

Control Systems for Accelerators, Operational Tools

Roland Müller*

Helmholtz Zentrum Berlin für Materialien und Energie, Berlin, Germany

Acronyms

ACOP	Advanced Component Oriented Programming
ALS	Advanced Light Source, Berkeley, USA
ANL	Argonne National Laboratory, Chicago, USA
API	Application Programming Interface
APS	Advanced Photon Source, ANL
ASIC	Application Specific Integrated Circuit
BESSY	Berliner ElektronenSpeicherring Gesellschaft für SYnchrotronstrahlung, Berlin, Germany
BOY	Best Operator interface Yet
BEAST	The Best Ever Alarm System Toolkit
CAMAC	Computer Automated Measurement And Control
CERN	Conseil Européen pour la Recherche Nucléaire, Geneva
COACK	Component Oriented Advanced Control Kernel
CSS	Control System Studio
DESY	Deutsches Elektronen Synchrotron, Hamburg, Germany
dm2k	Display Manager 2000
DMZ	DeMilitarized Zone
DSP	Digital Signal Processor
edm	Extensible Display Manager
ELI-ALPS	Extreme Light Infrastructure (ELI) - Attosecond Light Pulse Source
EPICS	Experimental Physics and Industrial Control System
FPGA	Field Programmable Gate Array
GIS	Geographical Information System
GTACS	Ground Test Accelerator Control System
HDL	Hardware Definition Language
HERA	Hadron–Elektron-Ring-Anlage, DESY
HMI	Human–Machine Interface
IBIC	International Beam Instrumentation Conference
ICALEPCS	International Conference on Accelerator and Large Experimental Physics Control Systems
IOC	Input–Output Controller
ISOLDE	Isotope Separator On Line DETector, CERN
JaCOW	Joint Accelerator Conferences Website

*E-mail: roland.mueller@helmholtz-berlin.de

jddd	Java DOOCS Data Display
KEK	kō-enerugī kasokuki kenkyū kikō, High Energy Accelerator Research Organization
KVM	Keyboard Video Mouse
LCLS	Linac Coherent Light Source, SLAC, Stanford, USA
LINAC	LINEar ACcelerator
LTS	Long-Term Support
LVDS	Low-Voltage Differential Signaling
medm	Motif Editor and Display Manager
NSLS	National Synchrotron Light Source
PCaPAC	PC at Particle Accelerator Conference
PCB	Printed Circuit Board
PLC	Programmable Logic Controller
PVSS	ProzessVisualisierungs- und SteuerungsSystem
RCP	Rich Client Platform
RDB	Relational DataBase
SACLA	SPring-8 Angstrom Compact Free Electron Laser
SerDes	Serializer Deserializer
SLS	Swiss Light Source, PSI, Switzerland
SoC	System on Chip
SSC	Superconducting Super Collider, Texas, USA
VME	Versa Module Eurocard
VXI	VME eXtension for Instrumentation

Abstract

Today engineers designing, building, modifying or migrating control systems for accelerators or large physics experiments have free access to very mature, adaptable, reliable, and high-performance control system toolkits, which have been developed by professional, global collaborations. Interconnectivity between the different control systems is well developed, even industrial vendors provide their devices “control system ready”. With respect to software and networking, an abundance of solutions and applications is already available. But the collaborative power of sharing, effort and success fades as it comes to non-standardized hardware. FPGA-based boards and recent attempts, like the “open hardware initiative”, might move this frontier and open up uncharted waters.

Introduction, Overview

Accelerators serving light sources and free electron lasers consist of many sophisticated sub-systems fulfilling the specifications required by the target parameters of the facility: a variety of technologies and components contribute. These are high-precision beam guiding and shaping magnets as well as sophisticated synchrotron radiation generating tunable magnetic structures.

Beam conditioning elements, like active or passive RF cavities, beam pipe structures, vacuum pumps and more, are needed to provide and preserve beam energy, intensity and lifetime. Diagnostics, cooling, power supplies, timing, protection systems, etc. all have to be orchestrated to reach the operational state the accelerator has been designed for.

Being one of the major light-source building blocks, the *control system* denotes the *domain-specific IT infrastructure*,¹ which allows one to remotely control and to survey all devices and mandatory subsystems, essential or useful for the operation, states and transitions of the accelerator complex. It comprises a complex and tailored architecture of hardware, networking, application software, a mixture of software libraries, tools, protocols, and development techniques to address the facilities various requirements efficiently.

The control system has to mediate change requests in status and set points of the controllers involved and it has to make all sensor read-outs available. Software components have to allow for fault detection, reporting and recovery, and to provide various levels of automation, consistency, and documentation, eventually mapping all the different requirements of the various operation modes and their transitions. Acquired and stored process data have to be complete enough to describe the long-term experiment of operating the light source, allowing for event analysis, comparisons and trending, even correlating minutes from years before or later.

First of all the control system framework has to allow for the commissioning of the light source, i.e., accelerator and insertion device tuning. Then, for the usability of the provided light, the control system has to provide operational sequences and surveillance tools; it has to enhance monochromator performance and provide triggers to the data acquisition systems and normalization capabilities. In a holistic way the control system framework has to stretch from the electron source to photon beam property manipulation and sample illumination to the data acquisition and sample environment. To have even the technical infrastructure (cooling water, air conditioning, electrical power) included is desirable, but this is often postponed due to resource issues.

Origins: A Community Evolves

Early accelerator projects underestimated the importance of a functional control system. Consequently the control part of the accelerator was frequently subject of improper functionality, failure of timely delivery or cost overrun, sometimes even preventing the commission of the otherwise complete accelerator. It was not before 1985 that control system specialists recognized the need to exchange ideas, solutions, experiences on an international scale. Starting with workshops in Brookhaven, Los Alamos, and an EuroPhysics conference in Villar sur Ollon, 1989, the first International Conference on Accelerator and Large Experimental Physics Control Systems (ICALEPCS [2015](#)), Vancouver, Canada initiated a biennial series of conferences intended to:

- Facilitate the interchange of ideas and information between control system specialists working on large experimental physics facilities around the world (accelerators, particle detectors, tokamaks, telescopes, etc.);
- Create an archival literature of developments and progress in this rapidly changing discipline;
- Promote, where practical, standardization in both hardware and software;

¹In the following *control system* is frequently used as a synonym for the complex *domain-specific IT infrastructure*, even if it has a wider scope.

- Assist colleagues from technically less developed countries in the realization of all of these goals.

In the ICALEPCS (2015) context the term “control systems” is broadly interpreted to include all components or functions, such as processors, interfaces, field busses, networks, human interfaces, system and application software, algorithms, architectures, databases, etc., and all aspects of these components, including engineering, execution methodologies, project management, costs, etc.

Around 1996 IBM PC/AT technology, MS-DOS Windows 3.1-, 95- and NT-based control activities at KEK/PF-LINAC (COACK), CERN/ISOLDE, DESY/HERA (ACOP) were in contrast to the at that time common UNIX or VMS workstation and rack-mounted CAMAC, VME/VXI front-end computer platforms. Exchange of ideas and solutions within this cooperation materialized 2008 in a first PCaPAC workshop at DESY. These initial gatherings of this subgroup of control system specialists, who were inspired by the Microsoft/office technology, have been continued. The scope has been extended, the coverage of control topics being still more focused on accelerators and PC type solutions. With PCaPAC a branch of conferences complementary to ICALEPCS has formed.

Diagnostics, the specific subset of accelerator components and techniques measuring the properties of charged-particle beams produced in high-energy accelerators is of special relevance for the control system: data produced by diagnostic subsystems are a prerequisite of analysis, reasoning, and taking action in accelerator operation (Schlott 2015). At each accelerator project the control and diagnostic groups entertain a vivid relationship of balancing technical requirements and discrimination of responsibilities. For controls a specific understanding of diagnostics and beam instrumentation techniques is necessary. The diagnostic specialists created their own forum of knowledge exchange, establishing workshops on Beam Instrumentation and Diagnostics for Particle Accelerators, initiated 1989 in Brookhaven (BNL) and 1993 in Montreux. The following series of workshops and conferences, DIPAC and BIW, are now merged to the global IBIC conference. IBIC venues circle through the continents Asia, Europe, America, similar to the controls (ICALEPCS and PCaPAC) and the related accelerator physics (IPAC, FEL) series of conferences.

Experiences, achievements, best practices, and proposals relevant for the field and the respective points in time are documented within the proceedings of these series of conferences on the topics of controls and diagnostics. Proceedings have been published in book format, distributed by CD's or the conference web sites, and they found entrance into document databases (INSPIRE 2015). Now the conference proceedings are all stored and online available at the searchable JACOW (2015) repository, ICALEPCS (since 1999), PCaPAC (2008), DIPAC (1999), BIW (2008), IBIC (2012).

Of course, control experts benefit from stimulations and impulses given at related conferences: IEEE Real Time Conference (IEEE RT 2014), Better User Group Software (NoBugs 2012), Computational Accelerator Physics (ICAP 2012), Accelerator Reliability (ARW 2015), and Operations (WAO 2014) workshops.

Powerful Cooperations: New Framework Qualities

As soon as the early “roll-your-own” control systems had been documented and the findings published, various approaches could be compared, and similar structures could be identified. Several attempts to join forces and share successful approaches have been undertaken by individuals and laboratories, with varying relevance and durability.

Quality properties, like performance, reliability, portability, configurability, especially with respect to extension, adaptation, and integration capabilities made some approaches attractive. They showed re-usability at other facilities, and they attracted so many collaborators that the combined functionalities and improvements generated toolkits of unparalleled productive efficiency, specifically the *Experimental Physics and Industrial Control System* EPICS (2015) and TANGO (2015).

At ICALEPCS 89 a Los Alamos group presented their GTACS control system, seeking partners. The APS (ANL) control group embarked on the Los Alamos idea to use it, improve it, and add missing features, while the Los Alamos group incorporated the changes and used them themselves. Interest in this portable toolkit spread to other laboratories, especially when the SSC project decided to use it for the whole accelerator complex. The initial development teams in Los Alamos and Argonne had the EPICS ready for the 93 APS linac commissioning. Development of the EPICS collaboration succeeded in maintaining a very successful working style: hundreds of using and contributing laboratories and projects exchange status and ideas in regular meetings (EPICS Collaboration 2015), with problem solving via active and focused email lists providing answers and solutions quasi-real-time due to the large time zone span. A very professional core development team guarantees code integration, testing, releasing, and bug fixing. The result, a wealthy, powerful, portable toolkit covering a broad spectrum of hardware devices has significantly contributed and modeled today's control system options and choices. Today the EPICS (2015) collaboration exemplifies capabilities of committed people utilizing the proper means of coordination and creative freedom. The EPICS toolkit serves as control core framework (section “[Frameworks: Toolkits, Software Base of Controls](#)”) at various light sources (ALS, APS, BESSY II, SLS, Diamond, NSLS II, LCLS).

Similarly, the ESRF control system team was looking for partners. Their approach was based on the devices in a device server model as the key concept, very different from the channel access and real-time database concept of EPICS. As a proof of portability the control system has been modified to TACO and installed at the 26 m radio telescope of the Hartebeesthoek Radio Astronomy Observatory (South Africa). Modified and augmented to Tango it has been presented at the 1999 ICALEPCS in Trieste. Soleil and later Elettra adopted TANGO (2015). These three institutes secured evolution and development of the system, ALBA and the Petra III beam lines joined, followed by MAX IV, Solaris, ELI ALPS and others (section “[Frameworks: Toolkits, Software Base of Controls](#)”). The heart of the Tango development is the Tango consortium; use cases and adopters disseminate. The exchange of achievements and requirements is also coordinated by regular meetings (TANGO Collaboration 2015).

Other open-source frameworks have a similar potential for light sources control systems, yet with a much smaller outreach. We have, for instance, DOOCS (2011), used for Flash I/II (DESY) and the XFEL, [TINE](#) for HERA and Petra III (DESY). MADOCA (Tanaka et al. 1997) was created for Spring8 and is further developed for SACLA (ICFA 2008) and Spring8 II (Spring-8-II Conceptual Design Report 2014).

With several competing choices available the babylonization of control systems (Duval et al. 2003) was a temporary menace. But in the mean time connectivity between the different toolkits is provided by gateways, protocol converting stubs etc. Today selection of one of the toolkits for the core framework of the control system typically does not include specific risks for a project (for the considerations involved see section “[Design Considerations](#)”) and still allows one to integrate subsystems readily controlled by an alternative toolkit.

Industry Support, Commercial Partners

In many industries, like oil and gas, electrical power, water and wastewater, building automation, semiconductor and electronics, automotive, pharmaceutical, food and beverage industries, the necessity of automation, supervision, and process control is also ubiquitous. Thus numerous commercial suppliers and vendors² of Supervisory Control and Data Acquisition (SCADA) systems exist.

Like control system implementations of accelerators and light sources, SCADA systems monitor, control, and alarm the plant and/or regional facilities operating systems from a centralized location. It includes the communication of information between a SCADA central host computer, many scattered units, and/or Programmable Logic Controllers.

Nevertheless which SCADA system could be utilized and to what extent depends on the specific circumstances of an accelerator-based project. Biased by the prevalence of SCADA systems within their industrial subsystems, CERN chose in 1997 PVSS (now named SIMATIC WinCC Open Architecture (2015)) for all four LHC detectors, and made it a CERN standard for all SCADA related tasks – probably the largest SCADA-based control system within the world of accelerators. As a paradigm it is not necessarily advisable for a small facility (Smale et al. 2013).

