Astronomical Data Analysis Software and Systems XIII
ASP Conference Series, Vol. 314, 2004
F. Ochsenbein, M. Allen, and D. Egret, eds.

Software reverse engineering and development: the VST TCS case

P. Schipani, M. Brescia, D. Mancini, L. Marty, G. Spirito

INAF - Osservatorio Astronomico di Capodimonte, Salita Moiariello 16, I-80131 Napoli, Italy

Abstract. The 2.6 m VST telescope is going to be installed at Cerro Paranal (Chile) as a powerful instrument for optical surveys. It is a joint project between the INAF - Osservatorio Astronomico di Capodimonte (OAC) and ESO. This paper deals with Telescope Control Software (TCS) technical aspects and software engineering design and development strategies.

1. Introduction

In the framework of a Memorandum Of Understanding jointly signed by ESO and OAC, the OAC has been committed to design and realize the VST telescope, software included. Nevertheless after the installation and commissioning the VST will be managed by ESO. Therefore in order to simplify the future maintenance of the software by ESO staff, it has been agreed to develop the VST software in the most "VLT-compliant" way. This constraint translates into pros and cons for OAC side, because many software utilities developed for VLT are available, but their migration to VST case requires a deep knowledge of all details, usually known to the original software developers only. In this context a reverse engineering approach, combined with the traditional software development for the VST specific subsystems, is the only solution.

2. Architecture

As the VLT is a large project involving many staff people for many years, a strong standardization has been introduced by ESO in the software development and hardware platforms. The VST software is based on the same standard architecture: distributed workstations running HP-UX and VME-bus-based Local Control Units (LCU) running the VxWorks real time multitasking operating system, connected together via Local Area Networks. LCUs interface with field electronics and electro-mechanics; the applications they run is a low level software, used to control and monitor the hardware devices, modularly distributed in processes and machines. The workstations coordinate LCUs and run the graphical user interfaces and "high level" application software; the workstation software has time execution requirements less urgent than the LCU's one, and is the one the operators usually interface with. The basic ideas in the architecture are modularity and standardization. Each LCU is devoted to control a spe-

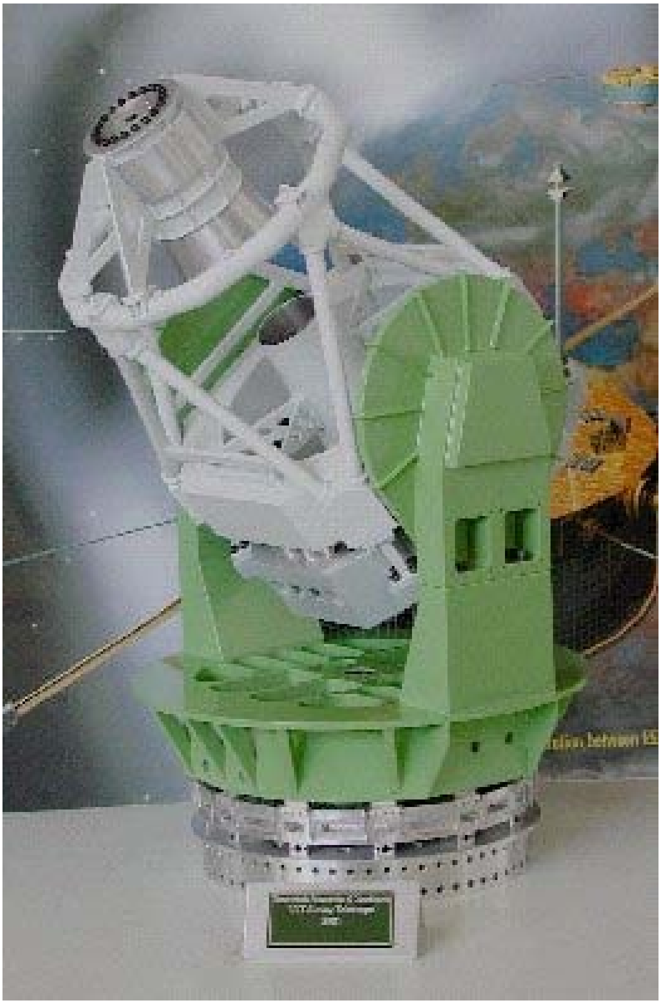

Figure 1. VST Telescope Model

cific subsystem, so the hardware devices are driven independently by different LCUs, using whenever possible the same VME boards and the same software. The programming languages on HP-UX platform are basically C++ for the core control modules, and Tcl/Tk for graphical user interface development. The LCU modules are developed in C. A key role in both workstation and LCU software architecture is played by an On Line Data Base, which allows the on-line information sharing between modules, monitoring the status of the system and of commands execution. Between the operating system and the application layer there is another software layer developed by ESO for VLT and reused for VST, the VLT Common Software, providing a lot of utilities, tools and common services to the application programmer.

