Control System Software Development Environment in ELI Beamlines

Pavel Bastl , Ondrej Janda, Alexey Kruchenko, Petr Pivonka, Birgit Plötzeneder,Saud Saldulkar, Jiri Trdlicka (ELI-BEAMS/Institute of Physics of the ASCR, Prague)

The poster presents the control system software development environment in ELI Beamlines – a combination of tools and approaches. The main building block in our software are components – which are created using a model-based development process based on the Iterative Unified process.

Control System in ELI Beamlines

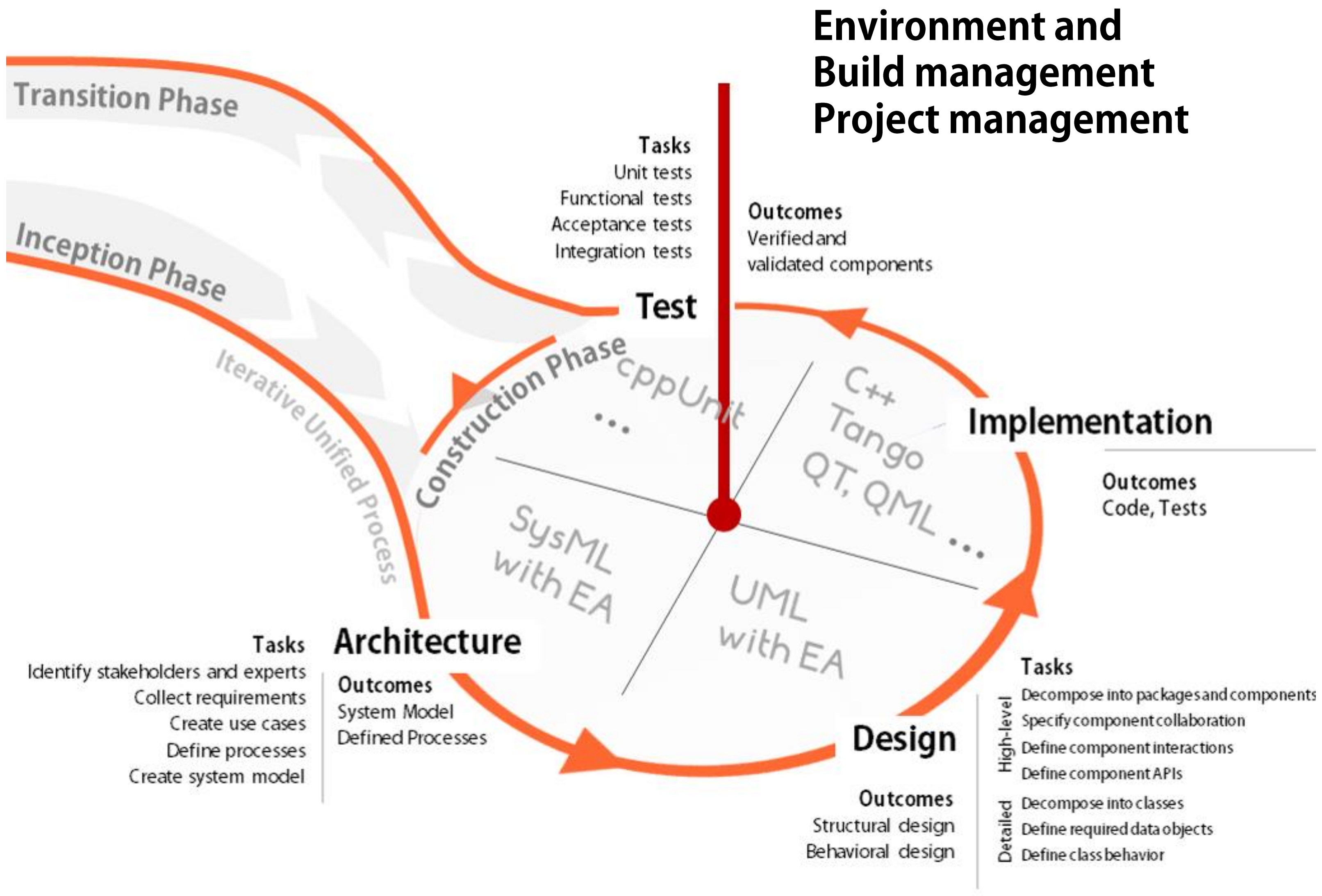
View of the server room / main control room as of 10/2017