The Vsystem control system framework originated in the accelerator world. Although of different origin, EPICS and Vsystem have a common root in AT Division at Los Alamos National Lab (LANL). Developed following customer requirements it is similar in some conceptual aspects to EPICS (Clout et al. 1997). The vendor Vista controls (Vista Control Systems 2010) received some orders based on the idea of buying the set-up and maintenance resources and getting a turn-key solution implemented.

Today several feature-rich frameworks are freely available and based on mature, easy to configure, and well maintained open software. Consequently out of commercial, domain-specific suppliers only those have become relevant who support one or various frameworks of choice (EPICS 2015; TANGO 2015) reasonably (Cosylab 2001–2015; I-Tech, Instrument Technologies).

National Instrument products hold a significant portion of application within most light-source control systems, frequently close to the area of diagnostic and experimental data acquisition (Fig. 1). This success is based on the LabVIEW (2015)-specific programming paradigm, which is especially attractive to engineers, due to the abundance of drivers for diagnostic equipment.

Control System: Definitions and Scope

A confusingly huge variety of hardware choices, software options, and configurations is suited to fulfill the remote control requirements of light-source facilities efficiently. A simplified view, the standard model of controls (Kuiper 1991) as well as extensions to this model (Thuot and Dalesio 1993), has been a very helpful guide in the early days of control system design.

The picture identifies a common 3-tier architecture (see Fig. 2): the *console layer* offering the human–machine interface, typically in the control room (see section “[Supervisory: Console, Command](#)”), a *network layer* mediating the process data transmission (see section “[Network: Cabling, Data Transmission, Protocols](#)”), and the *equipment access layer* providing digital and analog command and control of the devices (see section

²An arbitrary list in no particular order: Intellution, Rockwell Software, Wonderware, Citect, GE/Fanuc, Siemens, Iconics, USDATA, National Instruments, Indusoft, Think & Do.

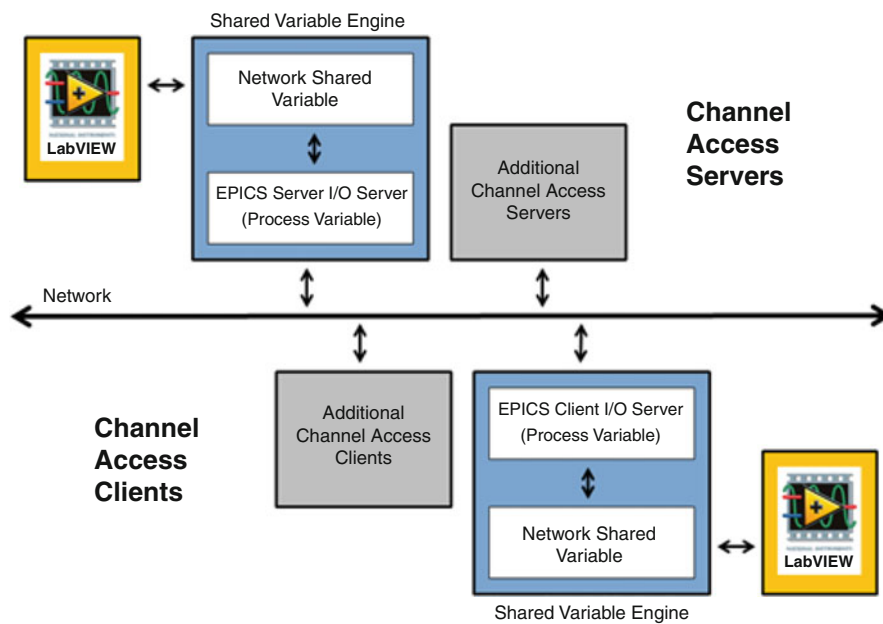


Fig. 1 LabVIEW datalogging and supervisory control and the LabVIEW real-time modules attached to the EPICS framework (National Instruments 2012)

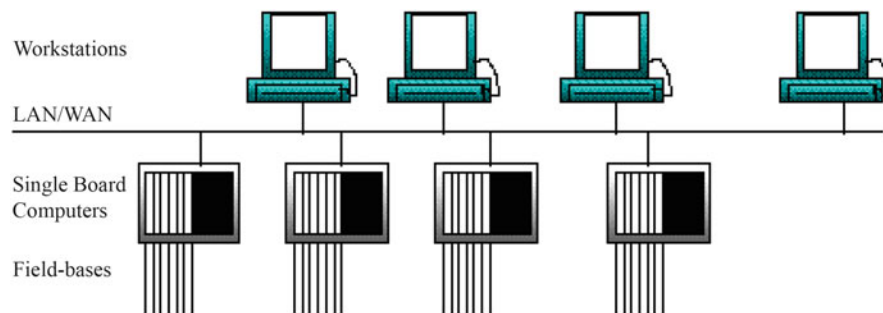


Fig. 2 The 3-tier standard model of distributed control systems (Thuot et al. 1996)

“Front End: Analog, Digital I/O”). The standard model is still the dominant structure of midsize, intermediate developed control systems. Even in complex heterogeneous systems and in nearly autonomous control subsystems the three functional tiers are usually identifiable.

Design Considerations

The art of control system design is a proper balancing of various figures of merit. Engineering projects, like complex control systems, should foresee a proper *project management* over life-cycle phases of design, fabrication, test, and operations, and they include schedule, cost, and quality assurance. This should pay attention to interface management, validation and verification processes, reliability, availability and maintainability, high risk impact aspects, and construction, commissioning, and operational logistics. Control system design goals are beneficial with long-term effects on development and maintenance costs. A reasonable perspective for continuous future improvements accompanying the long-term plans is mandatory.

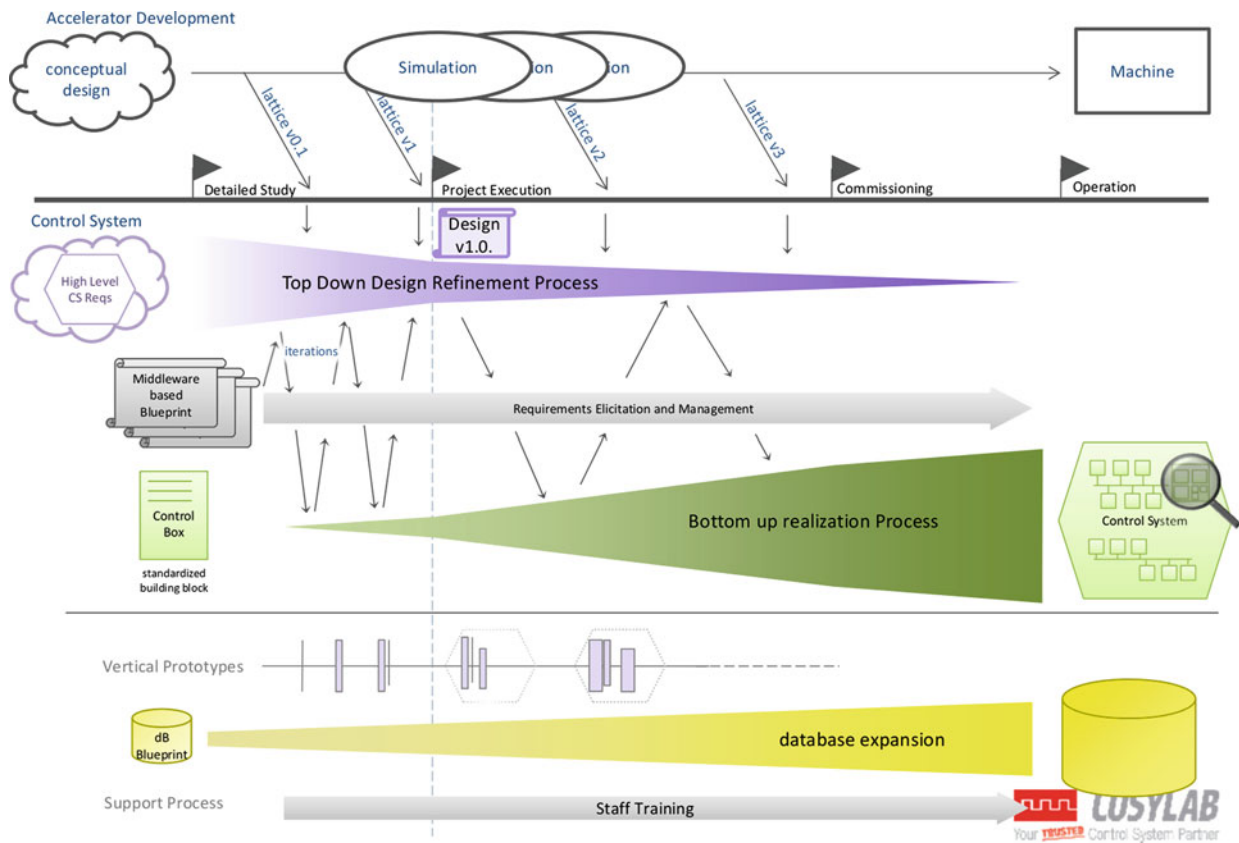


Fig. 3 Project management, down from control requirements, up from devices to control and integrate, all governed by middleware connections and configuration data base (slide from CosyLab Plesko 2014, a commercial partner in light-source projects of varying scope)

Resulting *systems and software requirements and specifications* need to be structured consistently, prioritized, traceable, and in relation to the various stakeholders perspectives. The requirements' structure needs to be suitable for concurrent top-down, bottom-up and middle-out developments (Verstovsek et al. 2007; Plesko 2014), reflecting starting points such as feature requests and use cases, equipment integration requests, and service-centered approaches (see Fig. 3).

A distinguished goal for light sources is a generic, but feature-rich tool set that can be applied both to the *accelerator domain* and the *beam lines*, addressing common experimental or scientific requirements. It is a matter of resources which features (scientific analysis and operation requirements), quality requirements (integration, performance, services, reliability, security, safety), and other issues will be emphasized in the course of the project, and when they will.

During the long-lasting period of light-source operations, the control system will be built by the aggregation of heterogeneous *components*, that will oscillate between *home-made*, hardware, software, and middleware exactly tailored to the needs and those shared between laboratories to various extents or coming from the open-source community. On the other end *components off-the-shelf* (COTS) from commercial companies promise saving of resources. Using COTS depends on the level of customization needed, to-date best practices, political decisions, financial or resource considerations. But also it depends on personal preferences as regards how these ingredients

are put together. Obviously, a potential for *cost reductions* lies in consistent *standardization*. It reduces the number of system types the staff has to cope with, minimizing staff training, and development and maintenance diversity. When choosing a control system framework (see section “[Frameworks: Toolkits, Software Base of Controls](#)”) for specific technical reasons, a sufficiently detailed requirement analysis is needed, especially when subsystems with known real-time demands are involved, and a proper validation should be finalized in due time.

Ultimately control systems at light sources should allow one to focus undistractedly on the intended delivery of photons on sample. This emphasizes the importance of trust, confidence in a *self-evident operability of the system* at hand, in all aspects and properties: dependability (availability, robustness, and recoverability), fault tolerance (isolated failures should not compromise operation), predictability (precision, reproducibility) extensibility (open, scalable, easy to expand, integration friendly, and allowing one to implement enabling technologies), and usability (learnability, task efficiency, memorability, understandability, and subjective satisfaction).

Control System, the Domain-Specific IT Infrastructure

In setting up the control system infrastructure *standard IT technologies, tools and methodologies* for optimizing performance, managing resources, and addressing off-normal situations across the infrastructure of networks, processing nodes, data storage systems, and databases apply. Power usage, virtualization, unified operating system installation, and control system computer configuration techniques have to be analyzed at this general level of standard IT.

Specific to the control domain are issues connected with managing and accessing large archived data sets (MapReduce [2011](#); Hadoop [2014](#)), in view of scalability and extendibility, custom arrangements for cyber security (section “[Access Control](#)”), extensions embracing a broader variety of inputs as encapsulated by the concept of the *Internet Of Things* (IoT [2015](#); Hardion et al. [2013](#)), incorporation of cloud computing (section “[Mobile Devices, Cloud Computing](#)”), all this has to be included into the coarse IT infrastructure to be set up for controls. System structuring properties such as hierarchy concepts, object orientation, browsing services, redirection, gateways, aliasing, etc.; they all impact practical complexity handling.

Standard Model: Components, Hardware Inventory

At the upper two layers (servers, consoles as well as network components) the baseline control system infrastructure usually differs not very much from systems provided and maintained by the central IT division of the laboratory. The central IT department generally covers offices, administration, communication etc. Thus, for economical reasons, whenever control system units comply with central IT standards, the call for tender, purchasing, initial configuration, repair, and replacement is frequently handled by their groups (network e.g., Janousch [2015](#)).

Apart from the domain-specific usage (limited identity management, restricted system updates and upgrades, only as far as operation allows, and where consistency and functionality is proven and tested) and the elevated requirement for availability, control IT equipment is no more special.

Selecting specific I/O front-end technology suited for the intended control system use, IT equipment becomes non-standard and has to comply with the design requirements. Dependent on the scope of the facility, the resulting choices for the general control system’s architectural

and structural elements are then frequently replicated for simplification reasons within specific subsystems, cryo system control, conventional facilities, sometimes parts or remains of a legacy system, etc. Gateway techniques might be necessary to provide the required connectivity and protocol conversion (proxys, stubs, APIs).

Customarily virtualization technologies are utilized by general IT to exploit ease of maintenance (cloning, snapshot management). For controls, virtualization enables a new paradigm of high availability (rapid, non-interrupting migration of running systems to sane or not serviced nodes) (Kapeller et al. 2013; Engel et al. 2013).

