United Kingdom Infra Red Telescope

Joint Astronomy Centre

Royal Observatory Edinburgh

Observation Chooser: Requirements

Alan Bridger and Gillian Wright

orac014-chor Version: 01 **Original: 3 June 1998 Modified: 17 June 1998**

This document sets out a specification and list of requirements for the Observation Chooser, part of the Observatory Control System, part of the ORAC (Observatory Reduction and Acquisition Control) project.

1.0 Introduction

1.1 Purpose

This document is intended to outline the specification of the Observation Chooser task in the UKIRT Observation Control System and provide a list of requirements. The requirements list will consist of both user requirements and software requirements. It is intended to be read by anyone with an interest in the UKIRT Observatory Control system, and in particular by the Michelle project.

1.2 Scope

The Observation Chooser is intended to form the initial user-interface to the Observatory Control System part of ORAC. A possible later replacement for it, the Observation Scheduler, will be considered in a separate document. The Chooser will read the *Observing Database* of *Observation Definitions* (created by the *Observation Preparation System*) and present it to the Observer, allowing him or her to select which Observation Definition is to be performed next. The Chooser will then extract this Observation Definition, translate it into an *Observation Definition Sequence*, understandable by the Observation Sequencer, and instruct the Sequencer to load it. The Chooser will also extract the information that forms the Instrument Configurations and Target Descriptions for this Observation Definition and make them available to the relevant systems. Finally the Chooser will form the User-Interface to the Observation Sequencer also, allowing access to the functions of that task and providing status feedback. A summary status from other systems may also be provided.

1.3 Overview

The rest of this document will briefly describe the UKIRT Observation Chooser and its relationships to other systems. A comparison with previous similar systems will also be made. The system context will be described and a brief design presented.

Finally a list of specific user and software requirements will be given.

2.0 General Description

2.1 Relationship to Other Projects

The Observation Chooser is part of the UKIRT Observatory Control System, which in turn is one part of the UKIRT Observatory Reduction and Acquisition Control (ORAC) project. References [O-1] and [O-12] give an overview of this project and summarises the relationships.

2.2 Predecessors

There is really no predecessor to the Chooser in the existing UKIRT system. One approximately equivalent function is the loading of an EXEC into the AIM task - but most of the functionality of the Chooser simply does not exist in that system.

2.3 Function and Purpose

In summary the main functions of the Observation Chooser are:

1. To present the User with a list of available Observation Definitions.
2. To allow the user to select one and have it executed by the Sequencer.
3. To translate the Observation Definition into forms readable by the other systems.
4. To allow access to the functions of the Observation Sequencer.
5. To provide feedback on the progress of the Observation Definition.
6. To provide a summary status of all the observing systems.

2.4 Environmental Considerations

The Observing Database that is to be read by the Chooser will hold files written in SGML by a UKIRT variant of the Gemini Observing Tool. This clearly places practical constraints on the Chooser, but also suggests that the Observing Tool is used an initial model for the interface. Given the existence of the SGML parsing code in the OT there is clearly a strong case for re-using this code if possible, which would mean that the Chooser should be written in Java.

The use of Java might imply that portability is not a concern, but in fact the possible need to communicate with DRAMA tasks will place constraints on the operating system used, and it is likely to be Solaris. Note that the existence of code to command Drama tasks from Java strengthens the case for using Java (the “TODD”, [X-14]).

2.5 Relationship to other systems

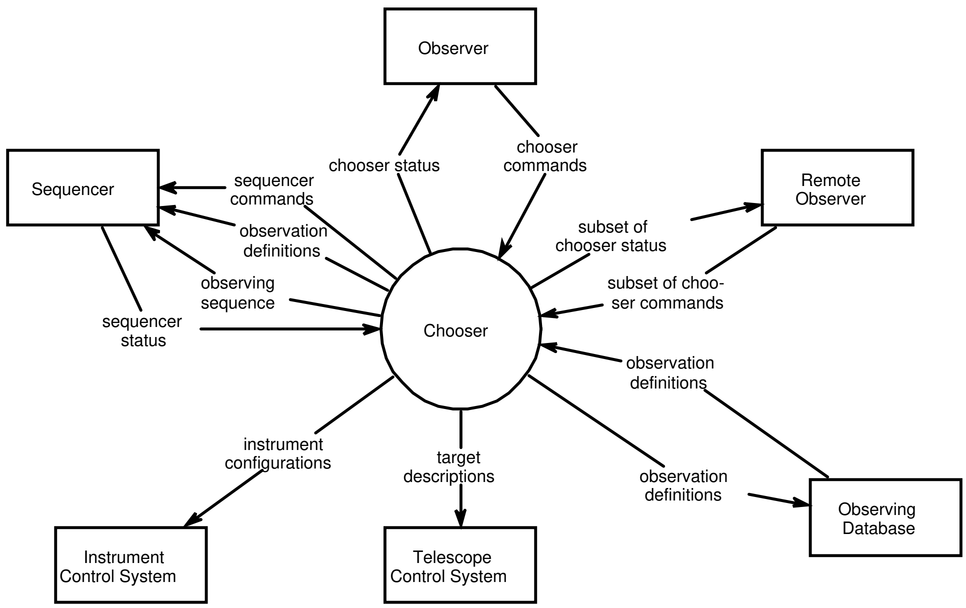

FIGURE 1. Observation Chooser context diagram

Figure 1 shows a context diagram for the Observation Chooser. In this diagram the external entities are:

Observer. This represents the Observer, issuing commands to the Chooser.

