TOWARDS A STATE BASED CONTROL ARCHITECTURE FOR LARGE TELESCOPES: LAYING A FOUNDATION AT THE VLT

R. Karban, N. Kornweibel, ESO, Garching, Germany
D. Dvorak, M. Ingham, D. Wagner, Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA, USA.

Abstract

Large telescopes are characterized by a high level of distribution of control-related tasks and will feature diverse data flow patterns and large ranges of sampling frequencies; there will often be no single, fixed server-client relationship between the control tasks. The architecture is also challenged by the task of integrating heterogeneous subsystems which will be delivered by multiple different contractors. Due to the high number of distributed components, the control system needs to effectively detect errors and faults, impede their propagation, and accurately mitigate them in the shortest time possible, enabling the service to be restored. The presented Data-Driven Architecture is based on a decentralized approach with an end-to-end integration of disparate, independently-developed software components. These components employ a high-performance standards-based communication middle-ware infrastructure, based on the Data Distribution Service. A set of rules and principles, based on JPL's State Analysis method and architecture, are use to constrain component-to-component interactions, where the Control System and System Under Control are clearly separated. State Analysis provides a model-based process for capturing system and software requirements and design, greatly reducing the gap between the requirements on software specified by systems engineers and the implementation by software engineers. The method and architecture has been field tested at the Very Large Telescope, where it has been integrated into an operational system.

CONTEXT

The European Southern Observatory (ESO) carries out an ambitious programme focused on the design, construction and operation of powerful ground-based telescopes for astronomy. Worldwide, Extremely Large Telescopes are considered one of the highest priorities in ground-based astronomy.

The European Extremely Large Telescope (E-ELT), a revolutionary new 40-metre-class ground-based telescope, will be the world's biggest eye on the sky, when it begins operating early in the next decade [1].

The E-ELT is characterized by a high level of distribution of control-related tasks and will feature diverse sets of data flow patterns and large ranges of sampling frequencies.

Within the E-ELT, the Telescope Control System (TCS) will maintain wave front quality throughout the duration of the observation. Considering the multitude and inter-relation of E-ELT distributed control loops (covering about 1000 moveable mirrors, and actuator stroke management), maintaining the wave front during the observation is foreseen to be more demanding than acquiring the target. This is a fundamental difference from our past experience on the Very Large Telescope (VLT) project - the essential complexity (the complexity associated with the underlying problem) has increased, and thus it is imperative to avoid adding any more complexity than necessary.

The TCS includes all hardware, software, and communication infrastructure required to control the telescope (including the dome), and to interface to subsystems down to, but not including, actuators and sensors, as they are typically delivered together with the respective sub-system. The main functions provided by the TCS are related to Control and Interlock (an automatic and immediate stop of a unit in case of a safety critical situation) handling, and fault management.

The **main challenges** can be summarized as follows:

- Large number of control points (the primary mirror alone encompasses 15000 actuators)
- Number of interfaces (15 subsystems, 9 focal stations, site operation)
- Large data volume (700 Gflops/s, 17 Gbyte/s in real-time in adaptive optics) of engineering data
- Multitude of interacting, distributed control loops (from 0.01Hz up to kHz rates)
- Software intensive distributed control strategy
- Integration of heterogeneous distributed components, provided mainly by contractors, adds additional complexity.

ARCHITECTURE

Separating the concerns of astronomy from those of technical implementation is a key goal of the architecture. Astronomy domain-specific knowledge should be isolated from the (technical) subsystem domain, which is typically contracted to an expert in that domain. The implementers of the subsystem should not be required to have knowledge of the astronomy domain. Another major architectural driver is the need to query knowledge of the state (e.g., temperatures, position of actuators, and taking into account uncertainties of sensor readouts) of the physical system and to control this state. Due to its high distribution and tight coupling this knowledge is essential to be able to deliver a functioning system.

The architecture must satisfy certain (domain independent) quality goals. The most important are related to the conceptual integrity. The conceptual integrity is the underlying vision that unifies the design of the system, at all levels. The architecture should do similar things in similar ways.

The **main goals** for the TCS **architecture** are:

- Define a framework and design rules to enable the software engineer to develop the domain specific applications which address and match the functional needs from the wave front control strategy.
- Contain system complexity.
- Promote modifiability and scalability (and long-term maintainability).
- Enable high availability and fault tolerance.

