measure the position of the secondary mirror on its articulating mechanism. The raw readings are displayed as integers in the range +- TBD microns and +- TBD arcseconds.

  --SECONDARY X Translation

  --SECONDARY Y Translation

  --SECONDARY X Tip (fast & slow)

  --SECONDARY Y Tilt (fast & slow)

  --SECONDARY Z Focus (fast & slow)

  --alternatively one could read out the values of the three raw encoders
- Secondary Module in Place

  --F6|F16

18.3.3.2 Limit Screen

- Focus limits.

  --LIMIT Z+

  --LIMIT Z-
- Translation Limits

  --LIMIT X+

  --LIMIT X-

  --LIMIT Y+

  --LIMIT Y-
- Tip Limits

  --LIMIT Rx-

  --LIMIT Rx+

  --LIMIT Ry-

  --LIMIT Ry+

18.3.3.3 Alarms Screen

- TBD

18.3.3.4 INTERLOCKS SCREEN

--TBD

18.3.4 M2 Tilt System

The SCS tip/tilt/focus unit is a physically separate unit from the f/16 secondary mirror positioning system.

The tilt system which the vendor supplies consists of

- a Heurikon MSE/12 VME crate, Motorola MVME-167 CPU, VME interfaces,
- fiber optic connection (20 meters) from VME modules to M2 Tilt Control Electronics module,
- M2 Tilt Control Electronics module
- connection between the M2 Tilt Control Electronics module and the M2 Tilt Sensor Electronics module
- connection between the M2 Tilt Control Electronics module and the M2 Tilt Mechanism
- M2 Tilt Sensor Electronics module
- connection betweenthe M2 Tilt Sensor Electronics module and the M2 Tilt Mechanism,
- the M2 Tilt Mechanism.

It is vendor’s choice what software to run on VME crate to do acceptance testing. Gemini uses EPICS running on top of Wind River’s VxWorks. If the vendor already has VxWorks expertise then we encourage its use.

The SCS interfaces to the vendor’s equipment as follows.

TABLE 18 - 1 **General Functions**

Function	Description	Range	Frequency	Type
Power	turn power on/off to entire unit	ON | OFF	< 0.1 Hz	digital switch
Health	monitor overall status of device	OK | BAD	1 Hz	digital o/p
Servo Parms	up/down load servo parameters	TBD	< 0.1 Hz	digital i/o
BW	target tip/tilt bandwidth (driven by available sampling)	< 50 Hz	< 0.1 Hz	digital i/p
Sample Freq	samples/sec available	1 - 200 Hz	< 0.1 Hz	digital i/p
HeartBeat	incrementing counter	32 bits	200 Hz	digital o/p

TABLE 18 - 1 **General Functions**

Function	**Description**	**Range**	**Frequency**	**Type**
WatchDog	shutdown secondary if host dies		200 Hz	digital i/o
IN POS	tolerance within which device is *in position*	+- TBD urad	< 0.1 Hz	digital i/p
Clock	supply clock to vendor VME crate			digital o/p

TABLE 18 - 2 **Chopper Functions**

Function	**Description**	**Range**	**Frequency**	**Type**
Chop Synch	waveform for chopper to follow (TTL or 3 state logic might be possible here)	+- 10 V	< 10 Hz	analogue i/p
Chop On/Off	start/stop chopping	ON|OFF	< 0.1 Hz	digital i/p
Frequency	set chop frequency	0.01 - 10 Hz	< 0.1 Hz	digital i/p
Amplitude	set chop amplitude	+- TBD	< 0.1 Hz	digital i/p
Chop Mode	set chop waveform	see below	< 0.1 Hz	digital i/p
Zero Point	set chop zero point	+- TBD	< 0.1 Hz	digital i/p
Angle	set demand chop angle	0 - 360 deg	20 Hz	digital i/p
Angle Velocity	set demand chop angle velocity	+- 1 deg/sec	20 Hz	digital i/p
Learn	adapt to input chopper waveform	ON|OFF	< 0.1 Hz	digital i/p

TABLE 18 - 3 **Monitoring Functions**

Function	**Description**	**Range**	**Frequency**	**Type**
Temp	return temperature probe data	+- 40 deg	1 Hz	digital o/p
Servo Error	actuator servo errors	+- TBD micron	200 Hz	digital o/p
Misc	any outputs such as accelerometers, etc	TBD	200 Hz	digital o/p
In Position	signals that servo error is less than "IN POS"	IN|OUT	20 Hz	digital o/p
In Position	same	TBD volts	20 Hz	analogue o/p
diagnostics	TBD			digital

TABLE 18 - 4 Tip/Tilt/Focus Functions

Function	Description	Range	Frequency	Type
X Tip	tip mirror about X axis	+- TBD urad	< 200 Hz	digital i/p
Y Tip	tip mirror about Y axis	+- TBD urad	< 200 Hz	digital i/p
Z fast piston	change mirror piston (focus)	+- TBD micron	< 200 Hz	digital i/p
Z slow piston	change mirror piston (focus)	+- TBD mm	< 0.1 Hz	digital i/p
{A1,A2,A3}	move individual actuators	+- TBD micron	< 200 Hz	digital i/p

The tip/tilt/focus functions have a varying range with frequency — larger motions at low frequency with smaller motions at higher frequencies. Gemini Optics Group can supply details.

TABLE 18 - 5 Momentum Compensation Functions (if used)

Function	Description	Range	Frequency	Type
X Tip	tip compensating mass about X axis	+- TBD urad	< 200 Hz	digital i/p
Y Tip	tip compensating mass about Y axis	+- TBD urad	< 200 Hz	digital i/p
Z fast piston	change compensating mass piston (focus)	+- TBD micron	< 200 Hz	digital i/p
{A1,A2,A3}	move individual compensating mass actuators	+- TBD micron	< 200 Hz	digital i/p
diagnostics	TBD			

It is up to vendor whether to use momentum compensation and, if used, whether to supply any external control. If used, vendor should supply diagnostic information relating to performance and system health.

18.3.4.1 NOTES

- Chop mode can be one of {Saw|2 Point|3 Point}
- very small latency associated with tip/tilt signals ~100 usec from time input signal present to time signal is applied to actuator
- probably need more than one data channel as slow commands (0-20 Hz) could come serially but fast commands need to come in parallel

18.3.4.2 TYPICAL USE OF VENDORS EQUIPMENT

Optical Alignment. As telescope is tracked across to sky SCS sends {X Tip, Y Tip, Z slow piston} commands at 20 Hz in order to maintain optical alignment.

Fast Steering Init. Before starting to fast steer the SCS downloads servo parameters appropriate for target bandwidth (availability of natural guide stars limits sampling bandwidth).

Fast Steering. As telescope is tracked the SCS sends {X Tip, Y Tip} signals at 200 Hz in order to keep target fixed in focal plane of telescope.

Fast Focus. As telescope is tracked the SCS sends {Z fast piston} at 20 Hz in order to maintain telescope focus.

Chop Initialization. Before starting to chop SCS downloads servo parameters as well as {Amplitude, Zero Point, orientation (to sky or to top-end)}

Analogue Chop. Chopper synchronizes itself with {Chop Synch} but use {Amplitude, Zero Point} to throw. SCS delivers {Angle, Angle Velocity} at 20 Hz while system is chopping.

Digital Chop. SCS sends {Frequency, Amplitude, Chop Mode, Zero Point} to chopper. SCS delivers {Angle, Angle Velocity} at 20 Hz while system is chopping.

Chop Adjust. While system is chopping SCS may send new values of {Amplitude, Zero Point} in order to fine tune chop.

Adaptive Chop (Optional). SCS sends {Learn On} to chopper and chopper must adapt and produce square wave output. When SCS sends {Learn Off} chopper stops adapting and run with new servo parameters. If SCS sends {Servo Parms Up} then chopper supplies adapted server parameters (Gemini intends to build up database so that adapted parameters can be predicted and downloaded).

18.3.5 M2 Positioning System Interface

TABLE 18 - 6 **5-axis System Functions**

Function	Description	Range	Frequency	Type
X/Y Position control	Change demand translation.	5 mm range in X, 10 mm range in Y	< 1 Hz	digital o/p
X/Y Position readback	Monitor translation.	5 mm range in X, 10 mm range in Y	< 1 Hz	digital o/p
X/Y Velocity control	Control velocity of translation.	TBD	< 1 Hz	digital i/p
X/Y Velocity readback.	Monitor velocity of translation.	TBD	< 1 Hz	digital i/p
Operating mode.	Select operating mode.	Operator or LUT	< 1 Hz	digital o/p

TABLE 18 - 7 **Mirror Support System Functionality**

Function	Description	Range	Frequency	Type
Loadcells	Monitor Loadcells.	TBD	< 1 Hz	digital i/p

TABLE 18 - 8 **Mirror Cover System Functionality [TBD]**

Function	Description	Range	Frequency	Type
Mirror Cover Monitor	Monitor mirror cover status.	TBD	< 1 Hz	digital i/p

TABLE 18 - 9 Monitoring & Metrology System Functionality

Function	Description	Range	Frequency	Type
Position sensors	Monitor position.	TBD	< 1 Hz	digital i/p
Temperature sensors	Monitor temperature.	TBD	< 1 Hz	digital i/p
Accelerometer	Monitor acceleration.	TBD	< 1 Hz	digital i/p
Gyro	Monitor Gyros.	TBD	< 1 Hz	digital i/p
Laser position-iong system	TBD	TBD	< 1 Hz	digital i/p
Laser position system power	Turn laser position system on/off	ON/OFF	< 1 Hz	digital o/p

TABLE 18 - 10 Baffle Control System Functionality

Function	Description	Range	Frequency	Type
Baffle Position control	Set baffle position.	Retracted | Inner | Outer	< 1 Hz	digital o/p
Baffle Position readback	Monitor baffle position.	Retracted | Inner | Outer	< 1 Hz	digital i/p

18.4 EXTERNAL BUS CONNECTIONS

18.4.1 Synchronization Bus

The secondary control system must be capable of accepting wave front sensor information and correcting the tip, tilt, and focus components of this.

18.4.1.1 REQUIREMENTS

The SCS accepts wave front sensor information as Zernike (or other orthogonal parameters) parameters $\{Z_1,Z_2,...,Z_N\}$ for the measured wave front in the focal plane. These parameters may be sent once or they may be continuously delivered at up to 200 Hz

The SCS calculates the change in the tip, tilt, and piston of the secondary needed to remove the tip, tilt, and focus components of the wavefront.

There are a number of different possible sources available that could generate wave front information. The secondary must be able to use any of them. The TCS instructs the SCS as to which wave front source to use.

The SCS generates a digital IN POSITION signal on the synchronization bus indicating that the servo errors of the Tip/TilT/Focus subsystem are within limits.

18.4.1.2 IMPLEMENTATION

The above targets are contained in the SCS as EPICS database values. These database values are updated via reflective memory from the wave front sources. Each wave front source has distinct database values in the SCS.

The IN POSITION value is an EPICS database value that is updated by the SCS. The action of the SCS writing to this EPICS value causes the reflective memory system to read it and update the other VME reflective memory systems.

18.4.2 Interlock System

The SCS must be capable of generating interlock requests as well as responding to them.

18.4.3 Time Bus

The time within the SCS VME crate is set to ±5 microseconds from the time bus.

18.4.4 Event Bus

18.4.4.1 REQUIREMENTS

The detector subsystems must be able to synchronize their readouts with the chopping and nodding of the secondary and telescope.

18.4.4.2 IMPLEMENTATION

The Secondary Tip/Tilt/Focus system accepts an analogue signal that sets the wave form shape of the chopper. The amplitude, angle and zero point of the chop are set by the TCS.

This signal normally comes from the TCS and is input to both the chopper and detector subsystems. For detectors that must generate this signal then the SCS accepts the signal from that source instead.

The Secondary Tip/Tilt/Focus system also generates an “in position” signal that is generated as an analogue signal.

18.5 SCS INTERNAL SUBSYSTEMS

18.5.1 5-Axis Articulation

The secondary can be manipulated in 5 degrees of freedom at slow speeds — the only degree of freedom not provided is rotation about the optical axis. The range of motion required for the M2 Tilt System is:

- Tilt: 2 degrees total in slow mode, 2.2 arcminutes total in fast mode.
- Focus: 15 mm total in slow mode, 25 microns total in fast mode.

The following values are the current target for the M2 Positioning System and represent ~2x what is understood at this time as the actual requirement:

- X translation 5 mm total.
- Y translation 10 mm total.

The calibration procedure for pointing and tracking generates a LUT that keeps the secondary fixed in the optical support structure reference frame from 0 to 75 degrees zenith angle.

As well this LUT must be calibrated for the temperature of the telescope (mostly the tube) in order to maintain focus. A change in secondary piston of 2.4 microns results in a blooming of the image of 0.01 arcseconds in 50% encircled energy. Given that the tube is some 11 meters long and has a CTE of 12 ppm, a 0.6 degree temperature change of the tube blooms the image by 0.165 arcseconds over a 1 hour exposure.

The secondary articulation mechanism responds to movement commands for each of the five DOF; each command may contain position and velocity commands that are acted upon until removed. The LUT is implemented in the position control system (aka TCS) which then issues commands to the secondary control system.

18.5.2 Mirror Support System

Unlike the primary the secondary support system is passive. However there is a desire to monitor the status of the support system to validate that it is performing properly. This is implemented via a series of tilt or position sensors.

The control system must provide for status requests and alarm conditions based on this metrology.

18.5.3 Fast Focus System

It is a goal of the project to provide a real time metric of focus at a 10-20 Hz rate during observing with all focus errors corrected by movement of the secondary. This metric is used to adjust the position of the secondary in order to null the focus metric via a servo loop running at 1/5 to 1/10 of the sample frequency. The deviation of the secondary focus position from that predicted by the LUT may be used as a metric of the change in curvature of the primary. Once a minute this metric is used to input a change to the primary radius of curvature.

If it proves unfeasible to provide the real time focus metric then this part of the work package must provide a more conventional means of focusing the system. Systems similar to this are in use on telescopes such as CFHT and take multiple exposures at different telescope focus positions in order to calculate the position of best focus. In this case the active optics sensor provides a focus metric that is applied directly to the secondary.

It may be necessary to use the metrology system discussed below to maintain the focus position of the secondary independently of any focal plane information.

18.5.4 Mirror Cover System (TBD)

There is a TBD mirror cover system for the secondary mirror. The only requirement is for the system to be able to monitor whether the cover is in place or not.

18.5.5 Fast Guider System

The lynch pin of the image quality error budget is based on the secondary's capability to remove wavefront tip/tilt (both global and atmospheric) by rapidly adjusting the tip/tilt of the secondary mirror.

As this system requires sampling of the tip/tilt at frequencies as fast as 200 Hz it is vital that a deterministic data channel be used between the source of the tip/tilt information (in general the A&G unit) where the fast guider sensor is located and the secondary mirror. The form of this interaction takes place in a number of ways.

- the position control system can monitor the tip/tilt information from the fast guider sensor, calculate the required motion of the secondary, and directly command the secondary motions
- the position control system can instruct the secondary control system to monitor a specific port and carry out a resident algorithm to calculate the required motions of the secondary
- the position control system can also issue commands to the fast guider system for other purposes, such as trailing objects, etc.

The fast guider system does not directly command the position of the secondary as this would require the fast guider system to know a great deal of state information about both the telescope and the secondary system.

18.5.6 Chopper System

At infrared wavelengths beyond 2.2 microns (and perhaps only at 10 microns and beyond) there is a requirement to "chop" the secondary between two positions on the sky in order to remove the sky background. This is a result of the fact that the sky background can be a factor of 106 brighter than the science object and so must be flat fielded to a very high degree. Due to changes in the background on timescales shorter than a typical astronomical exposure these infrared instruments are read out at several 10's of Hz and the secondary mirror is chopped between sky locations at 1-10 Hz over a distance of several 10's of arcseconds.

Due to the large size of the isokinetic patch at longer infrared wavelengths there may be a considerable gain in image size if the atmospheric tip/tilt is removed. There are a number of options for implementing this while chopping:

- chop on top of tilt — in this mode the tilt frequency is less than the chop frequency; i.e. tip/tilt correction at 1 Hz while chopping at 10 Hz. The tip/tilt sensor could measure the wavefront error during one half of the chop cycle
- tilt on top of chop — in this mode the tilt frequency is higher than the chop frequency; in effect the system is continuously adjusting the end points of the chopper throw. There are some quite challenging problems to be overcome in terms of the tip/tilt sensor in this mode as the object used for fast guiding is moving on the sensor due to the chopping.
- tilt half the time — in this mode the science object is placed on axis and a fast guide object is used to correct for tip/tilt. When the secondary chops "off source" there is no correction. One possible problem with this mode is that, if the science object is not chopped of the detector array, then one beam has a significantly degraded PSF relative to the other.

- slow chop — if the secondary is chopped slowly enough, then it might be possible to move the guide probe to reacquire the guide object and effectively do tip/tilt correction in both beams.
- slow nod — if there is not a requirement to remove the effect of the telescope but just the sky then it may be possible to move the telescope rather than the secondary if the rate is slow enough. In this case one would have to move the probe as in slow chop.
- post process — in this mode the fast guide information is not used to correct the secondary but is sent to the detector to be used for “shift and add” or equivalent. The A&G system must provide the information in the native coordinate system of the detector, most likely rows and columns.

The methods that are feasible are dictated by the number of wavefront sensors that are available and their field of view. It is likely that the numbers and fields of view is such (single WFS, small FOV), that there is only be a tip/tilt signal for 50% of the chop cycle.

18.5.7 Monitoring and Metrology

There is a requirement to monitor which secondary module and which telescope upper end is in place on the telescope. In addition there may be position sensors, tilt sensors, temperature probes, strain gauges, accelerometers, and gyros in place on the secondary subsystem.

It may prove desirable to directly measure the position of the secondary independently of a focal plane image. The baseline proposal for this is to use a HeNe laser system that incrementally measures the secondary position relative to the cassegrain rotator axis. For science observations where the HeNe would not interfere this system could be left on and used to servo the 5 DOF position of the secondary to correct for unpredictable static deflections. The SCS should be capable of handling this possible upgrade.

It might be necessary to periodically recalibrate this system during the night by going back to a bright star and using the primary active system.

18.5.8 Baffle Control System

Observations at different wavelengths require different kinds of telescope baffling. At optical and ultraviolet wavelengths the telescope must be fully baffled, but at infrared wavelengths there should be little or no baffling. The secondary mirror is equipped with a deployable baffle, whose control is the responsibility of the secondary control subsystem.

The system should be capable of moving the baffle to three pre-defined locations:

Retracted. The baffle is completely retracted — for observations at most infra-red wavelengths.

Inner. The baffle covers only a small annulus around the secondary mirror — for observations at short infra red wavelengths.

Outer. The baffle covers the maximum area around the secondary mirror — for optical and ultraviolet observations.

The system should also be capable of sensing the current location of the baffle.

Experimentation with some instruments may reveal a need to move the baffle to other positions. The deployable baffle may not be designed for extension of the range of positions. Fixed, removable baffles are also being considered for some scenarios.

The primary mirror is also equipped with a baffle, described in “A&G Baffle Sensing Subsystem” on page 19 - 21.

19 Details of the Acquisition & Guiding System

19.1 Function of the Acquisition & Guiding System

The Acquisition and Guiding (A&G) Control System makes possible the acquisition of target and guide star(s) and the subsequent calibration of the data. Precise acquisition is likely to matter for all GEMINI instruments. Not only do spectrometers use tiny entrance apertures, but even in the case of imaging instruments it is often desirable to return to the same field with excellent reproducibility.

The required functions of the A&G Control System include the enabling of acquisition so objects can quickly and reliably be placed in pre-defined positions in the focal plane; provide error signals, based on tracking guide star(s), so that servo loops involving telescope, secondary, and instrument rotator, can be closed; provide data on the incoming wavefront; and provide other common optical services, such as even focal plane illumination, field curvature correction, atmospheric dispersion compensation, polarization modulation, and filters.

19.2 A&G Control Overview

19.2.1 External Interfaces

The A&G Control System has the following external interfaces:

- Telescope Control System
- the A&G Control Console(s) - provided by the OCS
- the OCS screens system

- the A&G hardware

19.2.2 External Bus Connections

The A&G Control System is connected to the following external buses:

- Synchronization bus
- Interlock system
- Time Bus
- Event Bus
- Control LAN

19.2.3 Internal Subsystems

The A&G Control System consists of the following subsystems:

- A&G Guider Subsystem - includes the XY Guider and the peripheral WFS subsystems (Shack-Hartmann and wavefront sensors).
- A&G Science Fold Subsystem
- A&G Acquisition Camera Subsystem
- A&G High-resolution WFS Subsystem
- A&G Commissioning and Quality Control Subsystem
- A&G Baffle Sensing Subsystem
- A&G Field Curvature Correction Subsystem
- A&G Atmospheric Dispersion Compensation Subsystem (if in A&G)
- A&G Calibration Subsystem
- A&G Optical Polarization Modulator Subsystem [TBD]

Note that the A&G Field Curvature Correction Subsystem and A&G Atmospheric Dispersion Compensation Subsystem may be combined into one subsystem. This is TBD.

19.2.4 Internal Interfaces

The A&G Control System has the following major internal interfaces:

-

19.2.5 Internal Data Stores

The A&G Control System has the following major internal data stores:

- Information on shape of ???? surface
- Information as a function of field position
- Records of measured seeing for preceding 10 minutes or so
- Time-averaged data for WFS

19.2.6 Computer Hardware

The A&G Control System is controlled by two VME crates, the A&G VMEbus crate located on the Instrument Support Structure assembly and the Cassegrain Rotator VMEbus crate located on the external surface of the telescope center section. The A&G VMEbus crate is responsible for the A&G Guider Subsystem, A&G Acquisition Camera Subsystem, A&G Commissioning and Quality Control Subsystem, A&G Calibration Subsystem, A&G Slow Active Optics Subsystem, A&G Baffle Sensing Subsystem, and A&G Optical Polarization Modulator Subsystem.

The A&G Field Curvature Correction Subsystem and A&G Atmospheric Dispersion Compensation Subsystem are controlled by the Cassegrain Rotator VMEbus crate, which is also responsible for the Cassegrain Rotator Assembly.

It is possible to connect a workstation directly to the VME crates via ethernet in order to control the system in a stand-alone mode. This stand-alone user interface is provided through EPICS.

In operation the VME crates are connected to the telescope control network and there is no stand-alone console.

19.2.7 Software Philosophy

TBD

19.3 IMPACT ON A&G OF CHOPPING AND NODDING

In order to flat field data in the thermal infrared it is necessary to

- chop the secondary between two positions on the sky at up to 10 Hz
- nod the telescope between two positions on the sky at up to 0.2 Hz

The impact on the A&G depends on whether one or two A&G guide probes are used.

19.3.1 Single A&G Probe

If a single A&G probe is used then there is no signal available half the time for the tip/tilt/focus guider. This seriously degrades the removal of wind shake as the effective bandwidth of the servo is reduced to approximately 1/5 of the chop frequency.

In order to nod the telescope it is necessary to move the guide probe and the telescope in order to recenter both on the new sky position.

19.3.2 Two A&G Probes

In this situation there is a tip/tilt signal at both ends of the chop cycle so the removal of wind shake is improved.

In order to nod the telescope it is necessary to move both guide probes and the telescope in order to recenter both on the new sky position.

19.4 A&G and AO Wavefront Sensor Issues

There are 4 (or possibly 5) logical wavefront sensors used within the Gemini system. The first 3 are *facility* wavefront sensors in the A&G unit itself:

1. Fast Peripheral WFS; this is used to get tip/tilt/focus at up to 200 Hz by observing a star outside of the science field. This will likely be the equivalent of a 2 x 2 Shack-Hartmann. This information will be used to correct wind shake and atmospheric effects.
2. Slow Peripheral WFS; this is used to get low spatial order information on the figure of M1. In order to remove the effects of seeing it will integrate for about 60 seconds on a star outside the science field. This will likely be the equivalent of a 4 x 4 or 6 x 6 Shack-Hartmann. It will generate Zernike coefficients which will be passed to the M1 control system as corrections to the force distribution on M1. The M1 control system will have a simple algorithm which will look at the values of the Zernike coefficients and calculate the force distribution required to correct this. e.g. we know that for 1 wave of astigmatism we need to put the force vector (F1, F2, …, F120} on M1, as we have 0.15 waves measured we will multiply the vector by 0.15. This is done for each of the Zernike coefficients and the results summed to get a single force vector to be applied.

   This implies the A&G Control System and Primary Support Control System need to exchange information frequently.
3. High resolution WFS; this is used to get high spatial order information on the figure of M1. In order to remove the effects of seeing it will be integrated for a minute. This WFS works on axis and so precludes observing while it is in use. It will be the equivalent of a 20 x 20 Shack-Hartmann. As we expect that, at this resolution (400 ele-

ments across M1), the M1 surface will not be easily represented by zernike coefficients (one would have to go to quite high order in order to represent the effect of a single actuator sticking up) we will use a fitting technique to calculate the baseline set of forces to be used. This WFS will be used at the start of the night and perhaps one or two times during the night. This WFS may be treated like a scientific instrument. See “High Resolution Wavefront Sensor” on page 12 - 36.

4. *On-board* WFS on a scientific instrument; this is used to generate tip/tilt/focus at up to 200Hz but more typically at around 50Hz. This will be the equivalent of a 2 x 2 Shack-Hartmann. This WFS will be used to mainly correct atmospheric effects.
5. There may also be a wavefront sensor within the Adaptive Optics (AO) module.

Some cooperation between the ICS and the A&G Control System will be required, as will a rapid communication link between the A&G Control System and AO systems.

There is a need to switch between wavefront sensors depending on which instrument is in use. For instance, an instrument with an *on board* WFS may be switched out in favor of an instrument that needs to use a *facility* WFS.

There is also a need to use the same WFS’s for adaptive optics as well. Given the speed with which AO may need to run the WFS’s there may need to be a trade study to see if the method which works for A&G wavefront needs (< 200 Hz) is sufficient for AO needs.

The wavefront sensors will probably be commercial CCD cameras. The same detector control systems used for the science instruments could be used here. The sensors will behave just like separate instruments — generating images which need to be processed. Because of the time constraints, it is not sensible to have these images processed by the DHS. The A&G Control System will have its own built-in processors to do this. However, there is no reason why the A&G Control System cannot make use of library functions which have been tried and tested in the External Data Reduction System. The DHS could provide a means of saving and displaying the wavefront images for engineering purposes. This could be done by treating the A&G system exactly like an instrument.

19.5 EXTERNAL INTERFACES

19.5.1 Interface to Telescope Control System

19.5.1.1 REQUIREMENTS

The A&G Control System accepts positional/focus parameters $\{X,Y,Z,\dot{X}, \dot{Y}, \dot{Z}\}$ for the target position/focus and velocity of the two peripheral guide probes. These parameters may be sent once or they may be continuously delivered at up to 20 Hz.

In addition to the targets the A&G accepts the commands:

- STOP; halt and hold the current positions
- MOVE; go to the current target and stop
- FOLLOW; go to the current target and follow the target continuously
- STOP FOLLOWING; stop where you are
- PARK; move to a predetermined position and put A&G in park state
- PROBE CONFIG, configure a WFS probe
- ENCODER CONFIG; configure the encoders
- ZERO SET; zero different parts of the system

The A&G returns the following values:

- Number of Zernike coefficients
- $\{Z_1,Z_2,...,Z_N\}$, Zernike coefficients of measured wavefront
- the above must be corrected for detector pixel size
- time stamp for Zernike measurements (to allow “best before” processing)
- data quality measure of Zernike coefficients
- Medium transmission
- Guide error statistics
- FWHM of guide star
- Relative transmission
- magnitude of guide star
- Encoder readings
- Drive demand

- Servo errors
- Alarms
- Status (e.g. "in position")

19.5.1.2 IMPLEMENTATION

The above data values are contained in an EPICS database that communicates with the TCS via channel access.

The number of Zernike coefficients, the Zernike coefficients, the time stamp, and data quality are stored in EPICS database values that are connected to the reflective memory bus.

19.5.1.3 REQUIREMENT

The A&G system must be capable of inserting the high-resolution wavefront sensor into the telescope beam in order to generate corrections to the look up table used to correct the primary mirror figure. A capability to do an automated calibration test (similar to an automated pointing test) is desired.

The A&G system accepts the following commands from the TCS.

- CALIBRATE CONFIG
- START CALIB
- GET SHDATA, retrieve raw data

19.5.1.4 IMPLEMENTATION

The above data values are contained in an EPICS database that communicates with the TCS via channel access.

In turn, the OCS obtains these values from the TCS via channel access.

19.5.2 Interface to A&G Console

19.5.2.1 REQUIREMENT

The consoles provided by the OCS must have fine grained control capability of the A&G system. The following commands are available:

- *TBD*

19.5.2.2 IMPLEMENTATION

The TCS provides access to any A&G system console parameters to the A&G using EPICS Channel Access commands.

19.5.2.3 SAMPLE CONSOLES

The consoles shown below are samples of what the OCS provides to the operator for commissioning and interactive observing.

