Astronomical Data Analysis Software and Systems VI
ASP Conference Series, Vol. 125, 1997
Gareth Hunt and H. E. Payne, eds.

The VLT Data Flow Concept

P. Grosbøl and M. Peron

European Southern Observatory,
Karl-Schwarzschild-Str. 2, D-85748 Garching, Germany

Abstract. The size and complexity of the VLT require an integrated approach to the handling of science data associated with the operation of the observatory. In this paper, we present the VLT Data Flow concept: a strategy to fulfill the scientific requirements outlined in the VLT Science Operation Plan—namely, ensure optimal efficiency and a high, constant, and predictable data quality. The VLT Data Flow model has been designed following an object-oriented methodology. In the analysis phase, we were able to get a clear understanding of the problem domain, and in the design phase to partition the system into smaller parts with well identified responsibilities. Prototypes are being implemented, and will be tested on the New Technology Telescope (NTT). They will allow us to clarify the astronomical requirements, verify the system architecture, and determine whether new technologies, such as distributed object environments, are appropriate.

1. Introduction

Space experiments normally include software components for performing full calibration and reduction of the acquired data. This is less common for ground based observatories, due to the higher cost involved. However, when size and complexity become too great, even ground based observatories cannot be operated reliably without a global, consistent concept.

In this paper, we argue that the VLT exceeds the limit for facilities which can be managed in an *ad hoc* manner. An important part of the operational concept of an observatory is the flow of science data from the initial proposal to the reduced observations. The concept of a VLT Data Flow System and its architecture are discussed, including first considerations for their realization.

2. Why is a Data Flow Concept Needed?

The VLT is a very large, complex facility which consists not only of four 8 m telescopes, but also of several smaller auxiliary telescopes for interferometry. Two additional facts complicate the operation further, namely: *a)* that the four telescopes have to be considered both as individual units and in any combination, since a combined focus is foreseen, and *b)* that several sites (e.g., Paranal, Chile and ESO Headquarters, Germany) will share operational duties.

Major ground based astronomical observatories have an operational lifetime which exceeds 25 years. A facility like the VLT is likely to have at least this life-time to fully amortize the investment. This corresponds to several generations of electronics, computer hardware, and software. Thus, it must be expected that most of the control hardware and software components will be exchanged several times during the operational period.

The large investment made in the VLT by the ESO community places a large responsibility on ESO to operate it in the most efficient way. This can only be achieved by analyzing the system dependencies, and establishing a well defined work-flow which takes them in to account. The distributed nature of the operation makes this more difficult, and requires a global view based on basic astronomical requirements.

Finally, the observatory should not only work efficiently in a technical sense, but astronomers must also be able to use it in an optimal way. It will be difficult for normal users to develop a comprehensive overview of the operation, due to the relatively short observing time allocated to them. It is therefore important that a clear concept for the astronomical use is presented, so that observations can be specified in a simple but exact way, with predictable results.

3. What Will a Data Flow System Give?

The Data Flow System provides a top-down view of the flow of science data in the VLT environment, with emphasis on astronomical usage (Baade 1995). Several simple, but general concepts were derived by focusing on the scientific requirements.

The architecture, based on the astronomical tasks to be performed by the individual subsystems, makes the design more robust to changes over time (Peron et al. 1996). Clear responsibilities of the different components, together small interfaces between subsystems, improves the maintainability. Upgrading or replacement of components is also facilitated by the global view and its modularity.

4. General Concepts

An important issue for the Data Flow is to present a simple framework of concepts which both are easy for the standard user to grasp, and which hold the system together. The astronomer's view is as follows:

- Astronomical programs for the VLT are prepared and submitted using equipment description and instrument simulators. They are then evaluated in terms of scientific merits and technical feasibility. This functionality is provided by the *Program Handling* subsystem, which supplies the downstream flow with accepted observing programs. In addition, a long term schedule is established to ensure that the correct equipment and resources are in place when needed.
- For each program, the detailed description of the observations required to meet the science objectives must be specified. This is done by breaking the full program up in to *ObservingBlocks*, which are self-contained units. Each *ObservingBlock* specifies both a target and a set of exposures with a given instrument configuration. The astronomer is guaranteed that exposures inside an *ObservingBlock* are executed as an atomic unit in the

