

Libera

Control System Integration

Hinko Kočevar, Libera Workshop, Solkan, 15 October 2010

hinko.kocevar@i-tech.si

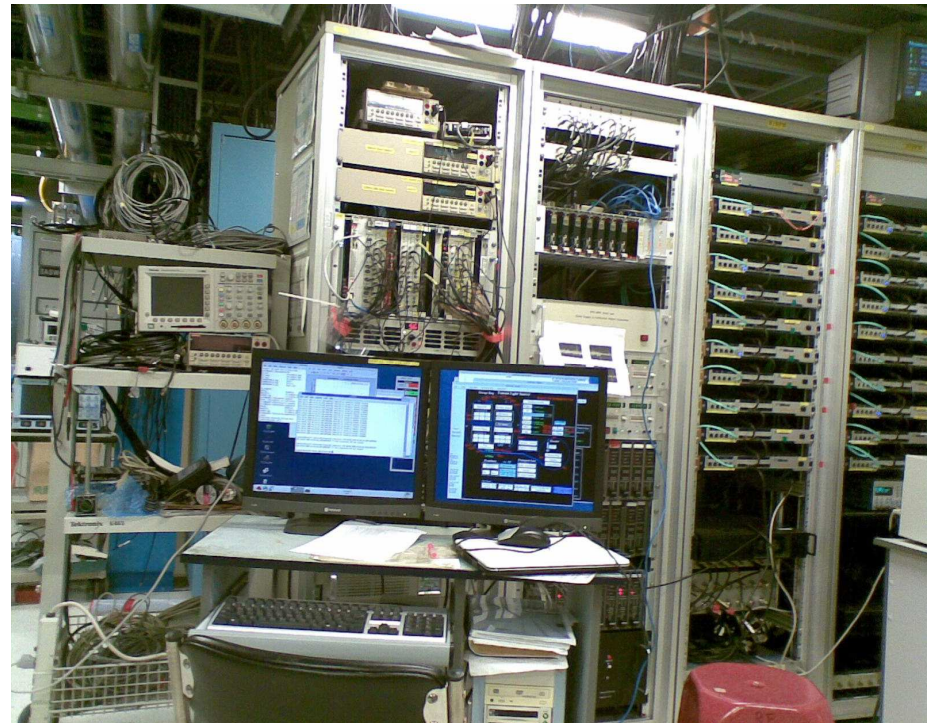
The Need for Integration



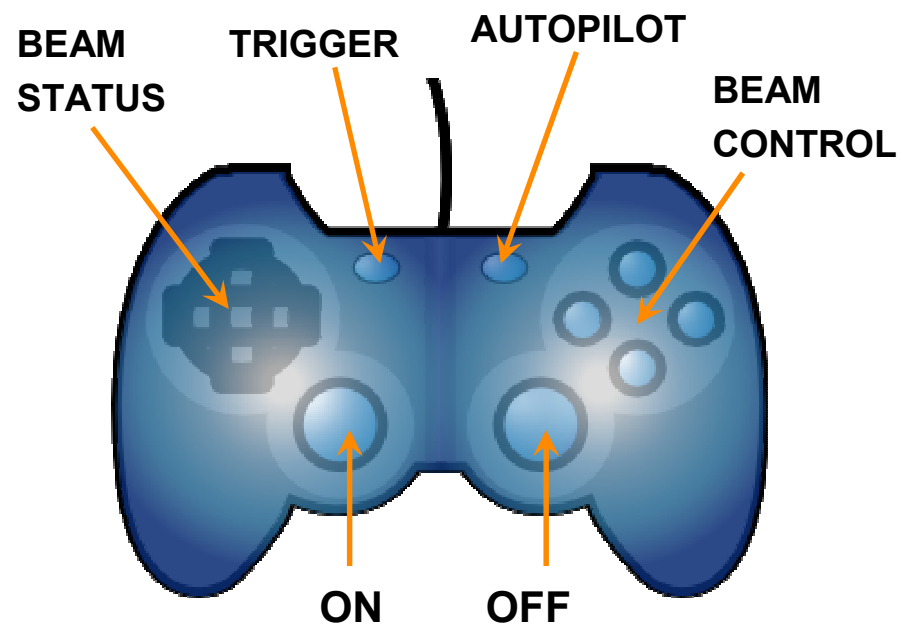
- **Why do we need to integrate?**
- **Part of the big picture**
- **Common view of the instruments**
- **Access to the instruments**
- **Control of the instruments**