FIGURE 19 - 1 **Main A&G console**

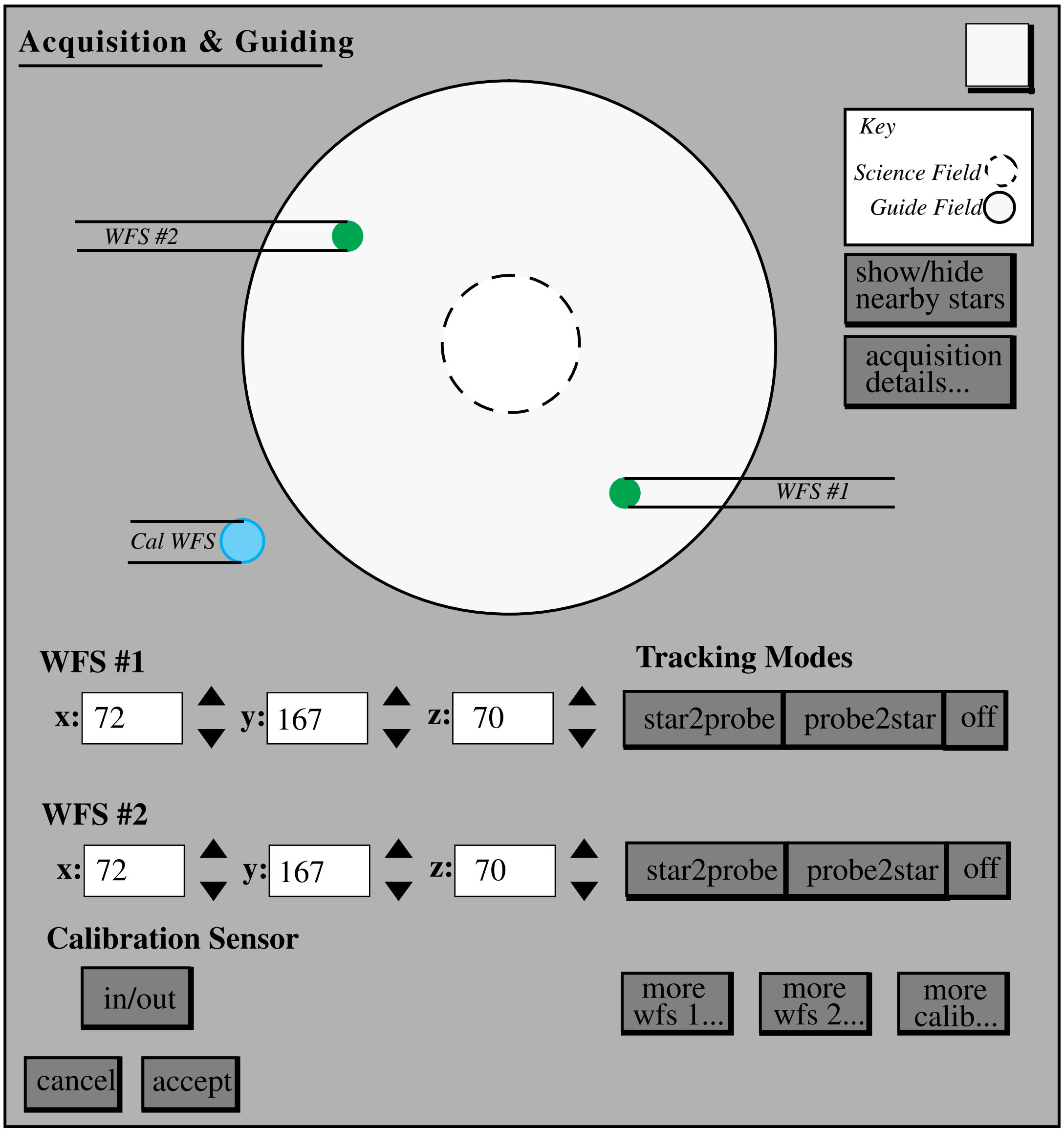


FIGURE 19 - 2 **A&G field acquisition console**

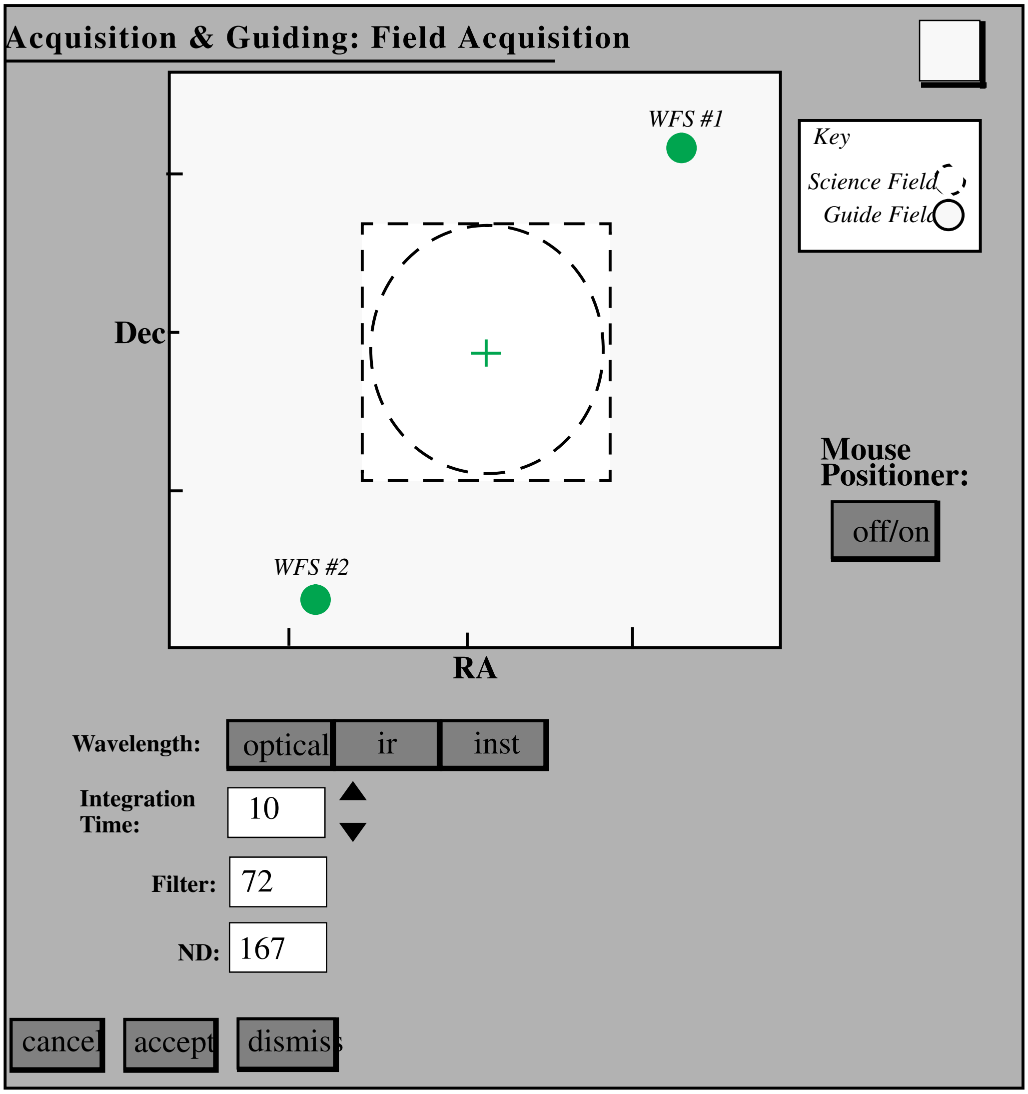


FIGURE 19 - 3 **WFS #1 Console**

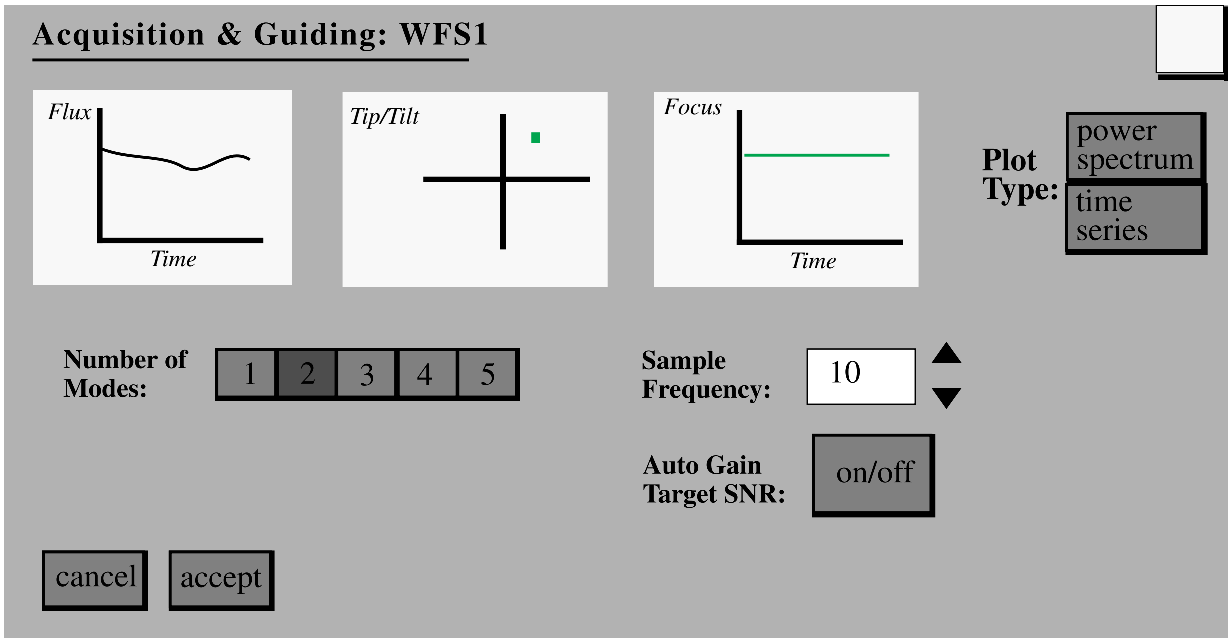


FIGURE 19 - 4 **WFS #2 Console**

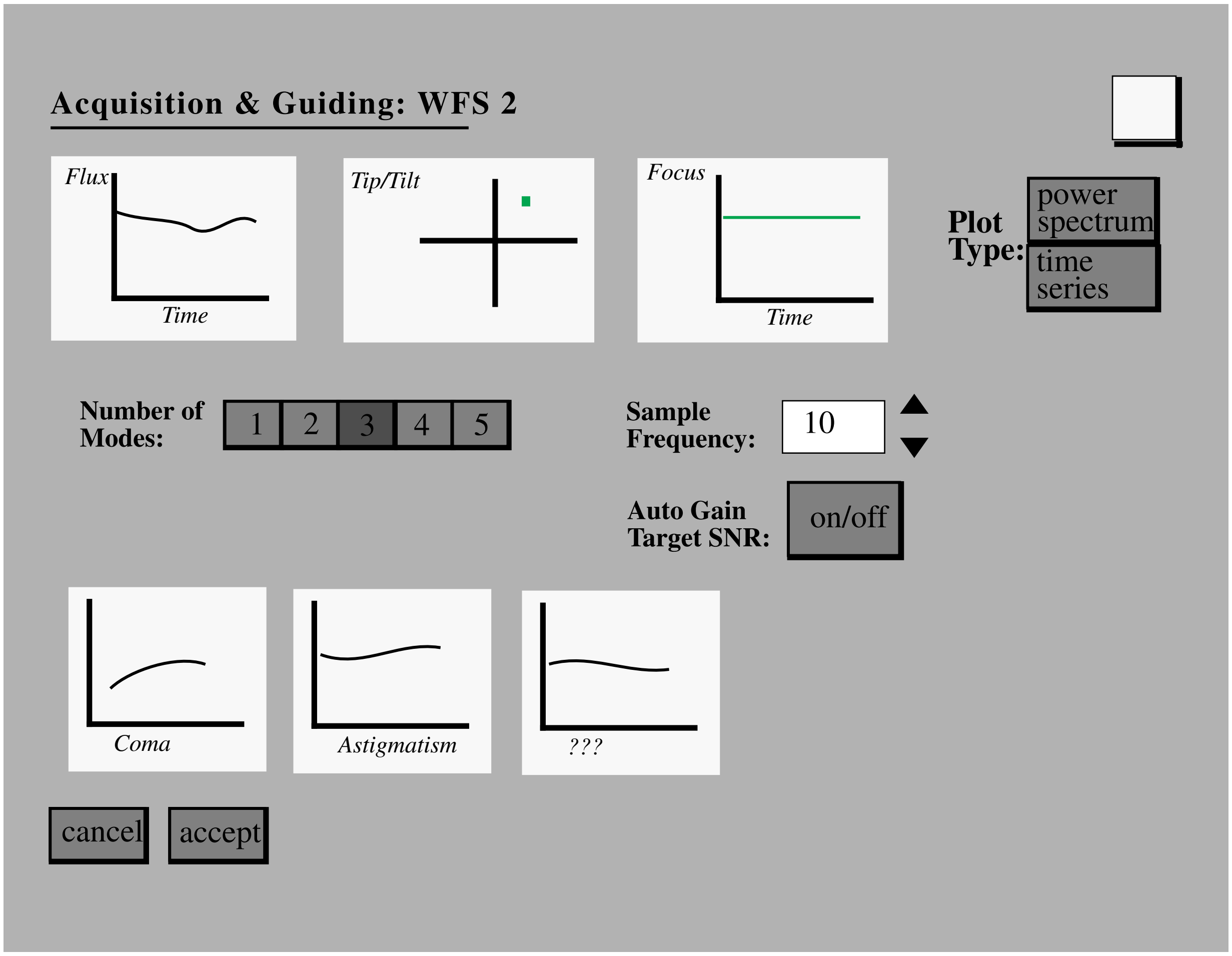
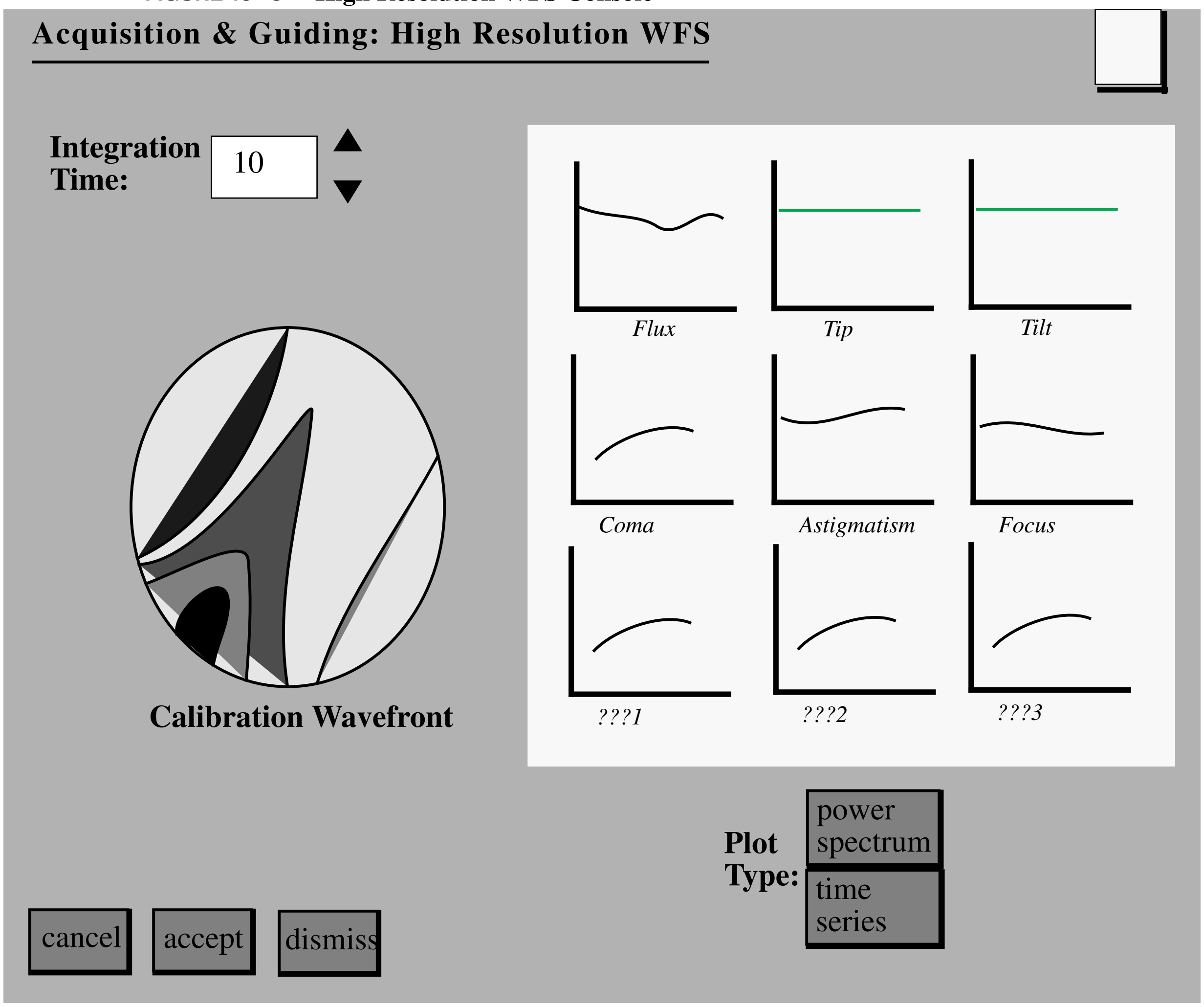


FIGURE 19 - 5 **High Resolution WFS Console**



19.5.3 Interface to OCS Screen Systems

19.5.3.1 ENCODER SCREEN

TBD

19.5.3.2 LIMIT SCREEN

TBD

19.5.3.3 ALARMS SCREEN

TBD

19.5.3.4 INTERLOCKS SCREEN

- Focal Station in Use.

  --Cassegrain

  --Fiber Optic to HIRES Lab

  --Steering Mirror to HIRES lab

  --TBD
- TBD

19.5.4 A&G Hardware Interface

19.5.4.1 A&G GUIDER

The A&G Control System provides for the following AGGS functions:

- XY Guider Select centroid algorithm
- XY Guider On/Off control
- XY Guider Position device
- XY Guider Set filter
- XY Guider Select shutter/dark slide
- XY Guider Take data (XY or image)
- XY Guider Set integration time
- XY Guider Select mode (continuous/single-shot)
- Fast Shack-Hartmann Sensor On/Off control
- Fast Shack-Hartmann Sensor Position device
- Fast Shack-Hartmann Sensor Select lenslet array
- Fast Shack-Hartmann Sensor Select filter
- Fast Shack-Hartmann Sensor Set integration time
- Fast Shack-Hartmann Sensor Select shutter/dark slide

- Fast Shack-Hartmann Sensor Take data (XY array or image)
- Fast Shack-Hartmann Sensor Select mode (continuous/single-shot)
- Wavefront Sensor On/Off control
- Wavefront Sensor Position device
- Wavefront Sensor Select filter
- Wavefront Sensor Select shutter/dark slide
- Wavefront Sensor Set integration time
- Wavefront Sensor Select output mode (Zernike coefficients or image)
- Wavefront Sensor Set Zernike count

19.5.4.2 A&G SCIENCE FOLD

The A&G Control System monitors and controls the science fold system. The following functions are required:

- Upper (AO and lower guide probe) Rotating Stage - position control
- Upper (AO) mirror - Deployed/Parked control
- Lower (Science Field and High Resolution Wavefront sensor) Rotating Stage - position control
- Lower (Science Field Fold) mirror - Deployed/Parked control
- Lower (Science Field Fold) mirror - Single-axis rotation control

19.5.5 A&G Acquisition Camera Subsystem Functionality

The A&G Control System provides for the following A&G Acquisition Camera Subsystem functions:

- Acquisition Camera On/off control
- Acquisition Camera Position device
- Acquisition Camera Set filter
- Acquisition Camera Select shutter/dark slide
- Acquisition Camera Set integration time
- Acquisition Camera Set framesize/binning

19.5.5.1 A&G HIGH RESOLUTION WFS

The A&G Control System provides for the following A&G High Resolution WFS Subsystem functions:

- Slow Shack-Hartmann Sensor On/Off control
- Slow Shack-Hartmann Sensor In/Out control
- Slow Shack-Hartmann Sensor Position device
- Slow Shack-Hartmann Sensor Select lenslet array
- Slow Shack-Hartmann Sensor Select filter
- Slow Shack-Hartmann Sensor Set integration time
- Slow Shack-Hartmann Sensor Select shutter/dark slide
- Slow Shack-Hartmann Sensor Take data (XY array or image)
- Slow Shack-Hartmann Sensor Select mode (continuous/single-shot)

19.5.5.2 A&G Commissioning and Quality Control

The A&G Control System provides for the following AGCQAS functions:

- TBD

19.5.5.3 A&G Baffle Sensing

The A&G Control System provides for the following A&G Baffle Sensing Subsystem functions:

- Baffle position sensor (fitted or not fitted)

19.5.5.4 A&G Field Curvature Correction

The A&G Control System provides for the following A&G Field Curvature Correction Subsystem functions:

- Field Curvature Corrector deployment In/Out
- Field Curvature Corrector position sensor

19.5.5.5 A&G Atmospheric Dispersion Compensation

If an AtmDC is incorporated, the A&G Control System provides for the following A&G Atmospheric Dispersion Compensation Subsystem functions:

- AtmDC Dispersion control

19.5.5.6 A&G Calibration

The A&G Control System provides for the following A&G Calibration Subsystem functions:

- Field Illumination On/Off control
- Wavelength calibration sources On/Off
- Flat field sources On/Off control
- Filters Positioning
- Polarization calibrators On/Off control
- High resolution wavefront sources On/Off control
- Optical reference beam On/Off control (required by the High Resolution Optical Spectrograph)

19.5.5.7 A&G OPTICAL POLARIZATION MODULATOR (TBD)

The A&G Control System provides for the following A&G Optical Polarization Modulator Subsystem functions:

- A&G Optical Polarization Modulator Select In/Out
- A&G Optical Polarization Modulator Set angular position

19.5.5.8 NOTES

TBD

19.5.5.9 TYPICAL USE OF EQUIPMENT

TBD

19.6 EXTERNAL BUS CONNECTIONS

19.6.1 Synchronization Bus

The A&G is connected to the synchro bus in order to pass wave front sensor information to the TCS and to the Secondary Control System.

19.6.2 Interlock System

The A&G must be capable of generating interlock requests as well as responding to them.

19.6.2.1 REQUIREMENTS

19.6.2.2 IMPLEMENTATION

19.6.3 Time Bus

The time within the A&G VME crate must be set to an external time reference.

19.6.3.1 REQUIREMENTS

19.6.3.2 IMPLEMENTATION

The A&G connects to the Gemini Time bus and uses the Gemini supplied EPICS implementation.

19.6.4 Event Bus

19.6.4.1 REQUIREMENTS

Not clear that A&G needs to be connected to the event bus.

19.6.4.2 IMPLEMENTATION

19.6.5 Control LAN

19.6.5.1 REQUIREMENTS

In order to send/receive values via EPICS channel access the A&G needs a TCP/IP connection.

19.6.5.2 IMPLEMENTATION

The A&G has a connection to the control LAN on the telescope.

19.7 A&G INTERNAL SUBSYSTEMS

19.7.1 A&G Guider Subsystem

By ‘guiding’ we mean not only the rapid tracking of image motion, but also of defocus; this system then includes the ‘fast active optics’ sensors as well as the XY guiders. The guiding subsystem must be able to sense

Zernike radial order 1 terms: (tip-tilt) at rates currently estimated at 200 Hz (dominated by wind-shake)

Zernike radial order 2 terms: (defocus) at about 20 Hz, (mostly wind driven); (astigmatism) at a TBD rate.

Zernike radial order 3 terms: (coma) at 5-200 seconds..

It is possible that the guiders may be required to operate in a part of the field which is vignetted in a complex way. This implies directly a loss of sensitivity of the guider. The guiding algorithms must be adjustable to find the optimum closed-loop bandwidth for the available guide stars and the conditions on the night.

19.7.2 A&G Science Fold

The science fold system directs the telescope beam into the various science instruments and into and out of the adaptive optics unit as required. This service is provided by a system of three rotating stages on which a number of probes and mirrors are mounted. These are the upper peripheral stage, the lower peripheral stage, and the science field stage that are each capable of being rotated to an arbitrary position. In addition, the lower peripheral stage and the science field stage have four index positions that correspond to the locations of the four pairs of ports for Adaptive Optics and Science field feed ports.

The upper peripheral stage carries a single guide probe and is not involved with beam redirection.

The lower peripheral stage carries a single guide probe and the Adaptive Optics feed mirror. The AO feed mirror is capable of being placed in either a deployed or a parked position.

The science stage carries the Science Field fold mirror. This mirror is capable of being placed in either a deployed or a parked position and can also be rotated about a single axis for selection of input beam source: direct, adaptive optics output, or calibration source.

The single High Resolution Wavefront sensor is mounted in the bottom of the instrument support structure. The wavefront sensor may be placed on a fixed position within the instrument rotator and not on any stage.

19.7.3 A&G Acquisition Camera Subsystem

The goal of the A&G Acquisition Camera Subsystem is to enable the acquisition of invisible objects onto 0.1 arcsec slits. This is the K-band spectroscopy case. See the functional specifications for more detailed goals. The acquisition camera should have the following approximate parameters (CCD or IR array assumed): 0.2 arcsec pixels, field of view 2 arcmin, readout rate 2 frames/second. It should be fed through re-imaging optics with an internal pupil for baffling. The coordinate system defined by the camera must remain in registration with coordinate systems fixed with respect to the instrument's entrance apertures, and with the coordinate system in which the location of the guide probe is known, to within 0.1 arcsec.

19.7.4 A&G Slow Active Optics Subsystem

To assist in the setup and checking of the primary mirror's active support as well as a detailed checking of collimation there shall be a slow active optics system. This shall be a high-order wavefront sensor (for example, a Shack-Hartmann with 20 x 20 sub-apertures) capable of deployment both on-axis and out to the maximum possible field radius. It should be capable of readout rates ranging from 'staring' modes to around 20 frames/sec, and would ideally deliver low-order aberrations at high temporal rates with little degradation compared with a sensor optimized for detection of such nodes.

19.7.5 A&G Commissioning and Quality Control Subsystem

To commission the A&G and to provide a check of its performance an A&G Commissioning and Quality Control Subsystem is required. It is necessary to deploy alignment telescopes and targets with the A&G area without interfering with any A&G functions. There are required simultaneous on- and off-axis guiding signals (up to Zernike radial order 3) with an on-axis sampling rate of at least 200 Hz and a off-axis rate of 20 Hz.

The A&G Commissioning and Quality Control Subsystem also requires a fixed and convenient location in the focal plane to act as a fiducial for pointing calibration and tests. Fixed, in this context, means at a known and stable offset from the intersection of the focal plane and the rotator axis.

19.7.6 A&G Baffle Sensing Subsystem

The telescope needs to be baffled when used at optical or ultraviolet wavelengths. Two baffles are used — a large baffle fitted around the hole in the primary mirror, and smaller baffle attached to the secondary mirror. The smaller baffle is deployed by the Secondary Control System (See "Baffle Control System" on page 18 - 22). The larger baffle is fitted and removed by the day crew and is not deployable. The A&G unit needs to be able to sense the presence or absence of this larger baffle.

19.7.7 A&G Field Curvature Correction Subsystem

When uncorrected, the telescope optics will not bring the whole field to focus in a perfectly flat focal plan. If a flat detector is placed at this curved focal surface distortions may be introduced at the edges of the field. The distortion is negligible for narrow field instruments but can be significant at wide fields. The Field Curvature Corrector is used to correct the distortion for wide field instruments. It should be deployable, because its presence would reduce the efficiency of narrow field observations. The FCC must be large enough to correct a field of view which includes the guiders and the science field.

19.7.8 A&G Atmospheric Dispersion Compensation Subsystem

Atmospheric dispersion is the variation in atmospheric refraction with color. This needs to corrected for at optical wavelengths. The correction is essential for instruments (such as the multi-object spectrograph) where accurate acquisition is critical over the whole field. Ideally it should correct dispersion to a small fraction of the best seeing, at a zenith distance up to 70 degrees. The AtmDC must be large enough to correct a field of view which includes the guiders and the science field.

Note that the FCC and AtmDC may be combined into one corrector, in which case the A&G Field Curvature Correction Subsystem and A&G Atmospheric Dispersion Compensation Subsystem would be combined into one subsystem. This is TBD.

19.7.9 A&G Calibration Subsystem

Provision of high-quality calibration can be expensive and often instruments overlap in their needs. We assume here that there is a case for provision of calibration as a general facility. The following calibration sources may be required:

- Uniform illumination of the largest science field to be used
- Wavelength calibration sources
- Flat field sources
- Filtration, order-sorting, and neutral density filters

- Polarization calibrators
- Calibration wavefronts for all wavefront sensors
- An optical reference beam to allow high-precision instruments (e.g. HROS) to correct for flexure.

19.7.10 A&G Optical Polarization Modulator Subsystem (TBD)

The study of polarization has not yet received much prominence within Gemini project reports, but is of obvious scientific importance. The existence of the A&G Optical Polarization Modulator Subsystem is, at this point, only a recommendation, not a requirement.

If present, this will exist only on the upward looking port.

20 Details of the Adaptive Optics Control System

20.1 Function of the Adaptive Optics System

The Gemini adaptive optics (AO) system makes fast corrections to the wavefront by utilizing a small deformable mirror. As a telescope device the AO system has a guide probe with a wave front sensor that is used to drive the deformable mirror and an internal tip/tilt optic.

The goal of the Gemini Adaptive Optics System (GAOS) is to deliver Strehl ratios greater than 0.5 at 1.6 micron wavelengths in median seeing conditions, and Strehl ratios greater than 0.2 at 0.7 micron wavelengths in the best 10% of conditions using only natural guide stars. Details of the specification of the GAOS are contained in *GAOS-1-FUNCT, "The Gemini Adaptive Optics System Science and Functional Specifications"* by Glen Herriot, Bob McClure and René Racine, [19].

20.2 Adaptive Optics Control Overview

20.2.1 External Interfaces

The Gemini AO System (GAOS) has the following external interfaces:

- Telescope Control System
- The AO engineering console
- The Instrument Control System
- The Data Handling System

- The AO mechanisms:
  --Optics
  --Deformable Mirror (DM)
  --Facilities Wavefront Sensor
  --Tip/Tilt Mirror Mount and Servo Control
  --Atmospheric Dispersion Compensator (AtmDC)

20.2.2 Other Systems Affected

- Observatory Control System (needs to provide an AO console and alarm screen)

20.2.3 External Bus Connections

The AO system is connected to the following external buses:

- Synchronization bus
- Interlock system
- Time Bus
- Event Bus
- Control LAN

20.2.4 Internal Subsystems

The AO system has the following major internal subsystems:

- The Optical Table subsystem, consisting of
  --Optics
  --Deformable Mirror (DM)
  --Facilities Wavefront Sensor
  --Tip/Tilt Mirror Mount and Servo Control
  --Atmospheric Dispersion Compensator (AtmDC)
- The AO Control Hardware and Software consisting of
  --GAOS control computer
  --Optical Table Control System
  --Reconstructor Unit

20.2.5 Internal Interfaces

The AO system has the following major internal interfaces:

•

20.2.6 Internal Data Stores

The AO system has the following major internal data stores:

•

20.2.7 Computer Hardware

The adaptive optics system is itself complex enough to warrant status as a scientific instrument. At the same time it interacts with the telescope sufficiently to be considered a telescope subsystem.

The AO computer system is housed in a VME chassis running the EPICS control system. It may prove that the computational speeds required are such that EPICS proves to be too large a performance drain on the system. In this case the high performance parts of the system are run on dedicated VxWorks CPU's and interface to an EPICS subsystem via the VME backplane.

In order for the TCS system to function in the presence of an adaptive optics system it is imperative that the AO system be interfaced to the reflective memory bus in order to inject wavefront information into the system.

A VME crate containing all the interfaces needed by the AO system is mounted on or near the instrument support structure at the cassegrain focus of the telescope.

The VME crate is contained in an enclosure with a heat extraction system. This enclosure is based on the standard developed by the Gemini Instrument Group.

20.2.8 Software Philosophy

The AO system is a completely distinct software entity similar to an instrument. The TCS only has a limited interface to this system as we expect that it is sufficiently complex that a separate operations console is required initially. Once the modes of operation have been established then it is possible for a more automated form of operation to be used.

20.2.9 Special Considerations

20.2.9.1 IMPACT ON ADAPTIVE OPTICS OF CHOPPING AND NODDING

In order to flat field data in the thermal infrared it is necessary to

- chop the secondary between two positions on the sky at up to 10 Hz
- nod the telescope between two positions on the sky at up to 0.1 Hz

20.2.9.2 IMPACT ON M2 OF AO TIP/TILT MIRROR

If the AO system's tip/tilt optic is damaged or turned off then it must be possible to use the tip/tilt secondary instead.

In this design there is nothing to stop the TCS from instructing the AO system to only remove focus and higher order turns and to instruct the secondary to remove tip and tilt.

20.2.9.3 REMOVAL OF DC TERMS

There are DC terms built up in tip/tilt, focus, and higher order terms. The DC tip/tilt and focus can be low pass filtered and passed to the secondary to remove. The DC higher order terms can be passed to the primary and removed.

20.3 INTERFACES

20.3.1 Interface to Telescope Control System

20.3.1.1 REQUIREMENTS

The AO control system accepts Zernike parameters {Zq,Z2,...,ZN} and positions parameters {X,Y,X',Y'} for the target wavefront and the target position and velocity of the AO guide probe. These parameters may be sent once or they may be continuously delivered at up to 20 Hz.

Instead of the above the AO system might be sent a vector or arrays of target deformable mirror target actuator positions {P1,P2,…,PN}.

In addition to the targets the AO accepts the commands:

- AO Lock, start AO system

- AO Config, configure AO system
- STOP; halt and hold the current positions
- MOVE; go to the current target and stop
- FOLLOW; go to the current target and follow the target continuously
- STOP FOLLOWING; stop where you are
- PARK; move to a predetermined position and put AO in park state
- ENCODER CONFIG; configure the encoders
- ZERO SET; zero different parts of the system

The AO system returns the following values:

- Number of Zernike coefficients (usually 3).
- $\{CZ_1,CZ_2,...,CZ_N\}$, Zernike coefficients of cumulative wavefront (usually tip, tilt and focus, which are C2, C3 and C4).
- time stamp for Zernike measurements (to allow “best before” processing)
- data quality measure of Zernike coefficients
- Medium transmission
- AO error statistics
- FWHM of guide star
- Relative transmission
- magnitude of guide star
- Encoder readings
- Drive demand
- Servo errors
- Alarms
- Status (e.g. “in position”)

20.3.1.2 IMPLEMENTATION

The TCS uses EPICS Channel Access to set the appropriate parameters in the AO system’s database and then issues an event that causes the AO system to match the system with those parameters. See ICD/7a.

The cumulative number of Zernike coefficients, the Zernike coefficients the time stamp, and data quality are stored in EPICS database values that are connected to the reflective memory bus.

20.3.2 Interface to the AO engineering console

20.3.2.1 REQUIREMENT

The engineering console provides complete control of all the AO system components. In addition the console must be capable of displaying a large number of diagnostics.

The following commands are available:

-

The following diagnostics are available:

- The detector photon counts (requires temporal filter).
- The photon counts integrated over all the subapertures (with a temporal filter).
- The wavefront sensor measurements (requires temporal filter).
- Mirror mode coefficients (requires temporal filter).
- Mirror voltages (requires temporal filter).
- Fried seeing parameter and atmospheric coherence time approximation (requires temporal filter).
- Residual wavefront phase variance and Strehl ratio (requires temporal filter).
- Scope-type display of mirror voltages against time.
- Scope-type display of WFS measurements against time.
- Scope-type display of WFS optical gain against time.
- Mean and variance of the WFS measurements.
- Interaction matrix eigenmodes.
- Power spectrum.
- Power spectrum of the electrode voltages.
- Power spectrum of the mirror mode coefficients.
- Transfer function frequency response of an electrode or of a correction mode.
- Step response of an electrode or of a correction mode.

The following statistical analysis functions and displays may also be available:

- Spatio-temporal cross correlation of the wavefront sensor measurements averaged in time.
- Spatio-temporal cross correlation of the electrode voltages averaged in time.
- The Point Spread Function (PSF)

- The time auto-correlation of the wavefront sensor measurements.
- The time auto-correlation of the electrode voltages.
- The time auto-correlation of the mirror mode coefficients.

20.3.3 Interface to the Instrument Control System

20.3.3.1 Requirement

The GAOS should be able to accept wavefront information from a wavefront sensor installed on board an instrument, so that light from an object close to the science object, or from the science object itself may be used as a reference.

The Instrument Control System and GAOS exchange wavefront information over the Synchronization bus using the mechanism described in ICD/5.

20.3.4 Interface to the Data Handling System

20.3.4.1 Requirement

The GAOS should be able to log errors and messages to the Data Handling System using the mechanism described in ICD/4.

The following information will be logged:

- Error messages
- Status messages
-

20.3.5 Interface to the AO mechanisms

20.3.5.1 Optics

20.3.5.2 Deformable Mirror (DM)

20.3.5.3 Facilities Wavefront Sensor

20.3.5.4 Tip/Tilt Mirror Mount and Servo Control

20.3.5.5 Atmospheric Dispersion Compensator (AtmDC)

20.4 Requirements on Other Systems

20.4.1 The AO System Console in the OCS

The OCS provides consoles to run the AO system during normal operations.

20.4.1.1 Requirement

The consoles provided must have fine grained control of the AO system. In addition the consoles must be capable of displaying a large number of TBD diagnostics.

The following commands are available:

- Start/Stop the correction process.
- Start/Stop various other functions TBD.
- Enable/Disable: full correction; partial correction; tip/tilt only correction; modal optimization; wavefront sensor gain optimization; PSF calculation.
- Commands to control the status display.
- Set the reference measurement vector, m_0.
- Set the wavefront sensor integration time.
- Set the wavefront sensor optical gain.
- Set the low frequency (<200 Hz) secondary mirror defocus update rate.
- Set the real-time filter time constants.

- Store the contents of the wavefront information buffer to a file on a local disk.
- Enter a control matrix, *D'*.
- Set the mirror mode gains, G_m .
- Set the mirror element gain, G_e .
- Set the automated more optimization enable/disable flags, G'_m .
- Set the wavelength to be used for the Fried seeing parameter, Strehl ratio and phase variance calculations.
- Set the Strehl ratio threshold value TBD.
- Set the conversion factors between internal units, arcseconds and microns.

I think much of this should be on the engineering screen, not the operator's screen. — Steven.

The following status information is available:

- The status of the close-loop (ON/OFF).
- The status of the modal optimization (ON/OFF).
- The status of the wavefront sensor gain optimization (ON/OFF).
- The status of the PSF calculation (ON/OFF).
- The offset voltages applied to the mirror electrodes when not in closed loop operation.
- The computed Point Spread Function.
- The average Fried seeing parameter, Strehl ratio and phase variance during the previous astronomical exposure.
- The atmospheric coherence time.
- Photon counts averaged over all sensors.
- The current integration time on the wavefront sensor.
- The current wavefront sensor optical gain.
- The reference measurement vector, m_0 .
- The results provided by the analysis facilities.
- The average detected photon counts (1 second average minimum).
- The control matrix, *D'*.
- The interaction matrix, *D*.

The following diagnostics are available:

- A set of individual or averages wavefront sensor measurements.
- The last telescope tracking and defocus commands sent to the secondary mirror.
- The update rate for the low frequency telescope tracking correction commands to the secondary mirror.
- The update rate for the low frequency defocus information provided for the TCS.