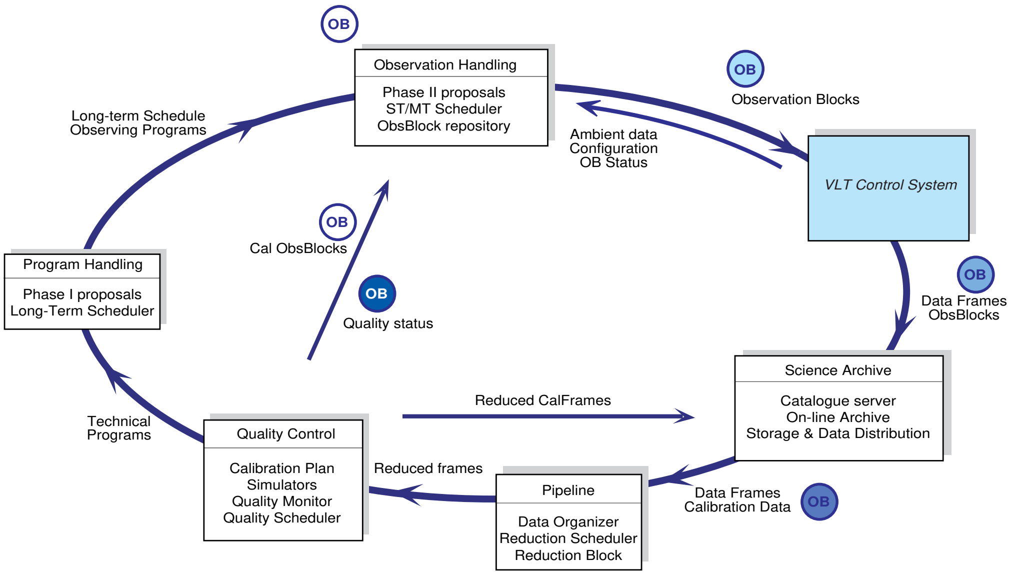

Figure 1. VLT Data Flow subsystem breakdown.

exact sequence specified. In service mode, a *Short Term Scheduler* provides to the ESO staff an ordered list of *ObservationBlocks* computed to take full advantage of weather conditions and instrumental configurations. The *Observation Handling* (Chavan & Albrecht 1997) provides functionality for handling *ObservationBlocks*, including possible scheduling, and forwards them for actual execution to the *VLT control* system.

- The external Data Flow view of the *VLT control system* is very simple, although internally it is extremely complex. From the outside, the *Observation Handling* subsystem gets environmental data and configuration parameters for instruments, while it provides the *VLT control system* with an optimized sequence of *ObservingBlocks* for execution. The resulting observations are then sent to the *Science Archive* subsystem, which keeps track of new data, log information, etc.

- The *Pipeline* subsystem receives the new data frames through the *Science Archive*. After classifying an incoming frame, a *ReductionBlock* is constructed by combining the raw data frame with necessary calibrations and an appropriate *Reduction Recipe*, specifying the actual reduction procedure. *ReductionBlocks*, like *ObservingBlocks*, are self-contained units which detail the standard pipeline reduction, independent of a given reduction system. The reduced data are made available to the user, and forwarded to *Quality Control* for validation.

- Finally, the *Quality Control* subsystem monitors the performance of the instruments by analyzing the reduced data from the *Pipeline* (Ballester et al. 1997). It generates calibration data and thereby establishes the quality level that can be achieved. Instrument simulators are maintained to reflect the current performance, for use by observers and the *Program Handling* subsystem. Calibration *ObservingBlocks* are also submitted to

Observation Handling according to a Calibration Plan, to ensure that specifications are met.

The main Data Flow subsystems are shown in Figure 1, where the general flow of science data is indicated. All along the flow, information is added to the *ObservingBlocks* so that a full record of events, from the original specification to the reduced data, is kept.

5. How Will it be Realized?

The analysis of the astronomical requirements was done by a mixed group of astronomers and software engineers to ensure a common view. An object-oriented methodology was applied since it provides a good mapping between the real world and software entities. The specific methodology, `OMT`, also gives a simple graphical representation of designs, which proved very useful in the discussions. An object-oriented approach also gives better modularity by strong encapsulation. The responsibilities of each component (i.e., class/object) are well defined.

The design concepts will be verified by prototyping on the NTT at La Silla. These implementations will test the general flow of information between subsystems and be used to analyze the merits of different choices. A baseline version will then be prepared for the VLT. At the same time, new technologies (e.g., OMG/CORBA and Java) will be evaluated.

6. Conclusions

A Data Flow concept is mandatory for the VLT due to its size and complexity. It provides the end user a simple view based on general concepts, like *ObservingBlocks* and *ReductionBlock*, which will yield a more homogeneous user interface and thereby help to increase efficiency. The tight *Quality Control* feedback loop will ensure that problems are detected early and corrected rapidly. The use of an object-oriented methodology has given a cleaner and more traceable system design. Finally, testing prototypes on the NTT is an important step in understanding the system behavior in a live, operational environment and in establishing a stable baseline for the VLT.

Acknowledgments. The VLT Data Flow concept presented here is based on work and discussions in the VLT On-line Data Flow Working Group and the Data Flow Project Team in ESO. We would like to thank all people involved in this process.

References

Baade, D. 1995, `VLT-PLA-ESO-10000-0441`, ESO

Ballester, P., Banse, K. & Grosbøl, P. 1997, this volume, 415

Chavan, A. M. & Albrecht, M. A. 1997, this volume, 367

Peron, M., Grosbøl, P. & van den Berg, R. J. G. 1996, in ASP Conf. Ser., Vol. 87, New Observing Modes for the Next Century, ed. T. Boroson, J. Davies, & I. Robson (San Francisco, ASP), 183