United Kingdom Infra Red Telescope

Joint Astronomy Centre

Royal Observatory Edinburgh

Observation Database: Concept

Alan Bridger and Gillian Wright

orac019-odbc Version: 01 Original: 25 January 1999 Modified: 22 February 1999

This document sets out a conceptual design of the Observation Database, part of the Observatory Control System, part of the ORAC (Observatory Reduction and Acquisition Control) project.

1.0 Introduction

1.1 Purpose

This document outlines the design of the Observation Database in the UKIRT Observation Control System. This design is intended to meet the requirements that can be distilled from those set out in the Chooser Requirements document [O-14] and the Observation Preparation System requirements [O-4] and those implicit in the overall system design. It is intended to be read by the implementers and maintainers of any related ORAC system.

1.2 Scope

The Observation Database is a crucial component of the overall ORAC system. As well as the actual database itself, and the basic database management system, the concept set out here is intended to cover the management requirements specific to the ORAC system and the tools that will be needed to maintain the database.

2.0 Overview

The rest of document will briefly describe the relationship of the Database to the rest of the ORAC system and the system context will be summarised. The major functions of the system are outlined, and a list of specific user and software requirements is provided. A conceptual design to meet these requirements is proposed.

3.0 General Description

3.1 Relationship to Other Projects

The Observation Database 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.

3.2 Predecessors

The approximately equivalent function in the current UKIRT system is the directory hierarchy of *Config*s and *EXEC*s, divided by *userid*, that exists at the telescope.

3.3 Function and Purpose

In summary the main functions of the Observation Database are:

1. To store Science Programmes which contain the Observation Definitions.
2. To store Observing Plans and Semester Plans
3. To allow authenticated storage and recall of these documents
4. To allow authenticated queries to be performed on the documents in the Database.
5. May be extensible to include storage of Phase 1 documents, depending on the UKIRT plans for electronic submission. (but this has not been costed, since not currently part of ORAC).

3.4 Environmental Considerations

The Observation Database should be designed to be portable so far as is reasonable. However, it and its server must be available on the network on a 24 hour basis. This may place practical restraints on the actual implementation. If a commercial database is chosen for the system then this too might place constraints.

3.5 Relationship to other systems

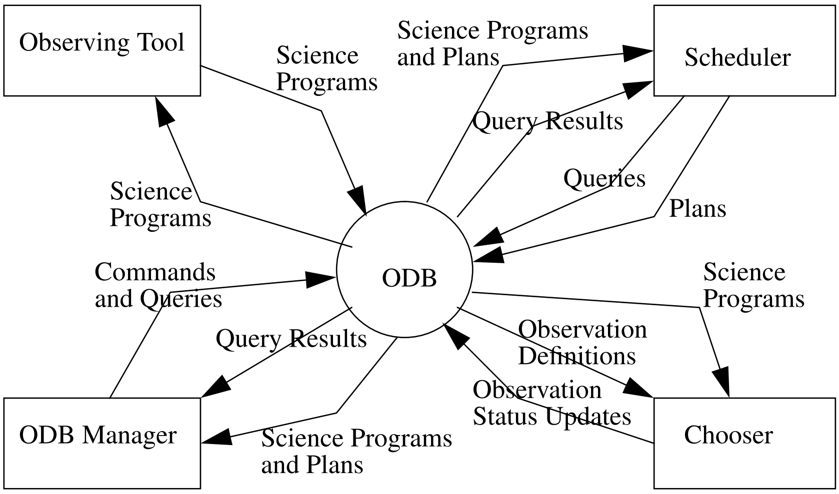

FIGURE 1. Observation Database context diagram

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

ODB Manager. This represents an Observatory Staff member acting as a manager of the ODB.

Observing Tool. This represents local or remote ORAC-OT clients.

Scheduler. This represents the Scheduler tool.

Chooser. This represents the Chooser part of the OCS.

The major data flows on Figure 1 are:

Science Programs. This is the flow of Science Programs to or from the ODB from or to clients.

Queries. This is a flow of queries sent to the ODB by a client.

