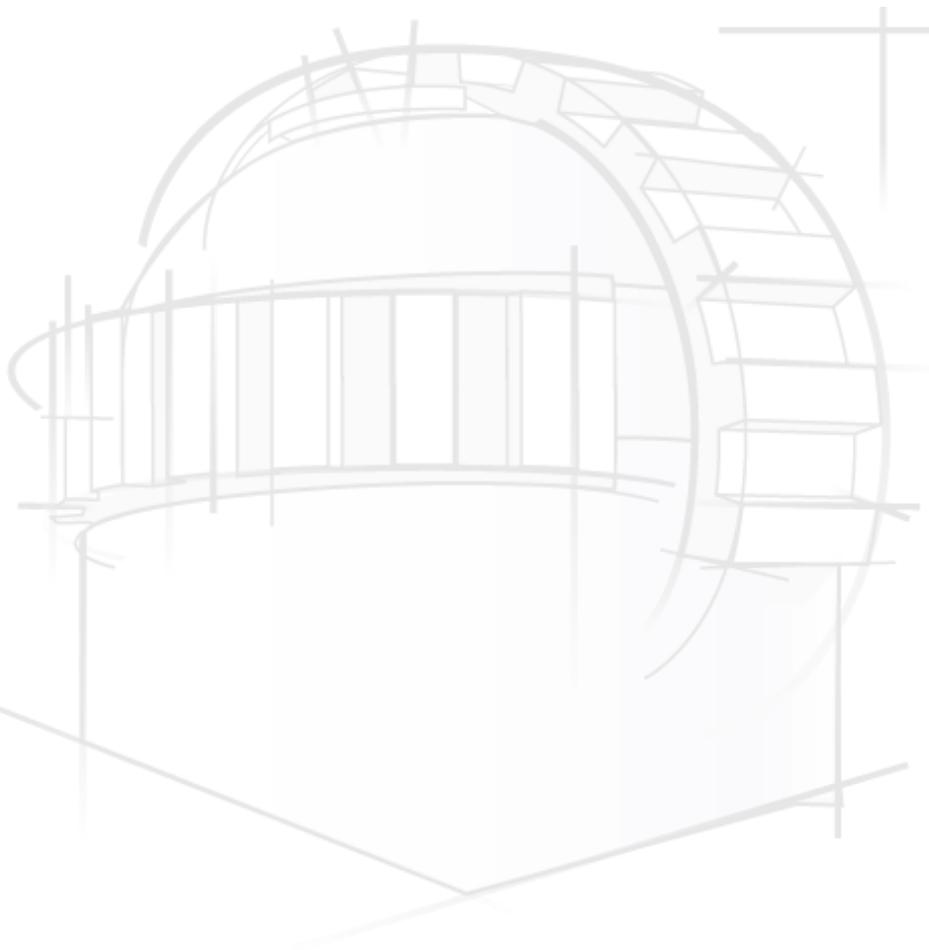


OCS Upgrades Program Top-Level Vision and Concepts

SCI-OCS3-OCD-03



Issued By: Science Group

Created By: Bryan Miller and the OCS Upgrades Science Team

Approved By: [Click here to enter text.](#)

Group Manager: Arturo Nunez



Version History

REVISION CHART			
Version	Author(s)	Description of Version	Date Completed
0.9	Bryan Miller	Based on OCD-01 v1.2	2019-Aug-09
1.0	Bryan Miller	Updated from comments on 0.9	2019-Sep-11