Supervisory: Console, Command

In the early days the choice of the proper computer hardware platform and the operating system (OS) tied in was of outstanding importance and had far-reaching consequences on the costs involved. Today basically all relevant control system frameworks (section “[Frameworks: Toolkits, Software Base of Controls](#)”) can be installed and function on all major platforms and operating system flavors.

Thus it has become more a matter of rack packaging, cooling, optimized distribution of virtualization host and guest systems, display technology (overview control room monitor wall (see Fig. 4), multi-screen, -purpose consoles, hand-held devices), noise control (KVM separation), support strategies for the OS (LTS), and proprietary software additions (drivers). Obvious advantages of disk array systems, storage virtualization (SAN 2015; Kapeller et al. 2013) and blade enclosures (Blade Server 2015) have proven very beneficial for the control system-specific requirements, availability, flexibility, reliability, and performance.

Advanced software techniques are relevant for the development tools provided: rapid prototyping, agile development (python, Qt), new programming languages (Scala, CoffeeScript, etc.), design and code for easy debugging, refactoring in practice, model-driven development (Beltran and Gonzalez 2003), domain-specific languages and code generation (Voelter 2011; DSL 2005–2014).



Fig. 4 Central control room for the BESSY II and MLS light sources

Usage of web services and service-oriented architecture (SOA), live status and data displays have become common to extend control room and facility site information space and allow for remote participation and on-call service support (WebPDA [2013–2014](#)).

Network: Cabling, Data Transmission, Protocols

The baseline network infrastructure has to comply with the standards of Structured Cabling ([2015](#)), especially with the ISO/IEC 11801 *IT, Generic cabling standard* covering both balanced copper and optical fiber cabling, which can be tailored to the needs of elevated electromagnetic compatibility (EMC) and radiation levels.

Specific needs of control subsystems enforce a correspondence in the network structure: these are the deterministic properties of shared data in reflective memory (RFM) applications, security encapsulation of PLC networks with their known vulnerability to off-label usage, sophisticated functionality optimized installations like the Diamond Light Source Fast Orbit Feedback Communication Controller (Uzun [2007](#)).

Field busses, the family of industrial real-time distributed automation and control network protocols (standardized as IEC 61158) are used at the lower accelerator process levels; formerly we had MIL-STD-1553, Bitbus, later Modbus, CAN, Profibus, WorldFib. These network segments have their specifics as regards maximal number of nodes, segment length, data transmission speed, multicast capability, and topology. Deterministic behavior, short cycle times $\leq 100 \mu s$, low communication jitter $\leq 1 \mu s$, and thus real-time automation requirements can also be met by utilizing the physical layer of the existing network structure ([EtherCAT](#)). Even for the transport of electrical power to the network nodes along with the data, standards and implementations are available (Power over Ethernet, PoE).

Wireless techniques (WLAN), ubiquitous elsewhere, are in the control system environment at best used for maintenance purposes (mobile consoles), by the light-source users within the data acquisition environment, or studied on test benches (Badillo et al. [2012](#)).

Front End: Analog, Digital I/O

At the process level, electrical, mechanical, analog and digital signals are handled by front-end input–output controllers (IOCs), typically mini or micro computers with modules plugged into the bus of the specific hardware platforms. IOCs are used to administer clock resources, memory, data transfer, analog to digital converters, and other I/O devices. With respect to hardware platforms there is a wide variety of choices, handling complexity, integration density, and performance (speed, reliability) as well as evolutionary developments, both in small development steps and in qualitative changes.

CAMAC was the standard bus and modular-crate electronics standard (standard EUR 4100, IEEE 583, defined 1972) for data acquisition and controls used in nuclear and particle physics experiments and in industry. Thus CAMAC was everywhere in early accelerator control systems, typically in connection with PDP 11 or VAX/VMS on the command level.

Later (1979) the VME computer bus standard, originally developed for the Motorola 68000 line of CPUs, became widely used for many applications and was standardized by the IEC as ANSI/IEEE 1014–1987. It is physically based on Eurocard sizes, mechanics, and connectors (DIN 41612). VME experienced many extensions on data sizes (VME32, VME64), application areas (VXI, instrumentation), (VXS, switched serial), (VPX, switched fabric), and general extendibility

(daughter boards, IP modules, piggyback controllers). The VME platform has the advantage of a huge offer of communication (serial, GP-IP, field bus) and configuration options. The spreading of EPICS, originally featuring IOCs based on the VxWorks realtime operating system running on VME, contributed to the acceptance of VME as the common entrance to other process levels.

COTS systems became increasingly attractive, both due to the improving stability of the underlying operating system environment and the common pressure on cost reduction. The once IBM PC compatible computer bus PCI began to be used for inexpensive, less critical solutions. The PCI-based platform has been significantly developed and extended since (compactPCI/PCIE, PXI/PXIe, ...). Today the PCI-based technology market for measurement and automation applications that require high-performance and a rugged industrial form-factor offers cost effective solutions covering a huge share of control system requirements.

Programmable logic controller (PLC) originated as replacement for relays, cam timers, and drum sequencers in the automation of industrial electromechanical processes. PLCs are armored for severe conditions (dust, moisture, heat, cold), and have the facility for extensive I/O arrangements. These connect the PLC to sensors and actuators. PLCs read limit switches, analog process variables (temperature, pressure, positions of complex positioning systems). On the actuator side, PLCs operate electric motors, pneumatic or hydraulic cylinders, magnetic relays, solenoids, analog outputs. The I/O arrangements may be built into a simple PLC, or the PLC may have external I/O modules attached to a computer network that plugs into the PLC. The functionality of the PLC has evolved to include sequential relay control, motion control, process control, distributed control systems, and networking. Programming is standardized by IEC 61131 (93). The data handling, storage, processing power, and communication capabilities of some modern PLCs are approximately equivalent to desktop computers.

A novel region of processing speed became accessible with ASICs and DSPs, while mechanical support and electrical connect of electronic components could be realized by printed circuit boards (PCBs). With the advent of the much more flexible and similar powerful FPGAs the covering of required I/O and processing functionality entered a new era. Configurable I/O capability is given by FMC (FPGA mezzanine card, Vita 57.1), VFC (VME FMC Carrier). Discrete logic components as well as embedded processors become obsolete with FPGAs. As a System-on-Chip (SoC), digital, analog, mixed-signal, and often radio-frequency functions are all configured on a single chip substrate.

The last decade's revolution in analog and digital communication technologies accelerated the need for changes. New standards became essential as technology developed. MicroTCA.4, xTCA, (Larsen 2012) has the potential to be the physics standard of choice for new projects, upgrades for the next 1–2 decades (see section “[Low-Level RF Control](#)”). Integrated SerDes communications obsolete parallel bus backplanes. Low-voltage differential signaling (LVDS)-balanced multi Gbps backplanes minimize discrete switch blades. Integrated self-diagnosis has become available.

Upgrade and maintenance strategies define when legacy systems, still existing and in use at the facility (NIM 1966; CAMAC 1976; FASTBUS 1986; VMEp 1998), are no longer cost-effective, supportable or even affordable.

For hardware platform or field bus *integration into the control system frame work*, generic interfaces exist (asyn (EPICS 2015), CDI (TINE), abstract classes (TANGO 2015), etc.). Driver availability can be sped up by using or customizing existing drivers, with the EPICS driver base (EPICS 2015) being the largest. Interoperability between protocols can be achieved by various methods (Duval et al. 2003).

Systems for safety (accelerator, personnel, vacuum, equipment, etc.) (see section “[Safety, Security](#)”) and synchronization (see section “[Timing and Synchronisation](#)”) typically use dedicated signal lines for hard real-time tasks; usually they can be integrated for control, monitoring, and archiving tasks reasonably well.

Networking: Configuration, Topology, Connectivity

Application of the network infrastructure for control purposes might require additional structures: e.g., bandwidth distribution and separation, suited to match data flows, sources, and sinks. Broadcast boundaries need to be set up or overcome, depending on the protocols in use. Overall throughput performance within certain process data subnets might benefit from built-in multicasting from each node (TINE), especially where high data load sources have several data sinks. An example of an application is an advanced, feature-rich GigE Video-based system (Weisse et al. 2007). Access control points have to be implemented to allow for device protection, protection from unintended, unauthorized usage, and general cyber security (firewalls, gateways; see section “[Cyber Security](#)”)

Subnetting

A *defense-in-depth* strategy is common practice to protect control systems from unintended, unexpected, detrimental usage, behavior, changes: this means layering security mechanism, so that the impact to one mechanism as a result of failure is minimized. It results in a multi-layer network topology, providing logical separation between corporate and control system networks. Behind the firewalls, control system networks for accelerators are typically further separated into private machine networks, tailored to the needs of the nodes attached. Legacy systems, field-bus segments, PLC communication links, and camera networks are treated specially.

Remote access from office and experimental user network to machine and beam-line networks should be restricted to dedicated gateways, realized e.g., via *ssh* comparable techniques, usable only for a well-defined list of users, possibly only allowed during times, where unintended side effects are acceptable, e.g., facilities shutdown or machine shifts. Appropriate privilege management, fault-tolerant design, and redundancy for critical components have to be in place. No direct communication between control system and campus network, e.g., no shared file server disk space, must be allowed (DMZ) (Fig. 5).

With respect to process data ubiquity, all signals are required to read only at the best possible rate literally everywhere at the facility. To be able to control the back direction of writing/setting control points and the load put on a control server, the network has to provide the necessary repeaters, gateways to allow for the implementation of role-based access (see section “[Role Based Access](#)”), and data concentration.

Middleware

To create data exchange software, serving interprocess communication and connections between front-end and console computers, has been one of the major and important development tasks in setting up a control system in early days. Today this data exchange is at the core of all major control system frameworks (see section “[Frameworks: Toolkits, Software Base of Controls](#)”).

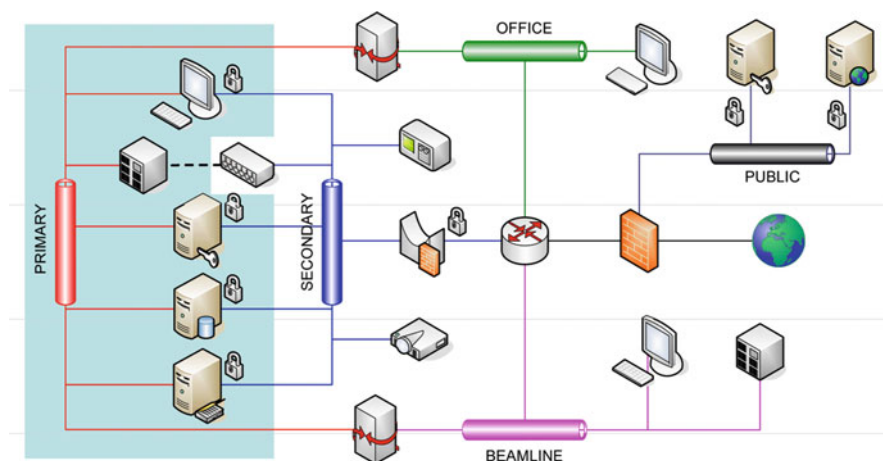


Fig. 5 Accelerator control system network security nesting (Leech 2007)

Typically all desirable middleware capabilities are available: Small library size, low memory and resource usage, scalability, certain performance characteristics (few to thousands of messages/sec, payloads may vary from few Bytes to several MB, variation of response times depend weakly on number of clients and messages). No, or only a few, external dependencies exist that can be linked with an application, additional services (e.g., brokers, global servers, daemons) are preferably not mandatory.

Available middleware belong to different categories (data-centric, object- and message-oriented) even the choice of messaging patterns is implemented (send/receive, call-back, request/reply and/or publish/subscribe, synchronous and asynchronous communication). Implementations feature a built-in Quality of Service (QoS): time-out management, message queues and priorities, various thread management policies, and an easy to trace peer-to-peer communication. They are open source, with a license allowing for redistribution, good documentation, and support from a large active community.

In the course of the evolution of control system frameworks, either the data model, e.g., devices and properties, base of remote device access (RDA) protocol at CERN, had to be mapped to the new middleware (here CDEV (Chen et al. 1995), otherwise CORBA (Böge and Chrin 2005; Smale et al. 2013; ZeroMQ 2007–2014). Or, when the data representation had to be augmented, new middle layer services needed to be added, to make better use of the proven technology (e.g., PVData/ EPICS V4 (Korhonen et al. 2013; Dalesio et al. 2013), multicast/Tango V8 (Goetz et al. 2013)).

In addition, specific subsystems require their own connectivity software, e.g., access to legacy systems, to the RDB, to PLC systems (Beckhoff ADS, ZeroMQ, Doocs-Server for XFEL undulator controls, or Siemens, OPC, CA-server in the EPICS world).

Frameworks: Toolkits, Software Base of Controls

Software, the application domain of the control system infrastructure, has to cover all digital information processing from data acquisition systems to scientific analysis and should fulfill the appropriate quality standard (ISO/IEC 9126). Basic requirement on the framework software is the capability to integrate the desirable hardware platforms, operating systems and programming

languages, the communication system with QoS and name resolution, and application development support (sketch of the TANGO framework Fig. 6).

For a status and snapshot of the present usage of sharable control system frameworks, even going through the list of light sources world-wide (Light Sources of the World 2015) one finds a colorful picture of homogeneous implementations, coexistencies, and evolutionary intermixtures of the major frameworks and toolkits.

An incomplete, arbitrary list of present control system framework prevalence can give a flavor of preferences and collaborations involved: EPICS (APS, BESSY, CLS, Diamond, LCLS, NSLS II, Sirius, SLS), Tango (ALBA, ESRF, Soleil, Petra III Beamline, MAX IV, Solaris, Elettra, FERMI), DOOCS (Flash, XFEL), Tine (Petra III), MADOCA (Spring8, SACLA). For the European XFEL a new framework Karabo (Heisen et al. 2013) aiming at data filtering and processing, the way HEP detectors are used to, is developed from scratch (Fig. 7).