These strategic goals can be achieved by applying certain tactics:

- Carry out the system design according to a well defined set of design patterns.
- Have a uniform way of designing closed loop control on the subsystem actuators, i.e. how are the sensor data processed, the state of the hardware system estimated, and the actuators commanded?
- Represent the state of the hardware system and the control system in a uniform way.

The design of the control system architecture brings together engineers from different domains: system, software, control, and electronics. Close collaboration between those disciplines is essential to close the gap between systems engineering and the other technical disciplines.

The data-driven architecture (explicitly formalizing the data and meta-data produced and consumed by a system, transported with messaging [10], see Figure 1) consists of three tiers organized in a decentralized manner in order to separate domain knowledge (e.g., tracking a celestial object vs. mirror control), integrate heterogeneous control systems, and support the implementation of flexible wave front control strategies.

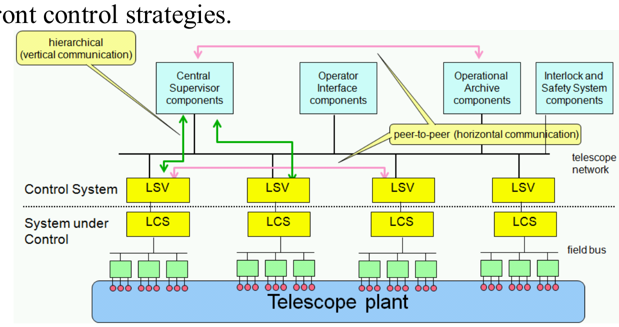

Figure 1: Conceptual Architecture.

The first tier provides robust and simple access to sensors and actuators of a subsystem and is called the Local Control System (LCS). The second tier adapts the LCS to the astronomy domain and is responsible for closed loop control of the sub-system's actuators and sensors. It is called the Local Supervisor (LSV). The third tier, the Central Supervisor, provides high level interaction with users and among subsystems.

STATE ANALYSIS AND MBSE

A Model Based Systems Engineering (MBSE) methodology is characterized as the collection of related processes, methods, and tools used to support the discipline of systems engineering in a "model-based" or "model-driven" context [7]. The State Analysis (SA) method [3] [8] is targeted to the control related domain, and focuses on behaviour, which is an often underestimated aspect in Systems Engineering. The SA method, which is founded on a state-based architecture and goal-based operation, defines a process for identifying and modelling the states of the physical system and their relationships. The methodology enables effective coordination, robust execution, and flexible response mechanisms in the system by defining goals (e.g., keep tracking error within a certain RMS error) for the states of the physical system instead of defining precise execution procedures up front.

State Analysis provides a uniform, methodical, and rigorous approach for developing control system architecture by:

- Discovering, characterizing, representing, and documenting the state variables of a system;
- Modelling the behaviour of state variables and relationships among them, including information about hardware;
- Identifying interfaces and operation; and
- Capturing the mission objectives in detailed scenarios motivated by operator intent.

In State Analysis, nominal and off-nominal states are given equal stature in the model. State Analysis provides the opportunity for (but does not obligate) earlier consideration of faults and off-nominal behaviour. Fault Detection, Isolation, and Recovery (FDIR, also known as Fault Management) becomes an integral part of the control system design. A strict boundary is drawn between the **Control System** being designed and the target **System Under Control**, as shown in Figure 2.

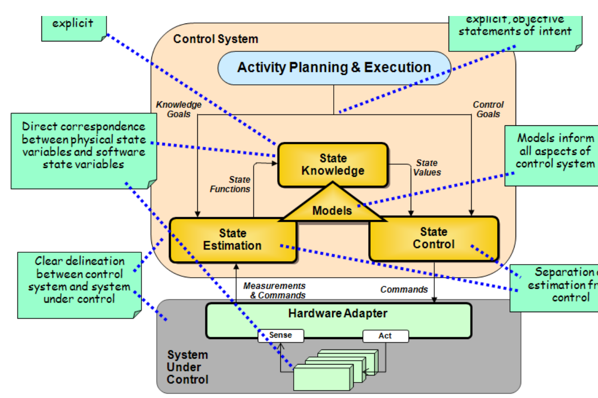

Figure 2: State Analysis Architectural Concepts.

State Analysis provides sound architectural principles and rules which help to achieve the requirements and goals of the E-ELT TCS. Thus, many concepts of the method have been baselined by the project, and adapted to the specific needs of the project (like where to draw the line between control system and system under control for contractual reasons).