Table of Contents

TABLE OF CONTENTS	3
PURPOSE AND SCOPE	4
APPLICABLE DOCUMENTS	4
JUSTIFICATION OF CHANGES	4
CONCEPT OF THE PROPOSED SYSTEM	4
CONCEPT DESIGN PROCESS	5
TOP-LEVEL GOALS	5
INSTITUTIONAL GOALS	5
TECHNICAL GOALS	5
SCIENCE STAFF GOALS	6
SCIENCE USER GOALS	6
NEW SYSTEM CONCEPTS	6
ITC INTEGRATION	7
USER-ORIENTED PARAMETERS	7
PHYSICAL CONDITIONS CONSTRAINTS	7
ENABLE SIGNAL-TO-NOISE AS A CONSTRAINT	7
SELECT INSTRUMENT/MODE BASED ON SCIENCE REQUIREMENTS	7
TIMING CONSTRAINTS	8
SEQUENCES	8
VALID OBSERVATIONS CREATED AUTOMATICALLY	8
AUTOMATIC CALIBRATIONS	8
PROMOTE SUFFICIENT CALIBRATIONS	9
PROGRAM TIMESCALES	9
NEW OBSERVATION WORKFLOW	9
GN/GS OBSERVATIONS IN THE SAME PROGRAM	9
REAL-TIME SCHEDULER	9
QUEUE FILLING AND LONG-TERM SCHEDULER	10
PROGRAM RANKING	10
TIME-CRITICAL OBSERVATIONS	10
TARGET OF OPPORTUNITY	10
OBSERVATION ASSOCIATIONS	11
APPLICATION PROGRAMMING INTERFACES	11
EASIER PROGRAM CHANGE REQUEST PROCESS	11
DISPLAY PROBABILITY OF EXECUTION	11
SIMPLIFIED AND INTEGRATED TELESCOPE CONTROL CONSOLE	12
INTEGRATED TOOLS FOR LOGGING AND SYSTEM ANALYSIS	12
EDITING OF SEQUENCES DURING EXECUTION	12

REFERENCES

1 Purpose and Scope

The purpose of this document is to provide the top-level concepts for the Observatory Control System (OCS) Upgrades Program. The background and motivation of the OCS Upgrades Program are given in the OCS Upgrades Program Mandate. In brief, the current software is difficult to maintain and update and many of the desired features are only possible with a fresh redesign and rewrite. The ideas presented here are based on recommendations from the 2020-2030 strategic plan, previous software planning, the 2015 Science Operations Review, user interviews, discussions of the 2017-2018 OCS Upgrades Working Group, meetings with the Gemini operations teams (eg. SOS, SUSD, TAC teams) during 2018, and the 2019 OCS Upgrades staff working groups.

The legacy system is described in Legacy OCS and Operations Concepts.

The OCS Upgrades Program encompasses several projects that will refresh all the major components of the OCS, see Figure 1. In its first establishment, the OCS Upgrades Program includes a new web-based Sequence Executor, a new Telescope Operations Tool to replace the existing TCC, an updated Proposal Evaluation System and the Gemini Program Platform - a system that replaces the existing Phase 1 Tool and Observing Tools adding an automated scheduler to relieve queue planning burden. These are shown in Figure 1 as chartered projects. In the future, we expect additional projects to fully implement the vision described here. We show some of these in Figure 1 as uncharted projects and will be refined as we complete the first part of the program. This document gives an overview of the entire program. This document is written so that Gemini staff and external reviewers can understand the motivation for the work and the top-level concepts of the full proposed system.