The central control system in ELI Beamlines connects, supervises and controls more than 40 complex subsystems with high demands on network, synchronisation, data acquisition, processing, and storage. For development, we use the following approaches:

- Use of already available frameworks (TANGO)
- Only two languages (C++, Python), user frameworks (Matlab, LabVIEW) are supported by providing bindings for all components
- Model-based development focussing on components and APIs
- Use of automatic software generation whenever possible

Software Development Process

Our software development process is based on the Iterative Unified Process. The approach breaks software into single components that go through iterations of architecture, design, implementation and testing. On top of the process are project management tools (RedMine) and build and environment management tools (cmake, git,..). Modern software engineering techniques such as continuous integration are used.



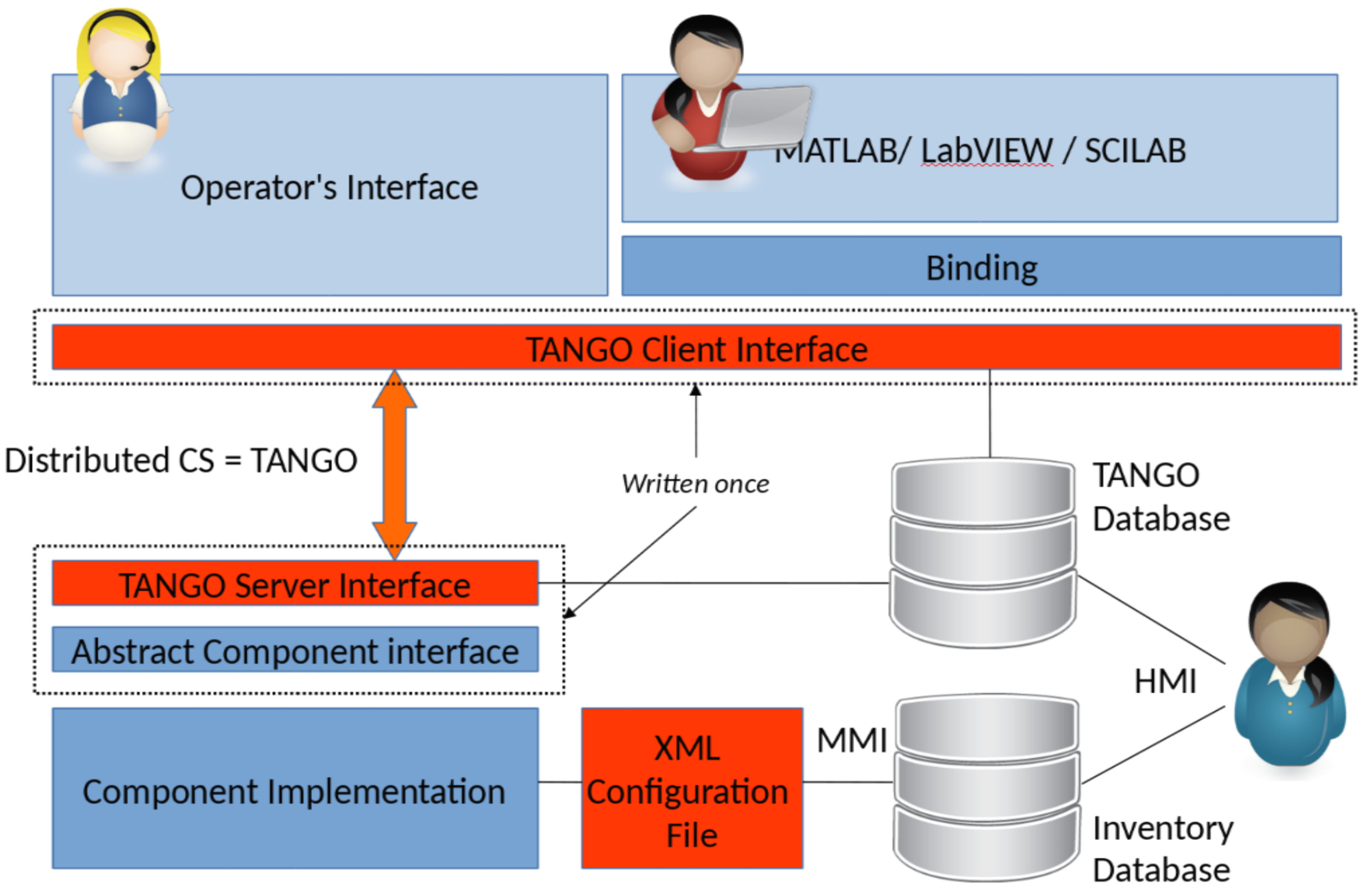
The ELI Control System Development Process