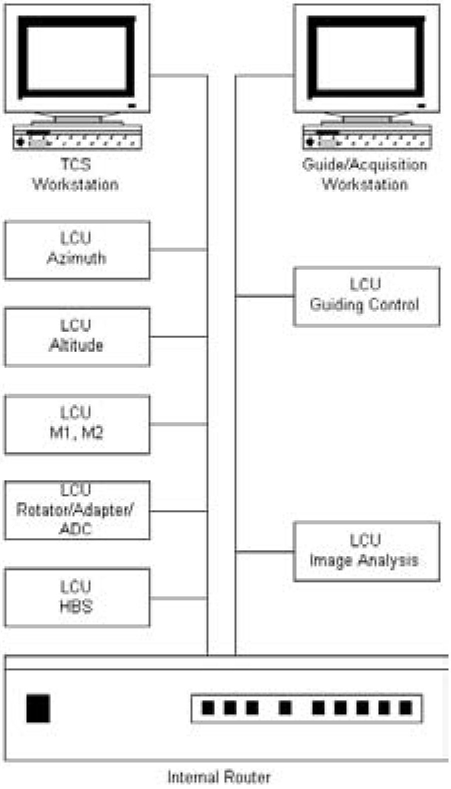

Figure 2. VST Telescope Control Network Architecture

3. Reverse Engineering

VLT software has been made available to Capodimonte staff, which was committed to reuse it as much as possible. This has been certainly an advantage because many parts could be recycled for VST. But obviously many other parts have been rewritten (most of the LCU software) or partially modified. Modifying a code that you do not know is often harder than rewriting it from scratch: thus a reverse engineering approach has been mandatory to let the old code run in the different (for telescope hardware, no. of machines, telescope functionalities) VST environment, to understand its operation in terms of simple functionality or in deeper details when needed, to change small or large parts preserving the interlacing between the many applications, running on many different machines. Sometimes a lot of time has been necessary to change just a few lines of code. ESO assistance in this hard work solved only partially the problem, because generally the help in this kind of work is fully effective only if persons work in the same location and can be easily grouped around the same table (otherwise, even a trivial - for the original developer!- problem can take a long time to be solved).

4. Work description

Unlike other telescope projects of similar size, usually committed to large organizations or consortia of many institutes, the whole VST design and realization has been committed to a small group (permanent research staff: 3 researchers + 2 graduated level technicians) of a small observatory, involved also in other parallel international programs. On software side, 2 permanent staff units have been available (1 has been deeply involved in other projects for the first years) + 2 short term young contractors, who unfortunately have changed many times along the years, moving to better remunerated and permanent position jobs: this has caused continuous loss of know-how and restart of the training with new contractors from scratch. It must be remarked that while in other work areas the start-up period can be relatively short, obviously the training in this kind of work cannot produce short term results, because in order to be operational it is necessary a good knowledge of computer science (e.g. Unix and Real-Time systems, C++ and Tcl/Tk languages, VLT software environment - used only inside ESO projects), control engineering (e.g. feedback control theory, control of industrial plants), very specialistic disciplines (e.g. active optics), daily telescope operation: no young engineer or astronomer with this skill is available at an affordable (for INAF-OAC) cost. It can be stated that VST telescope control software realization is an international level challenge faced with a low budget and with an undersized staff in comparison with similar projects usually carried out by larger staffs, generally spread within different research institutes.

5. Status of work

Most of the software is now (November, 2003) ready for tests with the telescope. Since this software interface with many external devices, it is foreseen a period of intensive tests with real hardware in order to let the overall control system (software + electronics) working at its best.

Acknowledgments. We are grateful to all those who have supported our work. S. Sandrock and R. Karban of ESO gave us precious suggestions in our migration from VLT to VST case.

References

Brescia, M., Mancini, D., Marty, L., Mazzola, G., Schipani, P., Spirito, G. 2002, Proc. SPIE, 4848, 553

Schipani, P., Brescia, M., Mancini, D., Marty, L., Spirito, G. 2001, Proc. of the 8th International Conference on Accelerator and Large Experimental Physics Control System, 579