ELI-ALPS CONTROL SYSTEM STATUS REPORT *

Lajos Jenő Fülöp, Sándor Brockhauser, Sándor Farkas, Veronika Hanyecz, Miklós Kiss, Miklós Tamás Koncz, Árpád Mohácsi, Kwinten Nelissen, Lajos Schrettner, Balázs Szalai, Péter Szász, Chris Turner, ELI-ALPS, Szeged, Hungary

Abstract

ELI-ALPS will provide a wide range of attosecond pulses which will be used for performing chemical, biological, physical or medical experiments by international research groups. It is one pillar of the first international laser facility for the scientific user communities. ELI-ALPS use the TANGO Controls framework to build up the central control system and to integrate the autonomous subsystems regarding monitoring and control. It will be also used for the implementation of some autonomous systems' control system while others will be implemented differently. The central control system and the integration strategy of the autonomous systems is designed. The centralization and integration needs are surveyed and the requirements are collected. Prototypes have been developed to clarify the requirements and to test the designs. Requirements elicitation, designing and prototype development follows a Lean-Agile approach and includes several fields: device drivers and simulators; integration logic; central supervision, archiving, logging and error recovery; graphical user interfaces and so on.

INTRODUCTION

ELI-ALPS is one of the three pillars of the European Extreme Light Infrastructure project. The Attosecond Light Pulse Source (ALPS) facility will provide a wide range of ultrafast light sources (such as coherent XUV and X-ray attosecond pulses) for performing material, condensed matter, surface science, chemical, biological, physical or medical experiments. Besides, the development of the technology for generating 200PW peak intensity pulses is also a main mission. As a research facility, the infrastructure will contain a large number of experimental devices and equipment which have to be managed and controlled by a robust and flexible system. We found that the TANGO Control system [1] is a good candidate, it is already used by several research institutions, mostly in synchrotrons, but also in laser projects [2]. The basic layout of ELI-ALPS is the following: different primary laser sources generate laser pulses with different characteristics that are delivered by the beam transport lines to the secondary sources, which then generate attosecond light pulses for the experiments.

In the first phase of the development of the ELI-ALPS Control System the fundamental concepts, the high-level architectural design and the frames of the development process have been identified. In this paper the most important requirements, concepts, aspects, and prototypes are summarized briefly.

REQUIREMENTS

The general requirements are to follow public standards [3] as well as in-house policies and handbooks that also declare the application of frameworks such as TANGO and Taurus [4]. In order to maximize efficiency, different techniques are planned to be used. Automation techniques will be applied as much as possible, e.g. automatically registering devices [5], generate and configure GUIs and so on. Simulation environments, mimicking the devices and some key relationships will allow to perform most of the software development and testing independently from the real hardware.

Scientific Systems

Laser sources will be delivered as black-box, turn-key systems together with hardware and software. The secondary source systems are addressed by dedicated projects. The requirements, the technical design, as well as the hardware shopping list are provided by expert institutes of the corresponding secondary source field. The beam transport systems are addressed by a dedicated in-house project, which provides the requirements, the technical design and the hardware. Software and integration will be covered by another dedicated project.

Central Control System

The Central Control System will include the following services. The *archiving* subsystem collects all the preconfigured local or central variables for later usage and trends. The *alarms* subsystem collects and shows all the local and central alarms that are the result of evaluating predefined formulas on attribute values. The *logging* subsystem collects all the logs.

The *Integration Platform* supports, manages and orchestrates the collaboration of the systems (lasers, secondary sources, beam transport, etc.). Each system provides a gateway for communication. The gateways should be accessible only from the central system. For the other systems the central control system defines a uniform interface in order to provide controlled access to the gateways of a connecting system. A locking mechanism of the central system is responsible for system (proxy/gateway) allocation (i.e. granting exclusive access, when a system can be allocated to only one other system at a time). Moreover, it is preconfigured which systems may allocate a certain system for monitoring and control purposes.

* The ELI-ALPS project (GOP-1.1.1-12/B-2012-000, GINOP-2.3.6-15-2015-00001) is supported by the European Union and co-financed by the European Regional Development Fund.
† https://www.eli-alps.hu/?q=en

