

# STATUS OF THE ESS CONTROL SYSTEM

Garry Trahern, European Spallation Source (ESS), Lund, Sweden

## Abstract

The European Spallation Source (ESS) is a high current proton LINAC to be built in Lund, Sweden. The LINAC delivers 5 MW of power to the target at 2500 MeV, with a nominal current of 50 mA. It is designed to include the ability to upgrade the LINAC to a higher power of 7.5 MW at a fixed energy of 2500 MeV. The Accelerator Design Update (ADU) collaboration of mainly European institutions will deliver a Technical Design Report at the end of 2012. First protons are expected in 2018, and first neutrons in 2019.

The ESS will be constructed by a number of geographically dispersed institutions, which means that a considerable part of control system integration will potentially be performed off-site. To mitigate this organizational risk, significant effort will be put into standardization of hardware, software, and development procedures early in the project. We have named the main result of this standardization the Control Box concept. The ESS will use EPICS, and will build on the positive distributed development experiences of SNS and ITER.

Current state of control system design and key decisions are presented in the paper as well as immediate challenges and proposed solutions.

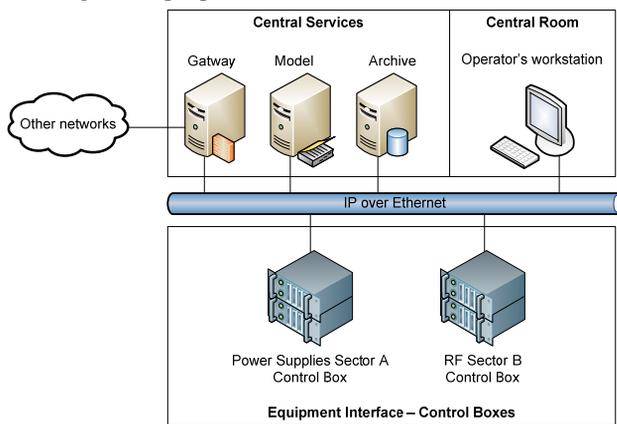


Figure 1: Three-tier architecture of the ESS control system.

## CONTROL SYSTEM DESIGN

For the ESS control system, a slightly modified three-tier architecture is suitable. The three tiers [1] depicted in Fig. 1 are:

- **User interface** (upper right in the figure). These are graphical and non-graphical user interfaces. Most of them will be in the control room, but some will be elsewhere, e.g., for site-wide monitoring of the ESS status, and for remote access.
- **Central services** (upper left in the figure). Computer services that need to run continuously irrespective of user's activities, e.g., archiving of process variables'

values, monitoring of alarm states, slow feedback loops, model of the machine as a whole, and management of activities that require coordination of several subsystems.

- **Equipment interfaces** (bottom in the figure). This tier is responsible for interaction with equipment and devices. It serves two purposes: to provide an abstract representation of equipment to higher layers through which the equipment can be monitored and controlled, and to implement real-time control loops.

## THE CONTROL BOX

The Control Box metaphor is based on the philosophy adopted by ITER [2]. In ITER terminology the Control Box philosophy is realized with the concepts Plant System Host, CODAC, mini-CODAC and plant system I&C. It provides a standardized solution for all collaborating teams and is a cornerstone in the 3-year design update phase. An example structure of a control box is shown in Fig. 2.

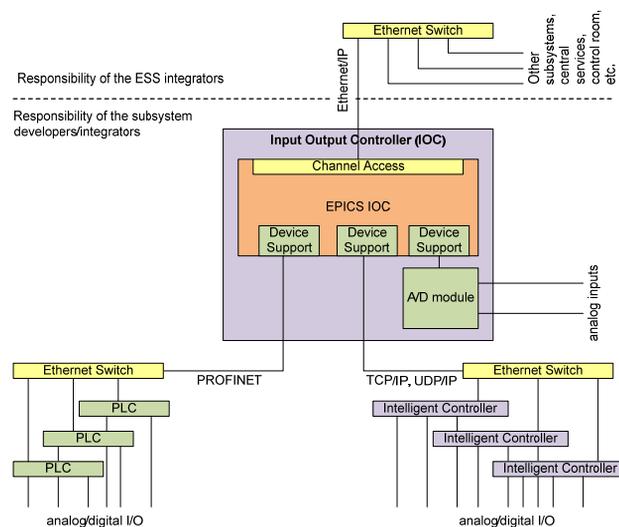


Figure 2: Schematic of Control Box components for a representative Control Box example.