Components as the building block

Based on the software development approach, a generic structure for software components has been developed. Every component requires a well-defined API and is configurable via XML file.

This approach has the following advantages:

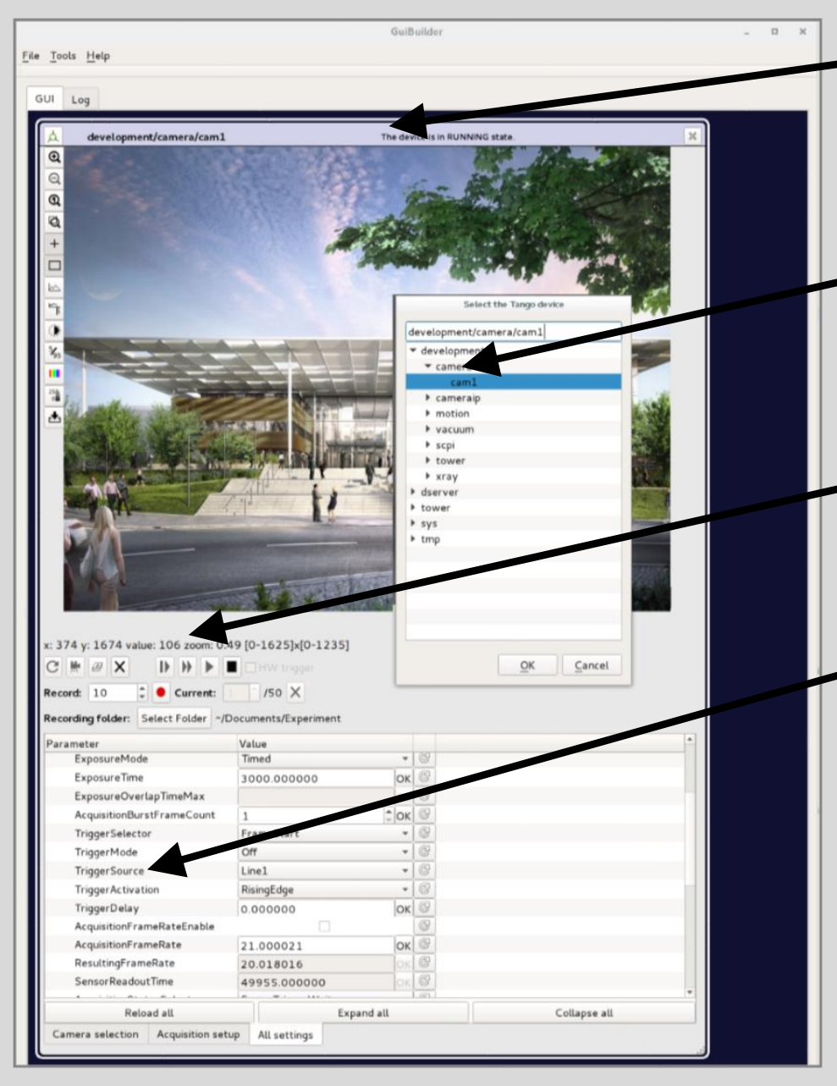
- TANGO interfaces are written exactly once per component / API
- Components can be defined as a plugin and loaded without compilation
- Configuration sources can be easily switched (Disk files, databases, streams,..)
- Configuration files can be parsed on-the-fly and GUIs can be created automatically to allow vendor-specific device configurations