ISBN 978-3-95450-148-9

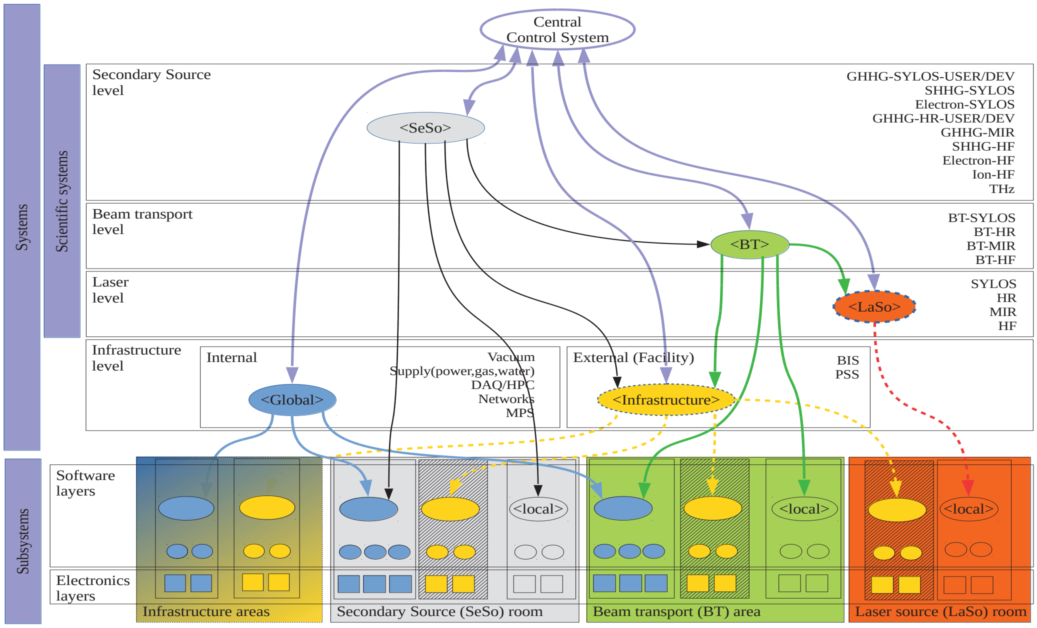

Figure 1: Control System Architecture.

The *Overview GUI* provides a general overview, monitoring all the systems. This GUI can be (semi-)generated based on the Gateways.

Data Acquisition

The data acquisition system will provide a framework for acquiring and processing experimental data originating from all of the experiments, as well as augmenting the data with metadata. Secondary sources and experiments should apply and adopt this framework. Data acquisition will use the common facility level timestamps for both triggering the experiments and timestamping the acquired data (the timing system is to be delivered by an external provider). Already existing solutions and results will be considered [6]. The experiment execution framework should have data processing capabilities at different granularities.

Each experiment will have a unique ID. An experiment consists of a series of **batches**, each batch has a unique batch ID. During a batch, low level data acquisition takes place that may involve low level reduction/ compression and pre-processing done in the low level device and electronic logics layers (e.g. by FPGAs).

After the completion of a batch, **higher level post-processing** can take place, maybe invoking scientific algorithms run on local machines or on a cluster accessible on the network (HPC services). The results of post-processing can influence the execution of the next or subsequent batches, this provides a feedback mechanism during experimentation (online data processing).

SYSTEM DESIGN

The following system design concepts have been identified based on the requirements and constraints given above.

Systems and Subsystems

Briefly, a subsystem incorporates the devices and electronic components (represented by rectangles at the bottom of Figure 1), as well as the subsystem specific software layers: device servers are implemented for physical devices on multiple levels (low level, compound, software logics). These are represented by ellipses in the Figure.

The higher level compound device servers implement gradually higher level functions. These device server components are organized in a layered structure, each layer providing services to the layer(s) above and using services of the layer(s) below.

Systems are composed by combining subsystems. These can be organized into different levels, indicated by boxes in Figure 1. The level name (e.g. Beam Transport Level) is indicated in the top-left corner, while the actual instances (the corresponding systems) are indicated on the right side of the box (e.g. BT-SYLOS): 4 laser sources, 4 beam transport lines and 11 secondary sources are listed.

The Scientific Systems include three levels: the *lasers* will be delivered by external suppliers as turn-key solutions; the *beam transport* systems use the services of beam transport-specific subsystems and the corresponding laser system; the *secondary source* systems use the services of subsystems located in a designated area, and the beam transport system (as well as the central system).

The Infrastructure systems (built on top of localised subsystems) are provided either internally (in this case subsystem device servers are white-box and accessible) or externally (in this case subsystem device servers are black-box and not directly accessible). The various systems comprise an architecture built on top of the TANGO bus. The design makes it possible to control devices located in a designated area of interest (e.g. secondary source room, beam transport area), and also to

ISBN 978-3-95450-148-9

control, monitor and manage all the devices of the systems supporting the same functionality (e.g. vacuum, power supply, computer and network monitoring, etc.)

Layers

Each system (and its subsystems) has some layers and each layer has to address hardware and software aspects (see Figure 2). The layers from bottom to top are the following.