VLT FIELD TESTING

As part of validating the technology decisions for the E-ELT, control systems at the VLT (Figure 3) [4] are being refurbished using technologies intended for use in the E-ELT control systems. The upgrades serve several purposes: field test technologies and methods in an operational environment (outside the lab), provide input into the E-ELT technology decisions, address obsolescence in the UT telescope control systems, and ready observatory technical staff for the construction of the E-ELT.

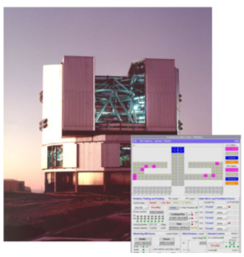

Figure 3: VLT UT.

As the first step in the upgrade and field test program, the enclosure (dome) control system (ECS) of one Unit Telescope (UT) was refurbished in 2010, using the SA method and architecture. It is now successfully operating at the Paranal observatory.

The Enclosure Control System Upgrade comprised:

- Dome and Windscreen;
- Seal, louvers, Observing Doors;
- Telemetry
- Thermal input/output (I/O);
- Approximately 1500 I/O points.

The refurbished control system interfaces to existing sensors and actuators in the field, and integrates transparently with the existing dome control interface at the supervisory level. The application of SA was therefore limited to the LCS and LSV.

The SA control pattern drove the development of the data model, which consisted of building a façade to the System Under Control, identifying and defining State Variables (SVs) and specifying Goals).

Though only a limited and introductory application of SA was made, the architectural rules of SA aided in improving the design of the control system software. Estimators (which generate state knowledge based on available evidence) read blocks of field data and publish data to State Variables. The Estimators were largely implemented with LabView.

During 2011 and 2012, the project will be working to upgrade the control system of the main axes. The goal of the project is to achieve the same or improved scientific performance while tackling obsolescence in the system by upgrading with specific technologies and architectures targeted for use in the E-ELT.

The main axes control system provides the means for telescope positioning (moving the telescope in either Altitude or Azimuth axis in order to point the telescope) and telescope tracking (coordinating the movements in both axes in order to track celestial objects given their position, motion and the current time).

The main axes upgrade applies more rigorously the SA method than the enclosure upgrade, together with other MBSE concepts. In particular, a SysML [5] profile for SA is being developed, allowing integrating the SA artifacts into an overall system model. The overall system model is built according to principles of the Object Oriented Systems Engineering Method (OOSEM [5]), which distinguishes a logical and physical model of the system, and also provides means to describe the system "as-is" and "to-be" and the relations between the two– particularly useful for an upgrade project. JPL has already defined an initial UML profile for SA [6], which serves as a starting point for the SysML profile.

Figure 4 shows a snippet of a State Effects Diagram (SED), which is a SA artifact used to identify SVs, Measurements, Commands, and the causal effects between them. The main focus in this diagram is the set of effects on and of the physical state variable "Azimuth Position And Velocity". Every box on the diagram corresponds to a physical SV which has to be estimated and possibly controlled.

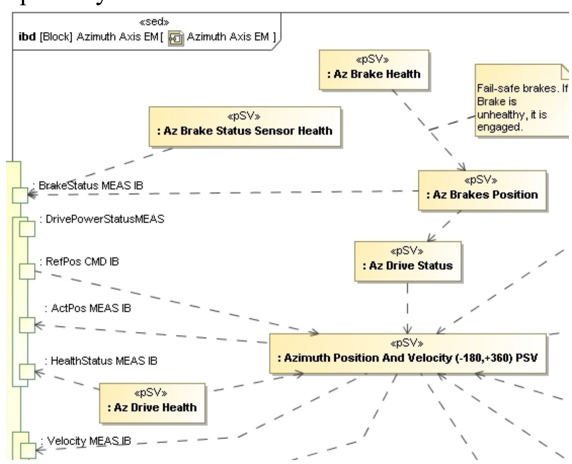

Figure 4: State Effects Diagram of Main Axes.

The abstract <<affects>> relationships are later on modelled in a more rigorous way using SysML State Charts, Parametric models or Activities, if and only if a more accurate description is needed.

The corresponding conceptual architecture (Figure 5) can be derived from the SA artifacts, including the SEDs. It shows two SVs "EncHeadHealth SWSV IB" and "AzPosVel SWSV IB" with their respective estimators and controller, interacting with the Azimuth Axis Local Control System. Since everything is contained in the same model, referring to the same model elements, the information on the system is all consistent.

Many artifacts created using SA can be re-used across projects and systems because they express domain concepts in an implementation-independent way. For example, SA artifacts from a previous design of an Antenna array [2] can be leveraged for the E-ELT.