## Benefits

The main benefits of the Control Box are:

- To allow independent and yet standardized subsystem controls development,
- To encourage and enforce consistency between subsystems (including target and experiments station),
- To facilitate testing of new equipment, e.g. EPICS drivers,
- To reduce risks early to prevent unexpected surprises at project integration time and
- To minimize throwaway hardware and software development.

## Development

The Control Box should not be completely defined and developed too early in the project as the technology landscape to support it is rapidly evolving. We therefore expect to have iterative Control Box development.

It is necessary to develop Control Box software and hardware in (e.g., yearly) cycles. The main strategy is to start with software-only aspects (which are easiest to develop, test and distribute), and as soon as possible deliver a Control Box with a simple standardized hardware interface such as infrastructure PLC control.

In future versions functionality will be added (e.g. hardware support, including timing and feed-forward), and tools and support will evolve with available technology.

## Hardware Platform

The Control Box concept enforces the usage of a common hardware platform for all teams and sub-systems to shorten the development time, support costs, etc. Therefore, the discussion on the selection of the hardware platform has been initiated early in the design stage and an iterative selection process has been agreed upon.

To further optimize the selection process a hardware selection table has been proposed where various important aspects of proposed platforms (VME, ATCA, PCI and all flavours) are leveraged against individual team requirements and scaled to the whole project size. Selection criteria for the platform includes:

- **Vendor support:** how many commercial vendors of crates and modules exist? A larger number implies that the probability of finding an off-the-shelf module for a particular task is higher.
- **Maturity:** how long has the platform been available, and how frequently is it used? Greater maturity implies lower risk and lower probability of backward-incompatible changes in the future.
- **Longevity:** how long is the platform expected to be available? Assessment should be given with the ESS's lifetime (several decades) in mind.
- **High availability:** how suitable is the platform for high-availability applications? (e.g., support for redundancy).
- **Software support:** how likely will software support (Linux driver, EPICS device support) be available for a device?
- **Prototyping vs. Production:** what is the cost of prototype development vs. production development, integration and support?
- **Risks and risk cost:** what are the risks connected to a specific platform selection and more importantly, what are the costs associated with these risks?

## BEAM LINE ELEMENTS DATABASE (BLED)

ESS Control system will be built around a central configuration database (codename: BLED) where all accelerator parts and the relationships between the parts are modelled, as shown in Fig. 3.

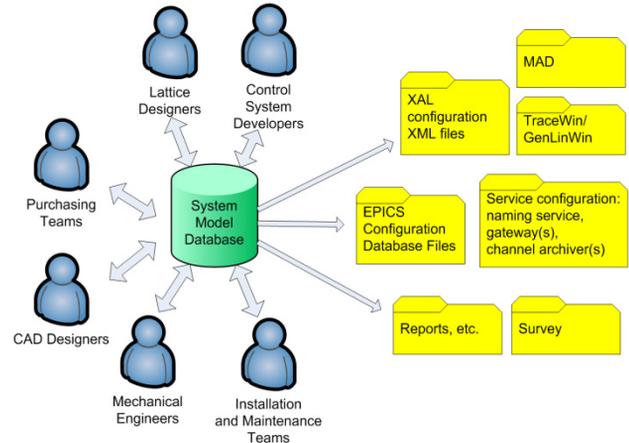


Figure 3: Users and main output of BLED.

This all-encompassing approach starts with the premise of where the equipment is located and includes data about the lattice as well as the equipment and devices. In addition, cabling information is stored in the database. The database also contains various high-level configuration parameters of the facility, all process variables, alarm definitions and alarm configurations and process variable archiving configurations. Basic information about the personnel who are making changes to the database is also included. Database entities are versioned. This makes it possible to recreate an older version of the facility model or some parts of the model, enables auditing and change tracking of the configuration and supports approval procedures for changes. The database should be constructed and operated so that it will provide a consistent model of the entire facility configuration.