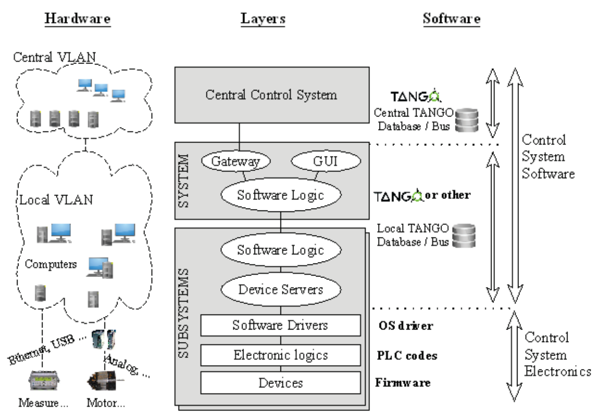

Figure 2: Control System Layers.

Control System Electronics. The *Devices* layer includes the elementary building blocks; it is an electronically controllable device, e.g. a detector or a motor. The *Electronic logics* layer controls the device and provides fast logics through an electronic controller, e.g. a relay or a PLC. The *Software drivers* layer connects the electronic and software worlds: a device or controller can be connected to a local or remote computer controlled by operating system or other drivers (e.g. LIMA, etc.).

Control System Software. The *Device Servers* layer makes the software devices available on the TANGO network and hides all the underlying layers. One computer can host several device servers on the local network, and one device server can provide access to several devices. The *Software Logic* layers provide functionalities on sets of related devices on subsystem and system levels, respectively. These are also implemented as TANGO device servers and are hosted on one or more computers. The *GUI* (Graphical User Interface) layer gives interactive access to the software logics and to the devices for the operators and users. TANGO provides generic GUI clients as well as toolkits to create custom ones. The *Gateway* layer is the access point towards to the central control system. It is implemented also as TANGO device server(s).

It is important to protect the different systems from each other in order to avoid unintentional interactions. A particular system should have its dedicated (virtual) computers, terminals and (Virtual) Local Area Networks; as well as it should have a dedicated TANGO database/bus. Similarly to the systems, the central control system will have its own (V)LAN and TANGO database/bus.

The layer concept can be applied only to the Beam Transport, Secondary Source and Internal Infrastructure systems (those that are not delivered by external suppliers as black-box systems).

PROTOTYPES

Two types of prototypes have been developed in order to validate the design concepts. A vertical prototype includes all layers of a small demonstrational system with physical devices. A horizontal prototype includes the software layers of almost all systems (laser sources, beam transport lines, secondary sources) with software simulated devices.

Vertical Prototype

A simplified optical system was assembled and gave the basis of a vertical prototype (see Figure 3). It was suitable to build up a prototype with TANGO. In the software logic layer there were two loops for stabilizing the manually pre-aligned beam. The GUI (Figure 4) displays these loops and also gives action buttons to the users.

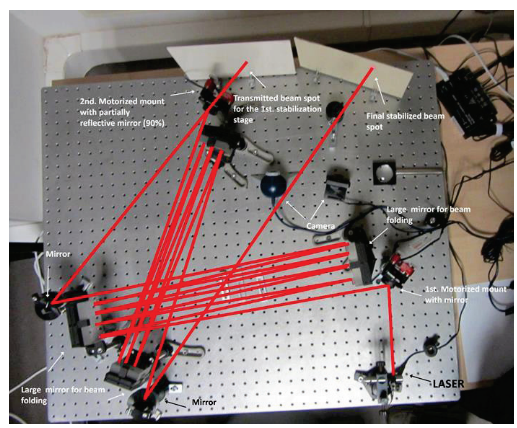

Figure 3: Layout of the vertical prototype.

ISBN 978-3-95450-148-9

Horizontal Prototype

This prototype includes a skeleton of the four laser sources and the ten secondary sources: the prototype is based altogether on 700 simulated devices. The prototype provides GUIs for each system, and a central one. The simulation was implemented on the *device servers* layer. This prototype was not generic enough and reusable directly for development and testing, therefore a simulation framework has been elaborated.

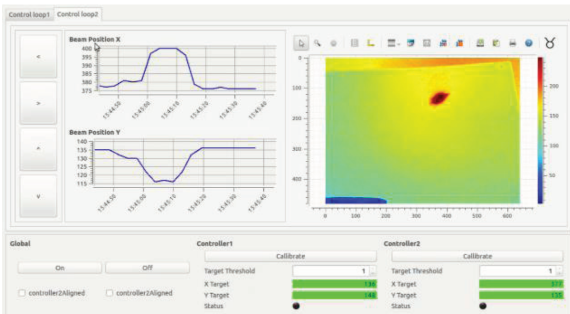

Figure 4: GUI of the vertical prototype.

Simulation Framework

Building on the experiences gained from the constructed vertical and horizontal prototypes, a framework has been designed that aims to cover as many aspects of the ELI-ALPS control system development as possible.