Integrating Instruments

- **Hardware integration**
 - Connect cables
 - Find location
- **Software integration**
 - Access
 - Configure
 - Use
 - Manage
 - Develop



The Need for Control



- Why do we need to control?
- Manage the instrument
- Observe the instrument
- Ease the use of the instrument
- Expose the instrument

Controlling the Instruments

- **Many instruments, many manufacturers, many flavors, many control aspects, many solutions**
- **Complex instruments with many functions**
- **Modern (networked) instruments**
- **Complex software for instrument access and control**
- **Open source or propriety software**

Libera Software Integration

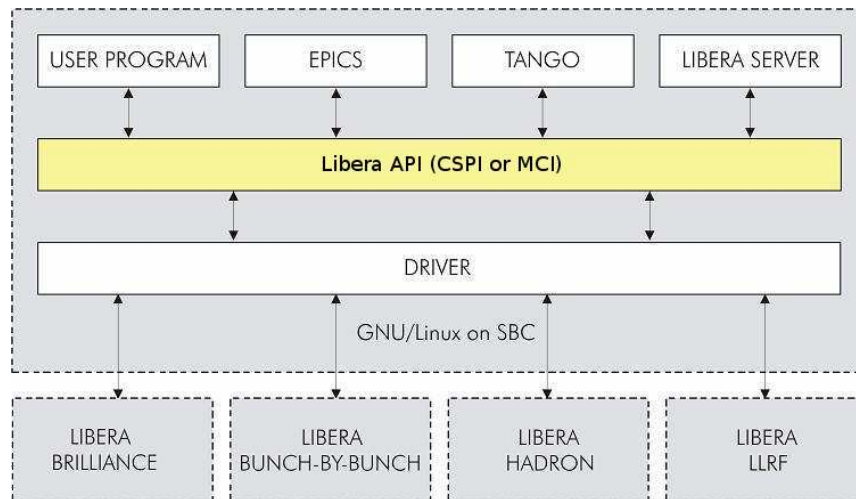
- **Two software API libraries:**
 - CSPI (Control System Programming Interface)
 - MCI (Measurement and Control Interface)
- **API centered for configuration and data transfers**
 - Local or network instrument access
 - Instrument management and use
- **Daemons running locally**
- **Open source software, GPL licensed**
- **Control system view vs. instrument view**

Customizable, extendable, supported

Libera Control System Interface

- **EPICS support out of the box (Instrumentation Technologies)**
 - EPICS IOC running locally
- **Tango (Alba, Elettra, ESRF, Soleil) and Tine (Desy) support from 3rd party**
 - Controlling server running remotely
- **Other control systems can also be interfaced**
 - Controlling server running either locally or remotely
- **EPICS also available from 3rd party (DLS)**
- **Integration support available**

Libera API Layer



- Open and close connection to the instrument
 - Local access
 - Remote access
- Set communication parameters
- Setup configuration and data connections
- Concurrent data stream acquisitions

Libera API Functions

- **Get and set configuration parameters and status**
 - Health parameters (temperature, fans)
 - Calculation parameters (pickup coefficients, offsets)
 - Limits (interlock)
 - Performance (DSC, AGC)
 - Other
- **Acquire desired data stream**
 - Raw ADC data stream
 - Turn by turn data stream
 - Slow data stream
 - Post mortem data stream

Libera API Implementation

- Network transparent API
 - Ability to run same code locally or remotely with minor modifications
- Designed to support variety of control systems
- C and C++ programming language
- Designed to be linked as a library into application binary
- Control system software takes care of m x n access and synchronization

Which Libera API Is for You?