Configurable generic tools, capable to cover the base operational requirements (see section “Base Operational Tools”), especially at day one, support customization of the base control system framework to the peculiarities of the facility very much. As sophisticated, very specific application programs are required, the software technique of providing tailored libraries, components, interfaces, and classes is frequently used (e.g., Abeans (Verstovsek 2002; Verstovsek et al. 2003), ACOP, COMA (Bobnar et al. 2007), ATKWidget) to enhance re-usability of code and consistency of the product.

Higher-level, model-based applications need the mapping of engineering units to physics space, i.e., power supply currents to magnetic fields, BPM readings to orbit data at their respective position, and the area of configuration management (see section “Configurations, Databases”).

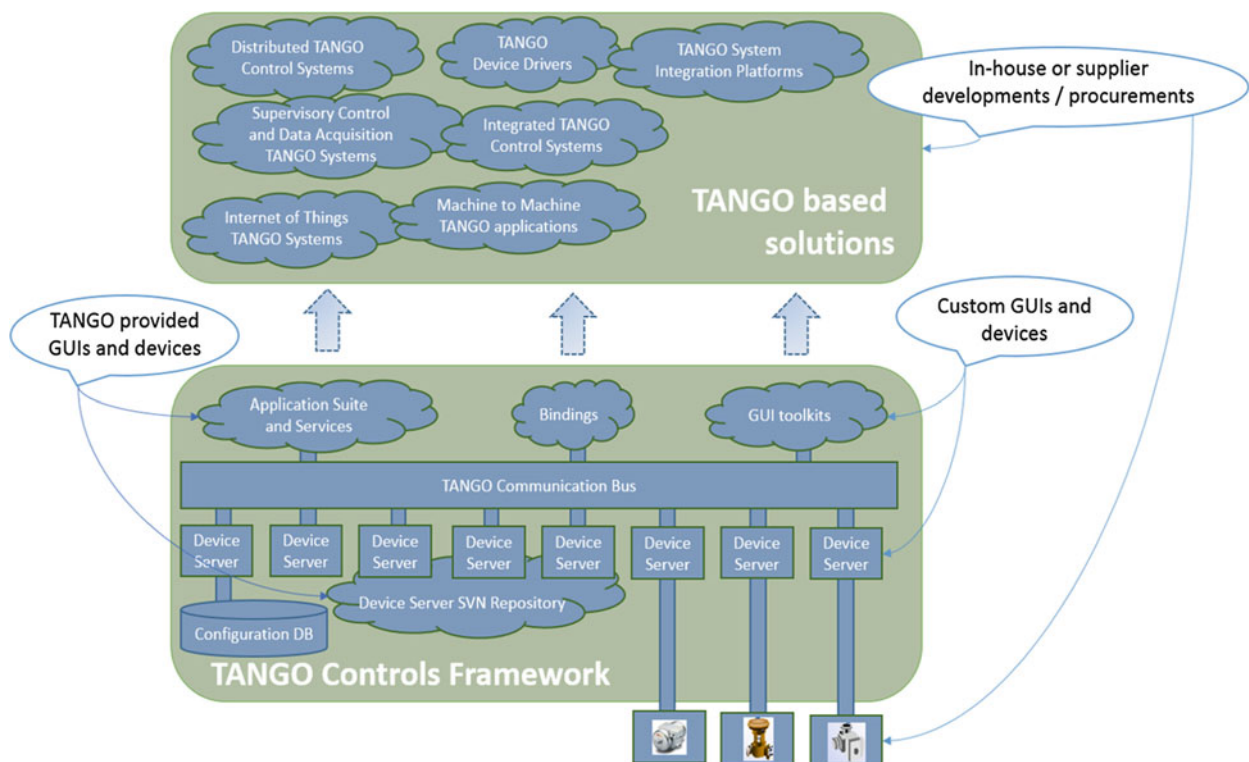


Fig. 6 The Tango control environment, featuring the integration of standard and custom devices, as well as the procurement support (TANGO 2015)

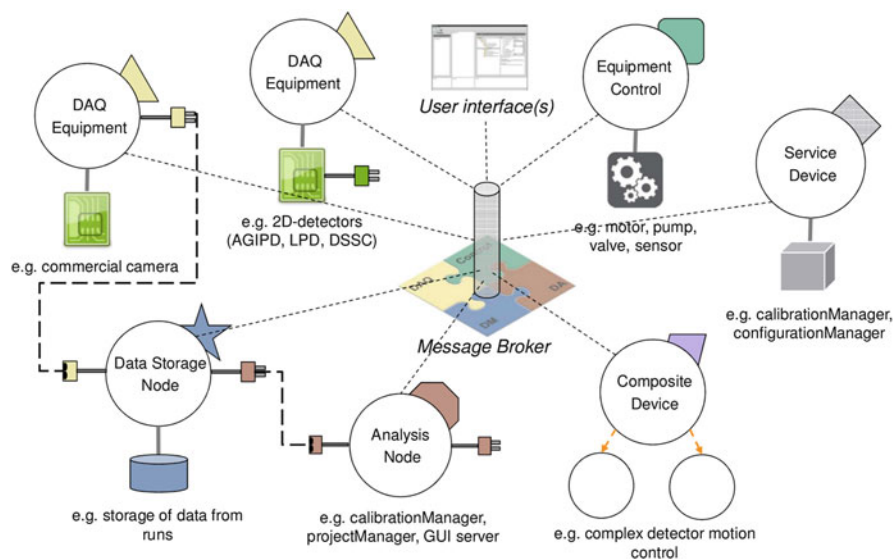


Fig. 7 The Karabo vision, emphasizing data flows at high rates and volumes, connecting data acquisition, analysis, and management. The control aspect is based on devices, command, attribute, property and plug-in technology, and self-description/configuration (Heisen et al. 2013)

A few software frameworks with varying portability have been created. XAL (Pelaia et al. 2007), an EPICS toolkit based on Java, light-weight XML configuration files, provides the list of all devices, their properties, their relationships within the system, high-level machine representation, connection management, control toolbox. In the widely used Matlab Middle Layer (MML) (Portmann et al. 2005) package the correspondence between the physics model and installed devices is provided by a code section central to the toolkit.

Specific software frameworks, attractive by their command line interfaces (CLIs) and capabilities to generate graphical output or build user interfaces (GUIs), either home-grown (SDDS) (Borland 1998), public domain (tcl/Tk, perl, python, Qt, ...) or commercial (IDL, Matlab, Mathematica), get their share within control system software, as multiple control systems or data sources (Tango, EPICS, SPEC) are supported. Examples for dedicated, adapted packages, contained in the framework and connectable with other frameworks, are TAURUS (2015), for control and data acquisition. SARDANA (2015), on top of Taurus, phrased a general software suite for Supervision, Control, and Data Acquisition in scientific installations.

Based on the Eclipse RCP, an ambitious integrated workbench approach becomes feasible. Control System Studio (CSS 2013) aims at an extendable, homogeneous provision of various services. Today CSS, a product of the collaboration between different laboratories and universities, provides tools to monitor and operate large-scale control systems, such as synoptic, archive, trend, alarm displays fundamental within the accelerator community (section “Base Operational Tools”).

All software that becomes a requisite part of the light-source operation needs an appropriate development environment: version management, test suites and procedures, secure and automated deployment, roll-back capabilities, collaboration tools for teams working together, even over geographic distances and time zones.

Real-time controllers require specific techniques for development and interfacing. Framework-specific code processors (VDCT (EPICS 2015), POGO (TANGO 2015), Server Wizard (TINE), etc.) help with the configuration and integration of distributed control servers. The evolution in the PLC area toward Programmable Automation Controllers (PAC) code running on a variety of

platforms like PC, embedded controller, FPGA chip, or mobile units, can be created and configured by commercial environments like LabVIEW(NI), WinCC(Siemens). Advances in real-time OSs, Arduino (2015), and embeddable programming languages (Matlab) constantly change the field.

Upgrades

Control systems often experience various evolutionary upgrade changes to allow for new experiments and to achieve maximum performance of the scientific installation (extension, enlargement, modification, implementation of new capabilities, fighting obsolescence in existing control systems or existing platforms and frameworks).

Upgrade activities scale from minor (e.g., vacuum controllers replaced by modern units) to intermediate (e.g., klystron to solid state power amplifiers, microtron to linac injector) and huge (SLAC to LCLS II, hybrid legacy/EPICS, Spring-8 to SACLA plus Spring-8 II). Expertise required in upgrade projects has to span old and new devices, and existing and future technologies, frameworks, and methods.

The integration of heterogeneous control systems, incorporation of heterogeneous components, re-engineering and maintenance of existing control systems, maybe isolation of legacy subsystems for the next migration step; all this requires specific considerations and discussing. For instance, can the low-level control components be integrated together and coordinated at a higher level? What level of integration between process control and the protection and safety systems is required? How can in the present era of fast-paced technological advances further evolution be considered? Especially at operational and running light sources, minimized impact on user service is mandatory. Here the change control process and the optimization of the transition to the upgraded systems are issues by themselves.

Software: Applications

Real usability of the control system for operation of the light-source facility (section “[Design Considerations](#)”) requires a few essentials: a comprehensible, self-explaining *human–machine interface*, a set of fundamental *base operational tools*, and automated *tuning procedures*.

Human–Machine Interfaces

Eventually human beings utilize the control system and have to interact with computer-based systems. Humans control hardware as well as interact with software tools. Thus, ensuring that the human–machine interface (HMI) contributes positively to successful, reduced-risk operation and ease of use, one has to account for human factors.

Given the pace and quantum leaps in digital commodity gadgets, HMI is an extremely rapid evolving field. About a decade ago it was innovative to provide configurable digital knob dials for fine-tuning purposes, today the extreme sensitive and precise touchscreen smart-phone apps support beam-line readjustments next to the hardware.

Even if HMI is a science by itself, sticking to a few basic rules provides adequate visual guidance (Bakker et al. 1999), like mandatory color coding, window frame layout, shortcut behavior,

confirmation prompts on potential damage, etc., all applied to notification and alarm levels, device class, the functional domain, and the installation area.

Interface building tools (such as edm, CSS, jddd, and web tools, section “[Display Manager, Device Control](#)”) help to generate a consistent look and feel. User-built synoptic displays combine dashboard functionality of rapid access to required actions with documentation (geographical, functional, logical views) and semantic nesting.

Data visualization tools are the key to comprehend large amounts of data (such as overview panels, even large display walls in modern control rooms, archive viewers, plotting tools, and high precision graphics)

The inclusion of mobile devices into the HMI eases maintenance activities, remote support, and monitoring. Novel interface trends (such as virtual displays, intelligent data display, and natural language processing) might emerge or simply pass by.

Base Operational Tools

Users of the accelerator control system are typically three different groups with diverging requirements: Operators need appropriate tools to provide a high-quality beam, they ask for ease of perception and automation. Physicists run measurements, evaluate data, and tweak the accelerator; here flexibility and opulence in available methods is required. Equipment specialists introduce, modify, and maintain the devices of their responsibilities, they need access to and protection of intrinsic functional details.

Proper shift handling requires coordination of all past and pending interactions, relying on coherent reporting tools: electronic log books (ELOG [2002](#); Shroff et al. [2013](#)), trouble ticketing systems (Trac [2003–2015](#)), and activity tracking (operator, program, device state changes). Apart from this, a set of standard operational tools is present at all light sources.

Display Manager, Device Control

Remote control even in the narrower sense of elementary device access implicates the need for a number of tools. We have buttons, switches, dials, sliders combined with status displays, meters, trending diagrams, text displays, all combined in windows meant for the matching level of user understanding (operator/expert). Here display managers called generic programs (medm, dm2k, edm, jddd) have proven to be very powerful and suited programs: in edit mode they allow one to compose actors and sensors, graphical resources, logical conditions, and simple mathematics as regards the desired control screens.

In execute mode, the generated configuration files allow one to access all devices and, applying macro substitution, replication and nesting, and device groups and collections. Navigation, documentation, and visual guidance become available as a by-product of embedded launchable related displays, and meaningful synoptics views, sticking to color codes in title bars, diagram lines, and compilations of status conditions. Similar functionalities come with the CSS plugin BOY, in this case even with the option to export the screens to a WEB display (see [Fig. 8](#)) (Kasemir and Chen [2013](#)).



Fig. 8 WEB display of a BOY screen, mobile and desktop version

Alarm System

Alarms, the mechanism to alert operators of off-normal conditions that require a response, are essential to maximize availability of the light-source service under promised and expected experimental conditions. Alarm conditions are detected by the control system, transmitted to the operator, and logged for correlation analysis. Operators are notified as soon as possible to perform actions to prevent or mitigate undesirable consequences.

If possible, alarm conditions should be selected that allow to react, while intervention is still possible, not when it is too late. Alarms must contain enough information. Alarm aggregation and filtering should prevent alarm floods, and confusing and nuisance alarms, and multiple notification of the same event needs to be eliminated. The alarm handler should allow one to identify the root cause, find the affected device, and open the related panel for reset and state change; it should implement a consistent method to determine and prioritize alarms based on impact, consequences, and required response time (Hollifield and Habibi 2007). Success of the alarm system is more often dependent on the proper configuration than on the underlying tool (alh (EPICS 2015), BEAST (Kasemir et al. 2009), PANIC (Rubio-Manrique et al. 2014)).