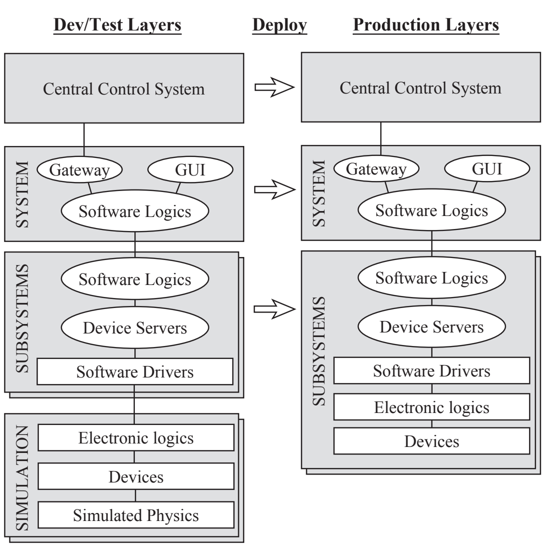

Figure 5: Architecture for development and testing purposes with simulation back-end.

The framework is based on the simulation concept, the system is clearly separated into two parts: the control(ling) environment (left side of Figure 5: top 3 grey boxes) and the controlled environment (left side of Figure 5: bottom grey box). The connection between the controlling and the controlled (simulated) parts can completely be described by a configuration file. This separation provides many advantages because most of the development can be done without the target hardware environment: unit testing, functional and integration testing, deployment test, prototyping, operator training etc. It is important to note that the simulation environment is not expected and does not have to exhibit the same real-time behaviour as the target environment in order to be useful. In fact, different simulation levels can be defined ranging from the provision of the expected interface to gradually higher degrees of realism.

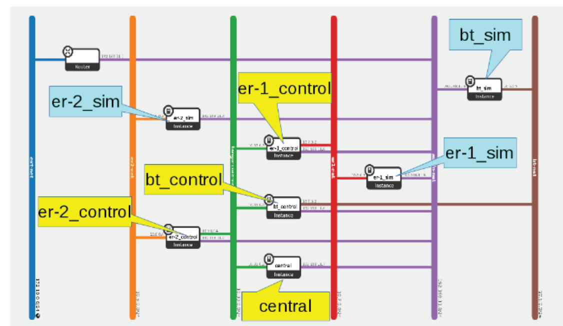

Figure 6: OpenStack environment of the demo system.

A demo system has been developed as a proof of concept for applying the simulation framework that utilizes OpenStack technology as a supporting infrastructure. The demo system consist of a laser source and its beam transport system, two secondary source (experimental) areas and a central control room. The OpenStack environment provides the necessary (virtual) machines for both the simulation (bt_sim, er-1_sim, er-2_sim) and control side (bt_control, er-1_control, er-2_control, central), and an isolated network area per system (Figure 6).

The simulation environment contains components for a sufficiently broad set of equipment: laser source, motors, motor controllers, mirrors, beam dumps, as well as photodiodes and CCD cameras as detectors. The connection to these equipment is described in a configuration file, which is fed into the controlling side to be able to connect the two together. The controlling side provides several functions accessible on GUIs: The central system provides overview status information for the operators. The beam transport system has setup functions as well as can be instructed to set its configuration so that the beam propagates to one of the secondary source areas. In these areas, it is possible to calibrate the local beamline (adjust beam target position and beam diameter, set delay step granularity), as well as start/stop a simplified pump-probe experiment involving an automatic scan through a range of delays between the pump and probe pulses.

CONCLUSIONS

The conceptual system design has been defined according to the requirements. Different kind of prototypes have been developed in order to check the

ISBN 978-3-95450-148-9

design. There is a lot of work in the future because the requirements of the scientific systems will be continuously changing and evolving in addition to new ones arriving.

REFERENCES

[1] "TANGO Controls Homepage," [Online]. Available: http://www.tango-controls.org/. [Accessed Oct 2015].
[2] M. Pina et al. (2013) "Controlling Cilex-Apollon Laser Beams Alignment and Diagnostics Systems with Tango", in Proceedings of ICALEPCS2013, San Francisco, USA
[3] K. Stouffer et al. (2015) "Guide to Industrial Control System (ICS) Security" U.S.: National Institute of Standards and Technology (NIST)
[4] "Taurus homepage," [Online]. Available: http://www.taurus-scada.org. [Accessed Oct 2015].
[5] S. Rubio-Manrique et al. (2011) "A Bottom-Up Approach to Automatically Configured TANGO Control Systems" in Proceedings of ICALEPCS2011, Grenoble, France
[6] R. Borghes et al. (2013) "A Common Software Framework for FEL Data Acquisition and Experiment Management at FERMI" in Proceedings of ICALEPCS2013, San Francisco, USA.

ISBN 978-3-95450-148-9