

Beamline Controls: Preliminary Design Report

Daron Chabot

11/09/2010

Contents

1	Overview	4
2	Coordinate Systems	4
3	Naming Convention	5
4	Electrical Cabling Convention	5
5	Sub-Systems	6
5.1	Input/Output Controller (IOC)	6
5.2	Standard Software Packages	6
5.3	Insertion Device	6
5.3.1	Integration with Monochromators	8
5.4	Front Ends	8
5.5	Vacuum	8
5.6	EPS/PPS	8
5.7	Motion	9
5.7.1	Generic Motion System	9
5.7.2	Calibration	11
5.7.3	Motor Controller	12
5.8	Optics	13
6	Data Acquisition	13
6.1	Detectors	14
6.2	Scanning	14
6.3	Scripting interface	14
6.4	Data acquisition applications	14
6.4.1	Generic Data Acquisition (GDA)	14
6.4.2	SPEC	15
6.4.3	BLISS	15
6.5	Remote access	15

7 Interfaces to Accelerator Systems	16
7.1 Network	16
7.2 Timing	16
7.3 EPS	16
7.4 PPS	17
7.5 Diagnostics	17
8 Beamline configuration database	17

1 Overview

With a capacity for up to approximately sixty beamlines at the NSLS-II, six project beamlines will be constructed during the initial phase of the project. Their identities, group leaders, and photon sources are:

1. IXS (Inelastic X-ray Scattering, Y. Cai) – IVU22 (baseline) or CPMU17 (future upgrade)
2. HXN (Hard X-ray Nanoprobe, Y. Chu) – IVU20
3. CHX (Coherent Hard X-ray, L. Wiegart) – IVU20
4. CSX (Coherent Soft X-ray, C. Sanchez-Hanke) – dual EPU45
5. XPD (X-ray Powder Diffraction, E. Dooryhee) – DW
6. SRX (Sub-micron Resolution X-ray Spectroscopy, J. Thiemes) – IVU21

Although the research conducted may be unique to each beamline, many mechanical and electrical components are common across most beamlines. Vacuum, optics, and motion components are generally required elements of any beamline. Due to the differing scientific regimes, beamlines frequently differ in their detector and data acquisition requirements.

Adoption of equipment standards across common beamline components is a practical necessity. This both reduces the number of devices requiring software development and reduces the number of different components for which spares must be maintained. Additionally, familiarity will be stronger with fewer devices, and this will aid configuration, maintenance, and trouble-shooting[9].

To this end, as much equipment as possible should be carried over from the accelerator systems for use on the beamlines. This includes components such as vacuum gauges and controllers, PLC equipment, motion systems, and control systems software.

The Experimental Physics and Industrial Control System (EPICS) software will be utilized for control of the NSLS-II accelerator, storage ring, and beamlines. The client/server architecture of EPICS permits access and control of instrumentation over standard TCP/IP networking protocols, and support exists within the EPICS community for a broad variety of devices.

The scope of coverage for beamline control systems extends from the photon source (insertion device (ID) or dipole wiggler (DW)), through the Front-End and optical components, up to and including the experimental endstation.

While specific requirements will be detailed separately for each of the initial six, project beamlines, the guidelines and recommendations detailed here will be generally applicable across all beamlines.

2 Coordinate Systems

A coordinate system definition shared across beamlines provides a common basis of reference and may enable software sharing as well, for those components with geometric dependencies.

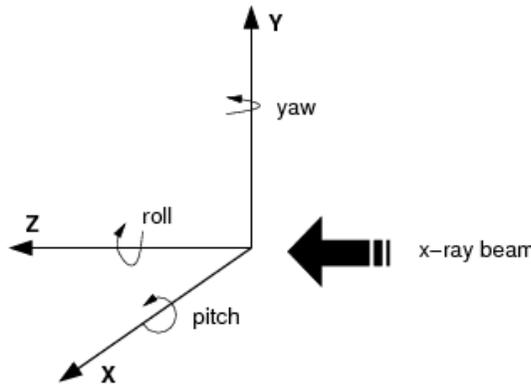


Figure 1: Coordinate system definition.