Telescope Control System. This is the existing UKIRT telescope control system.

Remote Observer. This represents a potential remote observer allowed to send commands to the Chooser. The commands sent and the status received may be a subset of the full range.

Instrument Control System. This is the system that has overall control of the instrument - it can configure it and instruct it to acquire data. The instrument includes any array controller and data acquisition system.

Observing Database. This is a database of Observation Definitions as prepared by the Observation Preparation System.

Sequencer. This is the Observation Sequencer ([O-13]), responsible for executing the Observation Definitions chosen.

The major data flows on Figure 1 are:

chooser commands. This represents the commands to select and perhaps edit observations that come from the Observer.

chooser status. This is a flow of status information - responses to **chooser commands** or displays of the observing status.

subset of chooser commands. This is a subset of **chooser commands** suitable for a remote observer.

subset of chooser status. This is a subset of **chooser status** suitable for a remote observer.

instrument configuration. This is how the component of an Observation Definition that represents an instrument configuration gets to the instrument system. This flow is likely to go via a temporary file.

Observation Definitions. This is the flow of **Observation Definitions** to and from the database, as created by the preparation system.

target description. This represents the *target description* component of an *Observation Definition*. As with an instrument configuration it is likely to be implemented using temporary files.

2.6 General Constraints

These are set out in Section 2.4.

2.7 Model Description

The Observation Definitions in the Observing Database will be held in the original SGML documents written by the Preparation System (which is a version of the Gemini Observing Tool). These could be single files for complete observing programmes, though there is nothing to stop the use of the system in ways that would create files relating to much smaller blocks.

The initial model for implementation is to reuse at least part of the Gemini Observing Tool. This will allow significant code reuse and provide a familiar interface to users. The user will thus be presented with the list of Observation Definitions as created by them using the OT. They may select one of them, and then request that it is the next to be "executed". The Chooser will then translate the Definition into the components required by the other systems (viz. Observing Sequence, Instrument Configuration, Target Description) and also create a sequence version of the Observation Definition (an *Observation Definition Sequence)* for use by the Sequencer. These outputs will be stored as temporary files and the Sequencer then instructed to Load and Run (the latter perhaps after confirmation by the Observer) the Observation Definition Sequence.

It is envisioned that the same container frame will provide windows monitoring the status of the other systems, the status of the Observing Sequence and a control window for the Observation Sequencer. There will be one control screen and one status screen for each Observation Sequencer.

Question: Should the full functionality of the OT be present, or should this be "relegated" to a different screen to avoid clutter?

Finally it should be noted that the Observing Database will also require a set of tools that will enable it to be interrogated off-line and edited (e.g. old observation definitions removed). Some of these may come with the Gemini OT, some may be required. This needs to be taken into account.

3.0 Specific Requirements

3.1 Functional Requirements

FC1. The Chooser must present a list of Observation Definitions from the Observing Database that are available to the User.

FC2. The Chooser must allow the User to select an Observation Definition from the list presented.

FC3. The Chooser must be able to translate a selected Observation Definition into the following components: A Target Description, An Instrument Configuration, An Observing Sequence, An Observation Definition Sequence. The mechanics of the translation must be defined.

FC4. The Chooser must be able to store the translated components of an Observation Definition in appropriate places.

FC5. The Chooser must be able to inform the Sequencer of the Observation Definition Sequence name that is to be executed.

FC6. In creating the Observing Sequence the Chooser must automatically insert certain commands at appropriate places, as determined by its current setup (see FC7.)

FC7. It must be possible to alter properties of the Chooser that define where certain commands must be inserted automatically into the Observing Sequence. The exact commands and the allowed insertion points are TBD. (But are probably BREAK after certain commands, BreakPoint at “sensible” stopping places)

FC8. The Chooser’s status display of the Sequence must highlight “sensible” stopping points, but not show the actual command.

FC9. The Chooser’s status display of the Sequence must be able to “hide” detailed commands that are part of certain Standard Sequences (see Section 3.8 in[O-12]).

FC10. Despite FC9. it is likely that details of Library Sequences will need to be seen.

FC11. The Command Interface of the Sequencer must be available in a window

FC12. It must be possible to operate more than one Sequencer, with the appropriate displays, each connected to a different instrument.

FC13. Must be able to dynamically affect a very few parts of the observing sequence (e.g. increase number of pairs taken).

FC14. A summary status of other systems should be available (ICS,TCS,DHS) as feedback.

FC15. It must be possible to present a “next” target to the TCS to allow confirmation of availability and airmass, etc. In the first instance a simple answer is to provide a function that will create a “userfile” from all the targets present in an Observing Programme. Then the checking may be done as at present. A more intelligent approach is likely to be part of the Scheduler.

3.2 Security Requirements

SC1. The User must only have access to Observation Definitions he or she is authorised to view and use. Exactly what this means needs to be defined.

3.3 Documentation Requirements

This is a list of the documentation requirements for the system. Similar requirements are placed on other parts of the ORAC project, and the numbering is the same where they are duplicated.

DO1. A programmers guide is required both on paper and html.

DO2. A user guide is required on paper and html.

DO3. User guide to be accessed from “help” in GUI.

DO4. Link to instrument cookbook written by support scientist (*e.g.* CGS4 manual).