Query Results. This is the results of **Queries** sent back to clients.

Plans. This is the flow of Observing Plans or Semester Plans to the ODB from the Scheduler.

Science Programs and Plans. This is the flow of **Science Programs** or **Plans** from the ODB to a client.

Observation Definitions. This is the flow of **Observation Definitions** from the ODB.

Observation Status Updates. This represents the updating of an **Observation Definition** with status information following its execution.

Commands and Queries. This represents either **Commands** or **Queries** to the ODB from the ODB Manager.

3.6 General Constraints

These are set out in Section 3.4.

3.7 Model Description

Briefly, the Observation Database is the repository for all of the Science Programs submitted and for the Observing Plans and Semester Plans created by the Scheduler. Its specification also encompasses the management system that must be used to interact with the database. It will reside at the JAC, possibly at the telescope. The database forms a crucial part of the system and will be divided into a number of areas: The Science Program database, the Observing Plan database and the Semester Plan database. It may also encompass a Phase 1 database. In addition at least the Science Plan database will be further divided into “pre-active”, “active” and “retired” sections, indicating the status of the Science Program. During its stay in the active database a Science Program will be updated by the ORAC-OCS with information on its current status.

The Observation Database will be accessible by all UKIRT users and staff, but access will be restricted, requiring authentication to see Programs or Plans. Appropriate staff will normally have rights to see all of the database.

3.8 Scenarios.

This section provides a brief description of the use of the database by observers and staff.

When an Observer has prepared a Science Program or Observation Definitions using the ORAC OT (which is a UKIRT specific version of the Gemini OT) they will submit them to UKIRT, where they will be stored in a database. The user may also get local copies of their defined programs from the database so that they may modify them and re-submit. For classical observing and current operations practice the database could be very similar to the one in use for UKIRT_PREP, i.e. an ascii file structure based on the users name or perhaps on the PATT run number, with the user responsible for “keeping it tidy”. However for flexible scheduling there are additional requirements, which mean that something more sophisticated is needed. The SGML files which are submitted from the OT may describe a programme of observations of several targets, but the Scheduler may schedule only one of the targets on a given night. For example the RA range of a programme may be too large for one night, or each target may need different conditions or the K band spectroscopy gets scheduled for all targets but not the H, etc.). Therefore the programs in the database must also contain information which the Scheduler can use to determine whether or not a target has been observed and hence whether to schedule it. In addition both the users and UKIRT staff need a user-interface through which they can enquire about the status of programs, and change the status if need be. In addition there is a need to be sure that all programs in the database which the Scheduler is selecting from are up to date - i.e. that they refer to valid programs for scheduling (not the ones from this time last year) and that they describe approved programmes. Thus some form of validating programs in the database is required. Since programmes will be given a priority when they are awarded time this information needs to incorporated in the database, perhaps as well as relative priority of targets set by the user. Finally, as is the case with the current system, it is desirable for old programs to be kept in the database so that users can recall them and modify them to create new ones.

4.0 Requirements

This section first summarises those requirements explicitly on the database which have been included in previous ORAC documents. A list of functional database requirements based on these and the above scenario is then given.

4.1 Requirements from the Preparation System (orac004-opsr)

The remote submission of definitions also implies the setting up of a server and database at UKIRT and putting in place appropriate security and authentication protocols.

FP9. Format of stored components tbd. (ASCII?)

4.2 Requirements from the Chooser (orac014-chor)

The Observation Definitions in the Observation 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.

It should be noted that the Observation 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.

FC17. The Chooser must be able to write back to the Observation Database the state of the Observation Definition it has executed. This information should be written whether or not the Observation was successfully completed.

4.3 Functional Requirements

FDB1. Programmes must be submitted to the database from the OT, possibly running at the users home institute, “transparently” without the user needing any details of where it ends up.

FDB2. Simple checking for silly mistakes/incomplete programs is required. This could be done by the database, but is perhaps better handled in the OT.

FDB3. Some feedback when submitting programs is needed

FDB4. UKIRT staff may wish to check validity of submitted programs before they can be queue scheduled.