Process Data Archiver

The operation of a light source is a long-term experiment: performance varies due to changes of known and unknown parameters. In the quest for understanding observations it is essential to record as many process variables as possible with a meaningful rate. Therefore all control system frameworks provide a process data store, the archiver, and retrieval, display, analysis, correlation tools, and the archive viewers.

Storage formats vary from real time buffers (compress, archive record, strip chart) to complete, long-term history, possibly with data aging and reduction, retrieval tools depend on intended usage (automation, flexibility, performance, storable procedures). The CSS DataBrowser plug-in (Shroff et al. 2013) covers strip-chart functionality and allows one to go back in history and add history data from the archive. As volumes increase, high performance, scalable and distributed storage, and a processing system for structured and unstructured data gain interest (e.g., Bigtable, Hypertable Marsching 2013; Giacchini et al. 2011).

The fast-acquisition communications network at Diamond allows one to collect position data from 174 Electron Beam Position Monitors (BPMs), a number of X-Ray BPMs and RF data at an update rate of 10 kHz and a total aggregate data rate around 15 MB/s eventually providing fast post-mortem data covering 4 days (Abbott et al. 2011).

Facility State Control

Well-tuned, optimized, stable set-ups of the facility need to be re-achieved safely and fast. Thus a reliable snapshot channel save–restore–compare tool is an important component. It typically allows one to change operation modes, and recover from shutdown activities or power outages. In principle, data could be tagged in the archiver and retrieved from there (Larrieu 2015). Unintended data losses at the volatile front-end computers are prevented by IOC autosave and recover procedures, either using a local or network drive or re-reading settings from the buffering hardware. The compare option allows one to identify deviations from the actual set points from the intended values. Precision setting of magnetic beam guiding elements requires additional reproducible cycling or degaussing procedures.

Accelerator Development Environment

Characteristic for accelerator studies are light-weight sequence programming tools with access to temporary installations and full scope access to diagnostic equipment. A valid physics model helps with predictions and data evaluation (Portmann et al. 2005; Böge and Chrin 2003; Yang 2013).

A control system simulator (Shen 2013) should consist of all kinds of control system framework components (applications, services, local control servers, etc.), except that local control servers are internally either dummies or connected to a software-based dynamic physics model of the accelerator machinery. The simulator allows for dry-run capabilities and is advantageous for staff training and software verification and validation, as it enables a more realistic, complex system environment.

Process Tuning, Feedback Systems

Light source facilities are very complex machines that cannot be operated without the use of sophisticated systems executing tasks automatically, not manageable manually by physicists or operators.

Examples are automatization components that simply prevent maloperation or keep the required timings, like sequencer programs for energy ramping (Birke 2009) or beam switching between parts of the facility. Or one has optimization tools, ensuring the stability of critical parameters during operation (fill pattern, tune, path length), tune the machines, and improve their performance. For the FELs specifically constant adjustment of bunch charge, compression, energy has to succeed on a shot-by-shot basis and involves accelerator units, timing devices, and detectors at the experiment (Kammerling and Schmidt 2013). Requirements for timing experiments at storage rings, pulse picking, kick and cancel, bunch length control (Jankowiak et al. 2015; Timing 2015) as well as wavelength selection at FELs (Mirian 2014; Parc et al. 2013) correspond to specific user service flexibility, supported by science enabling control system tools (Scafuri et al. 2013).

Tuning and optimization techniques utilize feedback or feed-forward techniques, predictive and adaptive correction systems, fuzzy logic, neural networks, genetic algorithms, use models, and simulators. On model-based tuning, incorporation of modeling results to real-time control closed loops is feasible; this results in systems that are more flexible and adaptive to minor changes. In some cases, e.g., orbit correction variants, system performance requires dedicated real-time platforms with deterministic communication systems and the usage of FPGAs (see section “[Specific Subsystems](#)”).

The ultimate goal, and thus of particular interest, are elements improved by artificial intelligence: human-like capabilities such as learning and adaptation, which can make use of the acquired knowledge and previously gained experience to understand and interpret behaviors or phenomena, and eventually help humans to solve complex problems. Frequently only rudimentary rule-based decision making expert systems are implemented. All rules are well defined instead of being the result of an opaque learning process (Narasimhan 2007; Calvo et al. 2009).

Specific Subsystems

Orbit Analysis and Correction

The synchrotron radiation source point and pointing stability are essential quality features limiting achievable effective brilliance, imaging contrast, and signal-to-noise ratio. The beam position needs to be measured with high precision and speed. It has to provide the basis for long-term reproducibility, slow drift control, and fast feedback compensation. Turn-by-turn measurements support beam threading, bunch-by-bunch measurements allow for fill pattern specific accelerator physics reasoning. Custom data highways have to deliver the position data to a central or distributed feedback processor unit calculating corrective actions that have to be distributed to all beam-steering elements involved. Valuable data for post-mortem analysis and process tuning (see section “[Process Tuning, Feedback Systems](#)”) become available (Abbott et al. 2011). The orbit correction is a proficiency by itself, covered by a separate article in this book (Boege and Aiba 2015).

Bunch-by-Bunch Instability Control, Diagnostics

Multibunch instabilities generated in storage rings by high current fillings and their wake fields require damping mechanisms. Bunch-by-bunch feedback (BBFB) systems are set up with bunch position monitors (transversal or longitudinal), correction kickers, and fast analysis and correction controllers. Early systems featured analog feedback controllers; today fast and flexible digital systems are prevalent – for details see the appropriate chapter in this book (Teytelman 2015).

Well embedded into the control system, BBFB processors add novel diagnostic capabilities (bunch specific characteristics, post-mortem data) (Schaelicke et al. 2013) as well as bunch-manipulation options (resonant bunch excitation for pulse picking (Holldack et al. 2014), bunch clearing, and individual bunch current scraping).

Timing and Synchronisation

Issues connected with timing and synchronization differ for control system and data acquisition at light sources. Relevant for controls are integration aspects, and compute node, action and response synchronization; thus, network time protocol (NTP), standard protocols for timing systems (IEEE 1588, 802.1AS) and reference clocks (radio clock, GPS).

Data acquisition needs long-term stability and high precision, precision stability, and limited jitter ranging from femtoseconds to several seconds, possibly, in the wide area up to kilometers range; so do feedback units. Transmission technologies like cable, optical or wireless technology, sensitivity to hardware effects (EMC, radiation effects, transmission delays, temperature, and pressure effects), and available delay compensation techniques need to match.

Generated from a GPS receiver with an embedded high-quality Oven Controlled Crystal Oscillator (OCXO) a time reference with 1-ns accuracy can be guaranteed. This clock can be distributed using the Micro Research Finland (MRF) event generator (EVG) event receiver (EVR) timing system to the rest of the facility with a very high accuracy (25 ps jitter RMS) (Fernandez-Carreiras et al. 2013).

White Rabbit, developed at CERN, is a new timing solution, in operation at GSI and tested elsewhere (DESY). White Rabbit is a fully deterministic Ethernet-based network for general purpose data transfer and synchronization. It can synchronize over 1000 nodes with sub-ns accuracy over fiber lengths of up to 10 km (White Rabbit).

FEL routine operation with the need for global feedback specifically requires synchronized FEL instrumentation, experiment equipment, and data acquisition.

Low-Level RF Control

Regulating the cavity field to a high precision (LLRF) is relevant for controlling the impedance of circular machines, linac-based coherent X-ray light sources (Geng et al. 2014), photocathode drive lasers and end-station lasers, and deflecting cavities.

There is a commercial solution, provided by Instrumentation Technology, a digital RF stabilization system (Liberia LLRF, Digital RF stabilization system). LLRF control implies specific difficulties in real-time processing of thousands of RF signals with very challenging RF field detection precision, RF generation, RF distribution, reliable, maintainable RF cabling, low-noise analog power supply, ultra-high precision analog electronics for detection, powerful digital

processing units and high-speed real-time processing, and distribution of high-performance Local Oscillator (LO), RF, and low-jitter clock signals.

LLRF evolves at the frontiers of engineering and science; thus experts in the field need a packed four-day workshop (LLRF13 [2013](#)) to discuss open questions on specific needs and techniques involved in in-situ automated cavity testing/commissioning, integration with machine and system protection, integration with instrumentation and fast orbit feedback, behavior of and interfacing to high-efficiency class-F amplifiers, fast tuner control (e.g., piezoelectrics and microphonics), SoC in FPGA with and without embedded (hard or soft core) processor(s), new hardware, modularity paradigms, mechanical–thermal design ideas, and collaboration on Open Systems (hardware, gateway, and software).

Configurations, Databases

Accelerators being very complex, unique and constantly evolving arrangements of sophisticated technology have to be described in various reference databases. Starting from conceptional and technical design reports the necessary equipment has to be specified and put together. Even in the early process of placing a call for tenders, purchasing, and acceptance testing it is helpful to have a mandatory and well-worked out *naming convention* in place.

Second, the proper conversion factors, which map the physical units (magnetic field, duration etc.) of the simulation and design into the engineering and technical correspondents (currents, triggers, limits etc.) are needed. They connect the design with the implementation details and have to feed model-based high-level tools XAL (XML), MML (see section “[Base Operational Tools](#)”)

Then all command/control data flow configurations within the control system’s infrastructure have to be set up and maintained, either for handling of device multiplicity, or script-based generation of alarm, archiving channels, and synoptic views. Sensors, modules, instruments, cables, controllers, software, process variables, and ancillary application programs all have to work together in unison. A change to any element typically requires a coordinated change in other elements (e.g., adding an input module also requires adding field wiring and databases). Likewise, the failure of an element will cause other related elements to not operate properly.

Due to the importance of functional integrity and trust in the control system, not only the importance of prescriptive (what is to be installed) databases, but also of the descriptive approach (what is currently installed) is well understood. With IRMIS (Dohan [2007](#)) an automated synchronization of the installation status of process variables and their connections to applications and to devices through ports and cables with a database designed to track connectivity and changes has been attempted, addressing questions like: Where does each device get its power from *power hierarchy*? What does each device physically plug into *housing hierarchy*? How does this device contribute to the passing of information through the Control System *Control hierarchy*? A common dream is to unify the prescriptive databases used for installation with the descriptive findings of the true inventory.

At every facility it is common to have numerous auxiliary databases, even GIS-based (Ishizawa and Yamashita [2009](#); Nemesure and Fu [2013](#)), that allow one to organize reliability (spares, failures, repairs, exchange, re-use, etc.), capture measurements (beam-based or survey alignment data), etc.

In essence the minimal reference repository database of control system configurations rests on three pillars and relies on the cooperation of three groups: it needs to describe geometries (posi-

tions, lengths, sizes) typically generated in the drawing department and used by the installation and alignment team. It needs to provide all proper calibrations of volts, ampères, count rates to position, pressure, and losses, usually overseen by the theory and simulations group. Finally, it needs to provide the data for the real time databases handling the distributed digital to analog or command conversions of the I/O to and from the devices attached; this helps to ensure that control system users perform the intended actions.

The *naming convention* specifically can be seen as a mini-database, at first pointing to the device, its usage within a functional domain of the facility, the installation location (Nemesure and Fu 2013), the operational function, the engineering and maintenance group in charge, and an identifier enumerating equal units.

With respect to the remote control capabilities of the control system it is advantageous to align the naming convention with the framework involved. In the example of the European XFEL, where a few frameworks coexist, naming conventions involved translate as follows (Rehlich 2013):

DOOCS: facility/device/location/property
TINE: context/server/device/property
Tango: domain/family/member/property
EPICS: flat (name server required)
Karabo: device(= facility/device/location)/property

In the case of EPICS the flat process variable name space is only structured by the convention to separate the device name from the signal part by a commonly accepted colon, i.e., all operational tools within the EPICS context work with *channels* named like DEVICE:signal. At BESSY it has been realized (Birke et al. 2005) that it is extremely helpful to agree not only on the naming of the devices, but also on the signal part of the device control points: whether it is an analog set value or read back, a status command or information (ON/OFF, ERROR, RESET, etc.). The corresponding mini-database points to software or configuration atoms of the underlying framework (record support: EPICS, device class: Tango).

At BESSY a human or programmed parser analyzes DEVICE: signal according to semantic content of the DEVICE and the signal. As usual DEVICES are categorized like member/family/counter/subdomain/domain/facility (Mueller et al. 2011; Birke et al. 2005), signals at BESSY have a first part, the signal type, and a second part, the signal item. The signal type describes the signal in terms of the control system features (e.g., set point, switch, status, ramp, raw), the signal item gives a hint to the physical (or physicist's) meaning of the value (e.g., access mode, external input, mode selection, door status).

Safety, Security

For function safety, and the application in the specification, design, and life cycle at any facility, international standards apply (IEC 61508, IEC 51161). At the light source, safety is guaranteed by *protection systems*, ideally implemented in safety instrumented systems (SIS, IEC 61511, IEC 62061). Security provisions, precautions to keep the control system in a safe state and free from danger, are typically mapped to *access controls*.

Protection Systems

Machine protection systems have to be well embedded into the control system. These systems have to prevent damage with serious consequences for operation or investment, which might come from random hardware failures, systematic hardware failures, software errors, common-cause failures, incorrect specifications, omitted safety requirements, and environmental influences.

In addition inviolacy and health of employees, users, guests, and visitors have to be unconditionally guaranteed by *personal safety interlocks*. These systems have to be wired and programmed to take all appropriate measures to turn the accelerator into a safe state, insert beam dumps, prevent injection, and switch off and disable power transmitters.