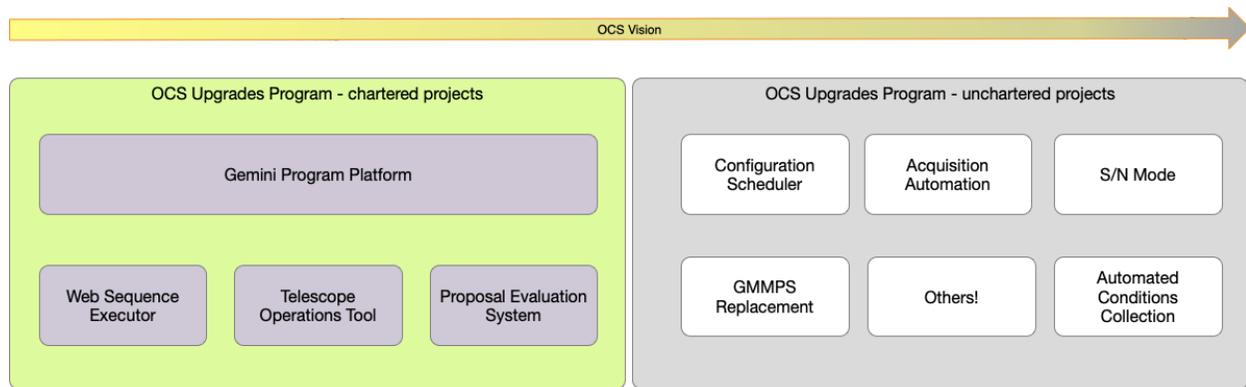


Figure 1. A visual representation of the different project components of the OCS Upgrades Program.

2 Applicable Documents

- Gemini Operations Model 2020s
- Legacy OCS and Operations Concepts
- OCS Upgrades Program Mandate

3 Justification of Changes

As mentioned in the Legacy OCS and Operations Concepts documents and the OCS Upgrades Program Mandate, the current OCS has a variety of limitations that need to be addressed. The main issues are that the tools can be difficult to learn and use, they are difficult and slow to maintain, and some of the processes are time-consuming and inflexible. In addition, new constraints and features required for automated scheduling and time-domain astronomy (TDA) cannot be implemented in the legacy system.

4 Concept of the Proposed System

This section describes the top-level concepts for the proposed system. The concepts are then illustrated by user stories or scenarios in the companion software requirements documents (SRDs). This must work within the context of the Gemini operations model.

4.1 Concept Design Process

We are using a “goal-aided” process as described in Cooper et al. (2014). This is basically the same process that we used at the beginning of the Phase I/II project in 2011-2012. Research is used to generate user goals, “personas”, and requirements from stories or scenarios.

The research has included:

- Brainstorming from previous projects such as Phase I/II, OCSINF, and OCSADV
- The results of the 2015 SciOps review
- Previous staff/user requests - “Future” JIRA tasks
- User surveys over the years
- 2017-18 OCS upgrades working group discussions. This group included Gemini staff and NGO and user representatives.
- Stakeholder interviews: individual and groups
 - Staff: SOS, astronomers, SUSL, FT/TAC groups, time accounting group
 - NGOs
 - Users (PIs)

The information that was collected can be found in the following locations.

2017-18 OCS Upgrades Working Group Drive Folder
2019 OCS Upgrades Staff Working Groups Drive Folder
SciOps Review 2015 Drive Folder
OCS Upgrades Issues Spreadsheet
Software Development JIRA (Gemini staff only)

4.2 Top-Level Goals

The following are the fundamental goals for the stakeholders of the system.

4.2.1 Institutional Goals

- Prepare Gemini for its role in the next decade as the prime facility for execution of flexible, innovative and efficient science programs
- Provide equitable treatment of all Observatory Participants.
- The observatory remains in demand by the communities as measured by proposal over-subscription rates
- The observatory remains productive compared to comparable observatories as measured by program completion statistics and publication rates.
- Support increasing rates of ToOs and Gemini's participation in time domain follow-up networks (e.g. AEON)
- Support new facility instruments and upgrades to existing instruments
- Encourage and support the use of visiting instruments
- Maintain or improve telescope system performance and avoid obsolescence
- Provide an interesting and productive workplace for staff in order to retain existing employees and to attract quality applicants

4.2.2 Technical Goals

- Modernize the software infrastructure to enable Gemini to support and sustain these software systems in the future with the planned software staffing levels.
- Minimize technical debt (avoid the short-term, quick solution rather than the proper solution)
- Have a secure software/IT environment that will allow only authorized access to user data and Gemini data and infrastructure.
- Provide automated regression testing as part of the compilation and deployment process.
- Make the code publicly available so that other observatories can view it, adapt it for their use, and even make contributions.
- Provide APIs for communicating with other Gemini systems such as the QAP, the science archive, acquisition tools, GEA, and the FRS.