## Accelerator Parameters

To ensure consistency of the information being used amongst all subgroups throughout the period of accelerator design and construction, a parameter list database and web interface have been proposed in reference [3]. The main objective is to provide tools to identify inconsistencies among parameters and to enforce an ethos for groups as well as individuals to work towards a common solution. Another goal is to make the Parameter Lists a live and credible endeavour so that the data and supporting information shall be useful to a wider audience such as external reviewers as well as being easily accessible.

### Control System Perspective

The use of the ESS Naming Convention in the configuration database for equipment and devices already provides the necessary information to identify all process variables in EPICS. However, in order to generate all EPICS database configuration files from BLED, a further extension of the BLED schema is needed. Therefore EPICS process variable fields, alarm server configurations and channel archiving configurations will be added to the schema to enable the generation of the EPICS database files, alarm configuration and channel archiving configuration files from the BLED configuration database. In the event that graphical database editing tools will leverage the configuration database, the schema will be extended with diagrams showing the positions of process variables on the diagrams.

### DEVELOPMENT ENVIRONMENT

The Control System of the ESS will be complex (i.e., it will consist of a multitude of subsystems that will need to be integrated) and its development will require the cooperation of many developers and controls experts.

To allow proper development procedures, artefact sharing, code storage, etc., vital central services need to be set-up. Figure 4 shows a graphical overview of the environment. These services must provide the ability to:

- Share artefacts (code, documents) between engineers (Mercurial, DocDB, Plone),
- Perform version control (Mercurial),
- Continuously integrate SW development (Hudson),
- Perform complex and time consuming operations (DMSC),
- Offer a standard development environment regardless of the platform (Virtual Machine),
- Build and package software (Maven),
- Deploy artefacts to user machines or IOCs (yum, rpm),
- Perform automated testing (JUnit) and
- Report bugs and issues (Bugzilla).

Therefore, standards and guidelines must be put in place for development. Development tools and platforms should also be standardized.

### ONLINE MACHINE MODEL AND SIMULATION

The user interface of the control system will most likely be based on XAL [4]. This software framework developed at SNS is a suite of Java/Python libraries and programs designed to interact with the EPICS services while providing human readable information to operate the machine. Through XAL it will be possible to change the state of the Linac by acting on all knobs and variables available to the EPICS system.

During the running of the machine, especially in the beam commissioning period, the operator has to decide which changes are needed to optimize the machine, for

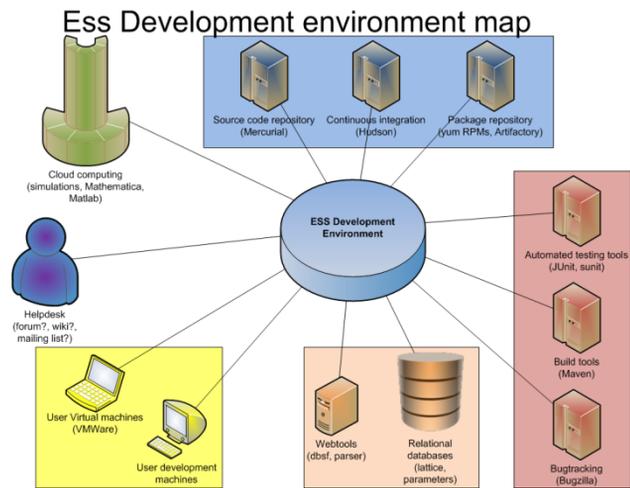


Figure 4: ESS Development environment map.

example to reduce losses, to steer the trajectory, etc. In this operation the user will be assisted by the Online Model: it will be a simulator of the Linac that interacts via XAL as a real machine and allows testing the parameter changes before applying to the real accelerator. Moreover the Online Model will be able to do automatic routine tasks such as loading the parameters from the running machine to find the optimal configuration for a specific problem, for example, to find the optimal configuration for correctors to steer the beam.

To have such instrumentality an optimal knowledge of the Linac is required, and this task can be achieved through a full campaign of simulations. Simulations are necessary to understand the dynamics of protons and the possible instabilities generated by errors in the productions of the Linac components during assembly or simply by random errors. The simulation code will be flexible enough to include unexpected phenomena arising during beam commissioning, and it will provide single and multi particle capabilities in order to predict the widest range of situations that the user can experience running the accelerator. The data to construct the simulation, e.g., systematic errors associated with component fabrication and alignment errors during installation will be maintained in the configuration database BLED.