Machine Protection Systems

Numerous interlock switches simply protect devices or sectors from failures related to their function: missing magnet or absorber cooling, RF power-line excess (power off), klystron arc detection (crow bar), superconductivity quench detection (cold bypass diode), and vacuum air in-rush (fast closing valve flapper). If set up with digital systems, the reaction time and decision delays matter. Closer connected to the control system are malfunctions of operational states, e.g., beam mis-steering, due to non-responsive corrector magnets, could stop the fast orbit feedback, based on software reasoning on inadequate amplitudes or responses, or it could dump the beam, based on excessive heat load detection, or local BPM reading. In any case, false positives or over-cautious triggers can have serious adverse effects on operation reliability. On the other hand, light sources are complex and unique, so true limits cannot be tested; only if damage occurs it is known that a limit was insufficient or the reaction inappropriate. Also in view of lessons learned, the process data archiver (see section “[Process Data Archiver](#)”) is of outstanding importance.

Personal Safety Interlocks

Radiation-protection systems (PSI) are mandatory both for accelerator sections and for beam-line surroundings and experimental hutches. Installed hardware, and programmed logic and intervention to the accelerator and hutches (beam inhibit, beam shutter closure, hutch door locks) have to comply with the legal requirements of the local authorities and is typically linked to the approval of operation. IEC 61511 is the relevant technical standard to ensure PSI safety, in Europe the 2006/42/EC Directive of the European Parliament concerning machinery and common safety level of machinery applies to the selection of components. Control system read-out of PSI status and details is essential to allow for the tracking of inhibit conditions and act for determined clearance.

Less common are automated systems protecting from hazards like electrical shock, oxygen depletion, etc. Here stand-alone systems, not connected to the control system or strict procedures for tunnel access, emergency OFF, and alarm buttons are common.

Access Control

Like in any IT infrastructure even physical access is a matter of concern by itself: thumb drive insertion into Windows XP-based oscilloscopes, replugging network cables, manipulations at

privileged service consoles within the accelerator service areas – there are countless possibilities to jeopardize otherwise best care and attention.

Role Based Access

Write-access restrictions are a common measure to prevent control system usage from user mistakes or programming errors. Usually it is sufficient to distinguish who (supervisor, operator, floor manager, experimental user, expert, developer) tries to do what (atomic control action, initiate sequences, start/stop processes), where (injector, storage ring, undulator, beam line, experiment) and from where (control room, hutch, office, home). For instance, an operator should not be able to change a monochromator setting, a user should not control the accelerator or the experiment at another beam line, all this can be relatively easy mapped into front-end access control lists (ACL) and proper gateway configurations. Usually excessive ACL fine graining is not necessary, based on trust in responsibility and expertise; operators in the control room normally can access the whole accelerator complex with only few exceptions.

Cyber Security

Malware- and intrusion-related security is more of a systems administration problem common to any IT infrastructure (see sections “[Control System, the Domain-Specific IT Infrastructure](#)” and “[Networking: Configuration, Topology, Connectivity](#)”). Nevertheless, control systems have a specific focus on security, vulnerabilities, and protective measures of front-end devices, VxWorks, PLCs, power supplies, and networked control hardware. Moreover, network security, network segregation, firewalling, SCADA security, authentication, authorization, remote and secure multi-user operation (*kiosk mode*; at the light source users change quite frequently), and expert interventions require tailored approaches. Due to the control system’s specific resilience requirements and the complexity of the topic, the 5th Workshop on Control System Cyber Security (2015) (CS2) in 2015 will again provide a well-attended forum to exchange ideas and best practices.

User Experiment Control

Within control systems *user experiment control* is a *domain-specific control system*. Facility user experiment control systems must interact with a variety of instrument hardware, sample environment equipment, detectors, and data acquisition electronics. These control systems must be specifically flexible and easy to use even within a heterogeneous, ever changing user community and the necessary quick turnaround. This is an emerging field of specific expertise.

A specific infrastructure has to be set up that is tailored to both flexibility and specialization of unique but varying experimental installations. It has to cover general-purpose computing equipment (like in section “[Standard Model: Components, Hardware Inventory](#)”), which can be backed by large-scale computing and storage resources (Rees et al. 2008). It is very well suited to utilize configuration and deployment advantages of virtual environments ([Spring-8 Beamline Control](#); Matias et al. 2011; Mauch et al. 2014). Easy data exchange, maintainability, and functional extendibility require adoption or accessibility of the accelerator software frameworks (see section “[Frameworks: Toolkits, Software Base of Controls](#)”).

In summary, user experiment control has to cover:

- Experiment automation, scanning, sequencing and run control (see sections “[Software Support, Work-Flow Management](#)” and “[Motor Control, Wavelength Scan](#)”)
- Sample environment control, including robotic sample changers
- User interfaces and remote monitoring
- Live feedback and online data reduction and visualization
- Detector and data acquisition interfaces
- Data formats and meta-data systems, long-term storage
- Data management, volume, and creation speed (see section “[Data Management](#)”)
- User information systems and databases

The variety of instruments to control is enormous: mirrors, lenses, slits, attenuators, intensity and position monitors, spectrometers, pump laser, screens, cameras, 2D detectors, pumps, gauges, etc.

Software Support, Work-Flow Management

Overbooking and shortage of beam-time results in the quest to prepare, utilize, and evaluate beam time most efficiently. This broad scope contains elements of scale.

First of all it is helpful to have tools in place that users are familiar with and that are well connected to the controls environment. Among those there would be (LabVIEW System Design Software [2015](#)) to allow for rapid adjustments and integration of their own brought-in devices, SPEC software, very common for instrument control and data acquisition in X-ray diffraction experiments, widely used data analysis tools, be it public domain (root, pythonxy, matplotlib, itk, NumPy, SciPy, TomoPy) or commercial (IDL, matlab, igor).

Then there are the tailored beam-line adapted provisions of sequencing, data analysis and visualization, embedded and using the underlying control system framework. This framework is not necessarily the framework prevalent at the accelerators (see section “[Frameworks: Toolkits, Software Base of Controls](#)”). Dependent on the beam-line environment or the origin of the experimental set-up, it may be using the other framework (e.g., Tango at Diamond/EPICS and Petra III/Tine). Integrated into and extending the control system frameworks, popular, powerful, and portable toolkits exist: synApps (EPICS [2015](#)), BLISS (Bakaikoa [2007](#); TANGO [2015](#)), [GDA](#), Dawn science ([2014](#)), Sardana ([2015](#)).

More recently the need to proceed from batch processing and sequencing toward full data work-flow support, including dry-run capability, automated data taking, and evaluation and storage has been understood. Ideally this would cover the full cycle from proposal, approval, scheduling via experiment, data cleansing, data analysis to publication record. Activities here are both small, Passerelle (Abeille et al. [2007](#))-based approaches of two laboratories (Soleil, ESRF), and broad cooperation and standardization attempts like PANdata ([2011](#)) – the Photon and Neutron data infrastructure initiative – bringing together thirteen major world class European research infrastructures to create a fully integrated, pan-European, information infrastructure supporting the scientific process (Fig. 9).

Elements of PanData are a common identity system, a federated data infrastructure for data management, analysis, discovery, and provenance, based on NeXuS/ HDF5 and the data catalog system ICAT. ICAT is open source meta-data management system designed for large facilities.

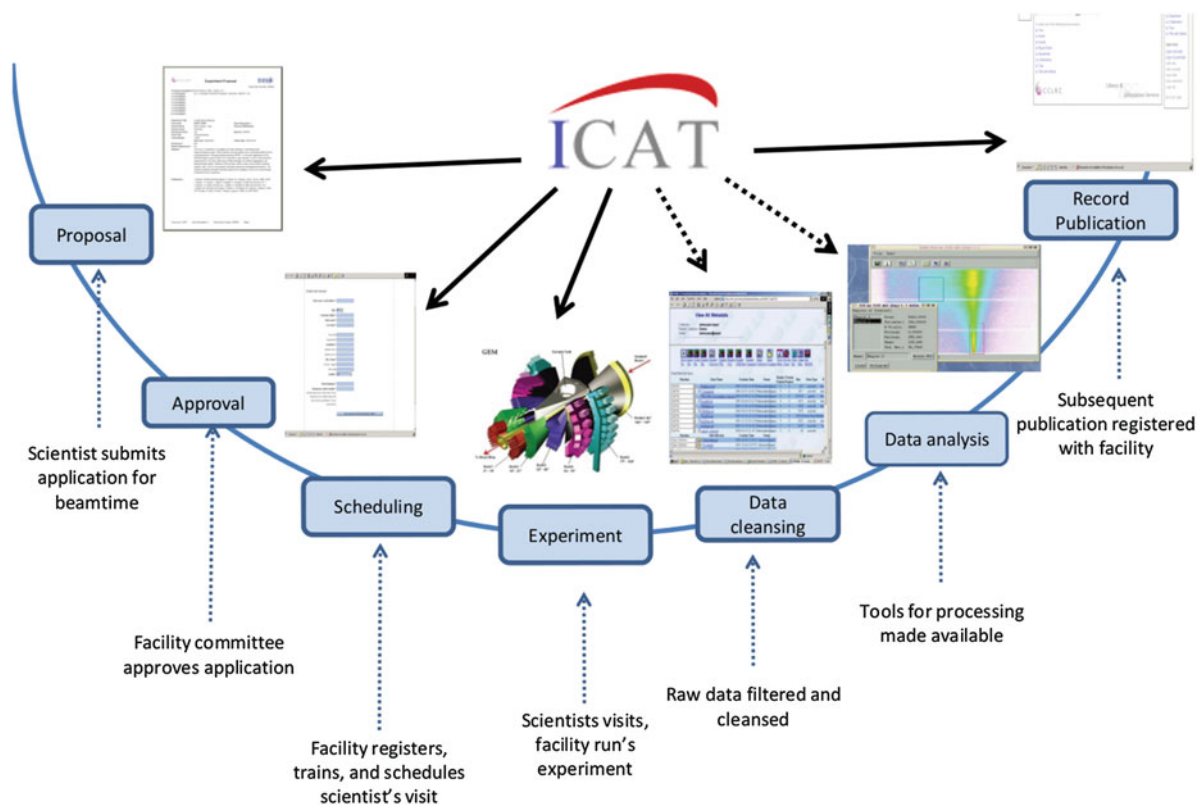


Fig. 9 Open data infrastructure for research: ICAT meta-data covering complete life cycles of research at light sources (Bicarregui 2013)

NeXus is developed as an international standard by scientists and programmers representing major scientific facilities in Europe, Asia, Australia, and North America in order to facilitate greater cooperation in the analysis and visualization of neutron, X-ray, and muon data. NeXus itself builds, on top of HDF5, data analysis tools like Mantid (Arnold et al. 2014) or Dawn science (2014), which support the standard data format and interface directly with ICAT.

Motor Control, Wavelength Scan

Motion control at light source facilities is an art by itself, both at the accelerator (cavity plunger, fluorescent screen, scraper, collimator, undulator gap and shift, laser mirrors for slicing, photo injector) and at the beam lines (slits, mirrors, gratings, sample positions, and angles). On the sample environment side, robotic support is requested by the increasing automation demands of high-throughput beam lines (ambient pressure, powder diffraction experiments).

Of course, the common design considerations, like system performance validation including reliability, behavior on radiation damage to encoders, in-house development versus industrial products, obsolescence management, etc. apply. Then the available technical solutions, kinematic transforms, complex trajectories, protection including collision avoidance, multi-axes and multi-controller synchronization, embedded and/or hosted motion features in software and in low-level hardware, make the difference. Experts in the field regularly exchange their experiences and define their path of standardization (MOCRAF 2013).

On the fly scans, i.e., wavelength scans with synchronized undulator gap, ideally coordinated polarization shift, monochromator mirror and grid control, and triggered data detector's acquisition, mean a true enhancement for light-source users with respect to signal-to-noise ratio, sample radiation and heat protection, and experimental throughput (e.g., Elettra (BACH 2015), Soleil (Joly et al. 2014), ALBA (Reszela et al. 2013; Fernandez-Carreiras et al. 2013)). Motion control challenges here are submicron positioning, continuous and synchronous motion on complex parametric trajectories, vibration avoidance, and feedback technologies.

Data Management

The amount and speed of data generated at the extreme brightness light sources by the fast and large hybrid pixel array detectors (Brönnimann 2015; Graafsma 2015) put a new quality demand on data processing due to the huge data rate, big data volumes, online analysis close to real time and long-term storage requirements.

Numbers from SACLA (Sugimoto et al. 2013; Okada et al. 2014), where the analysis procedure must be faster than the DAQ throughput of 5 Gbps, are simply impressive: instant visualization, real-space imaging from online reduced analyses, is used to determine the feasibility of carrying out the experiment and whether the experimental conditions need to be changed. Such instant visualization is performed in an interactive mode on a HPC system, consisting of an 80-node cluster system of 12.7 TFLOPS CPUs, and a single-node symmetric multiprocessing (SMP) system at 289 GFLOPS with 1 TB memory. First low-level filtering is implemented with a grid-based region-of-interest (ROI) statistical analysis, working on the cache storage systems (capacities of 200 TB and 250 TB). The archive of experimental data ranging over more than 1 year is a hybrid storage system of a 1 PB disk array and an automated tape-library system of 6 PB capacity in the SACLA computer room. Real-space 3D image reconstruction of large biomaterial would require two weeks on the HPC system. Therefore an external system, the K supercomputer with 10 petaFLOPS compute power, located at Kobe, about 100 km from SACLA, is used.

It is understandable that data streaming of all and reduced data downstream, including re-use of storage pipeline cells occupied by poor pulse data for reduction of data transfer and save amounts, is within the visions of the control and scientific computing environment Karabo (Heisen et al. 2013), designed and built from scratch at the XFEL.

Novel Approaches

Open Hardware

The Open Hardware Repository (ohwr) is a place on the web for electronics designers to collaborate on open hardware designs, much in the philosophy of the free software movement.