4.2.3 Science Staff Goals

- Enable robust and efficient execution of observations at night by improving and streamlining the observer and operator experience (e.g. acquisition improvements, sequence editing, access to finding charts, integrated logging)
- Enable, at the least don't preclude, single-person operation
- Reduce staff busy work via intelligent telescope scheduling (instrument changes, visiting instruments, classical and PV runs, etc)
- Reduce staff workload on routine tasks related to observation checking
- Make it easier to query information in the databases via APIs for custom reports and investigations, including time-use reports.
- Make the observation change request process easier and more transparent.
- Streamline the MOS mask tracking and checking process
- Have better interconnections between different observatory tools (e.g. logging, GEA, FRS, QAP) to avoid duplicating data entry and make system troubleshooting and time accounting easier and more accurate.

4.2.4 Science User Goals

- Have an easy way to identify the resources that match the user's science needs.

- Be able to include observations for both GN and GS in the same program and be able to execute them in a coordinated way while avoiding duplications.
- Use the ITCs more effectively for finding the correct exposure times or SNR and creating proposals.
- Specify conditions constraints using physical conditions on target instead of conditions bins.
- Have the option to use signal-to-noise ratio (SNR) as an observation constraint.
- Have a straightforward, easy-to-remember way of preparing basic observations
- Have an easier-to-use, more capable, and more integrated MOS mask design tool.
- Have more control of observation status, including being able to submit observations without a manual check process and to withdraw observations from the queue.
- Have more capabilities of interacting with the Gemini system programmatically.

4.3 New System Concepts

The new OCS has to meet the same overall needs as the current system, so the top level functional workflow is the same. However, in order to meet the goals the system needs to be organized in a more cohesive way, bringing together the separate tools and software models into a more unified system that allows the pieces to work more closely together. The processes should be automated wherever possible, but the user must always be able to operate the system manually. This section gives a description of some of the major changes envisioned for the system.

4.3.1 ITC Integration

The Integration Time Calculators (ITCs) should be incorporated as a service that can be used by any of the OCS tools. The OCS should be able to use the ITCs to calculate either S/N or exposure times, numbers of exposures, and coadds, and use those results in observations.

4.3.2 User-oriented Parameters

The legacy OCS abstracts many common concepts, such as conditions constraints, that are often hard for new users to understand. Also, information about basic instrument capabilities are distributed on various instrument web pages. This makes it difficult to determine what Gemini offers and whether that is appropriate for a given experiment. Therefore, a general concept of the new system is to use parameters that are more common to scientists and are directly related to the users' requirements.

4.3.2.1 Physical Conditions Constraints

Users should be able to use physical conditions on target (e.g. arcseconds for IQ, magnitudes of extinction, mm of water vapor, etc) instead of percentile bins. The system should allow the presentation and entry of the constraints using either continuous or discrete ranges (e.g. IQs of 0.3, 0.4, 0.5,... arcsec). The distributions may need to be mapped to percentile bins, so the system may need to store either the cumulative distributions or defined bin limits, which may change with time. Additional constraints such as Strehl ratio and contrast will be needed for some instruments.

4.3.2.2 Enable Signal-to-Noise as a Constraint

Gemini should move towards the long-standing goal of adding signal-to-noise ratio (SNR) as a constraint option for observations for which it is appropriate in order to provide more flexibility for scheduling and executing observations. The fundamental driver is to make more efficient use of telescope open time and to improve the completion rates of all programs, especially those in Bands 2 and 3, or equivalent.

4.3.2.3 Select instrument/mode based on science requirements

It should be possible for a user to enter a set of science-oriented requirements for their experiments such as observing mode (imaging/spectroscopy), bandpass or wavelength coverage, spectral resolution, etc and receive a list of instrument configurations that satisfy those requirements. If target information is available, then the ITCs can give a direct S/N or time comparison and even suggest what option has the highest likelihood of execution. For PIs preparing Phase 2 ToO observations the list should be restricted to the instruments/configurations that are currently mounted on the telescopes. Also related to ToOs, the system should be able to accept generic, instrument and site independent observing requests that can be executed at either site.