20.4.1.2 IMPLEMENTATION

The OCS communicates these commands to the TCS using the mechanism described in ICD/1. The TCS uses EPICS Channel Access to set the appropriate parameters in the AO system's database and then issues an event that causes the AO system to match the system with those parameters, as described in ICD/7a.

20.4.1.3 SAMPLE CONSOLES

The consoles shown below are samples of what the OCS provides for commissioning and interactive use of the AO system.

[use subset of UH IfA (BuzzGraves et al) console screens]

20.4.2 The OCS Alarms Screen

This will use the EPICS alarm manager to display alarms from the GAOS, as described in ICD/2.

20.4.3 Interface to the OCS Screens System

20.4.3.1 ENCODER SCREEN

20.4.3.2 LIMIT SCREEN

20.4.3.3 ALARMS SCREEN

20.4.3.4 INTERLOCKS SCREEN

20.5 EXTERNAL BUS CONNECTIONS

20.5.1 Synchronization Bus

The AO system is connected to the synchronization bus in order to make wave front sensor information available to systems that need to use it.

It is not at this time clear that the synchro bus is fast enough for the AO WFS's that are not colocated with the AO system. The problem is that the WFS's *on board* scientific instruments may need to switch between supplying information for fast steering and primary mirror surface (taking the place of the A&G *facility* WFS's) to supplying information to the Adaptive Optics module.

The best solution at present would be to have the *on board* WFS's connected to the AO VME crate directly via high speed VME modules. This VME crate would have a connection to the Synchro bus which would be used to supply WFS information to the system. The use of the WFS by the AO system would bypass the synchro bus. This solution has a number of disadvantages (most noticably that we would tie up the AO system when we were using the *on board* WFS's for A&G functions. This could be fixed, however, by having the AO supply such information to the TCS on a slower time scale.

The "solution of choice" for this is to swap the cable for the on-board WFS between the AO and A&G when required.

20.5.2 Interlock system

The AO system is connected to the interlock system in order to make and respond to interlock requests.

20.5.3 Time Bus

The AO system is connected to the time bus in order to maintain the system time of the VME system used.

20.5.4 Event Bus

The AO system is connected to the event bus in order to monitor the current position of the secondary in its chop cycle and the telescope in its node cycle.

It may also need to monitor whether the instrument has its shutter open, depending on how critical accurate synchronization with an instrument is.

20.5.5 Control LAN

The AO system is connected to the control LAN in order to support EPICS channel access.

20.6 ADAPTIVE OPTICS INTERNAL SUBSYSTEMS

20.6.1 The Optical Table Controller

This is responsible for

- The input shutter
- The X-Y-Z stages of the facilities wavefront sensor.
- The X-Y-Z stages of the artificial star and test target.
- The Atmospheric Dispersion Compensator insertion and rotation.
-

20.6.2 The Facility Curvature Wavefront Sensor Controller

This is responsible for

- The internal baffles required for observing in a bright background.
- The detector and its control electronics.
- The optical gain adjustment TBD.
- The field and pupil observing system TBD.
-

20.6.3 On-instrument Curvature Wavefront Sensor Controller

This is responsible for

- The internal baffles required for observing in a bright background.
- The detector and its control electronics.
- The optical gain adjustment TBD.

N.B. This is probably the responsibility of the Instrument Control System rather than the AO. Should the ICS provide a WFS controller server for the AO, just as the DHS provides a quick look server for the ICS? — Steven.

20.6.4 The Tip/Tilt Mirror Controller

This is responsible for:

- The tip/tilt mirror cell.
- The tip/tilt mirror cell gimbal mount.
- The tip/tilt mirror mount linear drive actuators.
- The tip/tilt mirror servo electronics and power supplies.

What are the options? Discuss this with AO people (e.g. Glenn). — Rick.

20.6.5 The Deformable Mirror Controller

This is responsible for:

- The bimorph mirror mount and cell.
- The adaptive mirror drive amplifiers.

20.6.6 The Adaptive Control Subsystem

The is responsible for

- The reconstructor system

21 Details of the Enclosure Control System

21.1 Function of the Enclosure System

In order to satisfy the Gemini Project Science Requirements it is necessary to provide for control of the telescope enclosure and its various subsystems. Besides providing the obvious role of protection from adverse environmental conditions, the enclosure carousel[1] subsystems are also used to improve both the throughput and dynamic image quality error budgets.

21.2 Enclosure Control Overview

21.2.1 External Interfaces

The Enclosure Control System (ECS) has the following external interfaces:

- Telescope Control System
- the ECS Control Console(s) provided by the Observatory Control System
- the Observatory Control Screens system
- the Contractor-supplied carousel control system
- the Contractor-supplied support facility control system

1. The "carousel" consists of the upper, rotatable parts of the enclosure.

21.2.2 External Bus Connections

The ECS is connected to the following external buses:

- Interlock System
- Time Bus
- Control LAN

21.2.3 Internal Subsystems

The Enclosure Control System is composed of the following major subsystems provided as three distinct groups:

Enclosure

- Carousel Rotation Subsystem
- Shutter Subsystem
- Air Flow Management Subsystem
- Carousel Shell Venting Subsystem
- Overhead Crane
- Platform Lift

Support Facility

- Thermal Regulation Subsystem
- Air Conditioning Subsystem

Interlocks

- Interlock Subsystem

21.2.4 Internal Interfaces

The ECS has the following major internal interfaces:

- Interaction between EPICS database and EPICS sequencer ‘state machine’ program(s).
- RS-232-C connection between VMEBus serial card and an Allen-Bradley 1785-KE module which enables communication with an Allen-Bradley PLC-5/25 based Carousel Control System.

- RS-232-C connection between VMEBus serial card and an Allen-Bradley 1785-KE module which enables communication with an Allen-Bradley PLC-5/25 based Support Facility Control System.

21.2.5 Internal Data Stores

The ECS has the following major internal data stores:

- EPICS database.
- Allen-Bradley PLC/5 programs.

PLC-5 programs are developed on a PC-based system which will download the resulting programs into the PLC-5 via the Allen-Bradley Data Highway/Data Highway Plus. This PC-based development system is standalone and has no connections to the Enclosure Control System.

21.2.6 Computer Hardware

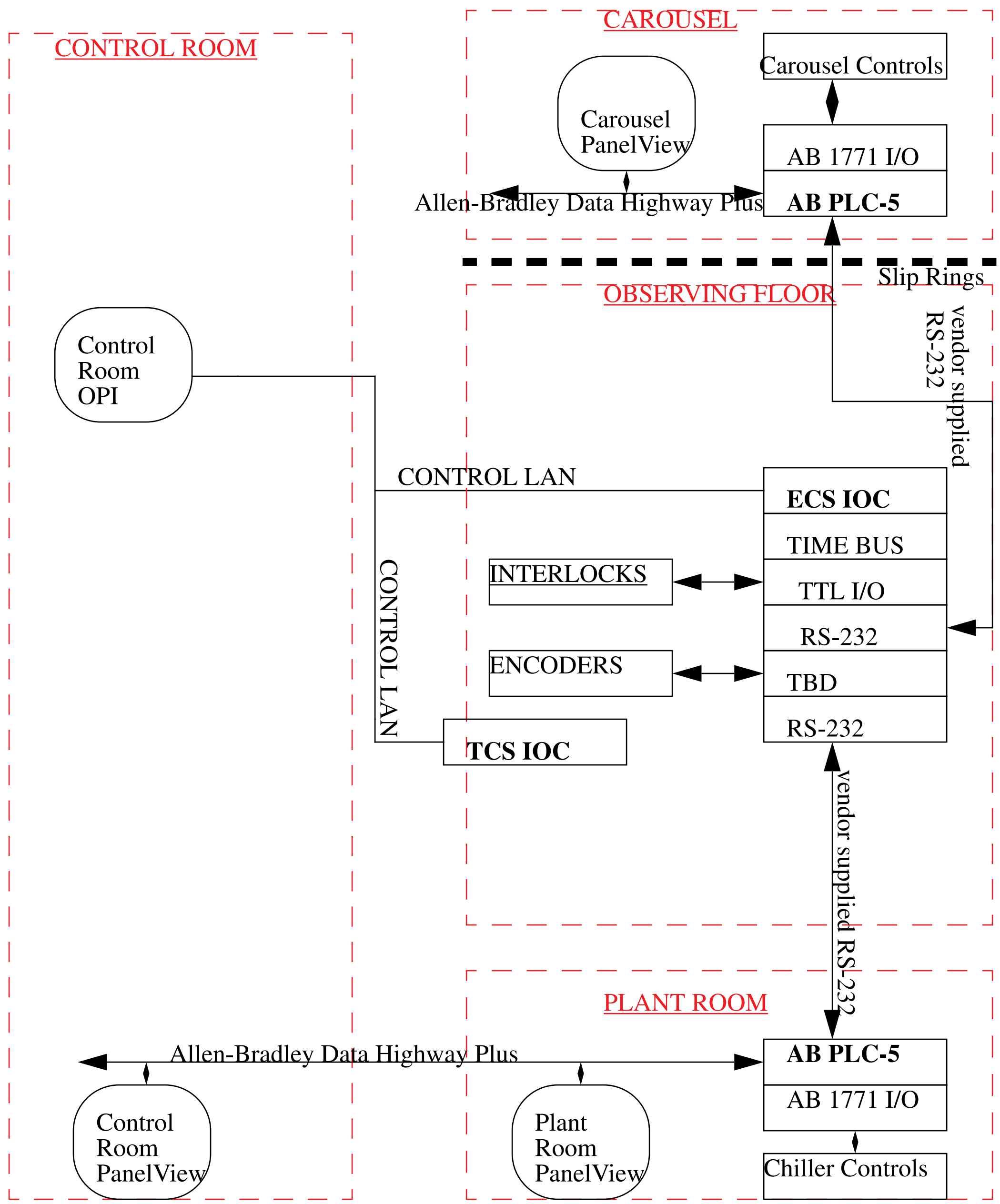
There is a VME crate dedicated to enclosure control on the observing floor. This VME crate may have multiple CPU's in order to handle all of functionality required. The VME crate control CPU is a M68040-based MVME-167.

The carousel and support facility contractors will each supply the complete communications path from their PLC systems to the Enclosure Control System IOC including RS-232-C lines and two connected Short-Haul Modems. The contractors' control system will present a RS-232-C interface to the IOC which will support the transmission of ASCII PLC-5 commands and data. The form of these commands are defined in the Allen-Bradley reference manuals *Data Highway/Data Highway Plus Protocol and Command Set* and *PLC-5 Programming Software.*

The contractors' systems will use a PLC-5/25 with the 1785-KE module providing the serial interface. It is the Allen-Bradley PLC/5 that interacts with the enclosure hardware via Allen-Bradley 1771 I/O modules. These I/O modules will all be 12-bit although 16-bit versions can be used if Gemini requires it.

An Allen-Bradley remote I/O link will extend from support facility PLC system to the Operations Room where it may be used to drive a Panel-View VDU. There will also be a Panel-View VDU located in the basement next to the PLC system.

FIGURE 21 - 1 Enclosure Control System Electronic Interfaces



21.3 INTERFACE TO TELESCOPE CONTROL SYSTEM

21.3.1 Requirements

The enclosure control system accepts azimuth parameters {A,A',A''} for the target position, velocity, and acceleration of the carousel. These parameters may be sent once or they may be continuously delivered at up to 20 Hz.

The enclosure also accepts the temperature profile of the enclosure, telescope, and primary mirror as well as the current wind speed and direction. These parameters may be sent once or they may be continuously delivered at up to 0.1 Hz

In addition to the targets the enclosure accepts the commands:

- STOP; halt and put the brakes on
- MOVE carousel; rotate carousel to the current target and stop
- DRIFT carousel; rotate carousel to the current target and drift with a specified constant velocity
- MOVE SHUTTERS; move upper and lower shutters to the current target and stop
- DRIFT SHUTTERS; move upper and lower shutters to the current target and drift with a specified constant velocity
- MOVE VENTILATION GATES; move ventilation gates to "best" position given environmental conditions
- FOLLOW; move enclosure, upper and lower shutters, and ventilation gates to the current target and follow the target continuously
- STOP FOLLOWING; stop where you are
- PARK; move to a predetermined position and put enclosure, upper and lower shutters, and ventilation gates in park state
- ENCODER CONFIG; configure the encoders
- ZERO SET; zero different parts of the system

The enclosure returns the following values:

- Encoder readings
- Drive demand
- Servo errors
- Alarms
- Status (e.g. "in position")

21.3.2 Implementation

The above data values are contained in an EPICS database that communicates with the TCS via channel access. See *ICD 1- The System Command Interface* for details.

21.4 SOFTWARE SYSTEM OVERVIEW

21.4.1 Computer Hardware

The ECS shall be implemented on a MVME-167 (M68040-based) VMEbus CPU which shall communicate via a RS-232 port with a commercial contractor supplied Allen-Bradley PLC-5 based local control system. This is true of both the support facility system (which includes chiller control) and the carousel system even though they are provided by independent contractors.

The ECS software will be based on the EPICS control system toolkit and will have a main goal of:

- keep parts of the enclosure out of the telescope beam
- to keep the enclosure well vented
- to protect telescope structure from wind to minimize image smear

The effects that the enclosure can have on the science being done can be seen from its impact on the various parts of the error budget.

21.4.1.1 STATIC IMAGE QUALITY

Error Source	Subsystems Affected
Primary Mirror Thermal Distortion	Air flow management, Thermal regulation
Primary Mirror Wind Buffeting	Shutters, Air flow management, Thermal regulation
Secondary Mirror Thermal Distortions	Air flow management, Thermal regulation
Secondary Mirror Wind Buffeting	Shutters, Air flow management, Thermal regulation

21.4.1.2 SELF INDUCED SEEING

Error Source	Subsystems Affected
Enclosure	Shutters, Carousel Skin Venting, Air flow management
Telescope Primary Mirror Delta T	Air Conditioning, Air flow management, Thermal regulation
Telescope Secondary Mirror Delta T	Air Conditioning, Air flow management, Thermal regulation
Telescope OSS Delta T	Air Conditioning, Air flow management, Thermal regulation

21.4.1.3 DYNAMIC IMAGE QUALITY

The enclosure has no effect on the dynamic image quality error budget.

IMAGE SMEAR

Error Source	Subsystems Affected
Wind Shake	Shutters, Air flow management

21.4.1.4

21.4.1.5 THROUGHPUT

Error Source	Subsystems Affected
Enclosure	Carousel
Shutter	Shutters

21.4.1.6 EMISSIVITY

Error Source	Subsystems Affected
Enclosure	Carousel

21.4.2 Interface to the Enclosure Console

21.4.2.1 REQUIREMENT

The console provided by the OCS must have fine grained control capability of the enclosure.

The following commands are available:

- TBD

21.4.2.2 IMPLEMENTATION

The TCS uses EPICS Channel Access to set the appropriate parameters in the ECS's database and then issues an event that causes the ECS to match the system with those parameters.

21.4.2.3 SAMPLE CONSOLE

The following figure shows one possible design for the enclosure console that would be provided by the OCS.

FIGURE 21 - 2 **Enclosure console**

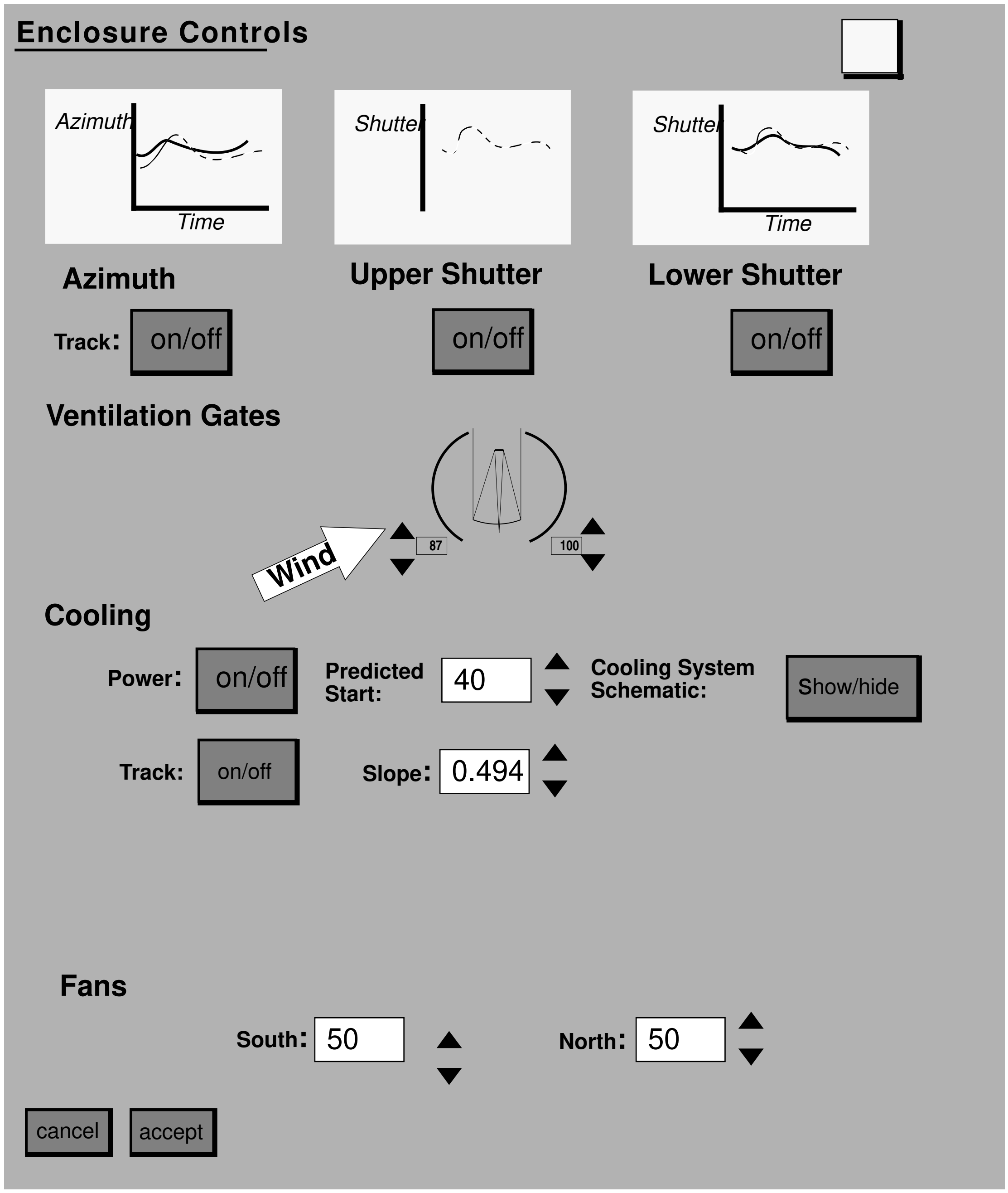
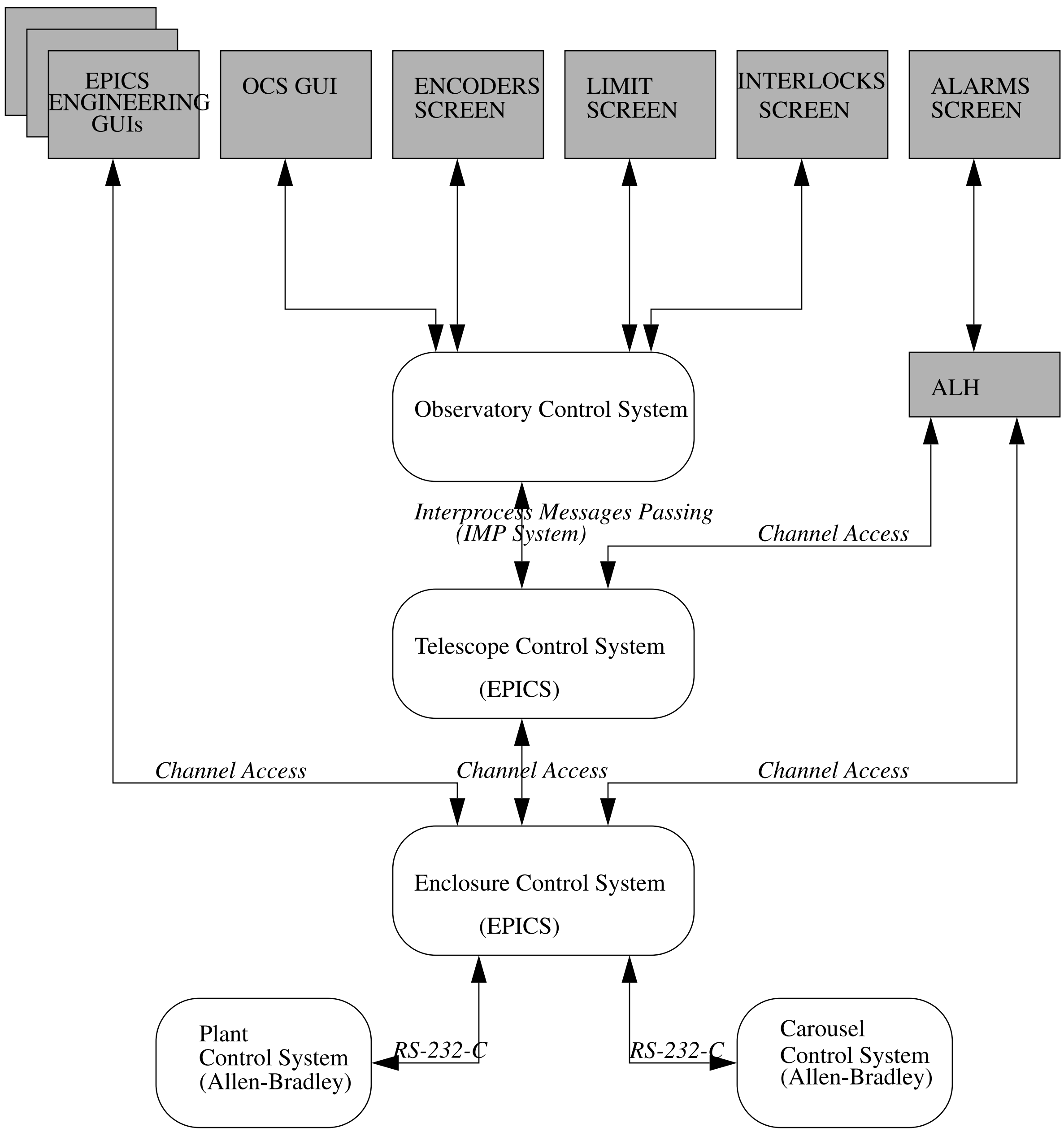


FIGURE 21 - 3 Enclosure Control System Software



21.5 INTERFACE TO SCREEN SYSTEMS

21.5.1 Encoder Screen

21.5.1.1 UPPER AND LOWER SHUTTERS

Tape encoders.

21.5.1.2 CAROUSEL ROTATION

Barcode encoder.

21.5.1.3 VENTILATION GATE POSITIONS

TBD encoders.

21.5.1.4 OVERHEAD CRANE POSITION

TBD encoder.

21.5.1.5 PLATFORM LIFT BALL SCREW POSITIONS

TBD encoders.

21.5.2 Limit Screen

21.5.2.1 UPPER AND LOWER SHUTTERS

Upper, lower, and collision limit switches.

21.5.2.2 VENTILATION GATES

Upper and lower limit switches.

21.5.2.3 AZIMUTH BOGIES

'Bottomed-out' limit switches.

21.5.3 Alarms Screen

21.5.3.1 CAROUSEL EMERGENCY STOP.

Normal state CLEAR.

21.5.3.2 OVERTRAVEL ALARMS.

Normally CLEAR; alarm state SET.

--Upper and Lower Shutters.

--Ventilation Gates

21.5.3.3 TORQUE TRIP ALARMS.

Shut down the drive until reset on the gallery control panel. Normally CLEAR; alarm state TRIPPED.

--Upper and Lower Shutters

--Ventilation Gates

21.5.4 Interlocks Screen

21.5.4.1 DRIVES

--Carousel (normally COMPUTER)

--Shutters (normally ENGINEER)

21.5.4.2 CONTROL LOCATIONS.

These show the state of the remote/local/off key switches on the gallery control panel. REMOTE means that the mechanism can be driven from the control room; LOCAL that it must be driven from Allen-Bradley Panel-View VDUs.

--Carousel (normally REMOTE)

--Upper and Lower Shutters (normally REMOTE)

21.5.4.3 POWER.

Normally ON; alarm state OFF.

--Carousel

--Upper and Lower Shutter

--Ventilation Gates

21.5.4.4 SHUTTERS

OPEN/CLOSED.

21.5.4.5 CAROUSEL INTERLOCK.

Normally DISABLED; alarm state ENABLED.

21.6 INTERFACE TO ENCLOSURE SUBSYSTEMS

The functional requirements of each of the enclosure subsystems are outlined here.

TABLE 21 - 1 General Functions (for all Enclosure mechanisms)

Function	Description	Range	Frequency	Type
Power	turn power on/off to entire unit	ON | OFF	< 0.1 Hz	digital switch
Health	monitor overall status of device	OK | BAD	1 Hz	digital i/p
Servo Parms	up/down load servo parameters	TBD	< 0.1 Hz	digital i/o
HeartBeat	incrementing counter	32 bits	20-50Hz	digital i/p
WatchDog	shutdown Enclosure systems if host dies		20-50Hz	digital i/o

TABLE 21 - 2 Carousel Rotation Functions

Function	Description	Range	Frequency	Type
Velocity	Control carousel rotational velocity	TBD	<20Hz	Allen-Bradley
Position	Monitor carousel absolute encoder	TBD	<20Hz	digital i/p
Set Brakes	Control carousel brakes.	ON|OFF	<1Hz	Allen-Bradley
Brake Status	Monitor carousel brake status	ON|OFF	<1Hz	Allen-Bradley
Zero degree reference	Monitor status of zero degree reference system	TBD	TBD	Allen-Bradley
Bogie Status (31)	Monitoring bottoming-out of bogie under heavy loading via microswitch.	OK | BAD	<1Hz	Allen-Bradley
Bogie Loading (31) [TBD]	Monitor bogie load via loadcell.	TBD	<1Hz	Allen-Bradley

TABLE 21 - 3 Upper and Lower Shutter Functions

Function	Description	Range	Frequency	Type
Velocity (2)	Control shutter velocity	OFF| SLOW| MEDIUM| HIGH	<20Hz	Allen-Bradley
Limits (3x3)	Monitor shutter limit switches: {upper shutter upper limit, lower shutter lower limit, both shutters collision limit} X {reduce to medium speed, reduce to slow speed, stop}	ON|OFF	<20Hz	Allen-Bradley
Position (2)	Monitor shutter tape encoder	TBD	<20Hz	Allen-Bradley
Velocity Readback (2)	Monitor DC tachometer	TBD	<20Hz	Allen-Bradley

TABLE 21 - 4 Air Flow Management Functions

Function	Description	Range	Frequency	Type
Ventilation Gates (2)	Control pairs of upper and lower ventilation gates.	0-100 % open	<1Hz	Allen-Bradley
Ventilation Gates Inner Screens (TBD)	Control screen position. Not provided Day One.	TBD	<1Hz	Allen-Bradley

TABLE 21 - 5 Enclosure Shell Venting Functions

Function	Description	Range	Frequency	Type
Constant Speed Fans	Control power.	ON| OFF	<1Hz	Allen-Bradley
Baffles	Control roof vent baffles.	ON| OFF	<1Hz	Allen-Bradley

TABLE 21 - 6 **Overhead Crane Functions**

Function	Description	Range	Frequency	Type
Cable Position	Readback cable drum encoding mechanisms.	TBD	<1Hz	Allen-Bradley
Parking Block	Control actuators.	TBD	<1Hz	Allen-Bradley

TABLE 21 - 7 **Platform Lift Functions**

Function	Description	Range	Frequency	Type
Ball Screws (4)	Readback encoders.	TBD	<1Hz	Allen-Bradley
Ball Screw Interlock	Status of passive interlock triggered by differential encoder values (tip/tilt of platform lift).	TBD	<1Hz	Allen-Bradley

TABLE 21 - 8 **Thermal Regulation Functions**

Function	Description	Range	Frequency	Type
Air Exhaust Fans (4)	Control fan power.	ON|OFF	<1Hz	Allen-Bradley
Air Flow Rate Sensor (2)	Monitor air flow rate.	TBD	<1Hz	Allen-Bradley
Air Exhaust Duct (2)	Control air exhaust duct opening and closing.	OPEN|CLOSE	<1Hz	Allen-Bradley
Motorized Damper (2)	Control damper opening and closing.	OPEN|CLOSE	<1Hz	Allen-Bradley
Motorized Damper Status (2)	Monitor damper open/close status.	OPEN|CLOSE	<1Hz	Allen-Bradley
Head Pressure Monitor (2)	Monitor chiller head pressure.	TBD	<1Hz	Allen-Bradley

TABLE 21 - 9 Air Conditioning Functions

Function	Description	Range	Frequency	Type
Air Conditioning Unit Power	Control power.	ON| OFF	<1Hz	Allen-Bradley
Air Conditioning Unit Setting	Control temperature set point.	TBD	<1Hz	Allen-Bradley

21.7 ECS INTERNAL SUBSYSTEMS

21.7.1 Carousel Rotation

The carousel rotates in order to allow an unobstructed path for the telescope beam while tracking a celestial object. The rotation controller shall be capable of synchronized movement with the telescope mount, independent movement, as well as accepting commands from a manual control.

21.7.1.1 ROTATIONAL POSITIONING

The carousel will be driven by a system of 4 azimuth drive bogies. The motors will be operated by 4 motor controllers operating in a master/slave relationship to provide synchronous motion. There shall be a single master controller driving the other 3 slave controllers. In the advent of a failure of the master controller, one of the slave controllers shall become the new master.

The maximum slew and track velocities of the carousel shall match those of the telescope mount's azimuth motion. Reference *Telescope Maximum Velocity,Accel.,Min. Decel. Requirements* (ICS-G-0005). The enclosure slew rate is 3.0 degrees per second. The maximum tracking rate is 2.0 degrees per second.

There is provided a system that indicates when the enclosure is at the zero degree azimuth reference point.

The enclosure's electrical and control signals are passed on a system of slip rings so there are no inherent limits to the enclosure rotation. The reference direction (zero azimuth angle) is due North with the positive rotation vector pointing toward nadir (opposite to zenith) so the azimuth angle increases in the clockwise direction.

The carousel's positional encoders are supplemented by DC tachometers. The carousel servo system will be designed to operate in the presence of wind torque on the structure.

A 'soft-start' shall be provided as part of the motor control to take out jerk in the drive system. To minimize power dissipation while tracking, the carousel will move in discrete steps and the power will be turned off while stopped. As long as the telescope is unvignetted by the carousel, reduced velocities and accelerations will be used to minimize wear on the drive mechanisms. The actual values will be determined during testing of the structure.

21.7.1.2 SYNCHRONIZATION MODES

The Enclosure Control System will provide the capability to automatically synchronize the carousel to the nominal telescope position and maintain a clear field-of-view during telescope tracking and pointing. The carousel position error when synchronized will be controlled by the software with the finest level of control being +/-3mm RMS when measured from the center of the enclosure opening to the nominal telescope optical axis.

The TCS shall decide whether to specify the actual pointing direction as the target to the enclosure or, in the case of rapid differentail movements, some 'mean' target position.

This synchronization will be provided in two different modes, minimum vibration and minimum scatter. In minimum vibration mode the carousel will be positioned so as to offer maximum viewing time before the carousel must be moved, at which time a synchronization signal will be made available indicating that movement will start in 5 seconds. The carousel will then move to the next "minimum vibration" position - at which time the synchronization signal will close.

Minimum scatter is intended to keep the relative telescope/carousel orientation fixed to +/-0.25 m in order to minimize the effect on the scattered light background. This is a science requirement.

21.7.1.3 BRAKES

In addition to the azimuth drive motors there will be provided on separate bogies electrically actuated fail-safe brakes to maintain the carousel position when not tracking. The brakes will be designed to prevent unwanted carousel motion in wind speeds up to the maximum observing conditions. The brakes will disengage when power is applied to the motors and will engage whenever the motors are stopped.

21.7.2 Upper and Lower Shutters

The enclosure has a shutter that acts as a weather, temperature and air seal around the opening through which the telescope beam passes. In windy conditions the upper and lower shutters can also be brought to a position adjacent, but outside,the telescope beam. The position sensor on the shutter shall not be adversely affected by daytime light levels. The proposed implementation of the position sensor is a tape encoder. The shutter controller shall be capable of synchronized movement with the telescope mount, independent movement, as well as accepting commands from a manual control.

21.7.2.1 POSITIONING

The maximum slew and track velocities of the shutters shall match those of the telescope mount's altitude motion. Reference *Telescope Maximum Velocity,Accel.,Min. Decel. Requirements* (ICS-G-0005).

The shutter's positional encoders is supplemented by DC tachometers. The shutter servo system will be designed to operate in the presence of a wind torque on the shutter structuresThe shutters operate at two speeds, one fast enough to fully open or close the shutters in less than 4 minutes and a slow rate of 3 degrees per minute. The shutters are also capable of incremental movement in 3 arcminute steps.

A 'soft-start' shall be provided as part of the motor control to take out jerk in the drive system. To minimize power dissipation while tracking, the shutters will move in discrete steps and the power will be turned off while stopped. As long as the telescope is unvignetted by the shutters, reduced velocities and accelerations will be used to minimize wear on the drive mechanisms. The actual values will be determined during testing of the structure.

21.7.2.2 SYNCHRONIZATION MODES

The Enclosure Control System will provide the capability to automatically synchronize the shutters to the telescope position and maintain a clear field-of-view during telescope tracking and pointing. The shutters position error when synchronized will be controlled by the software with the finest level of control being +/-3mm RMS when measured from the center of the enclosure opening to the telescope optical axis.

This synchronization will be provided in two different modes, minimum vibration and minimum scatter. In minimum vibration mode the shutters will be positioned so as to offer maximum viewing time before the shutters must be moved, at which time a synchronization signal will be made available indicating that movement will start in 5 seconds. The shutters will then move to the next “minimum vibration” position - at which time the synchronization signal will close.

Minimum scatter is intended to keep the relative telescope/shutters orientation fixed to +/-0.25 m in order to minimize the effect on the scattered light background. This is a science requirement.