There are 52 hardware projects, 39 Hardware Description Language (HDL) or Software projects on ohwr. Eleven institutes lead the different projects.³ Sixteen commercial companies are involved, who develop, produce and/or test open hardware products. Open hardware projects are e.g., a sub-

³CERN, Soleil, GSI, Brazilian Light Source, University of Bristol, Warsaw University of Technology, University of Zurich, University of Pavia, Rockefeller University, University of Cape Town, Heidelberg University

References

- M.G. Abbott, G. Rehm, I.S. Uzun, A new fast data logger and viewer at diamond: the fa archiver, in *Proceedings of ICALEPCS*, Grenoble, France, 2011
- G. Abeille, M. Ounsy, A. Buteau, A graphical sequencer for soleil beamline acquisitions, in *Proceedings of ICALEPCS*, Knoxville, 2007, <http://accelconf.web.cern.ch/AccelConf/ica07/PAPERS/RPPB20.PDF>
- Arduino, an open-source electronics platform based on easy-to-use hardware and software, 2015, <http://www.arduino.cc/>
- O. Arnold et al., Mantid – data analysis and visualization package for neutron scattering and μ SR experiments, in *Nuclear Instruments and Methods in Physics Research Section A*, vol. 764, 2014, pp. 156–166, <http://www.mantidproject.org>
- ARW, Accelerator Reliability Workshop, Knoxville, 2015, <http://neutrons-old.ornl.gov/conf/arw2015/>
- BACH, the Beamline at ELETTRA, 2015, <http://www.elettra.trieste.it/elettra-beamlines/bach.html>
- I. Badillo, J. Jugo, M. Eguiraun, D. Piso, Wireless network integration into epics systems, in *Proceedings of IPAC*, New Orleans, 2012, <http://accelconf.web.cern.ch/AccelConf/IPAC2012/papers/thppr008.pdf>
- V.R. Bakaikoa, Experiments at the ESRF, in *Tango Workshop at DESY*, 2007, http://plone.tango-controls.org/Events/meetings/september-2007/tango_workshop_DESY_18Sept07.ppt
- R. Bakker, T. Birke, R. Mueller, Is comprehensive and intuitively usable commissioning software feasible? in *Proceedings of the PAC*, New York, 1999, <http://accelconf.web.cern.ch/AccelConf/p99/PAPERS/MOP31.PDF>
- D. Beltran, M. Gonzalez, A proposal for modeling the control system for the Spanish light source in UML, in *Proceedings of ICALEPCS*, Gyeongju, 2003, <http://accelconf.web.cern.ch/AccelConf/ica03/PAPERS/TU501.PDF>
- J. Bicarregui, A brief talk on PaNdata and RDA, in *PaNdata ODI Presentations*, 2013, <https://indico.cern.ch/getFile.py/access?contribId=32&sessionId=4&resId=1&materialId=slides&confId=246453>
- T. Birke, Automated operation of the metrology light source storage ring, in *Proceedings of ICALEPCS*, Kobe, 2009, http://accelconf.web.cern.ch/AccelConf/icalepcs2009/talks/tua002_talk.pdf
- T. Birke, B. Franksen, B. Kuner, R. Lange, P. Laux, R. Mueller, G. Pfeiffer, J. Rahn, Beyond devices – an improved RDB data-model for configuration management, in *Proceedings of ICALEPCS*, Geneva, 2005, http://accelconf.web.cern.ch/AccelConf/ica05/proceedings/pdf/P1_078.pdf
- Blade Server, 2015, http://en.wikipedia.org/wiki/Blade_server
- J. Bobnar, I. Kriznar, P.K. Bartkiewicz, P. Duval, Honggong Wu, The ACOP family of beans: a framework independent approach, in *ICALEPCS*, Knoxville, 2007, <https://accelconf.web.cern.ch/accelconf/ica07/PAPERS/TPPA23.PDF>
- M. Boege, M. Aiba, Orbit trajectory control and feedback, in *Synchrotron Light Sources and Free-Electron Lasers*, ed. by E. Jaeschke, S. Khan, vol. 7 (Springer International Publishing, Switzerland, 2015)
- M. Böge, J. Chrin, Integrating control systems to beam dynamics applications with corba, in *Proceedings of the Particle Accelerator Conference*, Portland, USA, 2003, <http://accelconf.web.cern.ch/AccelConf/p03/PAPERS/TOPB010.PDF>

- M. Böge, J. Chrin, Developments to the SLS corba framework for high level software applications, in *Proceedings of the ICALEPCS*, Geneva, 2005
- M. Borland, A universal postprocessing toolkit for accelerator simulation and data analysis, in *Proceedings of the ICAP*, Monterey, 1998, <http://www.slac.stanford.edu/xorg/icap98/papers/F-Tu10.pdf>
- C. Brönnimann, Hybrid pixel photon counting X-ray detectors for synchrotron radiation, in *Synchrotron Light Sources and Free-Electron Lasers*, ed. by E. Jaeschke, S. Khan, vol. 1 (Springer International Publishing, Switzerland, 2015)
- I.V. Calvo, R. Barillere, G. Thomas, The virtual unicos process expert: integration of artificial intelligence tools in control systems, in *Proceedings of ICALEPCS*, Kobe, 2009, <http://accelconf.web.cern.ch/AccelConf/icaleps2009/papers/tup033.pdf>
- M. Cattin et al., CERNs FMC KIT, in *Proceedings of ICALEPCS*, San Francisco, 2013, <http://accelconf.web.cern.ch/AccelConf/ICALPCS2013/papers/wecocb01.pdf>
- J. Chen, G. Heyes, W. Akers, D. Wu, W. Watson, CDEV: an object-oriented class library for developing device control applications, in *Proceedings of ICALEPCS*, Chicago, 1995
- P.N. Clout, V.E. Martz, R.B. Rothrock, R.T. Westervelt, A comparison of Vsystem and EPICS, in *Proceedings of the ICALEPCS*, Beijing, 1997, <http://www.aps.anl.gov/News/Conferences/1997/icaleps/paper97/p118.pdf>
- Control System Cyber Security (CS2) Workshop, 2015, <https://indico.cern.ch/event/348695/>
- Cosylab, 2001–2015, www.cosylab.com/solutions/particle_accelerators
- CSS Control System Studio, 2013, <http://controlsystemstudio.org/>
- L. Dalesio et al., Distributed information services for control systems, in *Proceedings of ICALEPCS*, San Francisco, 2013, <http://accelconf.web.cern.ch/AccelConf/ICALPCS2013/papers/wecoba02.pdf>
- DAWN Science, (2014), <http://confluence.diamond.ac.uk/display/DAWN>
- D.A. Dohan, The IRMIS universal component-type model, in *Proceedings of ICALEPCS2007, TOPA03*, 2007, <http://www.aps.anl.gov/epics/irmis/>
- DOOCS, the Distributed Object Oriented Control System (DOOCS), 2011, <http://doocs.desy.de/>
- DSL, Domain Specific Languages, M. Voelter Collection of Talks, 2005–2014, <http://www.voelter.de/conferences/>
- P. Duval, Z. Kakucs, D. Golob, M. Kadunc, I. Kriznar, M. Plesko, A. Pucelj, G. Tkacik, *The Babylonization of Control Systems Part II – The Rise of the Fallen Tower*, Gyeongju, 2003, <http://epaper.kek.jp/ica03/PAPERS/TH513.PDF>
- ELOG, The ELOG Home Page, 2002, <https://midas.psi.ch/elog/>
- D.B. Engel, R. Mueller, P. Laux, P. Stange, Virtualization infrastructure within the controls environment of the light sources at HZB, in *Proceedings of ICALEPCS*, 2013, <http://accelconf.web.cern.ch/AccelConf/ICALPCS2013/papers/thppc005.pdf>
- EPICS, Experimental Physics and Industrial Control System (2015), <http://www.aps.anl.gov/epics/>
- EPICS Collaboration Meetings, (2015), <http://www.aps.anl.gov/epics/meetings.php>
- EtherCAT is the Open Real-Time Ethernet Network*, <http://www.ethercat.org/default.htm>
- D.F.C. Fernandez-Carreiras, F. Becheri, G. Cuni, R. Homs-Puron, G. Jover-Manas, J. Klorá, O. Matilla, J. Moldes, C. Pascual-Izarra, Z. Reszela, D. Roldan, S. Rubio-Manrique, X. Serra-Gallifa, Synchronization of motion and detectors and continuous scans as the standard data acquisition technique, in *Proceedings of ICALEPCS*, San Francisco, 2013, <http://accelconf.web.cern.ch/AccelConf/ICALPCS2013/papers/wecoab03.pdf>
- GDA – Software for Science, www.opengda.org/OpenGDA.html

- Z. Geng et al., Architecture design for the SwissFEL LLRF system, in *Proceedings of LINAC*, Geneva, 2014
- M. Giacchini et al., Hyperarchiver: an epics archiver prototype based on hypertable, in *Proceedings of ICALEPCS*, Grenoble, 2011, <http://accelconf.web.cern.ch/AccelConf/icalepcs2011/papers/mopkn012.pdf>
- A. Goetz et al., Tango V8 – another turbo charged major release, in *Proceedings of ICALEPCS*, San Francisco, 2013, <http://accelconf.web.cern.ch/AccelConf/ICALEPCS2013/papers/tucocb10.pdf>
- H. Graafsma, Integrating hybrid area detectors for storage ring and free-electron laser applications, in *Synchrotron Light Sources and Free-Electron Lasers*, ed. by E. Jaeschke, S. Khan, vol. 1 (Springer International Publishing, Switzerland, 2015)
- Hadoop, The Apache™ Hadoop® project develops open-source software for reliable, scalable, distributed computing (2014), <https://hadoop.apache.org/>
- V. Hardion, D.P Spruce, A.M. Otero, J. Lidon-Simon, M. Lindberg, A. Persson, The internet of things and control system, in *ICALEPCS*, San Francisco, 2013, <http://accelconf.web.cern.ch/AccelConf/ICALEPCS2013/papers/tucocb09.pdf>
- B.C. Heisen, D. Boukhelef, S. Esenov, S. Hauf, I. Kozlova, L. Maia, A. Parenti, J. Szuba, K. Weger, K. Wrona, C. Youngman, Karabo: an integrated software framework combining control, data management, and scientific computing tasks, in *Proceedings of ICALEPCS*, San Francisco, 2013
- K. Holldack et al., Single bunch X-ray pulses on demand from a multibunch synchrotron radiation source, in *Nature Communications*, vol. 5 (2014), <http://www.nature.com/ncomms/2014/140530/ncomms5010/full/ncomms5010.html>
- B. Hollifield, E. Habibi, *Alarm Management: Seven Effective Methods for Optimum Performance*, 2007. ISBN-13:978-1934394007, ISBN-10:1934394009
- ICALEPCS, International conference on accelerator and large experimental physics control systems, 2015, <http://www.icalepcs.org>
- ICAP, *International Computational Accelerator Physics Conference*, Rostock-Warnemünde, 2012, <http://accelconf.web.cern.ch/AccelConf/ICAP2012/index.htm>
- ICFA beam dynamics newsletter, No. 47, theme section: control systems, Oct 2008, http://icfa-usa.jlab.org/archive/newsletter/icfa_bd_nl_47.pdf
- IEEE Real Time Conference(RT), 2014, <http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=7086910>
- INSPIRE, the High Energy Physics information system. Search entry ICALEPCS, PCaPAC, DIPAC, 2015, <http://inspirehep.net/collection/Conferences>
- IoT, *Internet of Things*, 2015, https://en.wikipedia.org/wiki/Internet_of_Things
- IoT EU, *Internet of Things*, 2010, <http://www.internet-of-things.eu/>
- Y. Ishizawa, A. Yamashita, Life cycle management system for accelerator equipment, in *Proceedings of ICALEPCS*, Kobe, 2009, <http://accelconf.web.cern.ch/AccelConf/icalepcs2009/papers/wep020.pdf>
- I-Tech, Instrument Technologies, www.itech.si
- JACoW, Joint Accelerator Conferences Website, CERN, KEK (2015), <http://www.jacow.org/>
- A. Jankowiak, J. Knobloch, P. Goslawski, N. Neumann (eds.), *Technical design study BESSY VSR – variable pulse length storage ring upgrade of BESSY II*. Helmholtz-Zentrum Berlin für Materialien und Energie GmbH, 2015. doi:10.54442/R0001, <http://dx.doi.org/10.5442/R0001>
- M. Janousch, Overview of the SwissFEL project at PSI, in *Proceedings of the EPICS Collaboration Meeting, Spring 2015*. Michigan State University, 2015, <https://indico.fnal.gov/getFile.py?access?contribId=48&sessionId=0&resId=0&materialId=slides&confId=9718>