Users who know what instrument and mode that they need, including staff doing day checks or engineering, should be able to define and start configuring an observation directly.

4.3.3 Timing Constraints

It should still be possible to have multiple timing windows for each observation. In general there also need to be timing constraints between observations (e.g. do between M and N days after a previous observation) for cadence or monitoring observations. Airmass and hour angle constraints are also considered timing constraints since they limit the times when an observation can be observed on a given night.

4.3.4 Sequences

Sequences should include both acquisition and science steps, with acquisitions inserted as needed. The OCS should also be able to trigger actions within an instrument, e.g. a configuration or internal calibration process, that may not produce a FITS file.

Some sequence steps may be dependent on each other, for example a science spectrum and the required calibration, or an on-source exposure with the necessary sky. There should be ways defining these associations as well as other rules for how the sequences are constructed, scheduled, and executed.

4.3.5 Valid Observations Created Automatically

The OCS should always create valid observations. The defaults will be based on templates and parameters defined by the instrument teams. Options that users are allowed to change should be valid for the instrument and context. The system should produce an error, and not allow an observation in the queue, for configurations that are not possible and provide a warning or advice if a configuration is valid but not recommended. There should be an observation validation service for users submitting requests via APIs. Invalid observations will not be accepted. This will greatly reduce, and hopefully eliminate, the need for human Phase 2 checks.

4.3.6 Automatic Calibrations

The OCS should also have rules and templates for the calibrations required for each configuration. “SmartGCAL” GCAL configurations should continue, enhanced by automatic observations for photometric, Telluric, spectral response, and other baseline calibration observations. The system should automatically generate these and be able to pick optimal calibration targets from provided catalogs. The ITC should be used to set the exposure times. The configuration of the calibration observations should regenerate or update to match any changes made to the science observations. Staff should be able to disable calibrations.

Investigators should be able to add program standards and specify how they should be scheduled with respect to the science observations.

The OCS should also be able to generate lists of daily or weekly calibrations such as biases or darks based on the observed configurations.

4.3.7 Promote Sufficient Calibrations

There is often resistance to preparing baseline calibration observations that a program may not need immediately. This is mostly to save time in preparing the observations. However, these calibrations may be useful to future archive users and should be taken. One goal in automating the creation of calibration observations is to eliminate the resistance to taking baseline calibrations. This will be easier on the current teams and make future archive use easier since all calibrations will be available. However, there should also be a way for staff to override the default calibrations for the cases in which the calibration plan interferes with the science.

The observatory should also promote the taking of other types of calibrations, such as program calibrations. Many of these can be shared between programs. One suggestion is that adding program standards should not require a change request as long as the proprietary period is waived, the total time required is less than 10% of the total allocated time, and there are no duplications with science observations.

4.3.8 Program Timescales

The OCS should be able to handle proposals and programs on any timescale. It should be able to support multiple call-for-proposals simultaneously, with different capabilities and visibility constraints. Deadlines should be settable and not restricted by the underlying system. It should be possible to set the start and stop dates for when a program is active and these should also be clearly visible in the program.

4.3.9 New Observation Workflow

If observations are always valid or have sufficient automatic checks and calibrations are created or configured automatically to match the science then there is much less need for manual checks of every observation. Also, with increasing numbers of ToOs it will be more efficient for the program teams to be able to more directly manage their observations. The goal is to eliminate the requirement for manual observation checking (it can be optional) and allow PIs to submit observations directly to the scheduling queue and withdraw them if needed.

4.3.10 GN/GS Observations in the Same Program

There has been a longstanding need to include GN and GS observation in the same program or have an easy way to move observations between sites. It was decided in October, 2018 that a fundamental requirement of the OCS is that a program must be able to contain observations for both GN and GS instruments. This allows for more flexibility of where equivalent observation can be executed and makes it easier to schedule the two telescopes in a coordinated manner. The latter is considered important for TDA and MMA follow-up. This decision has a variety of implications, the main one being that the database and scheduler would be in a single location and may not be local to either of the sites.