Generic structure for a software component in our control system

EXAMPLES

Camera Component

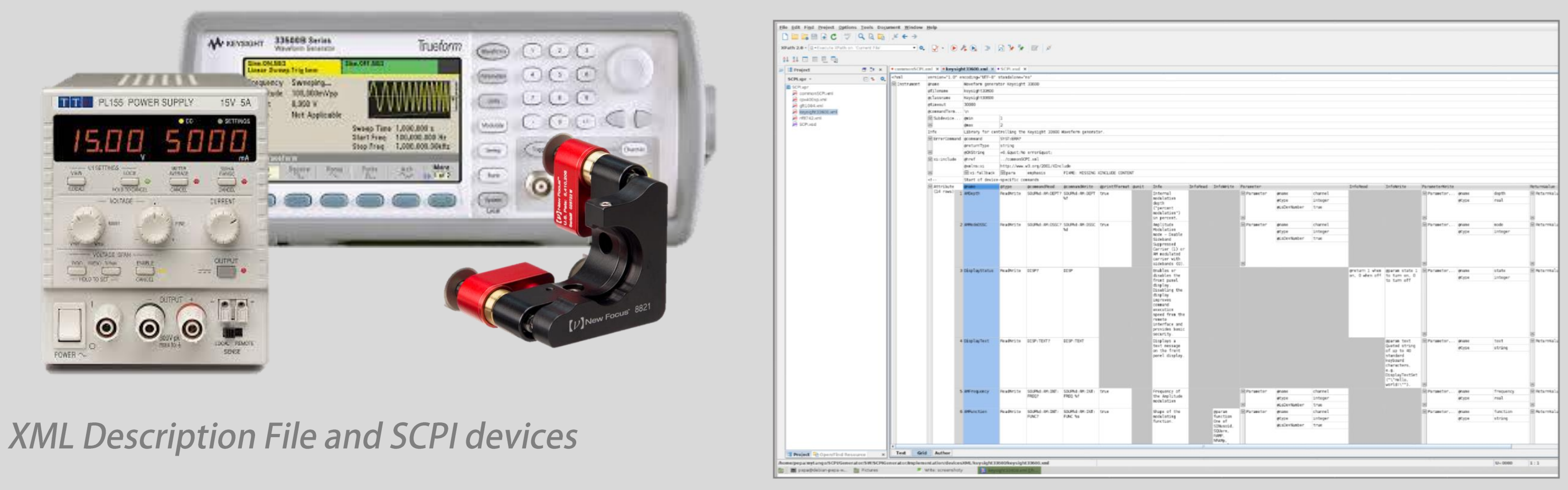

Generic Device GUI
TANGO – Database Browser
Standard Camera API Interface
Specific parameters, GUI automatically created from XML obtained from component

Details: Cameras in ELI Beamlines:
A Standardized Approach (this conference)

ELI Camera Tango Server

Drivers for SCPI/WX11 based devices

All SCPI commands are described in an XML – file and the device driver is automatically generated – up to the TANGO server level



XML Description File and SCPI devices

ELI:EXTREME LIGHT INFRASTRUCTURE, Institute of Physics ASCR
Extreme Light Infrastructure, CZ.1.05/1.1.00/02.0061