- L. Joly, E. Otero, F. Choueikani, F. Marteau, L. Chapuisb, P. Ohresser, Fast continuous energy scan with dynamic coupling of the monochromator and undulator at the DEIMOS beamline. *J. Synchrotron Radiat.* **21**, 502 (2014). doi:10.1107/S1600577514003671, <http://journals.iucr.org/s/issues/2014/03/00/ve5024/ve5024.pdf>
- R. Kammering, Ch. Schmidt, Feedbacks and automation at the free electron laser in Hamburg (FLASH), in *Proceedings of the ICALEPCS*, 2013, <http://accelconf.web.cern.ch/AccelConf/ICALEPCS2013/papers/thppc121.pdf>
- R. Kapeller, R. Krempaska, C. Higgs, H. Lutz, From real to virtual – how to provide a high-availability computer server infrastructure, in *Proceedings of ICALEPCS*, 2013, <http://accelconf.web.cern.ch/AccelConf/ICALEPCS2013/papers/thmib03.pdf>
- K. Kasemir, X. Chen, Bringing control system user interfaces to the web, in *Proceedings of ICALEPCS*, San Francisco, 2013, <http://accelconf.web.cern.ch/AccelConf/ICALEPCS2013/papers/thcoaab03.pdf>
- K. Kasemir, X. Chen, E. Danilova, The best ever alarm system toolkit, in *Proceedings of ICALEPCS*, Kobe, 2009, <https://accelconf.web.cern.ch/accelconf/icalepcs2009/papers/tua001.pdf>
- T. Korhonen et al., EPICS version 4 progress report, in *Proceedings of ICALEPCS*, San Francisco, 2013, <http://accelconf.web.cern.ch/AccelConf/ICALEPCS2013/papers/tucocb04.pdf>
- B. Kuiper, Issues in accelerator controls, in *Proceedings of the ICALEPCS*, Tsukuba, 1991
- LabVIEW System Design Software, 2015, <http://www.ni.com/labview/>
- T. Larrieu, Design and implementation of an archiver-based save and restore system, in *To be Presented at the ICALEPCS*, Melbourne, 2015
- R. Larsen, Introduction to MicroTCA, SLAC MicroTCA Standards Review, Stanford, 2012, https://portal.slac.stanford.edu/sites/ad_public/events/mtca_llrf_jun_2012/Published_Documents/Introduction%20to%20mTCA.pdf
- M. Leech, Accelerator control-system network security at diamond light source, in *Proceedings of the Control System Cyber Security Workshop*, Knoxville, 2007, <http://indico.cern.ch/event/13367/contribution/14/material/slides/1.pdf>
- Libera LLRF, Digital RF stabilization system, http://www.i-tech.si/accelerators-instrumentation/libera-llrf/modular-building-blocks_10
- Light Sources of the World (2015), <http://www.lightsources.org/regions>
- LLRF13, The sixth low-level radio frequency workshop, Lake Tahoe, 2013, <http://llrf13.lbl.gov>
- MapReduce: Simplified Data Processing on Large Clusters (2011), <http://research.google.com/archive/mapreduce.html>
- S. Marsching, Scalable archiving with the cassandra archiver for css, in *Proceedings of ICALEPCS2013*, San Francisco, 2013, <http://accelconf.web.cern.ch/AccelConf/ICALEPCS2013/papers/tuppc004.pdf>
- E. Matias et al., Phase II and III the next generation of cls beamline control and data acquisition systems, in *Proceedings of ICALEPCS*, 2011, <http://accelconf.web.cern.ch/AccelConf/icalepcs2011/papers/mopmu013.pdf>
- V. Mauch et al., OpenGL-based data analysis in virtualized self-service environments, in *Proceedings of PCaPAC*, Karlsruhe, 2014, <http://accelconf.web.cern.ch/AccelConf/PCaPAC2014/papers/fco202.pdf>
- N.S. Mirian, Two-color free-electron laser via two orthogonal undulators, in *Proceedings of FEL*, Basel, 2014, <http://accelconf.web.cern.ch/AccelConf/FEL2014/papers/tup006.pdf>

- MOCRAF – Motion Control Applications in Large Facilities, 2013, <http://www.synchrotron-soleil.fr/Workshops/2013/motioncontrol>
- R. Mueller, D. Engel, P. Laux, BDNLS – Bessy device name location service, in *Proceedings of ICALEPCS*, 2011, <http://accelconf.web.cern.ch/AccelConf/icalepcs2011/papers/mopkn027.pdf>
- S. Narasimhan, Automated diagnosis of physical systems, in *Proceedings of ICALEPCS*, 2007, <http://accelconf.web.cern.ch/AccelConf/ica07/PAPERS/FOAA01.PDF>
- National Instruments, How LabVIEW Uses I/O Servers, 2012, <http://www.ni.com/white-paper/13865/en/>
- S. Nemesure, W. Fu, Accelerator control data visualization with google map, in *Proceedings of ICALEPCS*, San Francisco, 2013, <http://accelconf.web.cern.ch/AccelConf/ICALEPCS2013/papers/tuppc132.pdf>
- NoBugs, New Opportunities for Better User Group Software, 2012, <http://nobugs2012.org/NOBUGS/workshops.html>
- K. Okada, T. Abe, Y. Furukawa, T. Hatsui, Y. Joti, T.K. Kameshima, T. Matsumoto, T. Sugimoto, R. Tanaka, M. Yamaga, M. Yabashi, Upgrade of SACLA DAQ system adapts to multi-beamline operation, in *Proceedings of the PCaPAC*, Karlsruhe, 2014, <http://accelconf.web.cern.ch/AccelConf/PCaPAC2014/papers/wco205.pdf>
- PANdata – the Photon and Neutron Data Infrastructure Initiative, 2011, <http://pan-data.eu/presentations>
- Y.W. Parc et al., Simulation results of self seeding scheme in PAL-XFEL, in *Proceedings of FEL*, New York, 2013, <http://accelconf.web.cern.ch/AccelConf/FEL2013/papers/wepso47.pdf>
- PCaPAC, PCs and Particle Accelerator Controls, since 2008 in JACoW. First workshop Hamburg, Oct 1996, http://www-mpy.desy.de/pc_workshop/
- T. Pelaia II et al., XAL status, in *Proceedings of the ICALEPCS*, Knoxville, 2007, <http://accelconf.web.cern.ch/AccelConf/ica07/PAPERS/MOPB02.PDF>
- M. Plesko, Recent highlights from cosylab, in *Proceedings of PCa-PAC*, Karlsruhe, 2014, <http://accelconf.web.cern.ch/AccelConf/PCaPAC2014/papers/tco103.pdf>
- G. Portmann, J. Corbett, A. Terebilo, An accelerator control middle layer using matlab, in *Proceedings of Particle Accelerator Conference*, Knoxville, 2005, <http://accelconf.web.cern.ch/accelconf/p05/papers/fpat077.pdf>
- PVSS, now SIMATIC WinCC Open Architecture (2015), http://www.etm.at/index_e.asp
- N. Rees et al., Diamond computing, in *EPICS Collaboration Meeting*, Oct 2008, <http://agenda.infn.it/getFile.py/access?contribId=25&resId=&materialId=slides&confId=715>
- K. Rehlich, Control system interfaces, in *XFEL Collaboration Meeting*, 2013, <http://www.slideserve.com/tien/control-system-interfaces>
- Z. Reszela, F. Becheri, G. Cuni, D. Fernandez-Carreiras, J. Moldes, C. Pascual-Izarra, Implementation of continuous scans used in beamline experiments at alba synchrotron, in *Proceedings of ICALEPCS*, San Francisco, 2013, <http://accelconf.web.cern.ch/AccelConf/ICALEPCS2013/papers/tuppc060.pdf>
- S. Rubio-Manrique et al., Panic, a suite for visualization, logging and notification of incidents, in *Proceedings of PCaPAC*, Karlsruhe, 2014, <http://epaper.kek.jp/PCaPAC2014/papers/fco206.pdf>
- SAN, Disk array Storage area network, Network attached storage (NAS), Storage virtualization, 2015, http://en.wikipedia.org/wiki/Disk_array
- SARDANA, Sardana is a software suite for supervision, control and data acquisition in scientific installations. It aims to reduce cost and time of design, development and support of the control and data acquisition systems, 2015, <http://www.sardana-controls.org/en/latest/>

- C. Scafuri et al., Automation of the wavelength change for the fermi free electron laser, in *Proceedings of ICALEPCS*, San Francisco, 2013, <http://accelconf.web.cern.ch/AccelConf/ICALEPCS2013/papers/tuppc052.pdf>
- A. Schaelicke, F. Falkenstern, R. Mueller, Bunch-bybunch feedback and diagnostics at BESSY II, in *Proceedings of IBIC2013*, Oxford, 2013. <http://accelconf.web.cern.ch/AccelConf/IBIC2013/papers/tupc16.pdf>
- V. Schlott. Beam instrumentation and diagnostics, in *Synchrotron Light Sources and Free-Electron Lasers*, ed. by E. Jaeschke, S. Khan, vol. 2 (Springer International Publishing, Switzerland, 2015)
- G. Shen, Virtual accelerator at NSLS II project, in *Proceedings of ICALEPCS*, San Francisco, 2013, <http://accelconf.web.cern.ch/AccelConf/ICALEPCS2013/papers/moppc156.pdf>
- K. Shroff et al., Olog and control system studio: a rich logging environment, in *Proceedings of ICALEPCS*, San Francisco, 2013, <http://accelconf.web.cern.ch/AccelConf/ICALEPCS2013/papers/thcoaab09.pdf>
- N.J. Smale et al., The ANKA control system: on a path to the future, in *Proceedings of ICALEPCS*, San Francisco, 2013, <http://accelconf.web.cern.ch/AccelConf/ICALEPCS2013/papers/moppc099.pdf>
- Spring-8 Beamline Control, http://www.spring8.or.jp/en/about_us/manage_structure/jasri/control_system/upgrading/beamline_control/
- SPring-8-II conceptual design report, 2014, <http://rsc.riken.jp/eng/pdf/SPring-8-II.pdf>
- Structured Cabling, 2015, http://en.wikipedia.org/wiki/Structured_cabling
- T. Sugimoto et al., On-line and off-line data analysis system for sacra experiments, in *Proceedings of ICALEPCS*, San Francisco, 2013, <http://accelconf.web.cern.ch/AccelConf/ICALEPCS2013/papers/tuppc015.pdf>
- R.Tanaka et al., Message and database oriented control architecture (MADOCA), in *The First Operation of Control System at the SPring-8 Storage Ring. Proceedings of ICALEPCS*, Beijing, 1997
- TANGO control system, a free open source device-oriented controls toolkit, 2015, <http://www.tango-controls.org/>
- TANGO Collaboration Meetings, 2015, <http://www.tango-controls.org/community/events/>
- TAURUS, Taurus is a python framework for control and data acquisition CLIs and GUIs in scientific/industrial environments. It supports multiple control systems or data sources: Tango, EPICS, SPEC, 2015, <http://www.taurus-scada.org/en/stable/>
- D. Teytelman, Beam instabilities, feedback systems, in *Synchrotron Light Sources and Free-Electron Lasers*, ed. by E. Jaeschke, S. Khan, vol. 1 (Springer International Publishing, Switzerland, 2015)
- M.E. Thuot, L.R. Dalesio, Control system architecture: the standard and nonstandard models, in *Proceedings of the PAC*, 1993, http://accelconf.web.cern.ch/AccelConf/p93/PDF/PAC1993_1806.PDF
- M.E. Thuot, M. Clausen, L.R. Dalesio, T. Katoh, M.E. Kraimer, R. Mueller, H. Shoaee, W.A. Watson, The success and the future of EPICS, in *Proceedings of the LINAC*, 1996, <http://accelconf.web.cern.ch/AccelConf/I96/PAPERS/TH203.PDF>
- Timing, Workshop on Timing Modes for Low-emittance Storage Rings, MAX-lab, Lund, 2015, <https://indico.maxlab.lu.se/event/60/>
- TINE, Three-fold Integrated Networking Environment, <http://tine.desy.de/>
- Trac, An enhanced wiki and issue tracking system, 2003–2015, <http://trac.edgewall.org/>

- I.S. Uzun, *Diamond Light Source Fast Orbit Feedback Communication Controller*, 2007, http://controls.diamond.ac.uk/downloads/other/fofb_cc/doc/CTRL-FOFB-CC-0001.pdf
- I. Verstovsek, Abeans and CosyBeans, in *EPICS Collaboration Meeting*, 2002, <http://www.aps.anl.gov/epics/meetings/2002-11/talks/verstovsek.pdf>
- I. Verstovsek, M. Kadunc, J. Kamenik, I. Kriznar, G. Pajor, M. Plesko, A. Pucelj, M. Sekoranja, G. Tkacik, D. Vitas, ABEANS: application development framework for Java, in *Proceedings of ICALEPCS*, 2003, <https://accelconf.web.cern.ch/accelconf/ica03/PAPERS/WE203.PDF>
- I. Verstovsek, J. Kamenik, P. Kolaric, Management system tailored to research institutes, in *Proceedings of ICALEPCS*, Knoxville, 2007, <http://accelconf.web.cern.ch/AccelConf/ica07/PAPERS/RPPB15.PDF>
- Vista Control Systems, 2010, <https://www.vista-control.com/>
- M. Voelter, Trends in programming languages, in *Keynote, Proceedings of the ICALEPCS*, Grenoble, 2011, http://accelconf.web.cern.ch/AccelConf/icalepcs2011/talks/tuaaukp05_talk.pdf
- WAO, Workshop on Accelerator Operations, Mainz, 2014, <http://wao2014.desy.de/>
- WebPDA – Bringing Real-time Data to the Web, 2013–2014, <http://webpda.org/>
- S. Weisse, P. Duval, G. Trowitzsch, M. Lomperski, Status of a versatile video system at PITZ, DESY-2 and EMBL Hamburg, in *Proceedings of ICALEPCS*, Knoxville, 2007, <http://accelconf.web.cern.ch/AccelConf/ica07/PAPERS/WPPA31.PDF>
- White Rabbit, <http://www.ohwr.org/projects/white-rabbit>
- L. Yang, The design of NSLS-II high level physics applications, in *Proceedings of ICALEPCS*, San Francisco, 2013, <http://accelconf.web.cern.ch/AccelConf/ICALEPCS2013/papers/tuppc130.pdf>
- ZeroMQ, *Distributed Messaging*, 2007–2014, <http://zeromq.org/>