Astronomical Data Analysis Software and Systems IX
ASP Conference Series, Vol. 216, 2000
N. Manset, C. Veillet, and D. Crabtree, eds.

DASH — Distributed Analysis System Hierarchy

M. Yagi, Y. Mizumoto, M. Yoshida, Y. Chikada, G. Kosugi[1], T. Takata[1], R. Ogasawara[1]

National Astronomical Observatory, Osawa, Mitaka, Tokyo 181-8588, Japan

Y. Ishihara, H. Yanaka

Fujitsu Limited, Nakase, Mihama, Chiba 261-8588, Japan

Y. Morita, H. Nakamoto, N. Watanabe

SEC Co. Ltd., 22-14 Sakuragaoka, Shibuya, Tokyo 150-0031, Japan

Abstract. We have been developing an object-oriented data reduction and data analysis system *DASH* (Distributed Analysis Software Hierarchy) for efficient data processing for the SUBARU telescope. *DASH* consists of many objects distributed on CORBA. We abstracted astronomical data and reduction/analysis procedure as an object, PROCube (data reduction PROcedure Cube) so that user can easily construct a pipeline with GUI-based editor written in JAVA2. We also try to retrieve calibration data for the target object automatically with Observational Dataset.

The beta version of DASH system is now working and is being tested at SUBARU observatory. The final version will begin to operate in April 2000.

1. Introduction

DASH is an object-oriented data reduction and analysis system for 8.2-m SUBARU telescope at Hawaii. SUBARU obtained scientific first light in Jan. 1999, and produced 0.5 TB data from a few test instruments in this test year. It will soon produce more than ten TB data per year. DASH is designed as the SUBARU observatory system to manage, reduce and analyze such a huge amount of astronomical data produced by various instruments. We adopt CORBA as a distributed object environment and construct a distributed data reduction and analysis system (Takata et al. 1996). Four prototypes have been tested (Mizumoto et al. 1998a, 1998b, Yagi et al. 1999). The Prototype-5 will be tested at SUBARU Observatory in Dec. 1999 as a beta version. DASH project would be completed by May 2000, and we have almost achieved just as we had scheduled so far.

[1]Subaru Telescope, National Astronomical Observatory of Japan, Hilo, HI 96720, USA

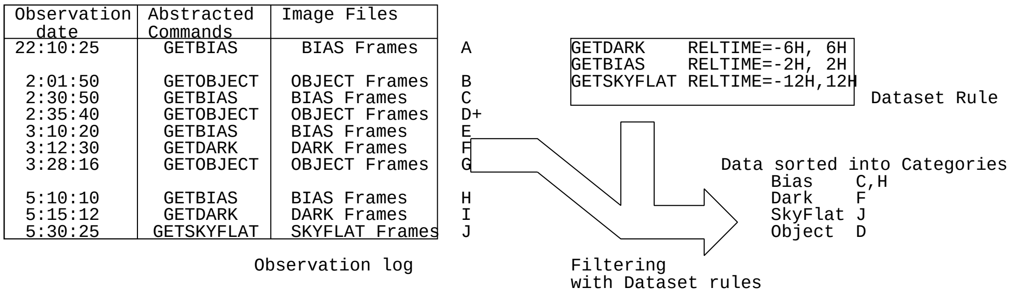

Figure 1. Schematic figure of Observational Dataset. When an user choose to reduce the frame "D", calibration frames needed are automatically selected according to the dataset rules, and sorted into categories. These categories are then set into skeleton PROCube.

2. PROCube and Observational Dataset

PROCube (data reduction PROcedure Cube) is an object that holds all information about data processing. It contains flow of reduction, pointers to engines, tunable parameters for each engine, and pointers to data. It also has a log of reduction, so that the result can be reproduced.

A PROCube which lacks pointers to the data but only have tags of data categories, such as "bias frames", "dark frames", or "object frames", is called a "skeleton PROCube". It describes how these data categories be reduced. When an user wants to reduce a data, the categories of skeleton PROCube should be filled with real data. We use "Observational Dataset" (Kosugi et al. 1998) to fill these data categories semi-automatically.

Observational Dataset consists of a set of rules of the relation among abstracted commands for telescope and instruments, and observational log which keeps a history of abstracted commands executed. When an user select his target, the abstracted commands listed in the log of observation are filtered by the dataset rules. The frames taken by the commands are then sorted into the data categories. Then the categories of the skeleton PROCube are filled with real frames. We can start the reduction with the filled PROCube and wait until the result comes out.

This framework is already coded and now being tested. We need to polish the dataset rules and the skeleton PROCubes.

3. UIs

There are three DASH UIs; DASHVisor, DASHEditor, and DASHMonitor.

DASHVisor is a supervisor of DASH environment of a user. It works as search viewer, and also as a controller of execution of PROCube. Editing PROCube, the user uses DASHVisor and DASHEditor. The user can search target object, observational dataset, and "skeleton" PROCube as well as engines with DASHVisor.