FDB5. Programs may be added to the database at any time - e.g. target of opportunity, service programmes.

FDB6. Users may also retrieve copies of their Programs from the database.

FDB7. When an Observation has been completed, the corresponding program will be updated to indicate this. Details of precisely what info is included (date? time? weather stats?) are to be determined in consultation with UKIRT staff.

FDB8. User interface to provide graphical/statistical information concerning completion status of programmes.

FDB9. Completion status of Observations and Programs may be changed by UKIRT staff.

FDB10. Nature of program (flexible, classical, service...) may be changed by UKIRT staff.

FDB11. Status information for their own programs to be available to the user.

FDB12. Whether or not the Observing Database user-interface will be a tool for users to retrieve data, or look at “thumbnails”, or be upgradeable to do this is tbd during preliminary design, in consultation with UKIRT staff.

FDB13. If the Observation Scheduler has been used to save observing plans for several nights/weeks for predicted weather conditions then the user-interface will provide an option to use these to provide predicted completion status for programmes.

FDB14. UKIRT staff can declare all programs as “retired” at end of semester.

FDB15. Old Programs with their Observation Definitions should be kept in the database, since users may wish to get them and copy information into new Programs.

4.4 Security Requirements

SM1. 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.

SM2. The TAG chairman will need access to the status information in the database too.

4.5 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 in html and on paper.

DO2. A user guide is required in html and a paper copy.

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

5.0 Proposed Concept.

In this concept the database consists of a number of related but distinct areas: Preactive, active and retired Science Programs, Observing Plans and Semester Plans. It should be noted at this point that Phase 1 proposals may also form a separate area, but Phase 1 handling is not part of the ORAC project. These areas (outlined further below) hold the documents necessary for flexible observation at the telescope. Apart from the ORAC client tools that will access the database it is also clear that a number of support tools will be required to aid the Observatory Staff in the management of the database. These tools are here collectively referred to as the ODB Management System.

In a typical envisioned use the following steps would be followed:

1. Science Programs will be placed in the “Preactive” area. These may be submissions from ORAC-OT clients or they may be skeleton Programs that have been created from Phase 1 submissions.

2. At some given time (to be identified) each Program will be "sanity-checked" by Observatory staff and moved in to the "active" area. It become available for scheduling.
3. The Scheduling Tool (ORAC-ST) will be used by Observatory staff to create Semester Plans, which will be stored in the appropriate area.
4. ORAC-ST will also be used to schedule the Observations in one or more Science Programs (depending on whether or not the Program is flexibly or classically scheduled the Observations may or may not be scheduled together) to create Observing Plans. These are also stored in the appropriate area.
5. Alternatively the Observations may be executed directly from the active Science Programs in a classical mode (using the Chooser).
6. During its stay in the active area each Science Program will be updated as Observations are taken and completed. The details of the information that is added is TBD.
7. At given times (TBD) Observatory staff will run a tool that can detect completed Science Programs and place them in the "retired" area.
8. Remote client ORAC-OT users may interrogate the active area to see the progress of their Science Programs. (May also access the retired area?)

5.1 The database areas

The areas identified above are here described in a little more detail.

1. Pre-active Science Programs. These are Science Programs that are not available for Scheduling. They will typically be either Science Programs derived from accepted Phase 1 submissions or they will be Science Programs submitted by Observers using ORAC-OT which have not yet been made active by Observatory staff. All Science Programs in this area are available for both reading and writing (modifying) by authenticated clients. Typically the clients will be ORAC-OT users or Observatory staff using the ODB Management System.
2. Active Science Programs. These are Science Programs that have been made active by Observatory Staff. This means that they are available for scheduling. Once in this area Science Programs may only be modified by authenticated users of the ORAC-OCS (specifically the Chooser) or by the ODB Management System. Other clients (ORAC-OT and the Scheduler Tool ORAC-ST) may only read from this area.
3. Retired Science Programs. This area is where completed Science Programs are placed, usually by an authenticated user of the ODB Management System. No other clients may write to this area, though authenticated ORAC-OT clients may read from it.
4. Observing Plans. These are the Observing Plans that are calculated by the scheduling tool. This area may only be read and modified by authenticated ORAC-ST clients and users of the ODB Management System.
5. Semester Plans. These are the Semester Plans that are calculated by the scheduling tool. This area may only be read and modified by authenticated ORAC-ST clients and users of the ODB Management System.