21.7.2.3 BRAKES

The shutters drive motors will be provided with electrically actuated fail-safe brakes to maintain the shutters position when not tracking. The brakes will be designed to prevent unwanted shutters motion in wind speeds up to the maximum observing conditions. The brakes will disengage when power is applied to the motors and will engage whenever the motors are stopped.

21.7.3 Air Flow Management

The enclosure provides control over the flow of ambient air into the observing area. This shall be accomplished by a system of ventilation gates that can be opened and closed in response to the external wind conditions.

There are two ventilation gates, on either side of the shutter, extending for 120 degrees each and opening ‘clamshell’ fashion horizontally (i.e. top half of gate moves up, bottom half moves down) to any opening position from 0-100% open.

21.7.3.1 CAROUSEL SHELL VENTING

The enclosure carousel skin shall have an active venting system driven by a set of constant-rate fans to drive the enclosure skin temperature towards that of the ambient air. These fans are located in the North exhaust duct and are only used when observing during periods of low (less than 5 mph) ambient wind velocity.

21.7.4 Thermal Regulation

The enclosure has a thermal regulation system that heats and cools it via four air exhaust fans capable of transferring air between the enclosure and the outside environment. The fans are grouped in pairs with each pair tied to an exhaust duct. The two exhaust ducts are placed north and south of the enclosure. Associated with each exhaust duct are a barometric damper and a motorized damper.

The barometric dampers are automatic and do not require any control or monitoring. The motorized dampers will require open/close control and monitoring. The chillers will require monitoring of their head pressure. The exhaust ducts will require open/close control.

The air flow direction will be assumed based on the placement of the air flow sensors relative to the intake into the exhaust tunnel.

21.7.5 Air Conditioning

The enclosure is air-conditioned during the daytime (non-observing) hours. The set point of the AC unit is the low temperature of the previous night. The AC unit does not run during observing as not to adversely impact carousel seeing. The AC unit may also be disabled any time the telescope mount is in motion if it impacts the ability to properly cool the mount bearing oil.

While the AC unit is running the damper in the observing floor plenum is closed.

21.7.6 Interlock System

21.7.6.1 DOOR LEADING TO EXHAUST TUNNEL FROM MECHANICAL PLANT ROOM

Hazard: Possible injury if personnel in tunnel when exhaust fans operate.

Remedy: Mechanical interlock provided on door leading into exhaust fan tunnel to disconnect fan electrical power before door is opened.

Hardware Provided By: M3 Engineering

21.7.6.2 PLATFORM LIFT INTERFACE TO THE CAROUSEL CRANE

Details: The carousel crane will be used inside the enclosure chamber to:

--lift the mirror out of the primary mirror cell and install it in the washout stand (centered on the platform lift) prior to translating the mirror into the re-coating facility.

--lift the mirror off of the washout stand and reinstall it in the mirror cell after the re-coating operation.

During lifting operations. the crane rigging is centered over the platform lift, and all cable motion must only occur in the vertical direction. The crane is equipped with a micro-drive to precisely control the acceleration/deceleration of the crane cable drum.

Hazards: **A.** If carousel azimuth drives operate while lifting operation in progress;

B. If crane trolley drive is activated while lifting operations are in progress (i.e. the crane cable drum trolley assembly moves in a horizontal fashion across the two trolley support beams);

C. If the crane cable drum drive acceleration/deceleration exceeds the safe limit;

D. If telescope azimuth/altitude drives operate while lifting operations in progress (telescope top end structure could snag lifting cables).

Remedies: A. Deactivate carousel azimuth drive servos during primary mirror lifting operations (key interlock). This must **not** be automatic.

B. Deactivate crane trolley drive servos during primary mirror lifting operations (Key interlock);

C. Deactivate all crane cable drum drive acceleration/deceleration capability other than that provided by the crane micro-drive (key interlock);

D. After the telescope is in the stowed position (zenith pointing), deactivate telescope altitude/azimuth drive servos during primary mirror lifting operations.

Hardware Provided By: Coast Steel (except D)

21.7.6.3 MIRROR LIFTING SHAFT IN THE STATIONARY CHAMBER FLOOR INTERFACE TO THE CAROUSEL CRANE

Details: While the primary mirror is in the re-coating facility in the enclosure basement, the carousel crane will be utilized to:

--lift the mirror out of the washout stand and install the mirror in the base of the coating chamber prior to re-coating operations;

--lift the mirror out of the base of the coating chamber and reinstall the mirror on the washout stand at the conclusion of re-coating operations.

The carousel crane cable drum assembly traverse two beams attached to the carousel arch girders. The mirror lifting shaft (located in the stationary floor of the enclosure) is centered over the location of the mirror washout area on the coating plant floor.

During lifting operations, the crane rigging must be centered over the lifting shaft, and all cable motion must only occur in the vertical direction. The crane is equipped with a micro-drive to precisely control the acceleration/deceleration of the crane cable drum.

Hazards: **A.** If carousel azimuth drives operate while lifting operation in progress;

B. If crane trolley drive is activated while lifting operations are in progress (i.e. the crane cable drum trolley assembly moves in a horizontal fashion across the two trolley support beams);

C. If the crane cable drum drive acceleration/deceleration exceeds the safe limit;

D. If telescope azimuth/altitude drives operate while lifting operations in progress (telescope top end structure could snag lifting cables).

Remedies: A. Deactivate carousel azimuth drive servos during primary mirror lifting operations (key interlock);

B. Deactivate crane trolley drive servos during primary mirror lifting operations (Key interlock);

C. Deactivate all crane cable drum drive acceleration/deceleration capability other than that provided by the crane micro-drive (key interlock);

D. After the telescope is in the stowed position (zenith pointing), deactivate telescope altitude/azimuth drive servos during primary mirror lifting operations.

Hardware Provided By: Coast Steel (except D)

22 Details of the Interlock and Safety System

22.1 Function of the Interlock and Safety System

The Interlock and Safety System (ISS) monitors the status of a large number of devices and, based on their current status, either enables or disables specific devices in the observatory. The Interlock and Safety System shall exist as a separate, parallel system to other observatory control systems. This system is intended to operate in a double safe mode — by this we mean that it is not sufficient to only detect the presence of a condition that causes an interlock, it also necessary to sense the absence of this condition.

The current baseline for this system is to use a programmable logic controller (PLC) to monitor interlock signals and to initiate actions based on these interlocks. It is necessary to define a standard interface, if possible, to which systems desiring interlocks can connect. It is intended that, where possible/practical, all interlock systems are self actuating and that only their status is monitored by the PLC system.

The philosophy is that the primary system that is desired to be interlocked performs this function independently of the PLC system. If there are secondary interlocks to be triggered from the primary system then these, in general, operate through the PLC system.

One of the most important observatory interlocks forces all telescope mount motion to cease. This, the Mount Emergency Stop, is the primary mechanism for bringing the observatory into a safe condition. This is accomplished by simultaneously removing electrical power to both the mount drives and brakes which has the effect of disabling the drives and engaging the brakes.

22.2 INTERLOCK AND SAFETY SYSTEM OVERVIEW

22.2.1 External Interfaces

The Interlock and Safety System has the following external interfaces:

- +TTL connections to all passive interlock sources
- +TTL connections to active interlock sources and recipients (includes VMEbus systems)
- Channel Access connections to non-TTL (network-based) software interlocks.
- Channel Access connections to the Observatory Control System.

22.2.2 External Bus Connections

The ISS is connected to the following external buses:

- Control LAN

22.2.3 Internal Subsystems

The Interlock and Safety System is composed of the following major subsystems:

- Interlock Management System (IMS)

  This is the PLC-based interlock control system.
- Safety Status System (SSS)

  This is the EPICS-based system which provides for

  --communication between the IMS and the OCS.

  --handling of network-based (non-TTL) software interlocks including the interaction with the IMS.
- Brake Monitoring System (BMS)

  This is the EPICS-based system that monitors the status of the brake system by reading a number of sensors. Control of the brakes is accomplished by triggering the Mount Emergency Stop interlock which is an IMS function.

22.2.4 Internal Interfaces

The ISS has the following major internal interfaces:

- RS-232 (Serial) link between the ISS IOC and the IMS PLC-5.

- CANBus connections to sensors associated with the brakes hardware

22.2.5 Internal Data Stores

The ISS has the following major internal data stores:

- Allen-Bradley PLC-5 programs.

  PLC-5 programs are developed on a PC-based system which will download the resulting programs into the PLC-5 via the Allen-Bradley Data Highway/Data Highway Plus. This PC-based development system is stand-alone and has no other connections to the Interlock and Safety System.
- EPICS databases

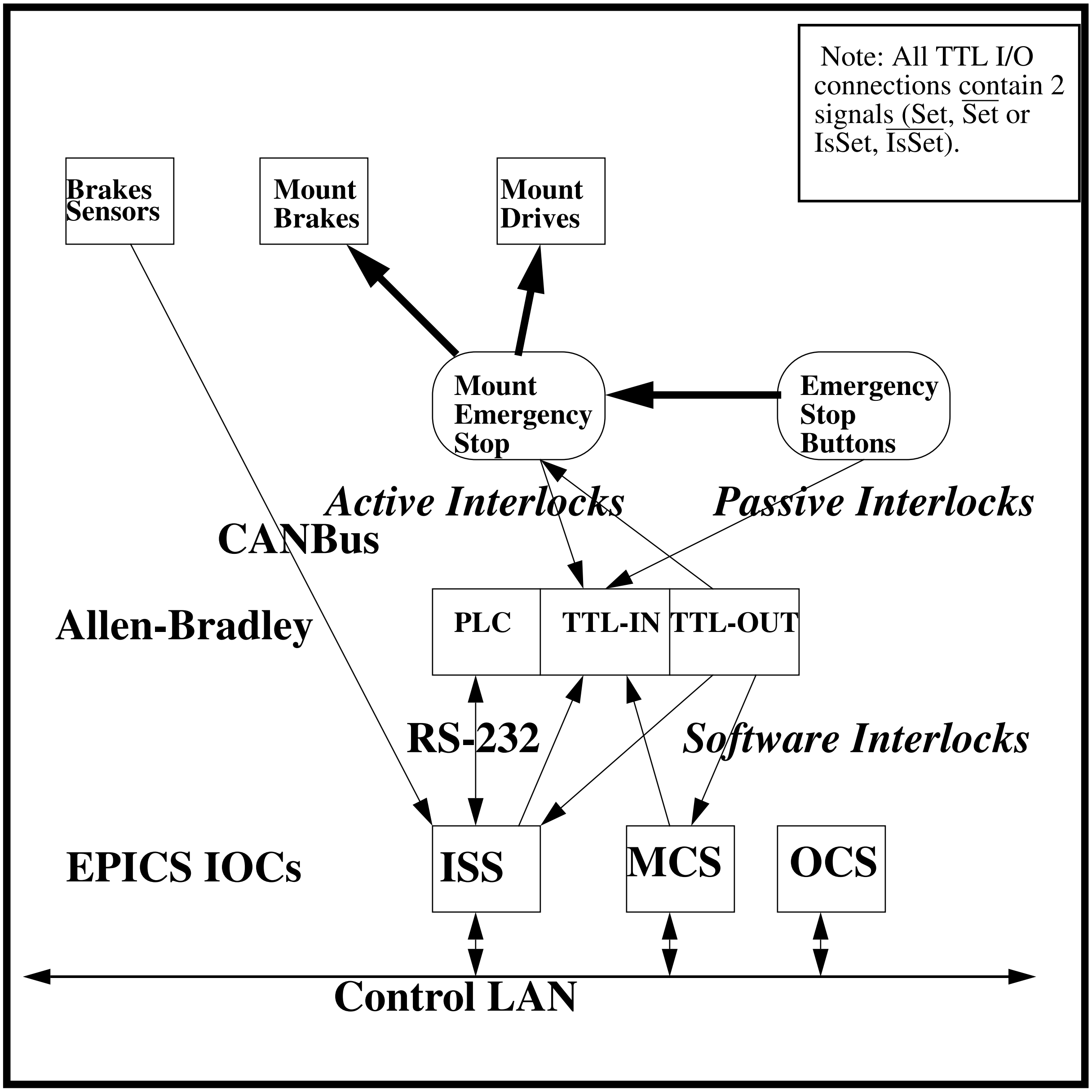
  The EPICS databases are developed on a SUN workstation and downloaded onto a VxWorks-based VMEbus system. This is accomplished by utilizing the standard VxWorks and EPICS development environment.

22.2.6 Computer Hardware

The Interlock Monitoring System will be implemented on an Allen-Bradley PLC-5 controlled 1771 I/O crate. Besides the PLC itself, this crate will contain a number of TTL input and output modules. These TTL I/O modules are the normal interface into the ISS.

The Safety Status System and the Brake Monitoring System will be implemented on an EPICS IOC. This VMEbus crate used to monitor the interlocks and to process network-based software interlocks and will be located on the mount base next to the Mount Control System IOC and dual-headed workstation. The VME crate control CPU is a M68040-based MVME-167. Communication between this VME system and the Allen-Bradley based IMS shall be across a RS-232 interface.

FIGURE 22 - 1 Interlock and Safety System Hardware Configuration



22.3 EXTERNAL INTERFACES

22.3.1 Interface to Passive (Hardware) Interlocks

22.3.1.1 REQUIREMENTS

The Interlock and Safety System in this case is completely passive meaning that it cannot trigger or clear the interlock. The only function provided is that of monitoring the state of the hardware interlock.

It is allowable to use this hardware interlock status to trigger other interlocks.

22.3.1.2 IMPLEMENTATION

The status of each passive interlock is sent to the ISS as +TTL signals indicating whether that interlock is set or cleared. The actual enforcement of the mechanism lockout is the responsibility of the hardware.

22.3.2 Interface to Active Interlocks

22.3.2.1 REQUIREMENTS

The Interlock and Safety System must be able to trigger interlocks based on request that originate from external hardware mechanisms. This allows a hardware subsystem to lock out another subsystem through the use of active interlocks.

22.3.2.2 IMPLEMENTATION

Active interlock requests are sent to the Interlock and Safety System as +TTL signals. These include the status lines arriving from hardware interlocks. The ISS then calculates which active interlocks are to be set based on the combination of hardware interlocks and active interlock requests currently triggered.

The resultant active interlock demands are output from the ISS as +TTL signals to the target interlock mechanisms.

22.3.3 Interface to Software Interlocks

22.3.3.1 REQUIREMENTS

Software systems (specifically EPICS-based VMEbus systems or IOCs) can present an interface to the ISS by using the same protocol as the active interlocks discussed above.

Each software system participating in the ISS shall also have a mechanism by which it indicates when it is non-functioning.

22.3.3.2 IMPLEMENTATION

Software interlock requests to the ISS can occur by two distinct methods.

The first, and preferred, technique, is for the IOC to assert an active interlock by asserting a +TTL output signals. These systems will also be required to have the ability to accept +TTL signals that indicate that the recipient system is under an interlock.

The second method, for use by systems that do not have the ability to generate or receive +TTL signals, is for the system to communicate with the Interlock and Safety System via the EPICS Channel Access protocol across the Control LAN. This may be used by the OCS and non-conforming instruments.

The specific meanings assigned to software interlocks are the subject of a trade study but may indicate the following:

- An interlock that a software system generates signals that it is no longer in a state where it can accept or send commands to the rest of the control system. It is unclear as to whether or not the system should be capable of indicating that it has an internal interlock triggered in the case where that internal interlock does not disable the complete system. If this function is required, then this indication of a non-critical internal fault must be distinquishable from the normal system-wide fault interlock.
- An interlock that is sent to a software system indicates a request from the Interlock and Safety System that the recipient system should be disabled. It is planned that this interlock should affect all subsystems controlled by the receiving processor.

The Channel Access communications library has the ability to sense when a particular IOC is disabled. This is indicated by an INVALID alarm state being raised on process variables managed by that IOC. The ISS will monitor specific process variables within each participating IOC to trigger the software interlock associated with that system.

22.3.4 Interface to the Observatory Control System

22.3.4.1 REQUIREMENTS

There should be an EPICS front panel monitoring the Interlock status with ability to query 'WHY'.

22.3.4.2 IMPLEMENTATION

All interlock status information contained within the ISS can be read by the OCS using EPICS Channel Access. It is conceivable that this could be implemented as part of the normal interface into the Observatory Status and Alarm system.

22.4 INTERFACE TO OCS SCREEN SYSTEMS

22.4.1 Encoder Screen

TBD.

22.4.2 Limit Screen

TBD.

22.4.3 Alarms Screen

TBD.

22.4.4 Interlocks Screen

TBD.

22.5 EXTERNAL BUS CONNECTIONS

22.5.1 Control LAN

The ISS is connected to the Control LAN via the ethernet port on the Woodsworth transition module which interfaces to the P2 connections of the MVME-167 VME-Bus CPU card. This ethernet connection is used for downloading of the VxWorks and EPICS systems at power-up and for EPICS Channel Access UDP/IP and TCP/IP based communications.

22.6 OVERVIEW OF INTERLOCK AND SAFETY FUNCTIONALITY

The functional requirements of each of the interlock subsystems are outlined here.

TABLE 22 - 1 **General Functions (for all ISS subsystems)**

Function	Description	Range	Frequency	Type
Power	turn power on/off to entire unit	ON | OFF	< 0.1 Hz	digital switch
Health	monitor overall status of device	OK | BAD	1 Hz	digital i/p
HeartBeat	incrementing counter	32 bits	20-50Hz	digital i/p
WatchDog			20-50Hz	digital i/o

TABLE 22 - 2 Interlock Management System Functions

Function	Description	Range	Frequency	Type
Monitor Interlock	Monitor the state of a specific passive or active interlock or collection of interlocks. Uses TTL I/O.	Set/Clear	<1Hz	TTL i/p
Request Interlock	Request that a specific active interlock be set or cleared. Uses TTL I/O.	Set/Clear	<1Hz	TTL o/p
Status	Returned status from last IMS Interlock Request. Uses Channel Access.	Accept/ Reject	<1Hz	binary i/p

TABLE 22 - 3 Safety Status System Functions

Function	Description	Range	Frequency	Type
Monitor Interlock	Monitor the state of a specific interlock or collection of interlocks. Uses Channel Access.	Set/Clear	<1Hz	binary i/p
Request Interlock	Request that a specific software interlock be set or cleared. Uses Channel Access.	Set/Clear	<1Hz	binary o/p
Status	Returned status from last SSS Interlock Request. Uses Channel Access.	Accept/ Reject	<1Hz	binary i/p

TABLE 22 - 4 Brake Monitoring System Functions

Function	Description	Range	Frequency	Type
Monitor Azimuth Brakes (4)	Monitor brake status (via pressure?). Uses CANBus.	TBD	1Hz	analog i/p
Check Azimuth Brakes (4)	Monitor dragging of brakes (via strain gauges?). Uses CANBus.	TBD	1Hz	analog i/p
Monitor Elevation Brakes (2)	Monitor brake status (via pressure?). Uses CANBus.	TBD	1Hz	analog i/p
Check Elevation Brakes (2)	Monitor dragging of brakes (via strain gauges?). Uses CANBus.	TBD	1Hz	analog i/p

22.7 INTERLOCKS AND SAFETY INTERNAL SUBSYSTEMS

22.7.1 Interlock Management System

The interlock management system monitors the status of a large number of devices and, based on their current status, either enables or disables specific devices in the observatory. This system is intended to operate in a double safe mode — by this we mean that it is not sufficient to only detect the presence of a condition that causes an interlock, it also necessary to sense the absence of this condition. The current baseline for this system is to use a programmable logic controller (PLC) to monitor interlock signals and to initiate actions based on these interlocks. It is necessary to define a standard interface, if possible, to which systems desiring interlocks can connect. It is intended that, where possible/practical, all interlock systems are self actuating and that only their status is monitored by the PLC system. The philosophy is that the primary system that is desired to be interlocked performs this function independently of the PLC system. If there are secondary interlocks to be triggered from the primary system then these, in general, operate through the PLC system.

22.7.1.1 HARDWARE ARCHITECTURE

Each point-to-point interlock cable shall contain the following TTL signals:

- Each system outputs the IsSet and $\overline{\text{IsSet}}$ status lines to the Interlock master PLC.
- Each system receives the Set and $\overline{\text{Set}}$ commands from the Interlock master PLC.

All active interlock systems shall set or clear interlocks based on the following logic table:

TABLE 22 - 5 **Set Command Logic**

Set\\$\overline{\textbf{Set}}$	**0**	**1**
0	Set	Clear
1	Set	Set

All interlock systems shall interpret another interlock system as being set or clear by the following logic table:

TABLE 22 - 6 **IsSet Command Logic**

IsSet\\$\overline{\text{IsSet}}$	**0**	**1**
0	IsSet	IsClear
1	IsSet	IsSet

Note: Can this be done with fiber optics?

22.7.1.2 INTERLOCK PHILOSOPHY

The following paragraphs are taken from section 13.3 of the Telescope CDR document [RPT-TE-G0018].

- Safety interlocks shall be provided for the protection of personnel and equipment.
- The primary purpose of the interlocks and safety systems is to prevent injury to personnel working on or around the telescope during operation, maintenance, repair, etc.... of the telescope and it's related components. The secondary purpose is to prevent damage to the telescope, instrumentation, or enclosure that would occur if a subassembly or system (e.g. telescope drive motors) is activated incorrectly or without activating required, interfacing equipment.
- Independent hard wired interlocks shall be provided on all telescope systems where necessary to prevent unsafe situations resulting from single component or subsystem failure.
- Wherever possible, passive systems shall be employed as the primary safety interlock (e.g. hydraulic preload device for drive motors that is pressurized with hydrostatic bearing oil-- drives cannot transmit drive torque if hydrostatic bearings are not fully pressurized.)
- Wherever possible two independent sensors wired in series shall be utilized as safety interlocks. These safety systems shall be designed to provide an electrical 'continuity' before allowing the desired system to activate-- i.e. a fault in either sensor shall disable the corresponding telescope system.
- All safety interlocks shall be monitored and controlled by the Gemini Position and Control System as outlined elsewhere in this document.
- The hazards identified are described, together with a description of the interlocks provided to prevent the hazard from occurring and a description of the hardware.

22.7.1.3 MOUNT EMERGENCY STOP INTERLOCK

The brake subsystem is responsible for maintaining the current telescope position in the presence of disturbing forces, most notably gravity and wind. The brakes can be applied either by operator command or in response to the current state of the observatory. The brakes are the target device for a large part of the interlock system — most interlock conditions remove the power from the telescope drives and apply the telescope brakes. In order to provide for fail-safe brakes it is necessary to bypass the computer system for activating the brakes in response to interlock conditions. It may be desirable to have some of the interlock logic on an electronic card near the brakes themselves. It is also desirable to have a brake watchdog that is reset by the Interlock and Safety IOC.

[6.3.4 (a)] The azimuth and elevation brake systems are designed to stop the telescope rotation during both normal operation (e.g. nightly shutdown) and emergency conditions. Both axes employ fail-safe brake calipers which activate (brake) when power is removed. In addition, the hydraulic bearing oil supply system is interlocked to the operation of the brakes so that a loss of hydraulic pressure will activate the azimuth and elevation braking systems.

[6.3.4 (b)] As shown in drawing #87-GP-0305-0001, the azimuth brake rotor is attached to the inside diameter of the azimuth track journal. The rotor is fabricated from ten steel rotor segments which are field-welded together at the time of telescope assembly. The brake calipers are attached to the bottom of the azimuth radial bearing brackets. The azimuth brake system is designed to provide a total of 273 kN-m of braking torque. This torque (approximately 4x the azimuth drive torque) will decelerate the telescope from 2 degrees/second in approximately 1 second. Maximum deceleration is approximately 2.3 degrees/second^2.

[6.3.4 (c)] As shown in drawing #87-GP-0211-0001, the two elevation brake rotors are attached to the outside faces of the elevation disks. Two calipers are attached to the mount column and provide a total of 140kN-m of torque about the elevation axis. This torque (approximately 4x the elevation drive torque) will stop the OSS at a maximum deceleration of approximately 2.5 degrees/second^2.

22.7.2 Safety Status System

The Safety Status System (SSS) has two distinct functions. The first is to serve as a route for systems not capable of interfacing to TTL signals to interface to the software interlocks managed by the IMS. The second is to provide status information concerning all interlocks known to the IMS to the Observatory Control System.

22.7.2.1 CHANNEL ACCESS BASED SOFTWARE INTERLOCKS

It is possible that a number of software systems including the OCS and some visitor instruments may not have an interface into TTL I/O. To allow these systems to participate in the software interlock protocol, the SSS will accept interlock requests andtransmit interlock demands via EPICS Channel Access.

The role of the SSS is purely that of a message passer. All software interlock logic will still be handled by the PLC-based IMS. Communication between the SSS and the IMS is via a RS-232 serial link.

22.7.2.2 INTERLOCK STATUS INFORMATION

The SSS will be capable of obtaining complete status information on all interlocks (passive, active, and software) by communicating with the IMS. These status values will be placed into the EPICS database maintained by the SSS and will be accessible by other Channel Access clients including the OCS.

22.7.3 Brake Monitoring System

The Brake Monitoring System (BMS) is used to supply readback information on a number of sensors connected to the mount altitude and azimuth axis brakes. The standard Gemini sensor network interface will be based on CANBus.

The two types of sensors required of the BMS are:

- Pressure Sensors

  There will be one (or a set of) pressure sensors for both mount altitude and azimuth brake systems. The purpose of the pressure reading is to serve as an indication as to whether or not the brakes are applied.

- Strain Gauges

  For each mount brake there will be one (or a set of) strain gauges. These values returned from these gauges will allow the control system to determine if the brakes are dragging.

It must be possible to connect a computer or a terminal directly to the brake subsystem in order to run diagnostics and tests. This interface point is at a TBD location.

22.8 INTERLOCK SOURCES

22.8.1 Enclosure Interlocks

22.8.1.1 DOOR LEADING TO EXHAUST TUNNEL FROM MECHANICAL PLANT ROOM

Hazard: Possible injury if personnel in tunnel when exhaust fans operate.

Remedy: Mechanical interlock provided on door leading into exhaust fan tunnel to disconnect fan electrical power before door is opened.

Hardware Provided By: M3 Engineering

22.8.1.2 PLATFORM LIFT INTERFACE TO THE CAROUSEL CRANE

Details: The carousel crane will be used inside the enclosure chamber to:

--lift the mirror out of the primary mirror cell and install it in the washout stand (centered on the platform lift) prior to translating the mirror into the re-coating facility.

--lift the mirror off of the washout stand and reinstall it in the mirror cell after the re-coating operation.

During lifting operations. the crane rigging is centered over the platform lift, and all cable motion must only occur in the vertical direction. The crane is equipped with a micro-drive to precisely control the acceleration/deceleration of the crane cable drum.

Hazards: **A.** If carousel azimuth drives operate while lifting operation in progress;

B. If crane trolley drive is activated while lifting operations are in progress (i.e. the crane cable drum trolley assembly moves in a horizontal fashion across the two trolley support beams);

C. If the crane cable drum drive acceleration/deceleration exceeds the safe limit;

D. If telescope azimuth/altitude drives operate while lifting operations in progress (telescope top end structure could snag lifting cables).

Remedies: A. Deactivate carousel azimuth drive servos during primary mirror lifting operations (key interlock). This must **not** be automatic.

B. Deactivate crane trolley drive servos during primary mirror lifting operations (Key interlock);

C. Deactivate all crane cable drum drive acceleration/deceleration capability other than that provided by the crane micro-drive (key interlock);

D. After the telescope is in the stowed position (zenith pointing), deactivate telescope altitude/azimuth drive servos during primary mirror lifting operations.

Hardware Provided By: Coast Steel (except D)

22.8.1.3 MIRROR LIFTING SHAFT IN THE STATIONARY CHAMBER FLOOR INTERFACE TO THE CAROUSEL CRANE

Details: While the primary mirror is in the re-coating facility in the enclosure basement, the carousel crane will be utilized to:

--lift the mirror out of the washout stand and install the mirror in the base of the coating chamber prior to re-coating operations;

--lift the mirror out of the base of the coating chamber and reinstall the mirror on the washout stand at the conclusion of re-coating operations.

The carousel crane cable drum assembly traverse two beams attached to the carousel arch girders. The mirror lifting shaft (located in the stationary floor of the enclosure) is centered over the location of the mirror washout area on the coating plant floor.

During lifting operations, the crane rigging must be centered over the lifting shaft, and all cable motion must only occur in the vertical direction. The crane is equipped with a micro-drive to precisely control the acceleration/deceleration of the crane cable drum.

Hazards: **A.** If carousel azimuth drives operate while lifting operation in progress;

B. If crane trolley drive is activated while lifting operations are in progress (i.e. the crane cable drum trolley assembly moves in a horizontal fashion across the two trolley support beams);

C. If the crane cable drum drive acceleration/deceleration exceeds the safe limit;

D. If telescope azimuth/altitude drives operate while lifting operations in progress (telescope top end structure could snag lifting cables).

Remedies: A. Deactivate carousel azimuth drive servos during primary mirror lifting operations (key interlock);

B. Deactivate crane trolley drive servos during primary mirror lifting operations (Key interlock);

C. Deactivate all crane cable drum drive acceleration/deceleration capability other than that provided by the crane micro-drive (key interlock);

D. After the telescope is in the stowed position (zenith pointing), deactivate telescope altitude/azimuth drive servos during primary mirror lifting operations.

Hardware Provided By: Coast Steel (except D)

22.8.2 Mount Interlocks

22.8.2.1 Elevation Drive Motors Driven when the Elevation Hydrostatic Bearings are not Fully Pressurized

[13.3.1 (a)] **Hazard:** If the elevation drive motors are driven when the elevation hydrostatic bearings are not fully pressurized damage could be caused to the drive rollers, elevation disks, and/or hydrostatic bearing system.

[13.3.1 (b)] **Interlocks:** Two safety interlocks provided to prevent this occurring:

1. If the elevation hydrostatic bearings are not fully pressurized, the elevation drive motors shall be prevented from receiving electrical power.
2. If the elevation hydrostatic bearings do not receive fully pressurized oil, the elevation drive motor preloads shall not be provided.

[13.3.1 (c)] **Hardware Description:** Two hardware systems are provided to prevent this occurrence:

1. Two pressure transducers which provide electrical contact (continuity) when a minimum, preset pressure is attained, shall be located at the two elevation bearing oil manifolds located on top of each mount column. The transducers shall be wired in series; a low oil pressure condition which is sensed by either one or both of the transducers shall prevent electrical power from reaching both elevation drive motor assemblies.
2. A hydraulic preload cylinder shall be used on each elevation drive motor assembly to provide the normal force required for drive traction. This cylinder shall receive pressurized oil from the elevation bearing oil manifolds.

22.8.2.2 ELEVATION DRIVE MOTORS ACTIVATED WHEN ELEVATION BRAKES ARE ON (ENGAGED)

[13.3.2 (a)] **Hazard:** Elevation drive motors activated when elevation brakes are on (engaged). This condition could cause damage to drive rollers, elevation disks, and/or brake system.

[13.3.2 (b)] **Safety Interlocks**: If the elevation brakes are on (applied), the elevation drive motors shall be prevented from receiving electrical power.

[13.3.2 (c)] **Hardware Description:** Elevation brakes are 'fail safe'; i.e. spring on (engaged), hydraulic powered off (disengaged). A pressure transducer, which provides electrical contact (continuity) when a minimum, preset oil pressure is attained, shall be fitted to the brake manifold. A low pressure condition which is sensed by the transducer shall prevent electrical power from reaching both elevation drive motor assemblies.

22.8.2.3 ELEVATION DRIVE MOTORS ACTIVATED WHEN ELEVATION LOCKING PIN ENGAGED

[13.3.3 (a)] **Hazard:** Elevation drive motors activated when elevation locking pins are engaged. This condition could cause damage to drive rollers, elevation disks, and/or locking pin system.

[13.3.3 (b)] **Safety Interlocks:** If the elevation locking pins are engaged, the elevation drive motors shall be prevented from receiving electrical power.

[13.3.3 (c)] **Hardware Description:** A electrical micro-switch shall be located on the elevation locking pin assembly. This switch shall provide electrical contact (continuity) to the elevation drive motor assemblies only if the locking pin assembly is disengaged (retracted).

22.8.2.4 ELEVATION DRIVE MOTORS ACTIVATED WHEN ELEVATION MANUAL DRIVE ENGAGED

[13.3.4 (a)] **Hazard:** Elevation drive motors activated when elevation manual drive system is engaged. This condition could cause injury to personnel working on the manual drive system. In addition this condition could cause damage to manual drive system and/or drive system.

[13.3.4 (b)] **Safety Interlocks:** If the elevation manual drive system is engaged, the elevation drive motors shall be prevented from receiving electrical power.

[13.3.4 (c)] **Hardware Description**: A electrical micro-switch shall be located on the elevation manual drive assembly. This switch shall provide electrical contact (continuity) to the elevation drive motor assemblies only if the manual drive assembly is disengaged (retracted).

22.8.2.5 ELEVATION END OF TRAVEL SENSED

[13.3.5 (a)] **Hazard:** Elevation drive motors activated when telescope tube has rotated to end-of-travel limits (zenith, horizon). This condition could cause damage to drive system and/or OSS assembly.

[13.3.5 (b)] **Safety Interlocks:** If an elevation 'hard' limit is sensed, electrical power to the elevation drive motors shall be removed.

[13.3.5 (c)] **Hardware Description:** A electrical micro-switch shall be located at each of the elevation 'hard' rotation limits.

22.8.2.6 AZIMUTH DRIVE MOTORS ACTIVATED IF AZIMUTH HYDROSTATIC BEARINGS ARE NOT FULLY PRESSURIZED

[13.3.6 (a)] **Hazard:** Azimuth drive motors activated if azimuth hydrostatic bearings are not fully pressurized. This condition could cause damage to drive rollers, azimuth journal, and/or hydrostatic bearing system.

[13.3.6 (b)] **Safety Interlocks:** Two interlocks are provided:

1. If the azimuth hydrostatic bearings are not fully pressurized, the azimuth drive motors shall be prevented from receiving electrical power.
2. If the azimuth hydrostatic bearings do not receive fully pressurized oil, the azimuth drive motor preloads shall not be provided.

[13.3.6 (c)] **Hardware Description:** Two pressure transducers which provide electrical contact (continuity) when a minimum, preset pressure is attained, shall be located at four of the azimuth bearing oil manifolds located underneath the mount base. The transducers shall be wired in series; a low oil pressure condition which is sensed by either one or both of the transducers shall prevent electrical power from reaching both azimuth drive motor assemblies.