4.3.10.1 Real-time Scheduler

Therefore, one of the primary deliverables of the GEMMA TDA component of the OCS Upgrades Program will be an automatic observation scheduler. The scheduler should run in real time and generate new observing plans for both telescopes within minutes as events occur. This will make the telescope much more adaptable to changes at night, naturally handling changes in conditions and ToOs, as well as significantly reduce the workload of the QCs and observers.

The real-time scheduler should be able to make schedules several months into the future based on forecasts, weather statistics, and the telescope calendar. This capability will also be used to run simulations of observing using different scenarios for weather, instruments, and weighting rules.

The scheduler should continue to a manual mode, like the current QPT, for engineering and other special needs. A mixed mode that would allow a QC to fix some observations in place and then have the scheduler fill in around them would be desirable.

4.3.11 Queue Filling and Long-term Scheduler

The queue filling algorithm used by the proposal evaluation process (NTAC and ITAC equivalents) will need significant changes to optimally handle SNR mode and observations that can be done at either site. In all cases the queue filling should take seasonal weather variations into account. That is, the amount of time available in the RA/Dec/conditions bins will need to be dependent on RA and site.

There is a need for a long-term scheduler that will analyze telescope calendar, scheduling constraints, instrument, mode, and target pressure for a configurable number of months and make recommendations for instrument swaps, classical and PV runs, etc. This will be usable by the queue filling process. For example, observations that cannot be done for logistical reasons, e.g. a timing constraint during an engineering shutdown, will be excluded. This will also be able to assist the HSOs and QCs by suggesting the dates for instrument swaps, scheduling blocks, and instrument component (e.g. MOS masks, gratings) changes. This tool should also be callable by the real-time scheduler so that the results can be used in observing simulations.

4.3.12 Program Ranking

The system should be able to support either ranking bands or continuous rankings. If ranking bands are used then the number and names of the bands may change with time.

4.3.13 Time-Critical Observations

Proposers should be able to request time-critical status for observations that have very strict timing constraints or unique opportunities that warrant the highest priority and cannot be interrupted by rapid, or interrupting, ToOs. Time-critical proposals should be given the highest ranking (band).

4.3.14 Target of Opportunity

A Target of Opportunity (ToO) observation is one that does not have a defined target when the proposal is submitted. ToO program status for programs should be kept if the OCS needs to respond differently to ToO programs or observations. Three types of behavior have been identified: 1) slewing and getting on target as fast as the system will permit, including interrupting an ongoing observation; 2) slew and observe as soon as possible, but do not interrupt an ongoing observation; and 3) treat as any other observation.

4.3.15 Observation Associations

Observations are often associated and it must be possible to reflect and control these associations. All the scheduling information transmitted now in notes should be captured in a way that is meaningful to the scheduler. The kinds of needed associations include groups of related observations, timing constraints, and AND/OR logic.

4.3.16 Application Programming Interfaces

It should be possible to interact with the system programmatically. This is usually done using defined Application Programming Interfaces (APIs). Some interfaces will be open to external users while others may be for internal use only.

The external APIs should include:

- Telescope status (e.g. open, closed, accepting ToOs) and available instrument configurations
- Program and observation information (information and status)
- Validation and submission of observation requests
- Receive a list of observing modes that match science requirements
- ITC results
- AGS results

The internal APIs should include:

- Target/instrument duplication checks against programs in the databases (esp. active programs)
- Generic search options for extracting information from the ODB for custom reports and graphs.
- Interfaces needed to interact with other observatory systems such as the QAP, seqexec and TCC servers, the science archive and FITS servers, LTTS, GEA, and the FRS.