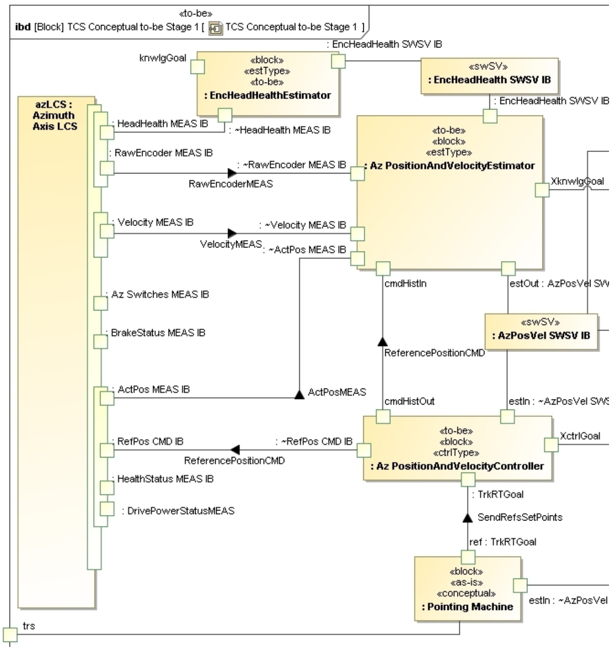

Figure 5: Conceptual State Based Architecture.

IMPLEMENTATION

JPL implemented the State Analysis method with its Mission Data System (MDS) architecture [9], which provides software libraries, classes, and behaviour reflecting the SA concepts (e.g., state variables, goals, etc). The current MDS implementation is based on method invocation: for example, a state variable is queried if its current constraint (its goal) can still be satisfied.

The current ESO prototype implementation is based on messaging, using the Data Distribution System (DDS) as its middleware. State Variables are realized by DDS topics. Consequently, controllers and estimators can simply publish and subscribe to a SV, making combination of multiple SVs in a control algorithm simple and without requiring knowledge of where the SV resides or what estimates it.

In this way, the full SA control pattern is implemented with the publish/subscribe mechanism, which allows runtime attaching and detaching of clients, for example to trace measurements and verify the behaviour of the control system.

SUMMARY AND FUTURE WORK

State Analysis is built on a sound theory which enables building an architecture for a distributed system, like the E-ELT, following well defined principles and rules. Although the scalability for very large systems needs still to be verified, the first practical experiences on the VLT give us confidence to pursue the path of applying SA for the VLT upgrade and eventually for the E-ELT control system.

Integration with other MBSE practices is fundamental to address all relevant system aspects, and requires, in particular, a SysML profile. The definition of the mapping between SA and SysML, and the formalization of SA concepts and relations in an ontology is part of the ongoing collaboration between ESO and JPL.

This research was carried out at the European Southern Observatory and at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

REFERENCES

[1] R. Gilmozzi, J. Spyromilio, „The 42m European ELT: status"", Proc. SPIE 7012, 701219 (2008)

[2] J. Choi, A. Coleman, D. Dvorak, J. Hutcherson, M. Ingham, C.Y. Lee, P. Wolgast, "Goal-Based Operations of an Antenna Array for Deep Space Communication", iSAIRAS. Los Angeles, CA. Feb 2008

[3] M. Ingham, R. Rasmussen, M. Bennett, A. Moncada, "Engineering Complex Embedded Systems with State Analysis and the Mission Data System", AIAA Journal of Aerospace Computing, Information and Communication. Vol. 2, No. 12, December 2005, pp-507-536

[4] K. Wirenstrand, "VLT telescope control software: status, development, and lessons learned", Proc. SPIE 2003, vol. 4837, p. 965 (2003(

[5] S. Friedenthal, A. Moore, R. Steiner, "A Practical Guide to SysML", Morgan Kaufmann OMG Press, 2009

[6] A. Murray, R. Rasmussen, "A UML Profile for State Analysis," IEEE Aerospace Conference, March 2011

[7] J. Estefan, "Survey of Model-Based Systems Engineering (MBSE) Methodologies," Rev. B, INCOSE Technical Publication, INCOSE-TD-2007-003-012008

[8] G. Jones, D. Wagner, D. Dvorak, A. Mishkin, "Human-Rated Automation and Robotics", JPL Technical Report D-66871, 2010

[9] Mission Data System, http://mds.jpl.nasa.gov

[10] G. Hohpe , B.Wool, "Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions", Addison-Wesley Longman, 2003