A hydraulic preload cylinder shall be used on each azimuth drive motor assembly to provide the normal force required for drive traction. This cylinder shall receive pressurized oil from the azimuth bearing oil manifolds.

22.8.2.7 AZIMUTH DRIVE MOTORS ACTIVATED WHEN AZIMUTH BRAKES ARE ON (APPLIED)

[13.3.7 (a)] **Hazard:** Azimuth drive motors activated when azimuth brakes are on (applied). This condition could cause damage to drive rollers, drive rings, and/or brake system.

[13.3.7 (b)] **Safety Interlocks:** If the azimuth brakes are on (applied), the azimuth drive motors shall be prevented from receiving electrical power.

[13.3.7 (c)] **Hardware Description:** Azimuth brakes are 'fail safe'; i.e. spring on (engaged), hydraulic powered off (disengaged). Two pressure transducers which provide electrical contact (continuity) when a minimum, preset oil pressure is attained, shall be fitted to the brake manifold. The transducers shall be wired in series; a low pressure condition which is sensed by either one or both of the transducers shall prevent electrical power from reaching both azimuth drive motor assemblies.

22.8.2.8 AZIMUTH DRIVE MOTORS ACTIVATED WHEN TELESCOPE MOUNT HAS ROTATED TO END-OF-TRAVEL LIMITS

[13.3.8 (a)] **Hazard:** This condition could cause damage to drive system and/or mount assembly.

[13.3.8 (b)] **Safety Interlocks:** If an azimuth 'hard' limit, an incorrect topple bracket position, or the azimuth cable wrap height microswitch is sensed, then electrical power to the azimuth drive motors shall be removed.

[13.3.8 (c)] **Hardware Description:** A electrical micro-switch shall be located at each of the azimuth 'hard' rotation limits.

22.8.2.9 ELEVATION LOCKING PIN RETRACTED IF PRIMARY MIRROR CELL IS REMOVED

[13.3.9 (a)] **Hazard:** Elevation locking pin retracted if primary mirror cell is removed. This condition would allow the unbalanced telescope OSS assembly to rotate inadvertently about the elevation axis. Injury to personnel working near the telescope would be possible. In addition, damage to the telescope, mirrors, and/or instrumentation would be possible.

[13.3.9 (b] **Safety Interlocks:** If the primary mirror cell is removed, the elevation locking pin shall be prevented from receiving electrical power, and therefore being retracted.

[13.3.9 (c)] **Hardware Description:** An electrical micro-switch shall be located on the elevation locking pin assembly. This switch shall provide electrical contact (continuity) to the primary mirror cell attachment assemblies only if the locking pin assembly is engaged.   KR: Not correct, primary mirror attachment is mechanical?

22.8.2.10 TOP END REMOVED WITHOUT ELEVATION LOCKING PIN ENGAGED

[13.3.10 (a)] **Hazard:** This condition would allow the unbalanced telescope OSS assembly to rotate inadvertently about the elevation axis. Injury to personnel working near the telescope would be possible. In addition, damage to the telescope, mirrors, and/or instrumentation would be possible.

[13.3.10 (b)] **Safety Interlocks:**   If the elevation locking pin is disengaged, the top end latching mechanism shall be prevented from receiving electrical power, and therefore being released.

[13.3.10 (c)] **Hardware Description**: This is implemented by a transferable castell key. Locking pin inserted enables removal of key; key inserted into the top-end enables top-end latches to be operated.

22.8.2.11 ELEVATION DRIVE MOTORS ACTIVATED AND/OR ELEVATION BRAKES DISENGAGED IF PERSONNEL ARE WORKING ON ELEVATION DRIVE MOTOR ASSEMBLIES

[13.3.11 (a)] **Hazard:** Elevation drive motors activated and/or elevation brakes disengaged if personnel are working on elevation drive motor assemblies, or near areas of the telescope that are subjected to ‘pinches’ if the OSS assembly were to rotate about the elevation axis. Injury to personnel working near the telescope would be possible. In addition damage to the telescope is possible.

[13.3.11 (b)] **Safety Interlocks**:   If the elevation maintenance key is removed, the elevation drive motors shall be prevented from receiving electrical power.

[13.3.11 (c)] **Hardware Description:**   A removable-type maintenance key shall be installed in the telescope local control panel located TBD. This key shall be electrically connected in series with the elevation drive motor electronics. Removal of the key shall electrically disconnect the drive motors.

22.8.2.12 AZIMUTH DRIVE MOTORS ACTIVATED AND/OR AZIMUTH BRAKES DISENGAGED IF PERSONNEL ARE WORKING ON AZIMUTH DRIVE MOTOR ASSEMBLIES

[13.3.12 (a)] **Hazard:** Azimuth drive motors activated and/or azimuth brakes disengaged if personnel are working on azimuth drive motor assemblies, or near areas of the telescope that are subjected to ‘pinches’ if the telescope mount were to rotate about the azimuth axis. Injury to personnel working near the telescope would be possible. In addition damage to the telescope is possible.

[13.3.12 (b)] **Safety Interlocks**: If the azimuth maintenance key is removed, the azimuth drive motors shall be prevented from receiving electrical power.

[13.3.12 (c)] **Hardware Description:** A removable-type maintenance key shall be installed in the telescope local control panel located TBD. This key shall be electrically connected in series with the azimuth drive motor electronics. Removal of the key shall electrically disconnect the drive motors.

22.8.2.13 INADVERTENT TELESCOPE ROTATION ABOUT AZIMUTH OR ELEVATION AXIS INJURING PERSONNEL AND/OR DAMAGING TELESCOPE, INSTRUMENTATION, ENCLOSURE

[13.3.13 (a)] **Hazard:** Inadvertent telescope rotation about azimuth or elevation axis injuring personnel and/or damaging telescope, instrumentation, enclosure.

[13.3.13 (b)] **Safety Interlocks:** Pressing any ‘Emergency Stop’ button shall:

1. Remove power from elevation drive assemblies.
2. Remove power from azimuth drive assemblies.
3. Engage (apply) elevation brakes.
4. Engage (apply) azimuth brakes.
5. Remove power from enclosure drive motors.
6. Remove power from enclosure shutter drive assemblies.
7. Engage (apply) enclosure brakes.
8. Sound audible and visual alarm inside enclosure and control room.

[13.3.13 (c)] **Hardware:** ‘Emergency Stop’ push-buttons shall be provided in the control room, on the local control panel, in the Cassegrain instrumentation area, near both of elevation drive motor assemblies, near the azimuth cable twister, TBD....

Push-buttons shall remain activated until manually reset.

22.8.3 M1 Interlocks

22.8.3.1 Air Pressure System Near Over Pressure Limit

Hazard: The sum of the 120 load cell forces is less than 10% of mirror weight.

Remedy: The following safety measures shall be taken:

- check air pressure sensor reading
- interlock air pressure regulator
- turn off air supply

22.8.3.2 Air Pressure System Failure

Hazard: The sum of 120 load cell forces is larger than 30% of the mirror weight. This will not harm the load cell system as it is designed to survive failure of the air pressure system.

Remedy: The following safety measures shall be taken:

- check air pressure sensor reading

22.8.3.3 Lateral Mirror Defining System Near Limit

Hazard: The displacement at any lateral position sensor is larger than TBD mm.

Remedy: The following safety measures shall be taken:

- interlock master cylinder

22.8.3.4 Axial Mirror Defining System Near Limit

Hazard: The displacement at any axial position sensor is larger than TBD mm.

Remedy: The following safety measures shall be taken:

- interlock master cylinder

22.8.3.5 Active Optics System Loading Near Limit

Hazard: The actuator load cell reading, or the support load cell reading, is larger than 90% of the load cell loading limit.

Remedy: The following safety measures shall be taken:

- unload actuator force
- interlock actuator pressure regulator

22.8.3.6 LATERAL LOAD CELL LOADING NEAR LIMIT

Hazard: The lateral support load cell reading is larger than 90% of the load cell loading limit.

Remedy: The following measures shall be taken:

- rotate the telescope to zenith pointing

22.8.3.7 RADIATING PLATE SYSTEM

Hazard: The coolant line pressure drops.

Remedy: Turn off the circulation pump.

22.8.3.8 PERSONNEL WITHIN MIRROR CELL

Hazard: A person is inside the mirror cell.

Remedy: The following safety measures shall be taken.:

- interlock the telescope elevation drive system at zenith pointing
- interlock the telescope azimuth drive system

22.8.3.9 M1 CELL ASSEMBLY REMOVAL

Hazard: The M1 cell assembly is being removed for primary recoating.

Remedy: The following safety measures shall be taken:

- interlock the telescope elevation drive system at zenith pointing
- interlock the telescope azimuth drive system
- turn on the M1 cell axial support to push the mirror against safety support
- interlock all the M1 cell assembly active systems

22.8.3.10 SEISMIC HAZARDS

Hazard: The acceleration on the mirror is greater than 0.5g.

Remedy: The following measures shall be taken:

- turn on the M1 cell axial support to push the mirror against safety support
- rotate the telescope to zenith pointing

22.8.3.11 M1 DRY GAS FLUSH SYSTEM INTERFACE

Hazard: The M1 dry gas system has developed a leak and is flooding the mirror cell area. Only dry AIR will be used in the mirror cell assembly which poses no safety issues (no nitrogen). This is for the active actuators, air seal, and air pressure support of the mirror. The only other gas to consider would be a relatively small amount of helium which passes through the mirror cell to the instrument area. If a major leak would occur, we may have a safety issue.

Remedy: The following measures shall be taken:

- the oxygen monitor will trigger an audible and visual alarm if the oxygen levels drop below a TBD value.

23 Control Scenario Walk-throughs

23.1 Introduction

The behavior of the Gemini Control System can be examined by describing the behavior of the system when operating under each of the reference scenarios introduced in "System Requirements" in Chapter 3 and [40]. This provides an *operational* check of the system functionality to complement the *narrative* description found here in the SDD.

These walkthroughs are now found in a separate document [41].

24 Glossary

24.1 Introduction

This chapter contains descriptions of the terms and abbreviations used in the Gemini software documentation in alphabetical order. The descriptions are presented in two sections:

Abbreviations and Acronyms. gives a brief translation of the abbreviations and acronyms used in the Gemini documentation. Pointers are given to the glossary section where an abbreviation refers to a term described there.

Glossary. contains a description of the terms used in the Gemini documentation.

24.2 Abbreviations and Acronyms

A&G	Acquisition and Guidance
A&GCS	Acquisition and Guidance Control System
aai	EPICS array analogue input record
AAO	Anglo-Australian Observatory (Siding Spring, Australia)
aao[1]	EPICS array analogue output record
ADAM	Astronomical Data Acquisition/Analysis Monitor

1. Not to be confused with AAO, which is “Anglo-Australian Observatory”.

ADC	Analog to digital signal converter
ai	EPICS analogue input record
ALH	EPICS Alarm Handler
ANL	Argonne National Laboratory
ANSI	American National Standards Institute
AO[1]	Adaptive Optics (see glossary)
aO	Active Optics (see glossary)
ao	EPICS analogue output record
AOCS	Adaptive Optics Control System
APD	Avalanche PhotoDiode
APS	Advanced Photon Source (at ANL)
AR	EPICS archiving tool (really a "logging" tool)
ARG	Argument structure processing routines. Part of SDS.
ARR	EPICS archive retrieval tool
ASCII	American Standard Code for Information Interchange
AtmDC[2]	Atmospheric Dispersion Compensator
AURA	Association of Universities for Research in Astronomy (USA)
AVL	Attribute/Value Layer
BNF	Backus Naur Form (see glossary)
BNL	Brookhaven National Laboratory
CA	EPICS Channel Access interface
CA client	A program that accesses an EPICS database through CA
CA server	A program that provides access to an EPICS database
CADC	Canadian Astronomy Data Centre
CASE	Computer-Aided Software Engineering
CATS	Collaborative Access Teams (APS instrument groups)

1. The meanings of AO, aO and ao must not be confused.
2. Not to be confused with ADC which is "Analogue to Digital Converter".

CCD	Charge Coupled Device
CCS	Configurable Control System (see glossary)
CD	Configuration Dealer (see glossary)
CDR	Critical Design Review
CEBAF	Continuous Electron Beam Accelerator Facility (Newport News)
CFHT	Canada France Hawaii Telescope (Mauna Kea, Hawaii)
CIO Record	Command Input Output Record (see glossary)
CLUI	Command Line User Interface
CLS	Command Layer Server (see glossary)
CRCS	Cassegrain Rotator Control System
CTIO	Cerro Tololo Interamerican Observatory (Chile)
DBMS	Database management system
DAC	Digital to Analogue Converter
DC[1]	Detector Array Controller (see glossary)
DCT	EPICS Database Configuration Tool
Dec.	Declination
DESY	Deutsches Elektronen-Synchrotron (Hamburg)
DFD	Data Flow Diagram (see glossary)
DHS	Data Handling System (see glossary)
DITS	Distributed Instrumentation Tasking System (part of DRAMA)
EPICS DM	EPICS Display Manager tool
DM[2]	Deformable Mirror
DOE	Distributed Objects Everywhere (a Sun utility)
DRAL	Daresbury and Rutherford-Appleton Laboratories (UK)
DRAMA	Distributed Real-time AAO Monitor for Astronomy
DSD	Data Structure Diagram (see glossary)

1. Not to be confused with DAC, which is “Digital to Analogue Converter”.
2. Not to be confused with the EPICS DM tool.

ECS	Enclosure Control System
EDD	EPICS database control screen editor
EPICS	Experimental Physics and Industrial Control System (see glossary)
ERD	Entity Relationship Diagram (see glossary)
ESO	European Southern Observatory (Garching, Germany)
FDDI	Fibre ??? ??? Interface (defined by ANSI standard X3T9.5)
FITS	Flexible Image Transport System
FOV	Field of View
FPA	Focal Plane Array
FWHM	Full Width at Half Maximum
GCS	Gemini Control System
GPO	Gemini Project Office
GSCG	Gemini Software and Controls Group
GTA	Ground Test Accelerator (at LANL)
GUI	Graphical User Interface
HDF	Hierarchical Data Format
HDS	Hierarchical Data System
HOS	High-level Operations Software
HROS	High Resolution Optical Spectrograph
HRWFS	High Resolution Wavefront Sensor
I/O	Input / Output
ICD	Interface Control Document
ICE	Inter-Client Exchange (protocol)
ICS	Instrument Control System (see glossary)
ID	Identifier
IDL	Interactive Data Language
IGPO	International Gemini Project Office
IEEE	Institute of Electrical and Electronic Engineers
IMP	Interprocess Message Passing System (part of DRAMA)

IOC	Input/Output Controller — an EPICS-based VME system
IPC	Interprocess Communication
IR	Infra Red
IRAF	Image Reduction and Analysis Facility
ISDN	Integrated Services Digital Network (see glossary)
JACH	Joint Astronomy Centre Hilo (Hawaii)
JCMT	James Clerk Maxwell Telescope (Mauna Kea, Hawaii)
KPNO	Kitt Peak National Observatory
LAN	Local Area Network
LANL	Los Alamos National Laboratory
LBL	Lawrence Berkeley Laboratory
M1	Primary mirror
M2	Secondary mirror
M3	An engineering company in Tucson — *not* a tertiary mirror
MCS	Mount Control System
MOS	Multi-Object Spectrograph
MSBI	Major Systems Baseline Interface
N/A	Not Applicable
NDF	Extensible N-dimensional Data Format
NFS	Network File System
NOAO	National Optical Astronomy Observatories (USA)
OCC	Optical Components Controller (see glossary)
OCS	Observatory Control System (see glossary)
ODB	Observing Database (see glossary)
OPI	Operator Interface workstation (EPICS)
OSS	Optical Support Structure
OT	Observing Tool (see glossary)
PCS	Primary (mirror) Control System
PDF	Packet Description File (see glossary)

PDR	Preliminary Design Review
PPARC	Particle Physics and Astronomy Research Council (UK)
PSF	Point Spread Function
PTC	Primary Thermal Control
PWFS	Peripheral Wavefront Sensor
RA	Right Ascension
RAM	Random Access Memory
RAL	Rutherford-Appleton Laboratory (Chilton, UK)
RDBMS	Relational database management system
RF	Radio Frequency
RGO	Royal Greenwich Observatory (Cambridge, UK)
ROE	Royal Observatory Edinburgh (UK)
ROM	Read-Only memory
RPC	Remote Procedure Call
RTAP	Real Time Applications Platform (a Hewlett Packard real-time database management system)
SAD	Status / Alarm Database (see glossary)
SC	Standard Control System (formerly "Standard Instrument Controller")
SCS[1]	Secondary (mirror) Control System
SE	Sequence Executor (see glossary)
SDD	Software Design Description
SDR	System Design Review
SDS	Self-defining Data System (part of DRAMA)
SIC	Standard Instrument Controller (no longer used – see "SC")
SIR	Status / Information Record
SIS	Sub-arcsecond Imaging Spectrograph
SQL	Structured Query Language (see glossary)

1. Not to be confused with "Standard Control System"

SRS	Software Requirements Specification
SSCL	Superconducting Super Collider Laboratory (Dallas, Texas)
STARCAT	Space Telescope Archive and Catalog
STD	State Transition Diagram (see glossary)
Synchro Bus	Synchronization Bus (see glossary)
TBD	To Be Determined
TBP	To Be Provided
Tcl	Tool Command Language
TCP/IP	Transmission Control Protocol/Internet Protocol
TCS	Telescope Control System (see glossary)
Tk	Tool kit
UDP/IP	User Datagram Protocol/Internet Protocol
UK	United Kingdom
UKIRT	UK Infra Red Telescope (Mauna Kea, Hawaii)
USA	United States of America
UV	Ultra Violet
VLT	ESO’s Very Large Telescope
VME	A real-time system obeying the ANSI/IEEE 1014-1987 Versatile Back-plane Bus standard.
VOS	Virtual Observatory System (see glossary).
VSI	Virtual System Interface (see glossary).
VUI	Visible User Interface (see glossary).
WAN	Wide-Area Network
WFS	Wave Front Sensor
WHT	William Herschel Telescope (La Palma, Canary Islands)
WI	Without Interruption
WiFOS	Gemini’s Wide Field Faint Object Spectrograph
WMKO	William M. Keck Observatories (Mauna Kea, Hawaii)
WYFFOS	WHT’s Wide Field Faint Object Spectrograph

24.3 GLOSSARY

References to other terms in the glossary are given in bold text.

ACK/NAK — Any communication protocol that mandates either a positive or negative acknowledgment to all messages.

Active Optics (aO) — A system which can adjust the shape of its optical surfaces to compensate for gravitational sagging or drifts in the optical alignment. It responds on a time scale of a few seconds to a few minutes.

Actual Configuration — The present configuration of a component of the Gemini System (see **"Configuration"**).

Actual Position — The position of a mechanism in its local coordinate system (corrected for any errors peculiar to the encoding system). This should be equal to the demand position if the mechanism is tracking or stopped in position.

Adaptive Optics (AO) — A system which can adjust its optical components rapidly to correct for atmospheric turbulence or wind shake. It responds on a time scale of a fraction of a second.

Agent — A process which acts as an intermediary between two systems. An agent process may act as an interpreter, translating the protocol understood by one system into that understood by the other. An agent can act as both a **"Client"** and a **"Server"**.

Alarm — Alarms are asynchronous occurrences in the control system that are critical or important to proper operation. For example, an alarm should occur if a power supply fails in the A&G system. Like the power supply example, the items monitored by alarms are often not directly related to the observing activity but can have a profound effect on observing. Alarms can have *warning* or *failure* severity.

Alarm Console — A window on the operator's screen that reports any alarms detected within the Gemini Control System.

Archiving — The saving of the data and information generated by the Gemini Control System in a standard form which can be interrogated and downloaded by future users. One common use of the archive will be to find out if Gemini has been used to observe a particular object or class of object, then download any relevant data.

Astronomer — The person to whom a science program belongs. Usually the Astronomer is same person as the Observer, although a different person can act as Observer on behalf of the Astronomer.

Attribute — An attribute is a textual description of some part of a Gemini based hardware or software system. An attribute has an associated value. (See **"Value")**.

Backus Naur Form (BNF) — A method of defining the relationship between data objects using a series of short-hand symbols to represent combination, choices and repetition. The symbols are

=	"consists of"
+	"and" (sequence)
[|]	"or" (selection)
{ }	"repeating" (iteration)
()	"optional components"
@	"identifier, key attribute"
-ref	"reference to (key of) another data object"
;	"end of sentence"

Bimorph mirror — A deformable mirror consisting of a piezoelectric bimorph wafer coated with a reflective surface, capable of correcting a wavefront in real time.

Bridging (network) — The use of hardware to isolate sections of a network, so messages with both source and destination on the same section are not transmitted to other sections.

Channel Access — A communication mechanism used for accessing the records of a remote EPICS database. A channel access server can make available the contents of its EPICS database to the outside world. A channel access client can read information from or write information to a channel access server's database, or it can register an interest in certain items in that database and be notified when they change. A channel access library is supplied by Los Alamos National Laboratories as part of the EPICS system.

Chopper — A device used to flip the secondary mirror rapidly between two or three predefined positions (the combination is called a "chopping secondary"). It allows a series of observations to be sandwiched between offset sky exposures to compensate for rapidly changing sky conditions. Chopping is especially useful for infra-red observations.

Chopping — The process of taking an observation consisting of pairs of object and sky observations flipped rapidly using a chopping secondary mirror.

Chopping Secondary — See **"Chopper"**.

Client — A process which uses a **"Server"** to carry out some action which it cannot otherwise do itself. The client commands the server and uses it as a system resource.

Coadded frame — Bulk data consisting of several raw frames co-added together.

Command field — The is a field set by the command routines in the TCS in order to instruct a mechanism to move, stop or zeroset.

Command Input Output (CIO) record — An EPICS database record type designed to handle the execution of commands. It includes fields for signalling command completion and status.

Command Layer Server (CLS) — A server provided by the OCS which converts command messages transmitted using IMP into EPICS channel access calls.

Core Instrument Control System — A template Instrument Control System which interfaces with the Observatory Control System but does not control any instrument hardware. In can be used as a template by instrument development teams. See also **"Instrument Control System (ICS)"**

Configurable Control System (CCS) — The Configurable Control System is the hub of the Observatory Control System with many tasks that are related to coordinating the activities of the Gemini Control System during an observing session. It provides the functionality required by the **"Visible User Interface (VUI)"**. In the software design it is the part of the OCS that interacts with the other software systems on behalf of the users.

Configuration — A description of the state of one or more components of the Gemini System. For example, an instrument controller configuration might contain the location of the filter wheel, position of the grating etc… A state described by a configuration can be dynamic. For example an instrument could be continuously changing the state of the chopper in one "configuration".

Configuration Dealer — A process which deals out configurations to the appropriate telescope and instrument queues.

Conforming Instrument — An instrument which conforms to the Gemini hardware and software requirements, i.e. it is based on VxWorks and EPICS.

Console — A graphical window used by the Observer or by an Engineer to interact with some part of the Gemini System.

Context Diagram — A top level **"Data Flow Diagram"** for a system. It shows where the boundaries between a system and the outside world lie and the net data flows into and out of the system.

Control LAN — Local Area Network in the Gemini system devoted to the movement of commands, alarms, and status information.

Control Tool — A tool the operator uses the control the observation procedure and monitor the observatory status. The telescope control console is part of the Control Tool. See also **"Scheduling Tool"**.

Data Handling System (DHS) — That part of the Gemini Control System responsible for displaying, processing, saving and archiving **"Science data"** in a standard way.

Database Access — A low level mechanism for accessing the contents of a local EPICS database directly. Database access is only used within a VxWorks device. A database access library is supplied by Los Alamos National Laboratories as part of the EPICS system.

Data Flow Diagram — A diagram showing how data is handled by a software system. The Gemini software uses the convention of Yourdon and De Marco with the real-time extensions of Ward and Mellor. Basically, a diagram consists of the following components:

Circular bubbles.	=	Processes in the system which handle data.
Arrowed lines.	=	Data flows within the system. Data flows in the direction of the arrow. A double headed flow means information flows both ways.
Rectangular boxes.	=	“Terminators”, or things in the outside world which provide or receive data.
Parallel lines.	=	Data stores, or places in the system where data may be stored and retrieved at a later time.
Dotted circular bubbles.	=	Control processes in the system, which handle events and control other processes.
Dotted arrowed lines.	=	Control flows within the system. These do not carry data, but carry only the knowledge that an event has happened. They may also be used to enable or disable a process.
Dotted parallel lines.	=	Event stores, or places in the system where an event may be stored and used to trigger a process at a later time.

Data LAN — Local Area Network in the Gemini system devoted to the transfer of bulk data.

Data-loss compression — Any compression algorithm that results in the loss of original data (resolution) after uncompression. Typical compression savings from these algorithms depend upon the amount of data loss that is acceptable, but can exceed 99%. Also see **“Loss-less compression”**.

Data Reduction — In the Gemini documentation this refers to the removal of telescope, instrument and atmospheric features from a set of data. See **“On-line Data Reduction”** and **“Off-line Data Reduction”**.

Data Reduction Queue — A first-in-first-out queue in the Data Handling System which enables observations to be reduced asynchronously without holding up data acquisition.

Data Reduction Recipe — A recipe describing how to reduce a particular type of data.

Data Reduction Rule — A rule describing how to search for a calibration observation of a particular type during data reduction.

Data Storage Server — A process which receives data from a remote client and stores it to disk on behalf of the client. (See **"Client"** and **"Server"**).

Data Structure Diagram — A diagram showing the contents of a piece of data, typically used to reveal the make-up of a data flow or show the contents of a data store. The diagram shows a hierarchical tree structure with each branch in the tree being made from the components below it. The conventions of Yourdon and De Marco are used. In the Gemini software design these diagrams are used internally, but they are converted to **"Backus Naur Form (BNF)"** when presented in the software design documents. Data structure diagrams consist of the following components:

Rectangular box.	=	A data item.
Lines connecting a box to one or more boxes below it.	=	The upper box is made out of the data items represented by the lower boxes.
Rectangular boxes with a circle in one corner.	=	The upper box is made out of a choice of one of the boxes with the circle in the corner.
Rectangular boxes with an asterisk in one corner.	=	The upper box is made out one or more copies of the box with the asterisk in the corner.

Data Transport — The act of removing data from the local disks at the summit to a more permanent location. It involves giving the Observer a copy of the data as well as copying the data to a remote archive site. See **"Alarm Console"**.

Demand Configuration — The configuration which a component of the Gemini System has been requested to match. See **"Configuration"**.

Demand position — The desired position for a mechanism in its local coordinate system, either updated continuously or set to a constant value if a mechanism is to be moved to a given position and stopped.

Detector Array Controller (DC) — That part of an **"Instrument Control System (ICS)"** responsible for sequencing and reading a detector array.

Dithering — The process of taking a series of observations with the telescope beam stepping through a sequence of pre-defined positions arranged to obtain an overlapped mosaic with small steps. See also **"Nodding"**.

Drifting — An observing mode where a series of observations are made with the telescope tracking switched off and the sky allowed to drift past. *Need to check this definition.*

Eavesdropping — Allowing an observer to watch an observation from a secondary site

Engineering Console — A console used mainly by engineers to take direct control of the EPICS system controlling an instrument or telescope subsystem. See **"Console"**.

Entity Relationship Diagram — A diagram showing how the various items of information used within a software system are related together. As an example an entity relationship diagram could be used to represent the statement

"Every science data file has one header and one header only. Each header contains one or more FITS items. Each FITS items is derived from an entry or combination of entries in the OCS, TCS and ICS configurations."

graphically. The conventions of Yourdon and De Marco are used. Entity relationship diagrams consist of the following components:

Rectangular box.	=	A data item.
A diamond shaped box.	=	A type of relationship.
Lines connecting rectangular and diamond shaped boxes.	=	The rectangular boxes are related by the relationship shown in the diamond shaped boxes. Symbols labelling the lines show whether the relationship is "one-to-one" or "many-to-one" etc.…
Rectangular boxes with an arrowed line pointing to a diamond shaped box.	=	These represent data items which are associated with the relationship.

EPICS Logging Record — An EPICS record used for recording logging information.

Event bus — A bus used to trigger and sense events on the Gemini System (such as a change in the chopper state). The implementation of this bus is TBD, pending a Standard Control System work package trade study. It is likely to support three or more states.

Experimental Physics and Industrial Control System (EPICS) — A database-driven, real-time control system originally co-developed by the Los Alamos and Argonne National Laboratories.

Epoch — Where a star has detectable proper motion, you need to specify the time at which the star was at the given position, and this is called the "epoch". Catalogs usually give star positions at the epoch which is the same as that for the mean equator and equinox, for example B1950 or J2000 (the B and the J are just slightly different ways of reckoning time). In such cases one says that the positions are for "equinox B1950 epoch B1950", or "equinox and epoch B1950". However, if one observed a star in (say) the middle of 1986, and did the astrometry relative to FK4 stars, then its position would be given as RA whatever Dec whatever "equinox B1950 epoch J1986.5".

Equinox — Star positions are usually expressed as "mean" RA,Decs. The equator and equinox of the RA,Dec system are, inconveniently, in gradual motion due to precession and nutation. Precession is the slow and steady part of this motion, and the "mean" equator and equinox are affected only by the precession component. When one gives a mean RA,Dec, one specifies the epoch and equinox it refers to by giving the date, or "epoch", that you feed into the precession formulas. The phrase "referred to the mean equator and equinox of epoch J2000" is, by convention, shortened to "equinox J2000".

Eric's — A place frequented by members of the Gemini 8m Telescopes project at lunch time.

Error Budget — The contribution to the image quality error which each component in the Gemini system is allowed to make.

Facility Instrument — An **"Instrument"** which is designed specifically for use on the Gemini telescope, or a pre-existing instrument which is adapted for use on Gemini. Facility instruments are the ones which are generally available to observers.

Fast guiding — The ability of the secondary mirror to move independently of the primary to keep a target appropriately positioned.

Flexible Scheduler — *Description to be provided.*

Fine Grained Commands — *Description to be provided.*

FITS format file — A file obeying the Flexible Image Transport System standard.

FITS header — A collection of header items whose names and values conform to the FITS standard. The items must be scalar and identified by a unique 8 character keyword. In the Gemini documentation the term “FITS header” is used to refer to those parts of the transmitted bulk data which will become the **“True FITS header”** (see below) when they are stored in a FITS format file.

FITS header information — This term is used in the Gemini software documentation to refer to the information to be collected together to form a FITS header. FITS header information is treated by the Observatory Control System as a set of **“Attribute”/“Value”** pairs.

Following — In a Telescope Control System context, when a mechanism is “following” it is updating its position continuously from a stream of demanded positions, velocities and accelerations (or whatever).

Handshaking (command) — A software protocol where subsystems acknowledge acceptance of a command. Unlike **“ACK/NAK”**, handshaking acknowledges acceptance of the message contents.

Health — Health is the highest level of status presented by the visible interface. It is defined to be a component’s well-being as determined by the component itself. Health can have one of three values: *good* means *I am normal, everything is okay*; *warning* means *I am operating but not normally*; and *bad*, which means *I am not operational*. Health should be viewed as a predefined mandatory type of alarm.

Health is defined recursively — a component is in good health only if all the components that it manages and relies upon are healthy too. By logically combining the health of all systems it is possible to find out if the entire telescope system is healthy.

Hierarchical data structure — A data structure built like a Unix directory structure, where related items of data can be grouped together into structures, and the structures themselves can be grouped into more complicated structures. The whole data structure resembles a tree, with the data units themselves appearing as leaves on the tree.

In-position radius — The in-position radius defines the tolerance within which observing is possible for a moving mechanism. If the position error is less than this value, then the mechanism is said to be ‘tracking’.

Interactive Observing — A mode of observing where each of the actions performed in the observing process are initiated directly by the individuals involved in the observing process. The next action in any sequence is determined just before it is executed.

Interlock System — A safety device used to lock out parts of the Gemini Control System when it would be unsafe to operate them.

Integrated Services Digital Network (ISDN) — A communications protocol which allows two or more ISDN-compatible devices to communicate using voice, video or packet information across a telephone line.

Instrument — A device, consisting of a detector and a means of dispersing and/or focussing light onto that detector, designed to extract a particular kind of information from the light gathered by the Gemini telescope. A **"Scientific Instrument"** is specially designed for astronomy research. The wavefront sensors on the Gemini telescope can be regarded as a kind of instrument.

Instrument Control System (ICS) — That part of the Gemini Control System which is responsible for controlling and operating a **"Scientific Instrument"**. There is a separate Instrument Control System for each available scientific instrument.

Instrument Sequencer — That part of an **"Instrument Control System (ICS)"** responsible for co-ordinating the actions of the **"Detector Array Controller (DC)"** and **"Optical Components Controller (OCC)"**.

Isoplanatic Patch — That small portion of the sky over which improvement of the image quality by wavefront correction of the light from a reference star is effective. Its size depends on the degree of correction required.

Limits — Software limits on position (positive and negative), velocity and acceleration which can be set for a mechanism.

Logging — The act of keeping a record of happenings during an observing session. "Logging" encompasses the observing log, system history records and any engineering logs made for the day crew.

Logging Client — A piece of hardware or software which is interested in the log generated by another piece of hardware or software.

Logging Device — A piece of hardware or software that generates a log that is of interest to a **“Logging Client”**. A Logging Device owns one **“Public Status Interface”**.

Loss-less compression — Any compression algorithm that results in no data loss on uncompression. Typical compression savings with loss-less compression are 25-75%. Also see **“Data-loss compression”**.

Major System — An obsolete name for **“Principal System”** which was removed because of a possible confusion with **“Major System Interface”.**

Major System Interface — Used in other Gemini documentation to refer to the interface between parts of the Gemini system built by separate groups. This is not a software interface but should not be confused with **“Principal System Interface”**, which is a software interface.

Membrane Mirror — A variable-curvature mirror consisting of a stretched membrane with a reflective coating.

Modal Correction — The correction of wavefront errors using orthogonal modes of an adaptive mirror, some of which are associated with well-known telescope aberrations, using a variable gain and bandwidth for each mode.

NAXIS — The number of axes a particular data set has. A 1-D spectrum would have NAXIS=1, a 2-D image NAXIS=3 and a 3-D data cube NAXIS=3. SDS can handle data up to NAXIS=7. NAXIS is also a standard FITS keyword having the same meaning.

Near-line Data Reduction — See **“On-line Data Reduction”.**

NDATA — The number of separate sub-arrays making up a complete data array. NDATA is *not* FITS keyword.