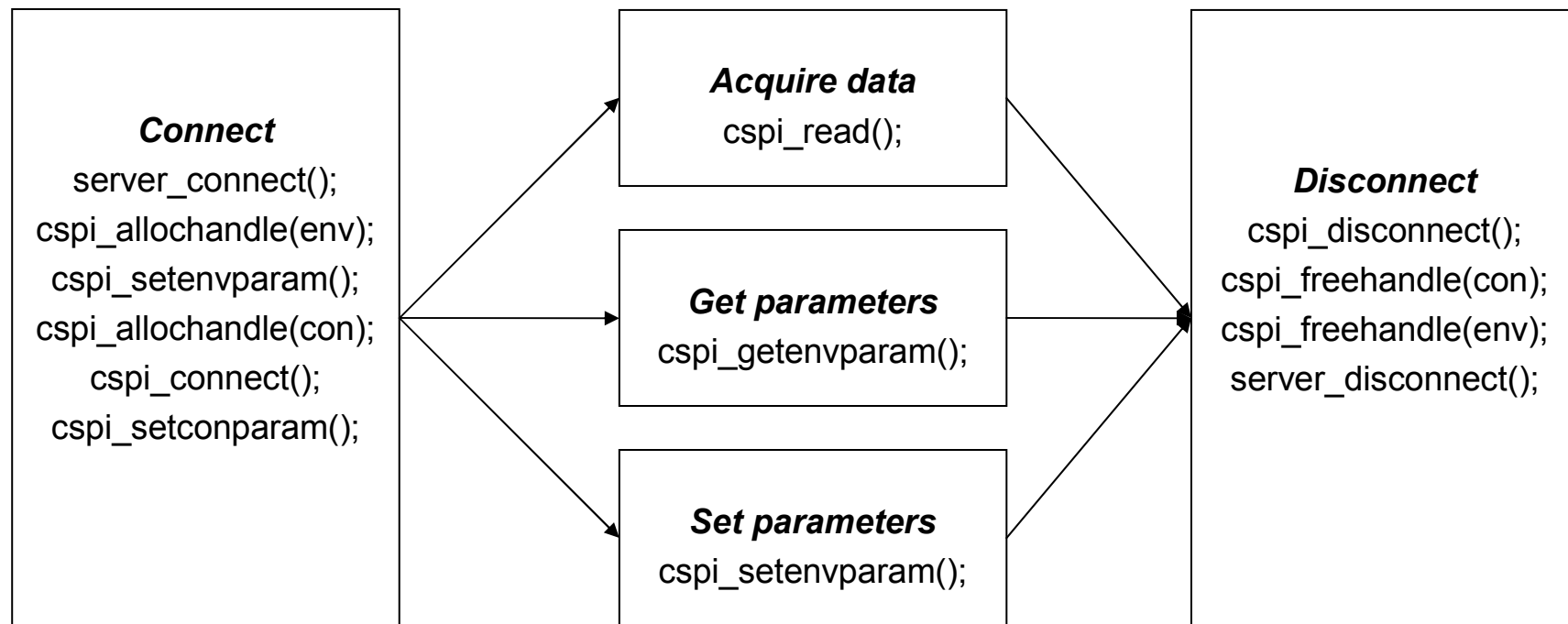
- Instrument dependent
- CSPI is for Platform A instruments (Libera Brilliance, Libera Photon,...)
 - First generation API
- MCI is for Platform B instruments (Libera LLRF, Libera Brilliance+,...)
 - Second generation API
 - Modularity
 - Networked by design
- Both APIs are actively developed and supported

Libera API Properties

- Performance
- Availability
- Thread safe
- Transparency
- Flexibility
- Modularity
- Scalability
- Interoperability



Example of Libera API Usage



What Future Holds?

- Active development
- Active support
- New functionality
- Adapting functionality