4.3.17 Easier Program Change Request Process

Certain changes to programs such as adding targets and using better than approved conditions or timing windows require the approval of the appropriate Head(s) of Science Operations (HSOs). Tools should be included to make this process easier. Any changes that require approval should generate warnings and the system should provide a convenient way to submit the change request, perhaps by doing it

automatically. All necessary people, for example contact scientists and HSOs, should be alerted. There needs to be an interface for HSOs to see requested changes, review automatically-run duplication checks, and approve or reject the requests. Reports for staff, especially QCs and CSs, should show timing windows and changes in constraints.

4.3.18 Display Probability of Execution

Investigators need to have good awareness of the probability that an observation will be executed within the start/stop dates of the program so that they can make informed and proper decisions. This is especially important as we move away from conditions percentile bins, which explicitly give information about the likelihood. Therefore, the OCS should calculate the likelihood of execution and present that clearly at any stage of the process where it is needed. It should be visible in the UI and available with an API. The algorithms will need to be defined but will likely include conditions distributions, target visibility, instrument and component availability, and science rank at Phase 2.

4.3.19 Simplified and Integrated Telescope Control Console

The Telescope Control Console (TCC) should be reorganized and restructured to make it easier to use. The design should not preclude future single-user operations. It should also be integrated with the rest of the OCS infrastructure to allow better coordination between systems and automation.

4.3.20 Integrated Tools for Logging and System Analysis

An important improvement to the daytime and nighttime operations would be to integrate the various tools that are used in order to eliminate the duplication of information and the need to copy/paste information from one application to another. One example of this would be to integrate the nightlog and the obslog into a common logging system. The nightlog and obslog could then be the results of queries to the logging system and the ODB. Likewise, there should be connections to the Fault Reporting System (FRS), the Gemini Engineering Archive (GEA), and the FITS servers.

These changes would allow the development of a timeline tool that would be useful to both day and night staff. This would show the observations taken during a night, comments, weather conditions, and the times of fault reports, and links to data. This would simplify any time accounting and data checks that need to be done.

4.3.21 Editing of Sequences During Execution

Observations and sequences should be organized to make sequence editing (e.g. to repeat problematic steps) easier and more robust. It should be possible to easily add steps without changing the visualization of what has occurred or altering the intention of the subsequent steps.

4.3.22 Acquisitions

The OCS should either incorporate or be able to interact with the acquisition tool via an API. The OCS will need to know when an acquisition is done so that it may proceed with the science steps. The acquisition tool should also support automated acquisitions in order to speed up that process.

4.3.23 Easier Access to and Use of Finding Charts

Finding charts should be associated directly with the observations for which they are needed. It should be possible to view the finding chart, a sky-survey image of the field, and the Gemini image at the same time. It should be possible to change the orientation (rotation/flip) and scale of displayed images.

4.3.24 MOS Mask Handling

MOS mask design files should be associated directly with the observations for which they will be used.

The MOS mask handling process should be integrated into the OCS to improve automation and reduce the number of tools that need to be used and the number of steps. The Gemini MOS Mask Preparation Software (GMMPS) is a Tcl/Tk application that needs to be rewritten. The new tool should be able to communicate directly with observations so that configurations can be read and updated. The stand-alone Instrument Configuration Tracking Databases (ICTD) should also be incorporated into the OCS in order to simplify mask handling process and the interactions with the scheduler.

4.3.25 Eavesdropping and Remote Observing

The system should make it easier for PIs and observers to manage eavesdropping calendars and make connections. The design should not limit the ability to enable remote observing in the future.

4.3.26 Undo/Redo Capability and History Tracking

There should be some level of change Undo/Redo built in.

There is a strong need to have a more detailed history of changes, including the identification of the person making the changes, than is available in the legacy system. If observations can be edited during execution then it would be very useful to be able to view earlier versions so that the original intent of the investigators is clear.

5 References

Cooper, A., Reimann, R., Cronin, D., and Noessel, C., 2014, *About Face: The Essentials of Interaction Design*, Fourth Edition (John Wiley & Sons: Indianapolis). Available on the Gemini Kindle's.