Nodding — The process of taking a series of observations with the telescope beam stepping through a sequence of two or three pre-defined positions. Nodding is normally associated with chopping, with the pre-defined positions arranged to be the same distance apart as the chopper throw. See also **“Dittering”** and **“Wobbling”**.

Non-conforming Instrument — An instrument which does not conform to the Gemini hardware and/or software requirements. Such an instrument is only allowed as a **"Visitor Instrument"**.

Observatory Control System — That part of the Gemini Control System responsible for coordinating the activities of the observatory. It interacts with the observer and operator and issues commands to the other Principal Systems.

Observer — The person responsible for ensuring the scientific objectives of a science program are met.

Observing Database (ODB) — A real-time database used by the Observatory Control System that contains information on the observations that are executing in the control system and the current state and status of the observatory and its operations.

Observing Modes — The "styles" of observing supported by the Gemini Control System

Observing Pool — A pool of pending observations.

Observing Tool — The tool used by the Observer to create and submit science programs and open consoles.

Off-line Data Reduction — This term described the **"Data Reduction"** carried out outside the scope of the Gemini System by the observer using an external data reduction package. Off-line reduction will be used by the observer to generate publication quality reduced data. In some situations this process will be made easier because the observer can make use of the data supplied by the on-line data reduction.

On-line Data Reduction — This term describes the semi-automatic **"Data Reduction"** which is carried out as soon as possible after acquiring the data. It is designed to allow the observer to make scientific decisions and not to produce data for final publication (as this may need calibrations which are unavailable at the time of observation or further refinement of the reduction process). On-line data reduction is also referred to as "near-line" data reduction because, although it is not off-line, it is not completely synchronous with data acquisition, which is allowed to proceed if there are delays in the data reduction.

Operator — The person responsible for the safety and integrity of the Gemini System, and for scheduling science programs to make efficient use of the resources available.

Optical Components Controller (OCC) — That part of an **"Instrument Control System (ICS)"** responsible for controlling the environment inside and optical path through an instrument.

Packet Description File — A file describing giving instructions on how to obtain a "packet" of information for a **"FITS header"**. A packet is a collection of FITS header items which logically belong together (e.g. all the parameters for one instrument). The file contains a list of FITS keywords and a location where the value of each FITS keyword can be obtained on the Gemini system. It may also contain a description of the time at which each FITS item should be sampled. The PDF concept was originally used on the William Herschel Telescope.

Plan View — The plan view is a graphical display of the observations and configurations that have been submitted to the **"Configurable Control System (CCS)"** for execution. The plan view shows how much time each observation should take and the ordering of the observations during the observing session.

Planned Observing — A mode of observing where the Observer plans the observations before the observing session starts. The plan is submitted to the system, scheduled by the Operator and then executed by the system.

Planner — A software-environment for developing observing software.

Primitive data type — The type of data representing a "leaf" in a hierarchical data structure. A primitive data type will be one of the types recognized by programming languages (such as "integer", "floating point", "integer array" etc…

Principal System[1] — At the highest level in the Gemini Control System decomposition, the software is divided into four kinds of software systems called "principal systems". The four types are:

- Observatory Control System
- Data Handling System
- Telescope Control System
- Instrument Control System.

1. Not to be confused with the term **"Major System"**, which is used for describing various kinds of work package.

There may be up to four concurrently executing Instrument Control Systems. A Principal System may be developed by one or more work packages. See **"Work Package"**. Not to be confused with **"Major System"**.

Principal System Interface — A software interface between two Principal Systems.

Private Status Item — A **"Status Item"** that is made available only with the permission of its owning **"Status Device"**.

Public Status Interface — That part of a status device's database which is available to any status client in the outside world. (See **"Status Device"**).

Public Status Item — A **"Status Item"** that is made available to any **"Status Client"**.

Quick Look Display — A display showing the latest frame obtained from the instrument, giving the observer an immediate indication of the quality of the data obtained.

Quick Look Display Server — A process which manages a quick look display and receives commands and data to display from a remote client. (See **"Client"** and **"Server"**).

Raw frame — Bulk **"Science data"** as generated by the Detector Array Controller, before it is processed by the Instrument Control System. A raw frame consists of data only, with no header information.

Raw science data — Bulk **"Science data"** as generated by the Instrument Control System, before any substantial data reduction. Raw science data consists of a data frame plus some instrument-specific header information.

Reduced science data — Bulk **"Science data"** generated by processing **"Raw science data"** with a data reduction recipe.

Reflective Memory — Random Access Memory which is copied into the address space of more than one CPU. Any change to one copy of the memory is automatically reflected in the other copies using signals transmitted along a fibre optic link. See also **"Synchronization bus".**

Remote observing — An observing session where the Observer is at a remote location instead of being in the control room with the Operator. The Operator is always in the control room.

Scenario — An example observing session conceived to test out the capabilities of the software design.

Scheduling System — The portion of the Gemini Control System that is responsible for dispatching Science Programs to the Sequencer for execution.

Scheduling Tool — A tool used by the Operator to schedule science programs. See also **"Control Tool"**.

Science Configuration — A Science Configuration is part of a Science Configuration. It describes the state of the Gemini Control System as it should be configured to allow the acquisition of one or more frames of science data.

Science data — This name refers to any bulk data in the Gemini Control System which contains scientific information required by the observer. It includes all the calibration frames, sky images, file headers, etc.… but does not include any commands, status, configuration or alarm information.

Science Observation — A Science Observation is part of a Science Program. A science observations consists of an ordered, related set of Science Configurations and additional information that is shared among all the observation's configurations.

Science Program — A formal description of an observer's plan for using the Gemini 8-m Telescopes, suitable for near-automatic execution. It usually consists of an unordered group of Science Observations.

Scientific Instrument — An **"Instrument"** specially designed for astronomy research.

Selected Program Pool — A dynamic set of observations which are available for viewing or inclusion in a plan.

Sequence Executor — An application that is part of the Configurable Control System in the Observatory Control System which executes a sequence recipe.

Sequence Recipe — A recipe of sequencer commands describing how to execute one science configuration.

Sequencer — The command interpreter for Science Programs and telescope control

Sequencer Commands — The commands issued by the sequencer to the Gemini major systems.

Sequencing System — The component of the Configurable Control System that manages the execution of observations.

Server — Any process which carries out some form of service for a client. Typically, a server may be on a remote machine and can therefore access things which the client can't. A server is completely subservient to a client. It responds to commands from the client but cannot itself initiate a dialogue with a client.

Service Observing — An observing session made by local staff on behalf of a remote Astronomer.

Servo constants — Some mechanisms in the TCS may require servo parameters to be adjusted and/or initially set depending on the type and range of motion desired.

Severity — Alarms and errors can have one of two severity levels. *Warnings* are alarms or errors that subsystems feel are important and should be brought to the attention of the user. A component should be able to continue working in spite of any events that cause warnings. Warnings can be acknowledged by users allowing operations to continue.

A *failed* severity indicates that a component is in a state that will not allow it to continue operations. The user or operator must solve the problem before continuing operations. Failed alarms or errors will not go away when acknowledged.

Standard Control System — The work package for providing drivers for observatory hardware which are not provided as part of the standard EPICS release (formerly the "Standard Instrument Controller").

Standard Instrument Controller — An old name for the **"Standard Control System"**.

State Transition Diagram — A diagram showing how an system behaves and changes its state in response to external events. The conventions of Ward and Mellor are used. State transition diagrams consist of the following components:

Rectangular box.	=	A particular system state.
Arrowed lines.	=	Allowed changes in the system state. An arrowed line emerging from the top edge of the diagram points to the initial state.
Horizontal lines and labels attached to an arrowed line.	=	The label above the line describes the external events which cause that change of state.
		The label below the line describes the actions carried out by the system when the change of state occurs.

Status — Status consists of the values hardware devices and software systems present to other hardware and software systems. These attributes are what define the state of a system. A component's *state* includes the attributes related to the purpose of the component, but will also include system related status attributes such as its operational state (busy, done).

Status/Alarm Database (SAD) — A special EPICS database which allows non-EPICS systems to communicate alarm, status and logging information without requiring a host-level channel access server.

Status Client — A piece of hardware or software which is interested in the state of another piece of hardware or software.

Status Console — A window on the operator's screen that reports information about the status of the Observatory.

Status Item — One piece of information provided by a **"Status Device"**.

Status Device — A piece of hardware or software that has state or information that is of interest to a **"Status Client"**. A Status Device owns one or more Status Items.

Status field — A field in the TCS which records whether a mechanism is: moving or stopped, following or not following, in position or not and zeroset or not.

Stopping radius — This is the tolerance within which a mechanism is set by a command to move to position and stop.

Strehl angle — The equivalent width of a star image calculated by modelling that star image as a perfect cylinder of height equal to the peak intensity of the image and area equal to the total photon count.

Strehl ratio — The ratio of the maximum intensity obtained from a point source image to the maximum intensity of a perfect diffraction-limited image.

Structured Query Language (SQL) — A standard language, common in commercial database management systems, used for specifying how to search a database for items matching certain criteria. The criteria can involve arithmetic and logical combinations of record properties (tuples).

Subsystem — A software system making up a well-defined unit within one **"Principal System"**. For example, the Telescope Control System contains the following subsystems:

- Acquisition and Guidance subsystem
- Adaptive Optics subsystem
- Cassegrain Rotator Control subsystem
- Enclosure Control subsystem
- Mount Control subsystem
- Primary Control subsystem
- Secondary Control subsystem

and the Instrument Control System contains the following subsystem:

- Detector Array Control subsystem

A subsystem is often the responsibility of one **"Work Package"**.

Synchronization bus — A fast bus used for the rapid exchange of wavefront sensing information, implemented as VMIC VMIVME-5578 Reflective Memory. This bus is used for any transactions which are too rapid for the Data LAN.

System — This word has two possible meanings in the Gemini software documentation:

(1) A system is a software or hardware entity with a well-defined function. Referring to something as a "system" in this way does not imply any particular hierarchy For example the "Data reduction system" is one part of the "Data Handling System".

(2) A piece of work with well-defined boundaries which can be issued as a work package (e.g. **"Standard Control System"**).

Telescope — In the Gemini software documentation the word "Telescope" is used to refer to the Gemini 8m telescope together with all of its associated peripheral devices. These include the telescope mount, the primary and secondary mirrors, together with their support and control structures, the Cassegrain rotator, and all the devices associated with acquisition and guiding and with the adaptive optics. Some of the latter devices can also be regarded as instruments.

Telescope Control System (TCS) — That part of the Gemini Control System which is responsible for controlling and operating the telescope and all its associated peripheral devices. It is also responsible for monitoring the environment.

Time bus — A bus used to distribute a time signal to all the Gemini systems.

Tip/Tilt — The process by which an image is moved in the focal plane by tilting the secondary mirror with respect to the optical axis.

Tip/Tilt Mirror — A servo-positioned mirror capable of high frequency 2-axis motion. A tip/tilt mirror may be used by the AO system to stabilize a star image by removing the second and third **"Zernike"** terms of the detected wavefront.

True FITS header — A FITS header as written to a FITS format file. This consists of a series of 80-byte ASCII card images. Each record contains an 8 character keyword identifying the item, the value of the item encoded as ASCII characters, and an optional comment describing the item. See **"FITS header"**.

Unix — A standard host-level operating system. The Gemini baseline for "Unix" is "Solaris" marketed by Sun Microsystems.

Value — The term "value" is used in the Gemini software documentation to refer to the data associated with a particular attribute (see **"Attribute"**).

Virtual Observatory System (VOS) — A model for designing the Gemini Control System that seeks to minimize both intersystem dependencies as well as hide subsystem implementation details from system level operations.

Virtual System Interface (VSI) — A layer of interface abstraction within the Virtual Observatory System that serves to isolate and hide implementation-specific interface requirements.

Virtual telescope interface — An interface to a "virtual telescope", which obeys every slew and offset command instantly. The real telescope attempts to match the configuration of the virtual telescope and catches up with it some time later.

Visible User Interface (VUI) — The Graphical User Interface tools used by the Observer and Operator.

Visitor Instrument — A self-contained **"Instrument"** which is brought to the Gemini telescope for temporary use.

VxWorks — A real-time operating system for VME hardware marketed by Wind River Systems.

Walk-through — A description of how the Gemini Control System would behave while carrying out an observing scenario.

Wobbling — Another name for **"Nodding"**.

Work Package — A well-defined unit of work, consisting of the development of a subsystem or a layer of utilities, which can be devolved to a collaborating group.

Zernike — A polynomial used to describe the shape of a wave front.

Zonal Correction — The correction of wavefront errors by a simple mapping of individual wavefront sensors onto corresponding individual elements of an adaptive mirror.

A1 Decomposition/ Dependency Descriptions

Introduction

This appendix provides the formal presentations of the decomposition and dependency descriptions for the Gemini 8-m Telescopes Control System (GCS). Only the top levels of the design are shown here, as the lower level design descriptions are the responsibilities of the Work Package Developers.

Top-level Diagrams

Figure A1-1 shows the environment in which the GCS operates and the types of interfaces between the GCS and this environment. Figure A1-2 shows the initial breakdown of the GCS into four basic systems:

- *Observatory Control System* - responsible for the interactions between the users of the system and the other subsystems - the OCS performs the scheduling and sequencing of the activities involved in performing observations
- *Instrument Control System* - responsible for the science instrumentation available on the Gemini telescopes
- *Telescope Control System* - responsible for control of the telescope and all associated subsystems
- *Data Handling System* - responsible for the movement of data, quick-look facilities, and interfacing to an external data reduction system

FIGURE A1 - 1. Top-level context diagram for Gemini Control System

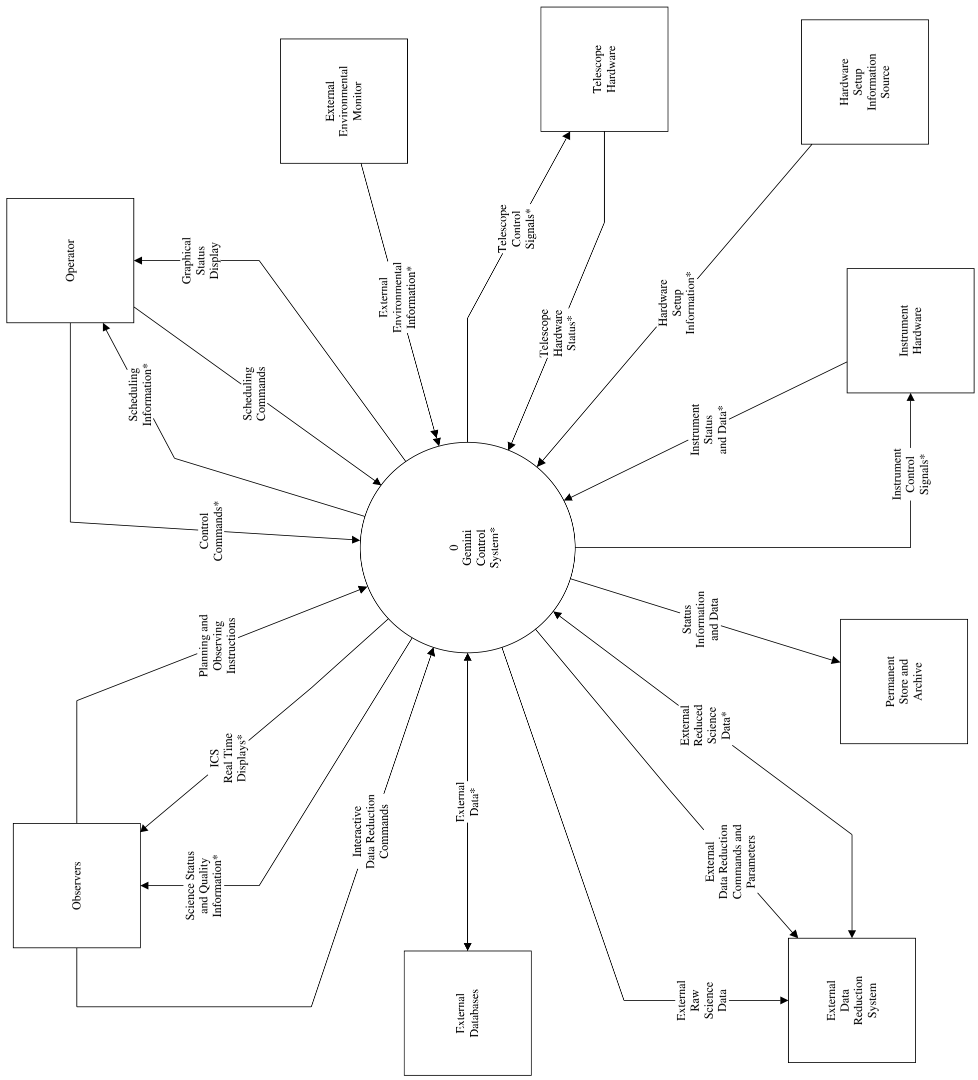
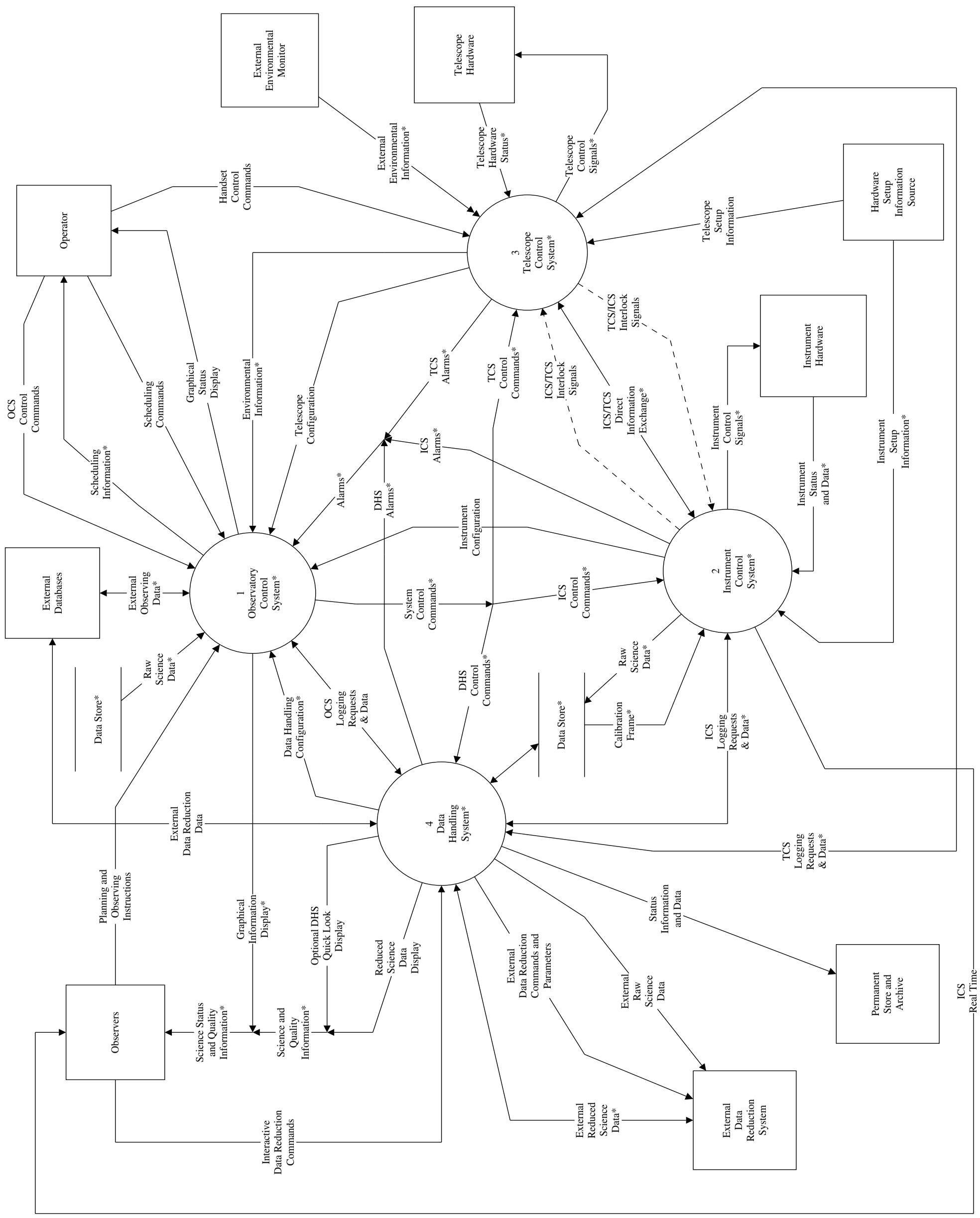