### STATUS

#### Design

There are three major advances in ESS controls design development that set stepping stones for further development:

- The **Conceptual Design Report** document has been drafted and describes the unified approach of controls covering the three main project segments: accelerator, target and instruments. All the development tools and services have been scaled accordingly to cover the additional segments such as

cryogenics and conventional facility infrastructure controls.

- The **Naming Convention** has been defined and approved. By learning as much as possible from ITER and SNS experiences, the naming convention was put in place early to allow proper enforcement and avoid pitfalls as the project advances.
- The initial **Costing Exercise** has been performed and provided the first estimation on the price of ESS controls. It is estimated to be less than 5% of total project cost.

### Control Box

There have been several advances regarding the Control Box as well.

- The discussion on **HW platform** has been initiated with the individual teams, e.g. beam diagnostics and RF.
- It has been accepted that **early prototyping** and as-common-as-possible hardware platform is the required approach to allow the learning process to start early. This should reduce complications in sub-system integration during the construction phase of the project.
- The Control Box development cycle has been fully integrated into the **Development Environment**, see Fig. 5.

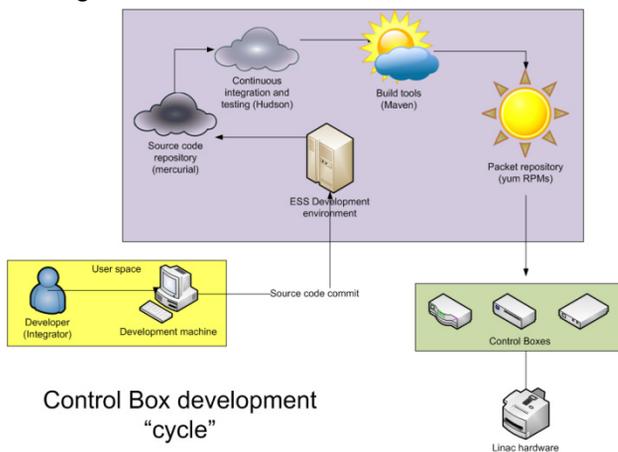


Figure 5: Control Box development cycle.

### BLED

Since the Beam Line Elements Database is a vital component of the control system the development has started as well.

- The main database schema has been finalized and agreed upon.
- The Accelerator Parameters and Naming Convention tools are currently both under development.
- Various parsers and lattice tools (DBSF) have also been modified and developed to be consistent with BLED requirements.

### Development Environment

The development environment as one of the key tools in the development process has been deployed in the DMSC, the Danish supercomputing cloud.

- Currently, the development environment is hosted on 8 CentOS service machines (3 designated for user's development) in the supercomputing cloud.
- It already provides an almost complete set of development tools: Mercurial, Hudson, MySQL database servers, Maven, Bugzilla, etc.
- It also provides a standalone virtual machine (based on Scientific Linux) that provides the required services for development across various platforms (Linux, Windows, Mac OS)

### Timing and Machine Protection

Because of the significance of Timing and MPS to all parties (accelerator, target, instruments, safety), gathering of requirements for both have started. The possible designs and approaches have been described in the CDR documents and regular iterative meetings with different teams are underway to enforce regular collaboration and convergence for these two project-wide systems.

## SUMMARY

The control system is a sophisticated network connecting all the various parts of the accelerator and is essential for the synchronization and day to day running of all the equipment. Its organization will inevitably also determine its efficiency and usability. A certain degree of flexibility is expected.

Therefore, the development of the ESS control system has been from day one focused on the following key issues:

- Collaboration and communication,
- Unified development, integration and support,
- Learning from existing projects, and
- Interaction and iterative decision-making.

## REFERENCES

- [1] Paul D. Sheriff, Fundamentals of N-tier architecture, Barnes & Noble, 2006.
- [2] F. Di Maio, et al., Development of the ITER CODAC Core Systems, Proceedings ICALEPCS2009, Kobe, Japan.
- [3] K. Rathsman et al., ESS Parameter List Database and Web Interface Tools, Proceedings of IPAC2011, San Sebastian, Spain.
- [4] A. Shishlo et al., The XAL Infrastructure for High Level Control Room Applications, Proceedings ICAP2009, San Francisco, CA.