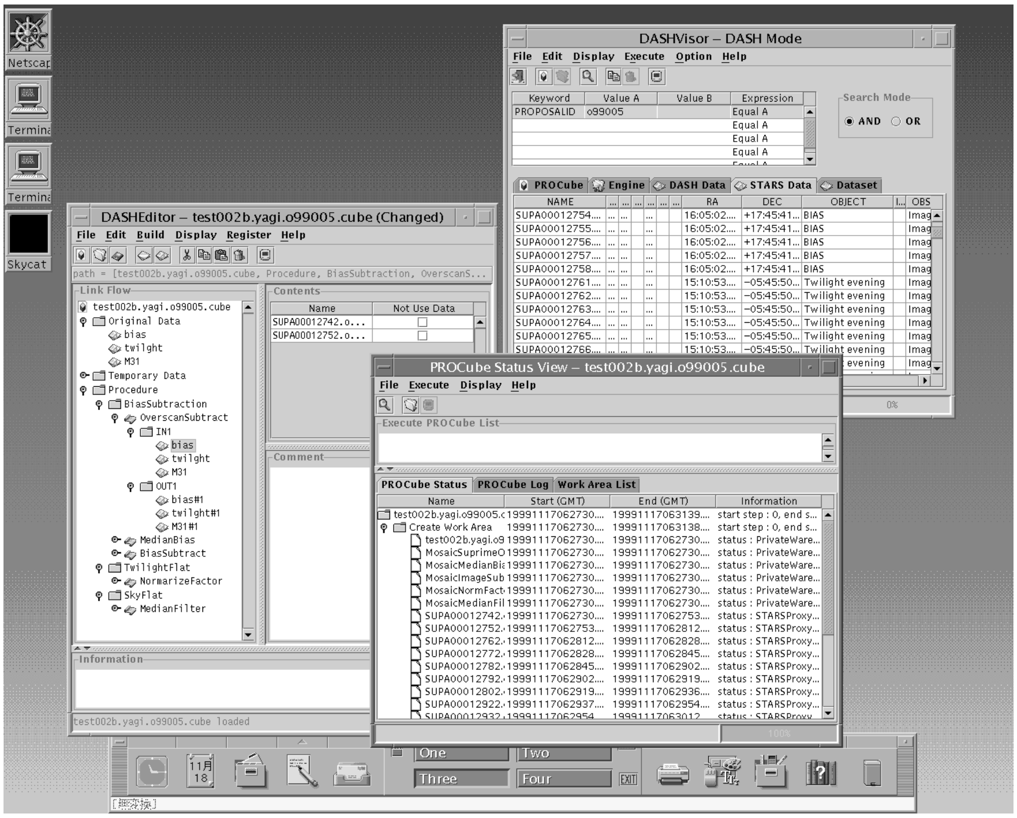

Figure 2. A snapshot of DASH environment. DASH GUIs. DASHEditor(left), DASHVisor(top-right) and DASHMonitor(bottom-right) are running on Solaris.

DASHEditor is a GUI-based pipeline (PROCube) constructor. We can construct a PROCube from scratch, or edit a PROCube in Data Bese. The components selected in DASHVisor are copied-and-pasted into a PROCube on DASHEditor.

After the built of the PROCube, the PROCube is "executed" from DASHVisor. The status of executing PROCubes is watched by DASHMonitor. DASHMonitor also sends a notice to the user by mail when the reduction ends.

All these components are written in JAVA2. We can use the GUIs not only on WSs but also on PCs.

As a image viewer, we adopt skycat, which is connected with DASHVisor, DASHEditor and DASHMonitor. We can browse the intermediate images with skycat while executing PROCubes.

4. Authentication and Access Control

The access to PROCubes, engines, and images in DASH is controlled under user/group authentication service of the SUBARU computer system. The authentication is done only when the user logs into the computer system (UNIX authentication).

There are three levels of access permission, USER, GROUP and PUBLIC. USER data (PROCubes, engines and images) are one's own data. Anyone but the user can see or use it. The user can delete the data if the data are not registered. GROUP data are shared in a group, based on proposal-ID, and only the members of the group can make use of them. PUBLIC data can be used by any user. GROUP and PUBLIC data are deleted only by administrators.

When an user logging into DASH, his user-ID and the list of groups to which the user belongs are automatically obtained from the authentication server. The user can search and use PUBLIC data, his USER data, and all GROUP data of the groups he belongs to.

5. SASH

SASH, a stand-alone system for single use extracted from DASH, is designed for visitors who want to continue their analysis in their home institute. It has the same UI as DASH but does not use CORBA, for SASH is in a single machine. DASH uses DA/DB(STARS (Takata et al. 2000) and DASH DB) and SUBARU authentication system, which are difficult to export to other institutes. SASH does not need DA/DB, nor authentication services. The user environment of SASH is designed to be the same as DASH. The reduction in DASH is easily detached and exported to SASH, and that of SASH could be easily re-attached to DASH.

References

Kosugi, G., et al. 1998, Proc. SPIE, vol. 3349, 421

Mizumoto, Y., et al. 1998a, in ASP Conf. Ser., Vol. 145, Astronomical Data Analysis Software and Systems VII, ed. R. Albrecht, R. N. Hook, & H. A. Bushouse (San Francisco: ASP), 332

Mizumoto, Y., et al. 1998b, Proc. SPIE, vol. 3349, 173

Takata, T., et al. 1996, in ASP Conf. Ser., Vol. 101, Astronomical Data Analysis Software and Systems V, ed. G. H. Jacoby & J. Barnes (San Francisco: ASP), 251

Takata, T., et al. 2000, this volume, 157

Yagi, M., et al. 1999, in ASP Conf. Ser., Vol. 172, Astronomical Data Analysis Software and Systems VIII, ed. David M. Mehringer, Raymond L. Plante, & Douglas A. Roberts (San Francisco: ASP), 375