FIGURE A1 - 2. Principal system components



OBSERVATORY CONTROL SYSTEM

FIGURE A1 - 3.Major components of the OCS

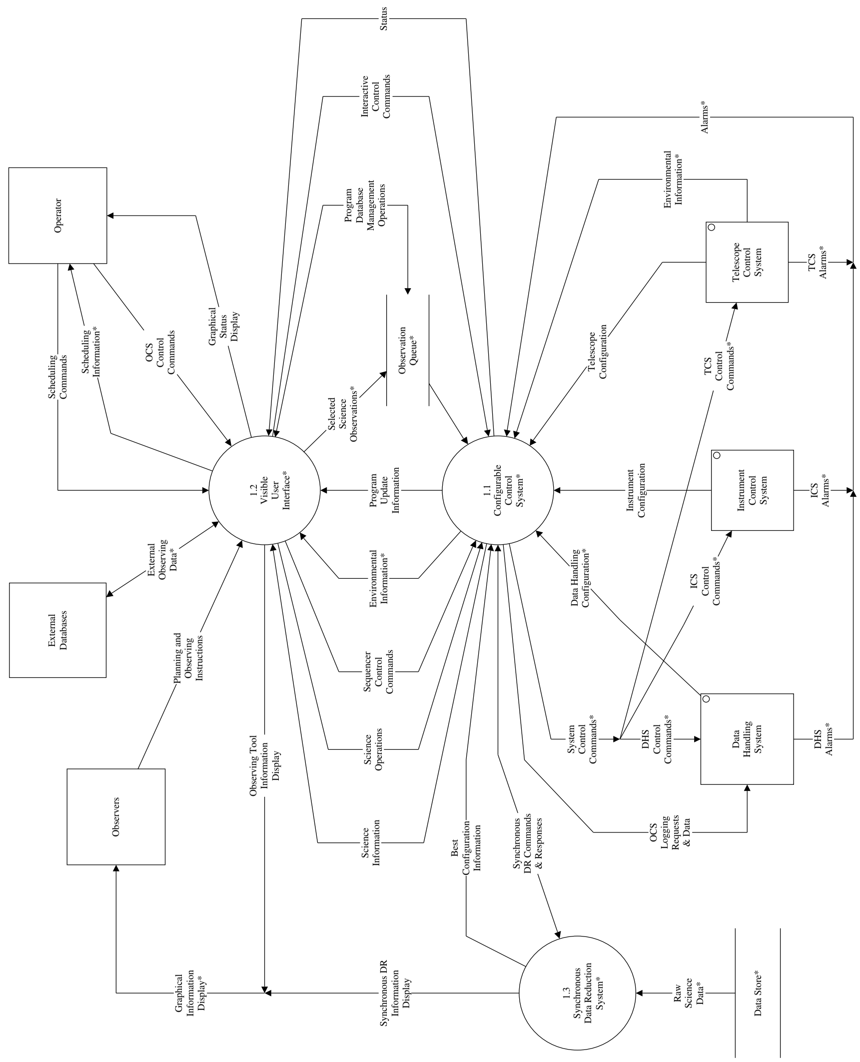


FIGURE A1 - 4. **The Configurable Control System**

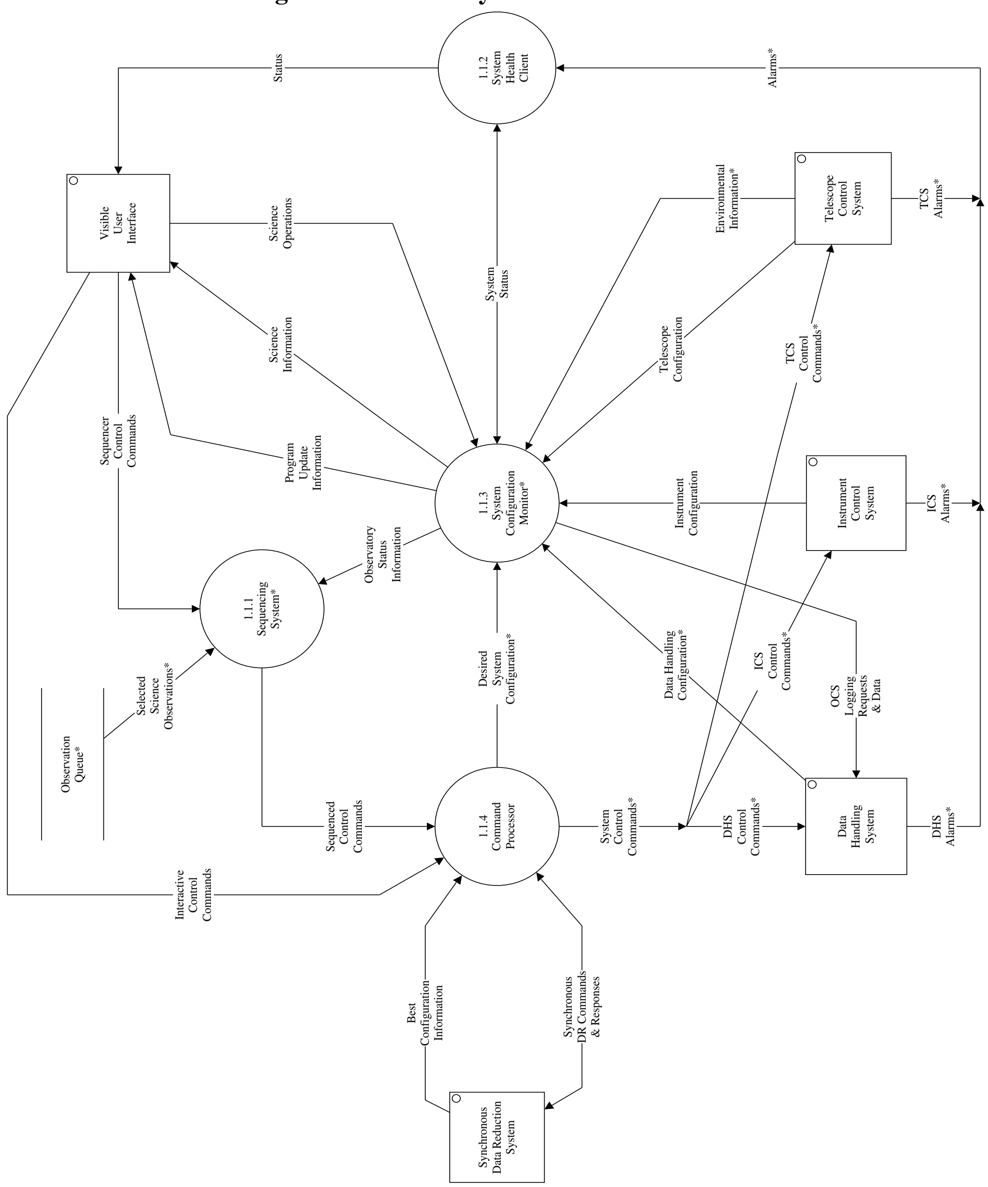


FIGURE A1 - 5. The Sequencing System

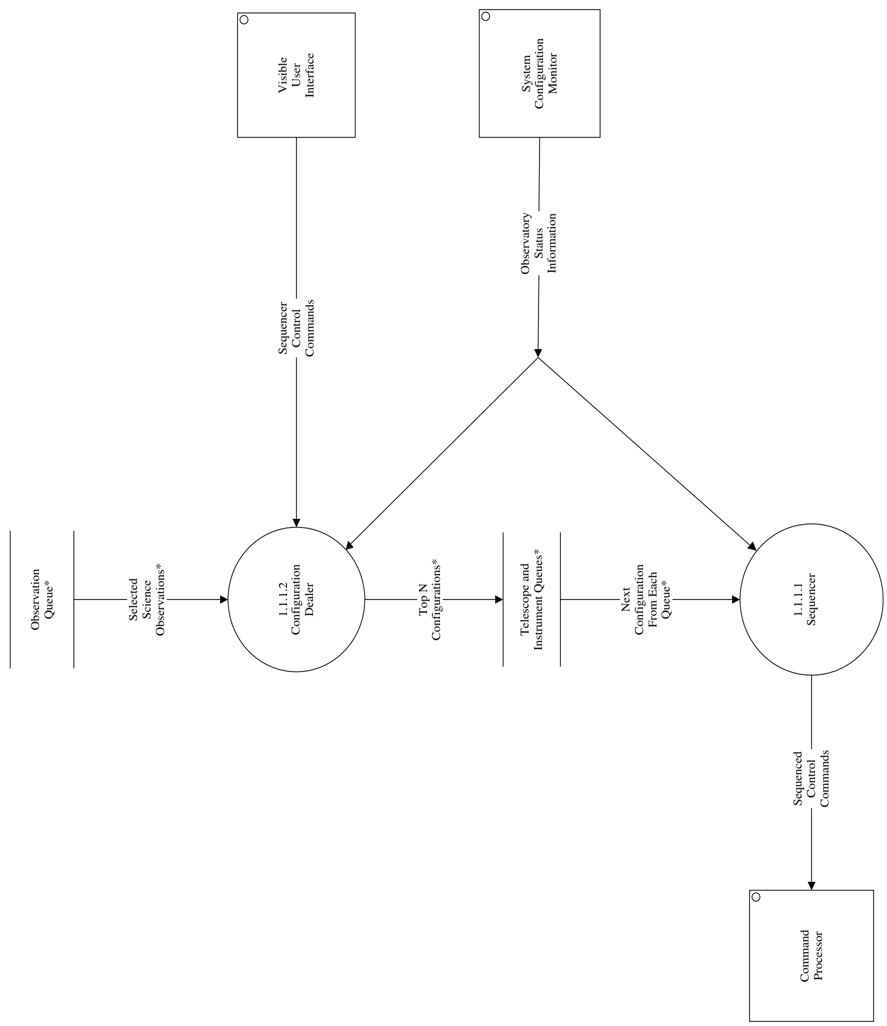


FIGURE A1 - 6. The System Configuration Monitor

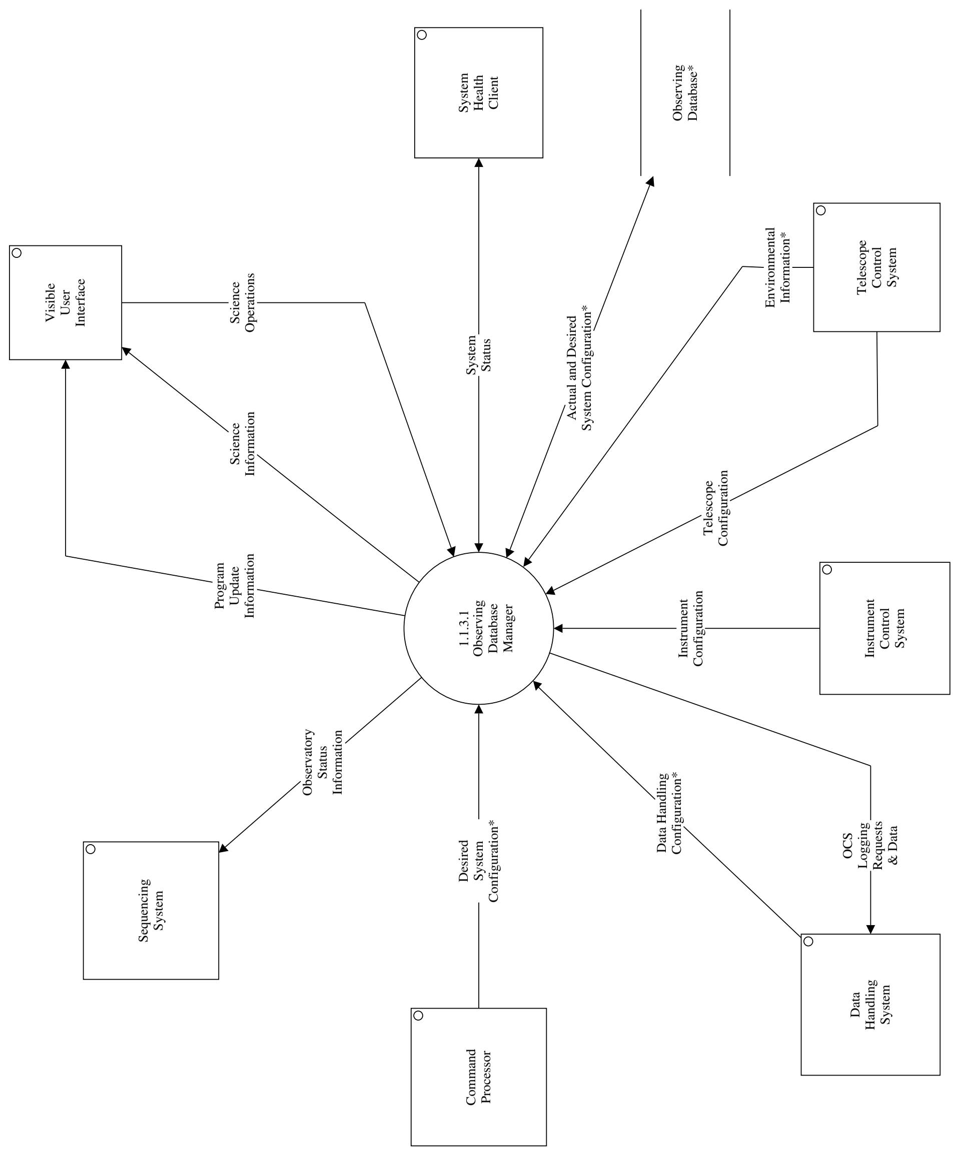


FIGURE A1 - 7. The Visible User Interface

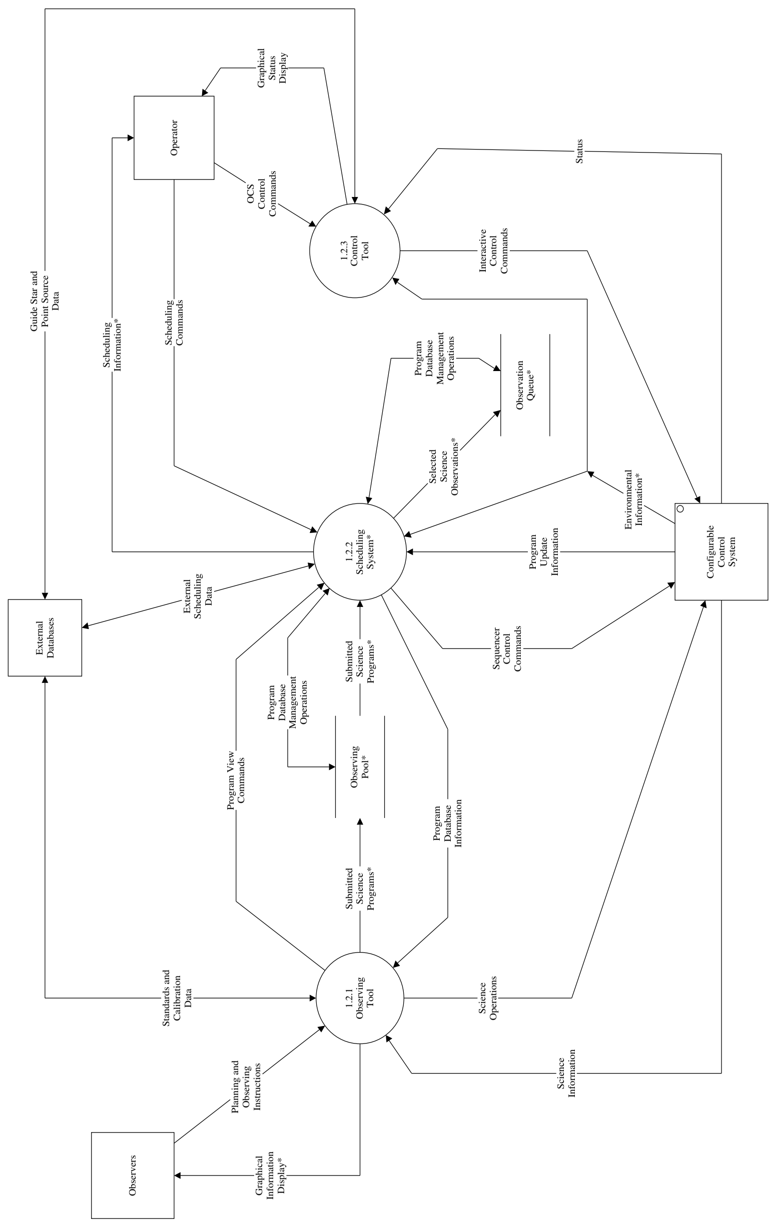
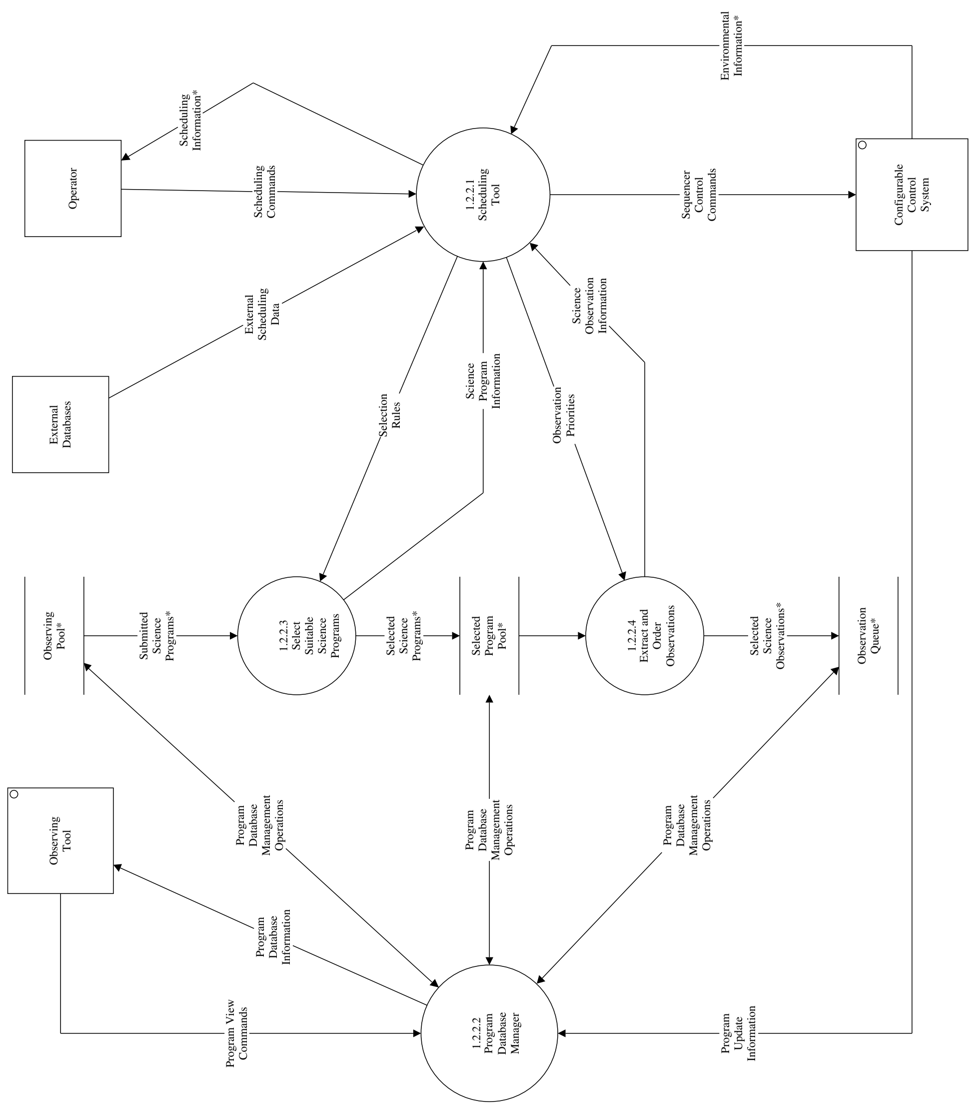
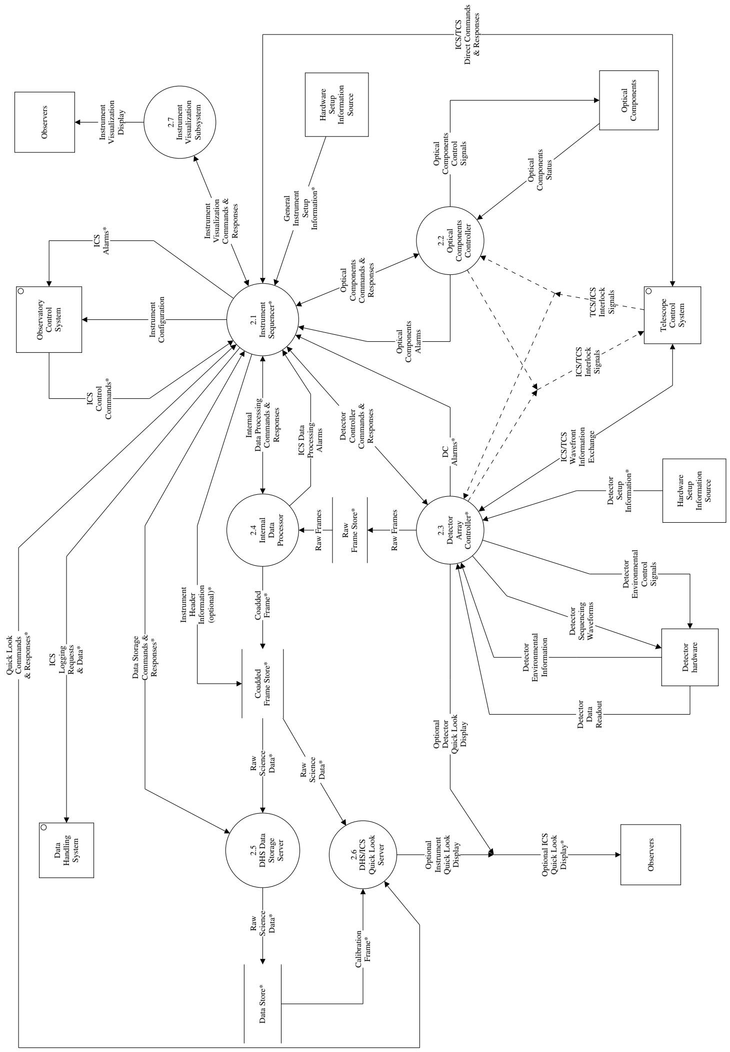


FIGURE A1 - 8. The Scheduling System



INSTRUMENT CONTROL SYSTEM

FIGURE A1 - 9. Components of the Instrument Control System



TELESCOPE CONTROL SYSTEM

FIGURE A1 - 10. Components of the Telescope Control System

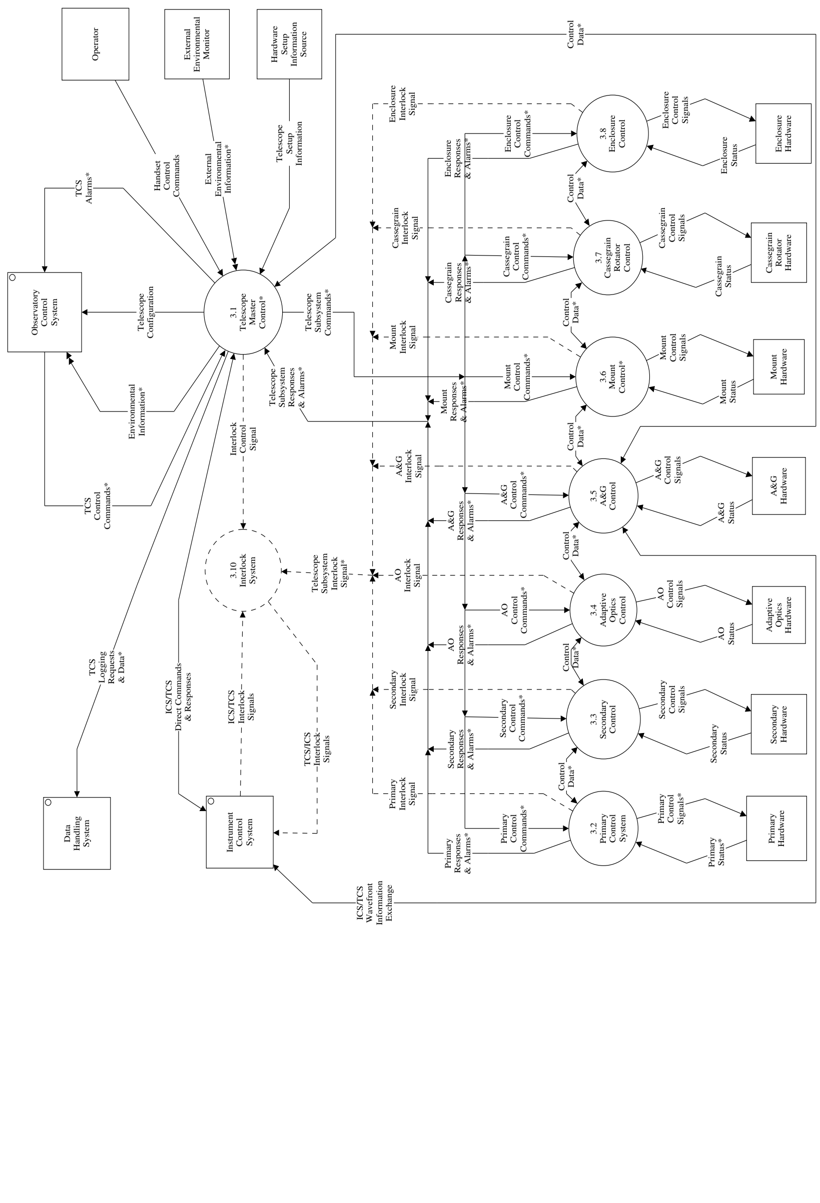
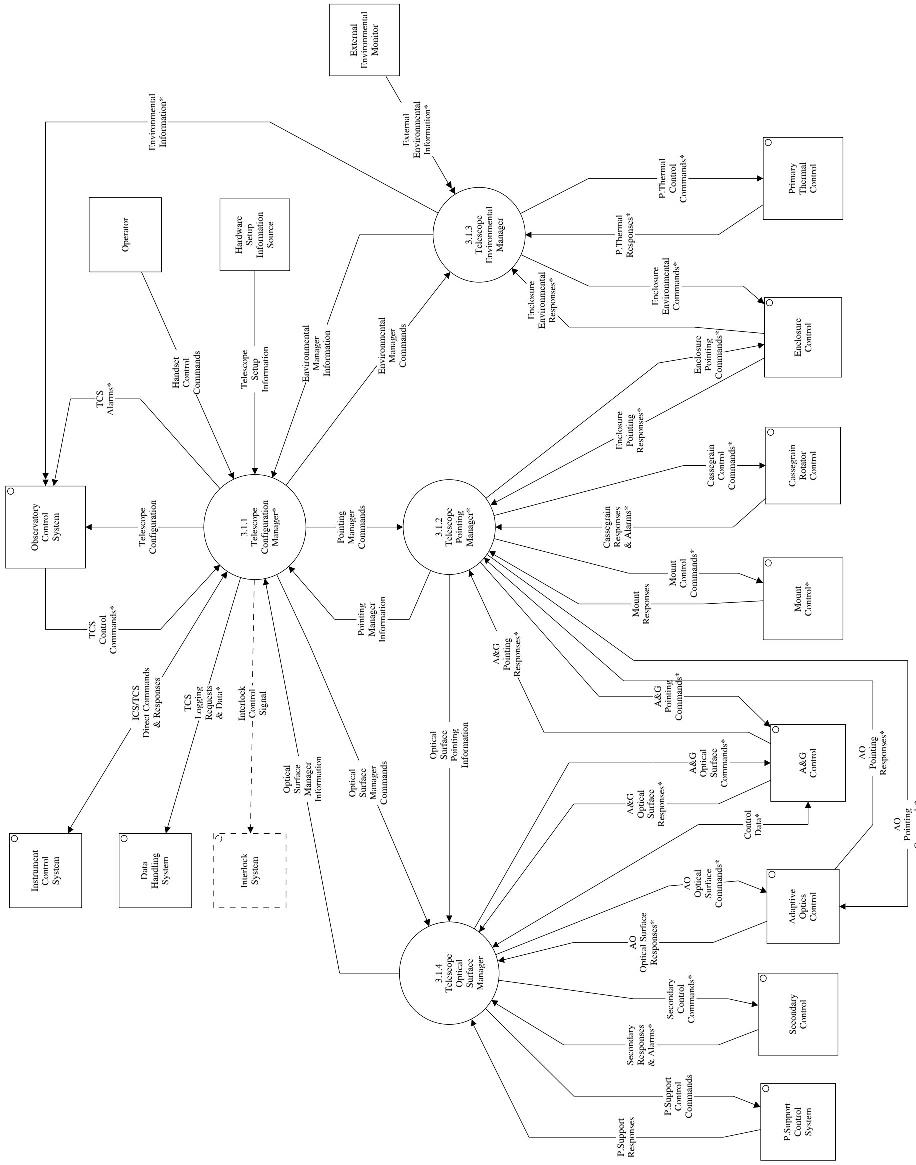


FIGURE A1 - 11. **Telescope Master Control**



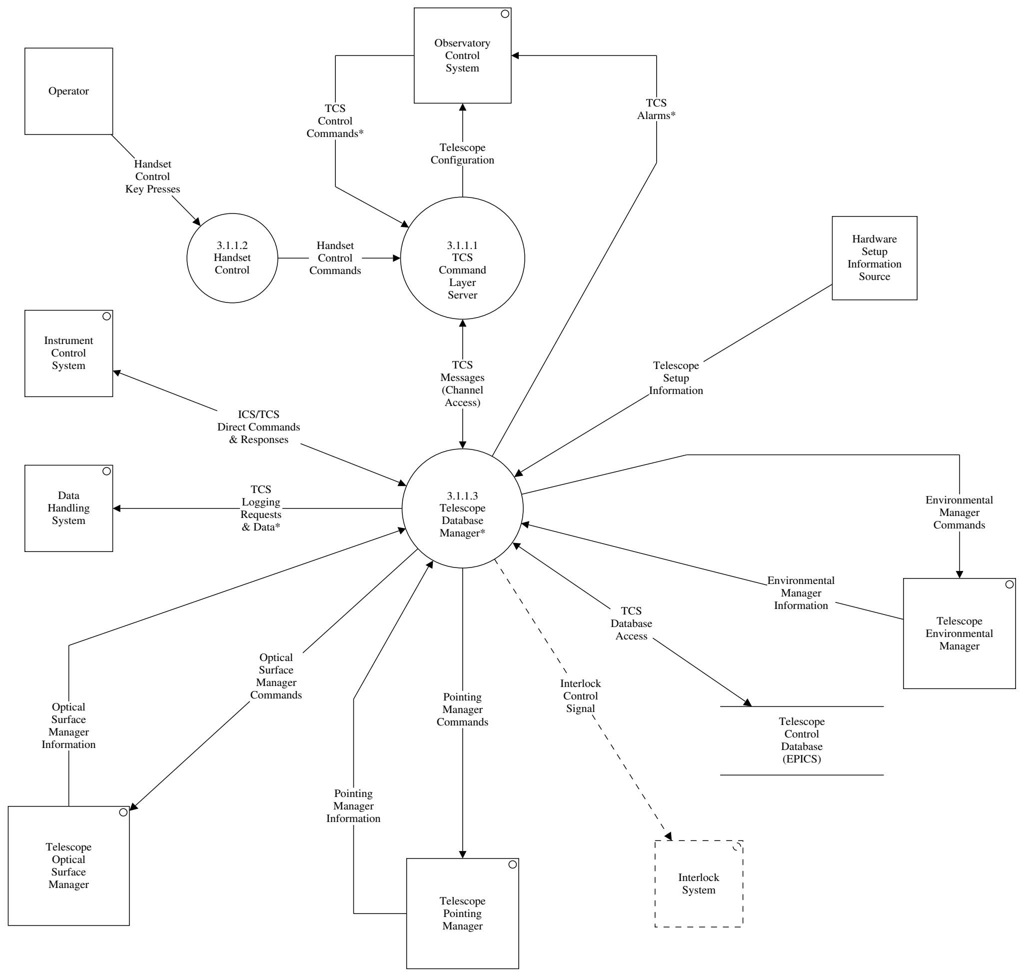
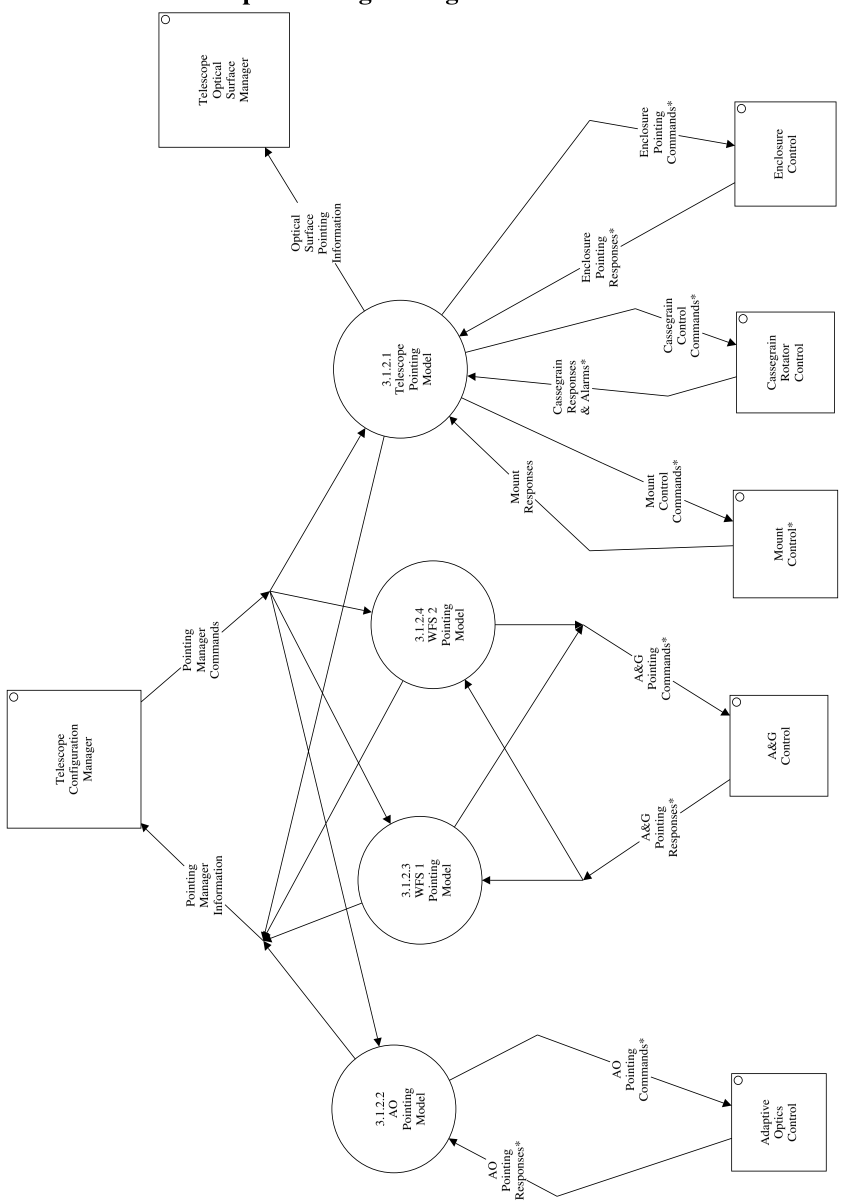


FIGURE A1 - 12.The Telescope Command Processor

FIGURE A1 - 13. **The Telescope Pointing Manager**



DATA HANDLING SYSTEM

FIGURE A1 - 14.**Components of the Data Handling System**

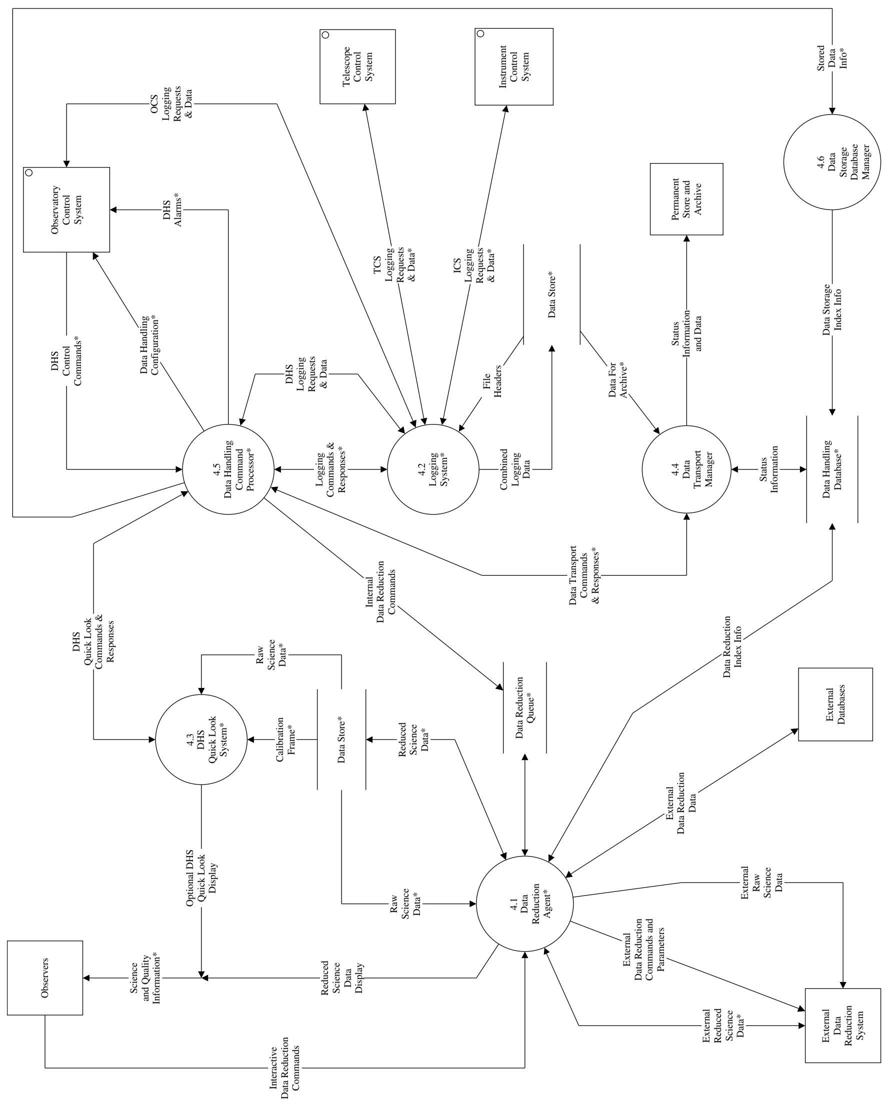
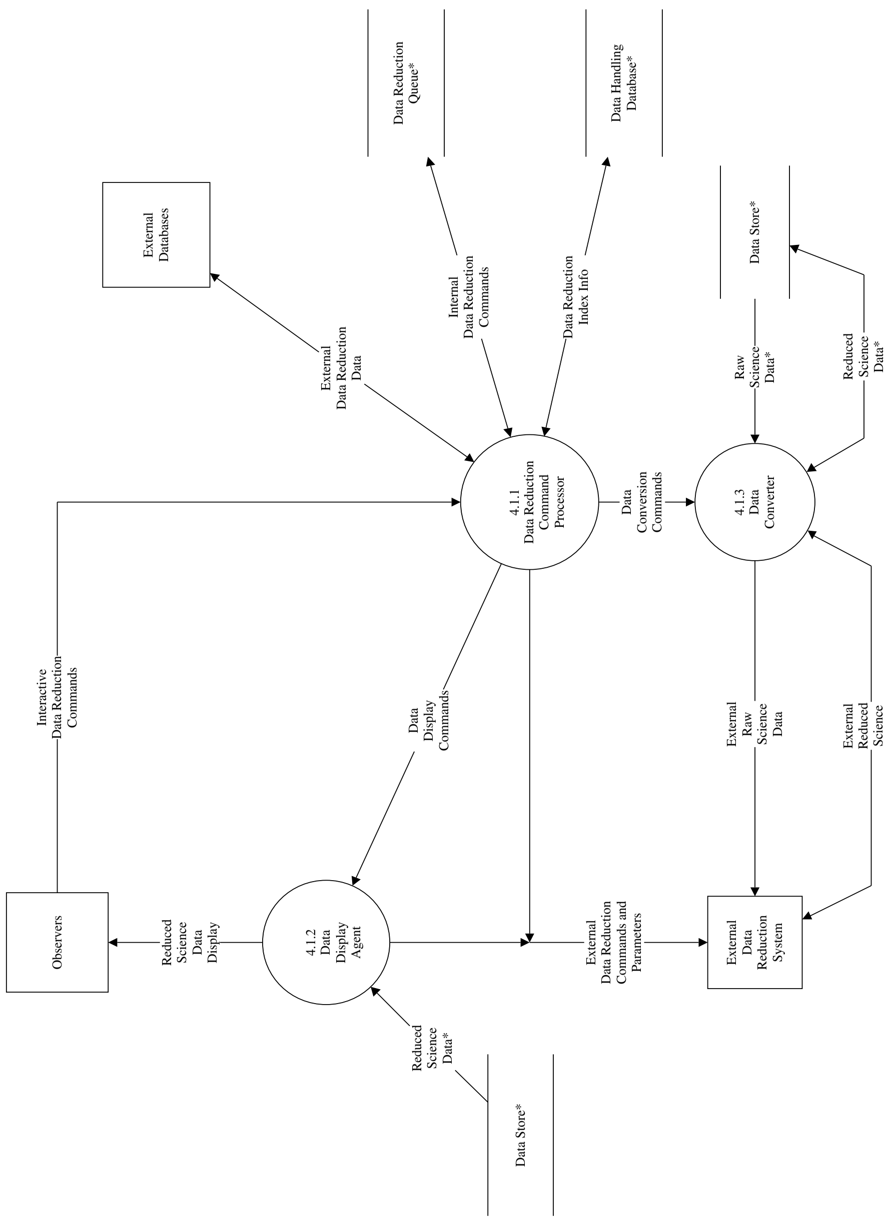


FIGURE A1 - 15. **Data Reduction Agent**



A2 DATA DICTIONARY

INTRODUCTION

This section introduces the data dictionary describing the high-level objects within the Gemini control system.

It is provided here as an aid to understanding the interfaces in the design.

Notation

The descriptions here are provided in Backus-Naur Form (BNF). Objects denoted by names that are in ALL CAPITAL letters denote *data types* (terminators). Data types have been chosen in a general and language-independent manner.