5.2 Authentication

Access to each area of the ODB must be authenticated. In general the appropriate observatory staff will have full access to all areas of the ODB. For other users (of any client) access to the ODB must be authenticated and will provide access only to Science Programs and Observing Plans that are owned by that user.

5.3 Science Programs

A Science Program is the output (in SGML) of the Observing Tool. It describes the Observations that make up the programme. The information required to fully define each observation is collectively known as an Observation Definition. Clearly a Science Program may contain many Observation Definitions. As the Science Program is set out in a hierarchical manner then some processing is often required before all of the information making up an Observation Definition is obtained.

5.4 Observing Plans

An Observing Plan is an ordered set of Observation Definitions taken from one or more Science Plans. A key detail to resolve here is how this is achieved. References across the database areas would seem to be unavoidable.

5.5 Semester Plans

A Semester Plan is a long-term schedule of nights, Observing Plans or Science Programs.

5.6 The Observation Database Management System

Not to be confused with a commercial DBMS, the ODB Management System is a set of tools that will clearly be required to handle interactions with the ODB. The following tools can be identified, but this list is not exhaustive:

1. A server to support access to the areas, and queries made on them (it is not clear to me how much will be supplied by the commercial system).
2. A tool to allow Observatory staff to browse the ODB and update or modify the areas “manually”.
3. A tool to provide reports on the various areas.
4. A tool to allow Programs and Plans to be moved between areas and also to be deleted.

6.0 Choice of Database Management System

As the requirements placed on the ODB are substantially more sophisticated than those placed on the current system for storing *EXEC*s and *Config*s it is clear that underlying the ODB Management System must be a commercial Database Management System. Some consideration as to the system to be used is given here, but no conclusions are reached.

There are two principal choices, relational (SQL-based) database systems and Object Oriented database systems.

Relational databases are better known, and already supported at the JAC. They are also far more widespread in the commercial world, having succeeded in the marketplace. However, as Science Programs (and by extension Observing and Semester Plans) are Objects, handled by an OO programming language it is not clear that an RDBMS, which uses tables as its fundamental storage unit, will be suitable. There are two possible approaches to solving this:

1. Store all of the OO classes as tables. However, for many classes this might become very difficult and result in very complex queries.

2. Extract all of the Observation Definitions and store them as separate entities. This might become problematic in attempting to relate each Observation Definition to its Science Program and in reconstructing Science Programs for export.

An Object Oriented Database Management System would fit much more naturally to the OO paradigm used in the tools, essentially providing a persistent store for the objects. However, OO systems have not succeeded in the market place, possibly because of their perceived complexity, perhaps because of the lack of a standard query language (nearly solved) and possibly simply because of the effort that would be required to move from an RDBMS. In addition the JAC has no experience in them.

From a program interface point of view it would seem that an OO system would be the best approach, but pragmatically it may be than an RDBMS is better (and the JAC has Sybase). Current plans in astronomy are split: Gemini and the Liverpool Telescopes are going for OO systems, the ESO VLT is going for a relational system (specifically Sybase) even though they use it to store Objects from their OO systems.

7.0 Conclusions

The choice of commercial database management system will need to be made following detailed consultation with the JAC, and also a technical evaluation of the choices. The post delivery support issues may be paramount and it is possible that UKIRT and JCMT will wish to use the same commercial software. For example, since JCMT have already invested effort in Sybase this may drive the choice. Once these discussions have identified which technology to pursue, study of the appropriate database plans and designs for other telescopes will be carried out as a basis for developing a preliminary design for UKIRT.

Given the issues identified here the costs of the ODB are necessarily highly uncertain. A blanket effort estimate for the system of 0.5 dsy, and a requisition estimate of £20000 has been put into the resource estimates outlined in [O-21].