THE DATA DICTIONARY

```
A&GActualProbePositions = { ProbeCoordinates } ;

A&GActualTargetPositions = { TargetPosition } ;

A&GAlarms = { A&GPointingAlarms } + { A&GOpticalSurfaceAlarms } ;

A&GControlCommands = A&GPointingCommands
                   + A&GOpticalSurfaceCommands ;
```

```
A&GDemandProbePosition = ProbeCoordinates ;

A&GDemandTargetPosition = TargetPosition ;

A&GOpticalSurfaceAlarms = { Alarm } ;

A&GOpticalSurfaceCommand = COMMAND ;

A&GOpticalSurfaceCommands = [ A&GOpticalSurfaceDemandParameters
                          | A&GOpticalSurfaceCommand ] ;

A&GOpticalSurfaceDemandParameters = A&GDemandOpticalCamera ?
                                  + A&GDemandInfra - RedCamera ?
                                  + A&GDemandCalibrationWaveFrontSensor ? ;

A&GOpticalSurfaceResponses = [ A&GOpticalSurfaceAlarms
                           | A&GOpticalSurfaceStatusFlags ] ;

A&GPointingAlarms = { Alarm } ;

A&GPointingCommand = COMMAND, constraint = One of STOP, MOVE, FOLLOW,
                     NOFOLLOW, GUIDE CONFIG, GUIDE, NOGUIDE,
                     STANDALONE, PARK, ENCODER CONFIG, ZERO SET ;

A&GPointingCommands = [ A&GPointingDemandParameters
                    | A&GPointingCommand ] ;

A&GPointingDemandParameters = A&GDemandTargetPosition
                            + One for each Guide Probe {
                              A&GDemandProbePosition } ;

A&GPointingResponses = [ A&GActualTargetPositions
                     | A&GActualProbePositions
                     | A&GStatistics
                     | A&GErrorSignal
                     | A&GPointingAlarms
                     | A&GPointingStatusFlags ] ;

A&GResponses&Alarms = A&GPointingResponses
                    + A&GOpticalSurfaceResponses ;

A&GStatistics = GuideErrors
              + FWHMofGuideStarImage
```

```
               + MedianRelativeTransmission
               + MagnitudeofGuideStarImage ;

A&GandMount = [ A&GtoMount | MounttoA&G ] ;

A&GandSecondary = [ A&GtoSecondary | SecondarytoA&G ] ;

A&GtoAO = WaveFrontSensingInformation ;

A&GtoMount = WaveFrontSensingInformation ;

A&GtoSecondary = One for each wave front sensor {
                 WaveFrontSensingInformation } ;

AOActualProbePosition = ProbeCoordinates ;

AOActualTargetPosition = TargetPosition ;

AOControlCommands = AOPointingCommands + AOOpticalSurfaceCommands ;

AOCumulativeZernike = Zernike ;

AODemandProbePosition = ProbeCoordinates ;

AODemandTargetPosition = TargetPosition ;

AOImageQualityAlarms = { Alarm } ;

AOLock = BINARY FLAG, constraint = ON or OFF ;

AOOpticalSurfaceCommand = COMMAND ;

AOOpticalSurfaceCommands = AOOpticalSurfaceDemandParameters
                         + ( AOOpticalSurfaceCommand ) ;

AOOpticalSurfaceDemandParameters = AOConfigurationInformation
                                 + AOLock ;

AOOpticalSurfaceResponses = AOCumulativeZernike
                          + AOImageQualityAlarms
                          + AOPointingStatusFlags ;

AOPointingAlarms = { Alarm } ;
```

```
AOPointingCommand = COMMAND ;

AOPointingCommands = AOPointingDemandParameters
                   + ( AOPointingCommand ) ;

AOPointingDemandParameters = [ AODemandTargetPosition
                           | AODemandProbePosition ] ;

AOPointingResponses = AOActualTargetPosition
                    + AOActualProbePosition
                    + AOPointingAlarms
                    + AOPointingStatusFlags ;

AOResponses&Alarms = AOPointingResponses
                   + AOOpticalSurfaceResponses ;

AOandA&G = [ AOtoA&G | A&GtoAO ] ;

AOandP.Support = [ AOtoP.Support | P.SupporttoAO ] ;

AOandSecondary = AOtoSecondary + SecondarytoAO ;

AOtoP.Support = DemandedSurfaceShape ;

AOtoSecondary = DemandedSurfaceShape ;

ActualDataConfiguration = DataHandlingConfiguration ;

ActualInstrumentConfigurations = { InstrumentConfiguration } ;

ActualSystemConfiguration = ActualDataConfiguration
                          + ActualTelescopeConfiguration
                          + ActualInstrumentConfigurations ;

ActualTelescopeConfiguration = TelescopeConfiguration ;

ActualandDesiredSystemConfiguration = ActualSystemConfiguration
                                    + DesiredSystemConfiguration ;

ActuatorForces = One for each actuator { ForceonActuator } ;

AdaptiveOpticsAlarms = { AOPointingAlarms }
```

```
                    + { AOImageQualityAlarms } ;

Alarm = EPICS ALARM ? ;

Alarms = [ DHSAlarms | ICSAlarms | TCSAlarms ] ;

AltitudeAcceleration = ANGULAR ACCELERATION ;

AltitudePosition = ANGLE, constraint = 0- 90 degrees ;

AltitudePositionandMotion = AltitudePosition
                          + AltitudeVelocity
                          + AltitudeAcceleration ;

AltitudeVelocity = ANGULAR VELOCITY ;

ArcFrame = ScienceDataFile ;

ArchiveRequestFiles = { ArchiveRequestFile } ;

ArchiveRequestStore = ArchiveRequestFiles ;

AstronomicalCalibrationFrame = [ SkyBackgroundFrame
                             | SpectralStandardFrame
                             | PhotometricStandardFrame
                             | RadialVelocityStandardFrame ] ;

Attribute = STRING, constraint = The name of an EPICS record field ;

AttributeValuePair = Attribute + Value ;

AvailableOpticalComponents = AvailableFilters
                           + AvailableStops
                           + AvailableSlits
                           + AvailableEtc... ;

AxisInformation = AxisLabel + AxisUnits + AxisValues ;

AzimuthAcceleration = ANGULAR ACCELERATION ;

AzimuthPosition = ANGLE, constraint = 0- 360 degrees ;

AzimuthPositionandMotion = AzimuthPosition
```

```
                         + AzimuthVelocity
                         + AzimuthAcceleration ;

AzimuthVelocity = ANGULAR VELOCITY ;

Bearing = ANGLE, constraint = 0- 360 degrees ;

BiasFrame = ScienceDataFile ;

CalibrationFrame = [ InstrumentCalibrationFrame
                 | TelescopeCalibrationFrame
                 | AstronomicalCalibrationFrame ] ;

CalibrationFrames = { CalibrationFrame } ;

CassegrainAlarms = { Alarm } ;

CassegrainCommand = COMMAND, constraint = One of STOP, MOVE, FOLLOW,
                    NOFOLLOW, DRIFT, UNWRAP, PARK, ENCODER CONFIG,
                    ZERO SET ;

CassegrainControlCommands = [ CassegrainDemandParameters
                          | CassegrainEncoderConfiguration
                          | CassegrainLoggingCommand
                          | CassegrainCommand ] ;

CassegrainDemandParameters = CassegrainDemandRotatorP&Motion ;

CassegrainDemandRotatorP&Motion = RotatorPositionandMotion ;

CassegrainEncoderReadings = EncoderReadings ;

CassegrainLoggingCommand = COMMAND ;

CassegrainResponses&Alarms = CassegrainEncoderReadings
                           + CassegrainServoErrors
                           + CassegrainAlarms
                           + MotionStatusFlags ;

CassegrainServoErrors = ServoErrors ;

CassegrainStatus = TBD ;
```

```
ChopperDemandParameters = ChopperThrow
                        + ChopperAngle
                        + ChopperSynch.Information ;

CloudCoverReading = CloudSkyCoverage + CloudTransparency ;

CloudSkyCoverage = FLOATING NUMERIC, constraint = RANGE 0.0-1.0 ;

CloudTransparency = FLOATING NUMERIC ? ;

CoaddedFrame = UniqueID + CoaddedPixelData ;

CoaddedFrameStore = CoaddedFrame
                  + InstrumentHeaderInformation ( optional ) ;

CommandName = STRING, constraint = The name of a sequencer command. ;

CommandParameters = { AttributeValuePair } ;

CompassDirection = 16 VALUE FLAG, constraint = N, NNE, NE, ENE, E, ESE
                  , SE, SSE, S, SSW, SW, WSW, W, WNW, NW, NNW ;

ControlCommand = CommandName + CommandParameters ;

ControlCommands = OCSControlCommands + HandsetControlCommands ;

ControlData = [ AOandA&G
             | AOandSecondary
             | AOandP.Support
             | A&GandMount
             | A&GandSecondary
             | OtherCommunicationTBD ] ;

CurrentCloudCover = CloudCoverReading ;

CurrentHumidity = HumidityReading ;

CurrentTemperature = TemperatureReading ;

CurrentWindVelocity = WindSpeedReading + WindDirectionReading ;

DCAlarms = { Alarm } ;
```

```
DHSAlarms = { TBDAlarm } ;

DHSControlCommands = { ControlCommand } + { StoredDataInfo } ;

DHSSharedDataBuffer = CoaddedFrame ;

DailyPlan = StartTime + Duration + ScienceObservation ;

DarkFrame = ScienceDataFile ;

DataArray = DataArrayDescription
          + { AxisInformation }
          + PixelInformation ;

DataArrayDescription = DataArrayTitle
                     + DataArrayLabel
                     + DataArrayUnits ;

DataForArchive = ProcessedLoggingData
               + RawScienceData
               + ( ReducedScienceData ) ;

DataHandlingConfiguration = DataStorageConfiguration
                          + DataReductionConfiguration ;

DataHandlingDatabase = IndexInfoandLoggingData + StatusInformation ;

DataReductionQueue = [ InternalDataReductionCommands
                   | InteractiveDataReductionCommands ] ;

DataStorageCommands&Responses = DataStorageCommands
                              + DataStorageResponses ;

DataStore = DataForArchive + CalibrationFrames + { FileHeaders } ;

DataTransportCommands&Responses = DataTransportCommands
                                + DataTransportResponses ;

DemandedSurfaceShape = Zernike ;

DesiredDataHandlingConfiguration = DataHandlingConfiguration ;

DesiredInstrumentConfigurations = { InstrumentConfiguration } ;
```

```
DesiredSystemConfiguration = DesiredDataHandlingConfiguration
                           + DesiredTelescopeConfiguration
                           + DesiredInstrumentConfigurations ;

DesiredTelescopeConfiguration = TelescopeConfiguration ;

DetectorControlSignals = DetectorSequencingWaveforms
                       + DetectorEnvironmentalControlSignals ;

DetectorSetupInformation = DetectorName
                         + DetectorType
                         + DetectorArraySize
                         + DetectorBadPixelInformation
                         + DetectorAvailableWaveforms ;

DetectorStatusandData = DetectorEnvironmentalInformation
                      + DetectorDataReadout ;

DirectionReading = [ CompassDirection | Bearing ] ;

DomeAcceleration = AzimuthAcceleration ;

DomeActualP&Motion = DomePositionandMotion ;

DomeDemandP&Motion = DomePositionandMotion ;

DomeFlatFrame = ScienceDataFile ;

DomePosition = AzimuthPosition ;

DomePositionandMotion = DomePosition
                      + DomeVelocity
                      + DomeAcceleration ;

DomeShutterActualPosition = ShutterPosition ;

DomeShutterDemandPosition = ShutterPosition ;

DomeVelocity = AzimuthVelocity ;

EPICSArchiveDataFile = [ EPICSArchiveByChannelDataFile
                     | EPICSArchiveSampleSetDataFile ] ;
```

```
EPICSArchiverStore = SavedEPICSLoggingData
                   + ExportedEPICSLoggingData ;

EPICSExportedDataFile = [ EPICSArchiveSampleSetDataFile
                      | EPICSArchiveSpreadsheetFile
                      | EPICSArchiveSpreadsheetForArraysFile ] ;

EnclosureActualHumidities = { EnclosureHumidityReading } ;

EnclosureActualTemperatures = { EnclosureTemperatureReading } ;

EnclosureAlarms = { EnclosurePointingAlarms }
                + { EnclosureEnvironmentalAlarms } ;

EnclosureControlCommands = EnclosurePointingCommands
                         + EnclosureEnvironmentalCommands
                         + ( EnclosureLoggingCommand ) ;

EnclosureDemandTemperature = TemperatureReading ;

EnclosureEncoderReadings = EncoderReadings ;

EnclosureEnvironmentalAlarms = { Alarm } ;

EnclosureEnvironmentalCommand = COMMAND, constraint = One of STOP,
                                FOLLOW, NOFOLLOW, ZERO SET ? ;

EnclosureEnvironmentalCommands = [
                                 EnclosureEnvironmentalDemandParameters
                               | EnclosureEnvironmentalCommand ] ;

EnclosureEnvironmentalDemandParameters = EnclosureDemandTemperature
                                       + WindGatesDemandPosition
                                       + FansDemandSpeed ;

EnclosureEnvironmentalResponses = InternalEnvironmentalInformation
                                + WindGatesActualPosition
                                + FansActualSpeed
                                + EnclosureEnvironmentalAlarms ;

EnclosureHardware = LowerEnclosure + Dome ;
```

```
EnclosureHumidityReading = HumiditySensorLocation + HumidityReading ;

EnclosureLoggingCommand = COMMAND ;

EnclosurePointingAlarms = { Alarm } ;

EnclosurePointingCommand = COMMAND, constraint = STOP, MOVE DOME,
                           DRIFT DOME, MOVE SHUTTER, DRIFT SHUTTER,
                           FOLLOW, NOFOLLOW, PARK, ENCODER CONFIG,
                           ZERO SET ;

EnclosurePointingCommands = [ EnclosurePointingDemandParameters
                            | EnclosureEncoderConfiguration
                            | EnclosurePointingCommand ] ;

EnclosurePointingDemandParameters = DomeDemandP&Motion
                                  + WindShieldDemandP&Motion
                                  + DomeShutterDemandPosition ;

EnclosurePointingResponses = WindShieldActualP&Motion
                           + DomeActualP&Motion
                           + DomeShutterActualPosition
                           + EnclosureEncoderReadings
                           + EnclosureServoErrors
                           + EnclosurePointingAlarms
                           + MotionStatusFlags ;

EnclosureResponses&Alarms = EnclosurePointingResponses
                          + EnclosureEnvironmentalResponses ;

EnclosureStatus = TBD ;

EnclosureTemperatureReading = TemperatureSensorLocation
                            + TemperatureReading ;

EncoderInconsistentFlag = BINARY FLAG ;

EncoderReading = INTEGER ? ;

EncoderReadings = RawEncoderReadings + ProcessedEncoderReadings ;

EnvironmentalInformation = InternalEnvironmentalInformation
                         + ExternalEnvironmentalInformation ;
```

```
EnvironmentalRequirements = SeeingRequired
                          + AirMassLimits
                          + WeatherLimits
                          + MoonPhaseLimits ;

ExportedEPICSLoggingData = { EPICSExportedDataFile } ;

ExternalData = ExternalDataReductionInformation
             + ExternalObservingInformation

ExternalEnvironmentalInformation = WeatherInformation
                                 + SeeingInformation ;

ExternalObservingData = StandardsandCalibrationData
                      + GuideStarandPointSourceData
                      + ExternalSchedulingData ;

FITSHeader = { FITSPacket } ;

FITSItem = FITSKeyword + FITSValue + ( FITSDescription ) ;

FITSKeyword = CHARACTER STRING, constraint = <= 8 characters ;

FITSPacket = { FITSItem } ;

FanSpeed = MULTI VALUE FLAG, constraint = FAST, MEDIUM, SLOW or
           whatever ;

FansActualSpeed = FanSpeed ;

FansDemandSpeed = FanSpeed ;

FileHeader = UniqueID + FITSHeader ;

FileHeaders = FileHeader ;

Flat = ScienceDataFile ;

ForceonActuator = FLOATING NUMERIC ;

GeneralInstrumentSetupInformation = OpticalComponentsSetupInformation
                                  + ProcessingComponentsSetupInformation ;
```

```
GraphicalInformationDisplay = TBD ;

HardwareSetupInformation = InstrumentSetupInformation
                         + TelescopeSetupInformation ;

HumidityReading = FLOATING NUMERIC ;

HumiditySensorLocation = CHARACTER STRING, constraint = Description of
                         location ;

ICS = ICS / TCSDirectCommands&Responses
    + ICS / TCSDirectWavefrontExchange ;

ICSAlarms = DCAlarms
          + ICSDataProcessingAlarms
          + OpticalComponentsAlarms ;

ICSControlCommands = { ControlCommand } ;

ICSDataProcessingAlarms = { Alarm } ;

ICSLoggingRequests&Data = ICSArchiverLoggingRequests&Data
                        + ICSHistoryLoggingRequests&Data ;

ICSRealTimeDisplays = InstrumentVisualizationDisplay
                    + ( OptionalICSQuickLookDisplay ) ;

InPositionFlag = BINARY FLAG ;

IndexInfo = DataReductionIndexInfo + DataAcquisitionIndexInfo ;

IndexInfoandLoggingData = IndexInfo + LoggingData ;

InstrumentCalibrationFrame = [ MaskFrame
                           | WindowFrame
                           | BiasFrame
                           | DarkFrame
                           | Flat - fieldFrame
                           | ArcFrame ] ;

InstrumentConfigurations = { InstrumentConfiguration } ;
```

```
InstrumentControlSignals = DetectorControlSignals
                         + OpticalComponentsControlSignals ;

InstrumentHardware = DetectorHardware + OpticalComponents ;

InstrumentHeaderInformation = UniqueID + FITSHeader ;

InstrumentQueue = { QueueEntry } ;

InstrumentSetupInformation = GeneralInstrumentSetupInformation
                           + DetectorSetupInformation ;

InstrumentStatusandData = DetectorStatusandData
                        + OpticalComponentsStatus ;

InternalEnvironmentalInformation = EnclosureActualTemperatures
                                 + EnclosureActualHumidities
                                 + OtherInternalInformation ? ;

LocationonPrimaryMirror = TBD ;

LoggingCommands&Responses = HistoryLoggingCommands&Responses
                          + CombineLoggingCommands&Responses
                          + ConvertLoggingCommands&Responses
                          + ArchiveRequestCommands&Responses ;

LoggingData = EPICSLoggingRequests&Data + Non- EPICSLoggingData ;

LoggingRequests&Data = DHSLoggingRequests&Data
                     + OCSLoggingRequests&Data
                     + TCSLoggingRequests&Data
                     + ICSLoggingRequests&Data ;

MaskFrame = ScienceDataFile ;

MotionLimitExceededFlag = BINARY FLAG ;

MotionStatusFlags = [ InPositionFlag
                  | MovingFlag
                  | StoppedFlag
                  | FollowingFlag
                  | GuidingFlag
                  | EncoderInconsistentFlag
```

```
                    | MotionLimitExceededFlag
                    | MotionTimeoutFlag ] ;

MotionTimeoutFlag = BINARY FLAG ;

MountAlarms = Alarm ;

MountCommand = COMMAND, constraint = One of STOP, MOVE, FOLLOW,
               NOFOLLOW, UNWRAP, DRIFT, PARK, ENCODER CONFIG, ZERO SET
               ;

MountControlCommands = [ MountDemandParameters
                       | MountEncoderConfiguration
                       | MountLoggingCommand
                       | MountCommand ] ;

MountDemandAltitudeP&Motion = AltitudePositionandMotion ;

MountDemandAzimuthP&Motion = AzimuthPositionandMotion ;

MountDemandParameters = MountDemandAltitudeP&Motion
                      + MountDemandAzimuthP&Motion ;

MountEncoderReadings = EncoderReadings ;

MountLoggingCommand = COMMAND ;

MountResponses&Alarms = MountEncoderReadings
                      + MountServoErrors
                      + MountAlarms
                      + MotionStatusFlags ;

MountServoErrors = ServoErrors ;

MountStatus = TBD ;

MounttoA&G = WaveFrontSensingInformation ;

NextConfigurationFromEachQueue = { ScienceConfiguration } ;

NumberofZernikeCoefficients = INTEGER ;

ObservationQueue = [ SelectedScienceObservations
```

```
                  | ProgramDatabaseManagementOperations ] ;

ObservingDatabase = ActualandDesiredSystemConfiguration
                  + SequencerExecutorInformation ;

ObservingPool = [ SubmittedSciencePrograms
               | ProgramDatabaseManagementOperations ] ;

OpticalComponentsAlarms = { Alarm } ;

OpticalComponentsSetupInformation = AvailableOpticalComponents
                                  + EncoderLookupTables ;

OptionalDHSQuickLookDisplay = TBD ;

OptionalICSQuickLookDisplay = [ OptionalInstrumentQuickLookDisplay
                             | OptionalDetectorQuickLookDisplay ] ;

P.SupportActualActuatorForces = ActuatorForces ;

P.SupportAlarms = { Alarm } ;

P.SupportCommand = COMMAND, constraint = One of STOP, MOVE SURFACE,
                   MOVE ACTUATOR, FOLLOW, NOFOLLOW, PARK ;

P.SupportControlCommands = [ P.SupportDemandParameters
                         | P.ThermalTemperatureMatrix
                         | MountActualAzimuthPosition
                         | P.SupportCommand ] ;

P.SupportDemandActuatorForces = ActuatorForces ;

P.SupportDemandParameters = [ P.SupportDemandShape
                          | P.SupportDemandActuatorForces ] ;

P.SupportDemandShape = P.SupportDemandZernike
                     + P.SupportDemandTranslation ;

P.SupportDemandTranslation = Translation ;

P.SupportDemandZernike = Zernike ;

P.SupportLoadCells = LoadCells ;
```

```
P.SupportResponses = P.SupportActualActuatorForces
                   + P.SupportLoadCells
                   + P.SupportAlarms
                   + P.SupportStatusFlags ;

P.ThermalAlarms = { Alarm } ;

P.ThermalCommand = COMMAND, constraint = One of STOP, FOLLOW, NOFOLLOW,
                   ZERO SET ? ;

P.ThermalControlCommands = [ P.ThermalDemandParameters
                         | P.ThermalCommand ] ;

P.ThermalDemandParameters = P.ThermalDemandTemperature ;

P.ThermalResponses = P.ThermalTemperatureMatrix + P.ThermalAlarms ;

P.ThermalStatus = TBD ;

P.ThermalTemperatureMatrix = { P.ThermalTemperatureReading } ;

P.ThermalTemperatureReading = FLOATING NUMERIC ;

PendingConfigurations = TelescopeQueue + { InstrumentQueue } ;

PhotometricStandardFrame = ScienceDataFile ;

PixelInformation = ArrayOrigin
                 + SignalValues
                 + ( VarianceValues )
                 + ( DataQualityValues ) ;

PlanView = Dates + DailyPlan ;

PrimaryAlarms = P.SupportAlarms + P.ThermalAlarms ;

PrimaryControlCommands = P.SupportControlCommands
                       + P.ThermalControlCommands ;

PrimaryControlSignals = P.SupportControlSignals
                      + P.ThermalControlSignals ;
```

```
PrimaryHardware = PrimarySupportHardware + PrimaryThermalHardware ;

PrimaryResponses&Alarms = P.SupportResponses + P.ThermalResponses ;

PrimaryStatus = TBD ;

ProbeCoordinates = Xi + Eta ;

ProcessedEncoderReadings = One for each encoder { EncoderReading } ;

ProgramDatabase = ObservingPool
                + SelectedProgramPool
                + ObservationQueue ;

ProgramStoreDatabase = ProgramData ;

QueueEntry = ScienceObservation + ScienceConfigurationPointer ;

QuickLookCommands&Responses = [ QuickLookCommands
                            | QuickLookResponses ] ;

RadialVelocityStandardFrame = ScienceDataFile ;

RawEncoderReadings = One for each encoder { EncoderReading } ;

RawFrameStore = RawFrames ;

RawFrames = { RawFrame } ;

RawScienceData = CoaddedFrame
               + InstrumentHeaderInformation ( optional ) ;

ReducedScienceData = [ CalibrationFrame | ScienceFrame ] ;

ReducedScienceDataDisplay = TBD ;

RotatorAcceleration = ANGULAR ACCELERATION ;

RotatorPosition = ANGLE, constraint = 0-360 degrees ;

RotatorPositionandMotion = RotatorPosition
                         + RotatorVelocity
                         + RotatorAcceleration ;
```

```
RotatorVelocity = ANGULAR VELOCITY ;

SavedEPICSLoggingData = { EPICSArchiveDataFile } ;

SchedulingInformation = PlanView + OtherInformation ;

ScienceConfiguration = InstrumentConfigurations
                     + TelescopeConfiguration
                     + DataHandlingConfiguration
                     + SequencerRecipe ;

ScienceDataFile = FileHeader + DataArray ;

ScienceFrame = ObjectFrame ;

ScienceObservation = ScienceObservationRequirements
                   + { ScienceConfiguration } ;

ScienceObservationRequirements = InstrumentRequirements
                               + EnvironmentalRequirements ;

ScienceProgram = ScienceProgramHeader
               + ScienceProgramStatus
               + ScienceProgramRequirements
               + { ScienceObservation } ;

ScienceProgramHeader = ProgramTitle
                     + Observer ' sNameandAddress
                     + TACcode
                     + AdditionalInformation ;

ScienceProgramRequirements = { ScienceObservationRequirements } ;

ScienceProgramStatus = ProgramSelected
                     + InstrumentsAvailable
                     + TACPriority
                     + ObjectsAvailable
                     + ProgramOpen ;

ScienceStatusandQualityInformation = GraphicalInformationDisplay
                                   + ScienceandQualityInformation ;
```

```
ScienceandQualityInformation = ( OptionalDHSQuickLookDisplay )
                             + ReducedScienceDataDisplay ;

SecondaryAlarms = { Alarm } ;

SecondaryCommands = COMMAND ;

SecondaryControlCommands = [ SecondaryDemandParameters
                         | SecondaryEncoderConfiguration
                         | SecondaryCommands ] ;

SecondaryDemandParameters = SecondaryDemandTranslation
                          + SecondaryDemandTilt
                          + ChopperDemandParameters ;

SecondaryDemandTilt = Tilt ;

SecondaryDemandTranslation = Translation ;

SecondaryEncoderReadings = EncoderReadings ;

SecondaryResponses&Alarms = SecondaryEncoderReadings
                          + SecondaryServoErrors
                          + SecondaryAlarms
                          + MotionStatusFlags
                          + CopperSynch.Info ? ;

SecondaryServoErrors = ServoErrors ;

SecondarytoA&G = One for each wave front sensor {
                 WaveFrontSensingInformation } ;

SelectedProgramPool = [ SelectedSciencePrograms
                    | ProgramDatabaseManagementOperations ] ;

SelectedScienceObservation = ScienceObservation ;

SelectedScienceObservations = { SelectedScienceObservation } ;

SelectedSciencePrograms = { ScienceProgram } ;

SensedZernike = Zernike ;
```

```
ShutterPosition = ANGLE ? ;

SkyBackgroundFrame = ScienceDataFile ;

SkyFlatFrame = ScienceDataFile ;

SpectralStandardFrame = [ RelativeFluxSpectralStandardFrame
                        | AbsoluteFluxSpectralStandardFrame ] ;

SpeedReading = FLOATING NUMERIC ;

StandardsandCalibrationData = TBD ;

StoredDataInfo = FileName + FITSHeaderInformation ;

SubmittedSciencePrograms = { ScienceProgram } ;

SystemControlCommands = [ DHSControlCommands
                        | ICSControlCommands
                        | TCSControlCommands ] ;

TCSAlarms = [ PrimaryAlarms
            | SecondaryAlarms
            | AdaptiveOpticsAlarms
            | A&GAlarms
            | MountAlarms
            | CassegrainRotatorAlarms
            | EnclosureAlarms ] ;

TCSControlCommands = { ControlCommand } ;

TCSLoggingRequests&Data = TCSArchiverLoggingRequests&Data
                        + TCSHistoryLoggingRequests&Data ;

TargetCoordinates = AltitudePosition + AzimuthPosition ;

TargetPosition = TargetCoordinates + RotatorPosition ;

TelescopeCalibrationFrame = [ SkyFlatFrame | DomeFlatFrame ] ;

TelescopeControlSignals = PrimaryControlSignals
                        + SecondaryControlSignals
                        + AOControlSignals
```

```
                        + A&GControlSignals
                        + MountControlSignals
                        + CassegrainControlSignals
                        + EnclosureControlSignals ;

TelescopeHardware = PrimaryHardware
                  + SecondaryHardware
                  + AdaptiveOpticsHardware
                  + A&GHardware
                  + MountHardware
                  + CassegrainHardware
                  + EnclosureHardware ;

TelescopeHardwareStatus = PrimaryStatus
                        + SecondaryStatus
                        + AOStatus
                        + A&GStatus
                        + MountStatus
                        + CassegrainStatus
                        + EnclosureStatus ;

TelescopeQueue = { QueueEntry } ;

TelescopeSubsystemCommands = PrimaryControlCommands
                           + SecondaryControlCommands
                           + AOControlCommands
                           + A&GControlCommands
                           + MountControlCommands
                           + CassegrainControlCommands
                           + EnclosureControlCommands ;

TelescopeSubsystemResponses&Alarms = PrimaryResponses&Alarms
                                   + SecondaryResponses&Alarms
                                   + AOResponses&Alarms
                                   + A&GResponses&Alarms
                                   + MountResponses
                                   + CassegrainResponses&Alarms
                                   + EnclosureResponses&Alarms ;

TelescopeandInstrumentQueues = [ NextConfigurationFromEachQueue
                               | TopNConfigurations
                               | PendingConfigurations ] ;
```

```
TemperatureReading = FLOATING NUMERIC ;

TemperatureSensorLocation = CHARACTER STRING, constraint = Description
                            of location ;

TemperatureValue = FLOATING NUMERIC ;

Tilt = Tilt.Rx + Tilt.Ry ;

Tilt.Angle = ANGLE, constraint = 0- some small number of degrees ;

Tilt.Rx = Tilt.Angle ;

Tilt.Ry = Tilt.Angle ;

TopNConfigurations = { ScienceConfiguration } ;

Translation = Xmotion + Ymotion + Zmotion ;

Value = STRING, constraint = Must translate to valid value ;

WaveFrontSensingInformation = NumberofZernikeCoefficients
                            + SensedZernike
                            + WavefrontUpdateTime
                            + WavefrontSensorQuality ;

WeatherInformation = CurrentWindVelocity
                   + CurrentTemperature
                   + CurrentHumidity
                   + CurrentCloudCover
                   + CurrentPrecipitation
                   + OtherWeatherInformation ? ;

WindDirectionReading = DirectionReading ;

WindGatesActualPosition = WindGatesPosition ;

WindGatesDemandPosition = WindGatesPosition ;

WindGatesPosition = ANGLE ? ;

WindShieldAcceleration = ANGULAR ACCELERATION ? ;
```

```
WindShieldActualP&Motion = WindShieldPositionandMotion ;

WindShieldDemandP&Motion = WindShieldPositionandMotion ;

WindShieldPosition = ANGLE ?, constraint = Perhaps less than 0-90
                               degrees ? ;

WindShieldPositionandMotion = WindShieldPosition
                            + WindShieldVelocity
                            + WindShieldAcceleration ;

WindShieldVelocity = ANGULAR VELOCITY ? ;

WindSpeedReading = SpeedReading ;

WindowFrame = ScienceDataFile ;

Xmotion = FLOATING NUMERIC ;

Ymotion = FLOATING NUMERIC ;

Zernike = 1 to N { ZernikeCoefficient } ;

ZernikeCoefficient = FLOATING NUMERIC ( DOUBLE PRECISION ) ? ;

Zmotion = FLOATING NUMERIC ;
```

A3 Interface Description

Introduction

This appendix presents some of the issues in the implementation details for the interfaces found in the Gemini Control System. The concentration is on those interfaces that need to be agreed upon by two or more system components (i.e. external interfaces that impact single subsystems are only considered briefly).

More details on the interfaces themselves can be found in the Interface Control Documents.

System Interface Overview

The interfaces in the Gemini control system can be categorized into several classes:

- *external* interfaces between the system and external systems
- *primary internal* interfaces between the major internal systems
- *secondary internal* interfaces between each internal system and its subsystems

External Interfaces

External interfaces connect the system to external systems, such as the observer using the system, as well as external databases, telescope hardware, etc. A general description of these interfaces can be found in Chapter 9 - “High-Level System Interfaces”, with details of interfaces in the chapters for the appropriate major system that use each external interface.

Primary Internal Interfaces

These are the interfaces between the major system components: the Observatory Control System (OCS), the Data Handling System (DHS), the Instrument Control System (ICS), and the Telescope Control System (TCS). All of these interfaces are designed using the ‘unidirectional database- state driven’ model described in the ‘High-Level System Design’ chapter.

Secondary Internal Interfaces

These interfaces are internal to the major system components and connect subsystems, or connect subsystems to specific hardware devices. Details of these interfaces are specific to the interface between a system and a subsystem and are found in the chapter describing the detailed design for that subsystem.

HARDWARE INTERFACES

The Gemini Control System includes the following hardware interfaces:

- Time Bus
- Synchro Bus
- Event Bus
- Interlock System Interface

Time Bus Interface

HARDWARE

Each VxWorks system requiring access to the Time Bus contains a Bancomm bc635VME and/or bc637VME.

FUNCTIONALITY

There must be an accurate way to synchronize and transmit time information throughout the Gemini Control System. This is provided by the *Time Bus*. This bus must interface between the EPICS systems and other portions of the system. The following functional requirements exist:

- *Database Timestamp Maintenance*

EPICS maintains a timestamp for every database record that holds the time when that record was last processed. For an IOC that has a Bancomm card installed, the value for the timestamp is to be read from that card.

- *Current UTC*

  A database programmer can define an EPICS database record that provides the current value of UTC as an accessible field. The field supplies the time using the standard EPICS timestamp format.

- *Periodic Interrupts*

  The driver and device layers support a programmable periodic I/O interrupt from the Bancomm hardware that can be used to initiate database processing and EPICS event generation within the IOC.

- *External Event UTC*

  A TTL signal may be connected to the Bancomm card and used to signal the occurrence of an external event. When the event signal occurs, the value of UTC is latched by the card and an interrupt is generated. Driver and device layer support is required so that a suitably initialized database record causes the event to generate an EPICS I/O Interrupt; thus initiating record processing and permitting the latched time to be read. The record processing provides the ability to generate an EPICS event as a direct result of the I/O Interrupt.

- *Hardware Status Reporting*

  The EPICS alarm reporting system is used to indicate error status conditions from the Bancomm hardware. Errors reported include the loss of an incoming time reference signal (GPS or IRIG-B as appropriate).

TIME BUS ISSUES

There are some unresolved issues with the Time Bus that require additional study:

- *Use of UTC versus TAI*

  UTC introduces problems that are associated with the occasional addition of leap seconds. Since UTC is synchronized from GPS, the hardware automatically applies leap second corrections. The same corrections must be made by the Telescope Control System at the identical moment to keep the sky position correct. An alternative to this necessary synchronization is to use Atomic Time TAI, which is never corrected by the addition of leap seconds.

- *NTP Support*

  Bancomm sells a complete GPS Network Time Server system that provides a source of IRIG-B time codes and also functions as a Stratum One NTP time server over TCP/IP. It should be possible to use this system in place of the bc637VME card.

- *VxWorks Support*

It may be desirable to provide time suppor to VxWorks using the Bancomm hardware, including possibly generating the VxWorks system clock interrupt this way.

- *Record Types*

  The datatype of an EPICS Timestamp consists of a pair of 32-bit integers, holding nanoseconds and seconds since January 1, 1990. EPICS does not currently provide a means of accessing or manipulating timestamps from within an EPICS database, thus some consideration is needed as to what type of record the Time Bus interface should support. There may be some call to provide a string record, but a numeric datatype is essential for some applications that need to manipulate the time.

PERFORMANCE REQUIREMENTS

The only specific performance requirement identified at this time is that the time in the VME crate be maintained to +- 10 microseconds.

Synchro Bus

The Synchro Bus provides a very high-speed interface between system components. A design goal is to make access to the Syncho Bus as transparent as possible, so that information may be moved using either the Synchro Bus or TCP/IP depending upon performance requirements.

HARDWARE

Each VxWorks system requiring access to the Synchro Bus requires a VMIC reflective memory card (VMIVME-5578).

FUNCTIONALITY

The Synchro Bus must be integrated with the EPICS database system and support EPICS records. The following functionality is required:

- *Output Records*

  A database programmer may declare an EPICS analog, long, string or waveform output record with a reflective memory address. Any value that is written into the VAL field of such a record is broadcast to all reflective memory input records that have the same reflective memory address.

- *Input Records*

  An input record that is declared within the reflective memory reports the value of the field at the time of record processing. The same record types are supported as for output records.

- *Status Reporting*

  The VMIC Reflective Memory hardware provides a manes to test the integrity of the fiber-optic cable under software control. The device driver performs periodic tests of the ring state and reports the results to the higher level EPICS software using the standard EPICS Alarm State/Severity mechanism.

SYNCHRO BUS ISSUES

There are some unresolved issues with the Synchro Bus that require additional study:

- *C Subroutine Interface*

  The ability to interface directly to the Reflective Memory from a C function may be useful. This same function interface is used between the EPICS device and driver layers, though not all of the features are needed for this task.
- *Use of Interrupts*

  The VMIC cardset provides the ability to send three different types of interrupts between memory nodes, either by broadcasting or to a specific node. There may be different ways to use this capability.
- *Reflective Memory Allocation*

  An EPICS record that is connected to the Reflective Memory must be associated with an address in the memory space. There are several possible ways in which this memory space can be partitioned and allocated to different records and nodes. In addition, some means of preventing simultaneous update (resulting in loss of data) must be chosen.

PERFORMANCE REQUIREMENTS

A series of 10 32-bit floating point values representing Zernicke polynomial coefficients must be transmittable between two IOCs at an update rate of 200Hz, with a maximum delay of 500 microseconds from source to destination.

The allocation of the 500 microseconds is as follows:

- 100 microseconds for originator (a wave front sensor) from end of integration to writing Zernikes out to local EPICS database values
- 200 microseconds for synchro bus to transfer EPICS database values from one system to the other
- 100 microseconds for remote EPICS system to read its local EPICS database values and write out commands to secondary control system
- 100 microseconds for secondary control system to generate a voltage command to its actuators

Event Bus

The Event Bus provides a means of transferring eventsignals throughout the distributed control system. Most of the design of this interface is still under development, but it may be possible to layer the Event Bus onto the Synchro Bus using TTL cards.

There are a number of requirements for this bus:

- transmit 3 state information from one source to multiple destinations - this is how the command to the secondary chopper (2 or 3 position chop) would be sent as well as how the secondary's *in position* signal would be propagated. It is unlikely that this need could stand the latency and jitter inherent in using a digitial bus so it would need to be analog
- transmit TTL information from one source to multiple desitnations (this would use the aforementioned TTL to synchro bus I/F)
- TBD

Interlock System

HARDWARE

The hardware for the interlock system is still an unresolved design decision.

FUNCTIONALITY

There are two requirements for the Interlock System:

1. A *Hardware Interlock Monitor* responsible for reporting the state of a hardware interlock to the control system software. The state of each interlock requires a *positive report,* both the interlock status and the correct operation of the interlock system can be monitored. This is accomplished by requiring two signals, one for interlock on, the other for interlock off, where exactly one of these signals must always be present.
2. An *Electronic Interlock Interface* that is completely separate from the EPICs system. This could, for instance, be based on a Allen Bradley PLC system. The combination of trigger states and demanded interlocks would be "burnt" into ROM presumably. If any of these were critical interlocks then an additional Hardware Interlock would have to be proper functioning of the Electronic Interlock System.

3. An *Interlock Interface to EPICS* providing a means for software to request that a particular interlock be set and thus prevent some activity from occuring that might be dangerous. The software must be able to read the status of the interlock. The information returned may include other data such as the reason for the interlock request and the current request state is from the software

INTERLOCK SYSTEM ISSUES

More work remains on the details of the Interlock System Interface.

PERFORMANCE

Reliability is the critical issue in the Interlock System. The system must operate with very little chance of undetected failure of the Interlock System.