The coordinate system convention adopted here follows a right-hand rule, with the positive z-axis defined as parallel to the direction of x-ray beam propagation, the y-axis is positive in the vertical direction towards the ceiling, and the x-axis as being orthogonal to both the y and z-axes. See Figure 1.

Common angular definitions may also be provided for *roll*, *pitch*, and *yaw*, where roll is rotation about the z-axis, pitch is rotation about the x-axis, and yaw is rotation about the y-axis. These angular conventions also follow a right-hand rule. A formal coordinate system document that covers beamline requirements will be developed in consultation with the beamline scientists.

3 Naming Convention

The equipment, signal, and process variable naming convention shall conform to the general prescription provided for the facility [8]. This document is being extended to cover the requirements for beamline system, device and signal naming.

It is anticipated that much use will be made of the process variable *aliasing* feature of EPICS 3.14.11 and greater. This will permit more user-friendly descriptors to be employed for often used control system channels. Note: the aliasing mechanism should be used in concert with a descriptive entry for each applicable record's '.DESC' field. The nomenclature standard will be extended to include the conventions for beamline PV aliases.

4 Electrical Cabling Convention

All equipment shall conform to the cabling and connection conventions as described in the document, *Beamline Systems Electrical & Control Interfacing Standard* [1].

In particular, motion system cabling follows the convention used at the ESRF, ASP, and Soleil: drive and limit/reference signals are routed in the same cable, and circular, industrial-style connectors are utilized. This arrangement provides durable connectivity, easy connectivity, and assures that end-of-travel limits follow the proper axis.

5 Sub-Systems

Generally, beamlines will contain the sub-systems illustrated in Figure 2. The Controls group will provide a seven or nine-slot VME 64x chassis, VME CPU, and Timing systems Event Receiver (EVR). It should be noted, however, that the Controls group prefers to minimize reliance on VME technology, and is inclined to support “network appliance” hardware wherever possible. Thus, PLC I/O modules shall be used preferentially, and, where appropriate, in roles that may have been traditionally occupied by VME (i.e. non-demanding digital and analogue I/O) or other field-bus technology, such as IEEE-488 (GPIB).

5.1 Input/Output Controller (IOC)

Each beamline shall be equipped by the Controls Group with a single, nine-slot VME-64x crate. The system controller and CPU board in these crates shall be an MVME3100 (PowerPC) or GE Fanuc V7768 (x86). The crate shall also contain an Event Receiver (EVR) board for interfacing with the accelerator timing system. The EVR form factor shall be either a VME module or a pluggable mezzanine card (PMC), which is mated with the crate CPU (available with the MVME3100 only).

Control system software on the crate CPU shall execute under the RTEMS operating system. The version of RTEMS utilized shall be 4.10 or higher. In the case that Linux is utilized on a VME CPU, it shall be based on Debian version 5.0 or higher, and the kernel version shall be 2.6.34 or higher.

The standard softIOC will be a 1U rack mount server running Debian Linux version 5.0 or higher with a kernel version of 2.6.34 or higher. Standard commercial hardware will be used for the softIOCs. The platform will be the same as that selected for the accelerator softIOCs.

5.2 Standard Software Packages

EPICS base

synApps packages: motorRecord, Asyn, optics, areaDetector (these should be available as discrete *.debs), etc

IOC 'health' diagnostics (per-IOC).

Autosave/restore.

CA Archiver.

Conserver + procServ for all IOC console access and logging.

5.3 Insertion Device

Whether a fixed or adjustable-gap design, insertion devices will serve as the photon source to many, if not all, NSLS-II beamlines. Given their location in the storage ring and the potential for facility-wide impact that introduces, insertion device operation requires the implementation of lockout policy whereby accelerator operators grant permission of ID use to beamline staff only when the machine state permits [3].

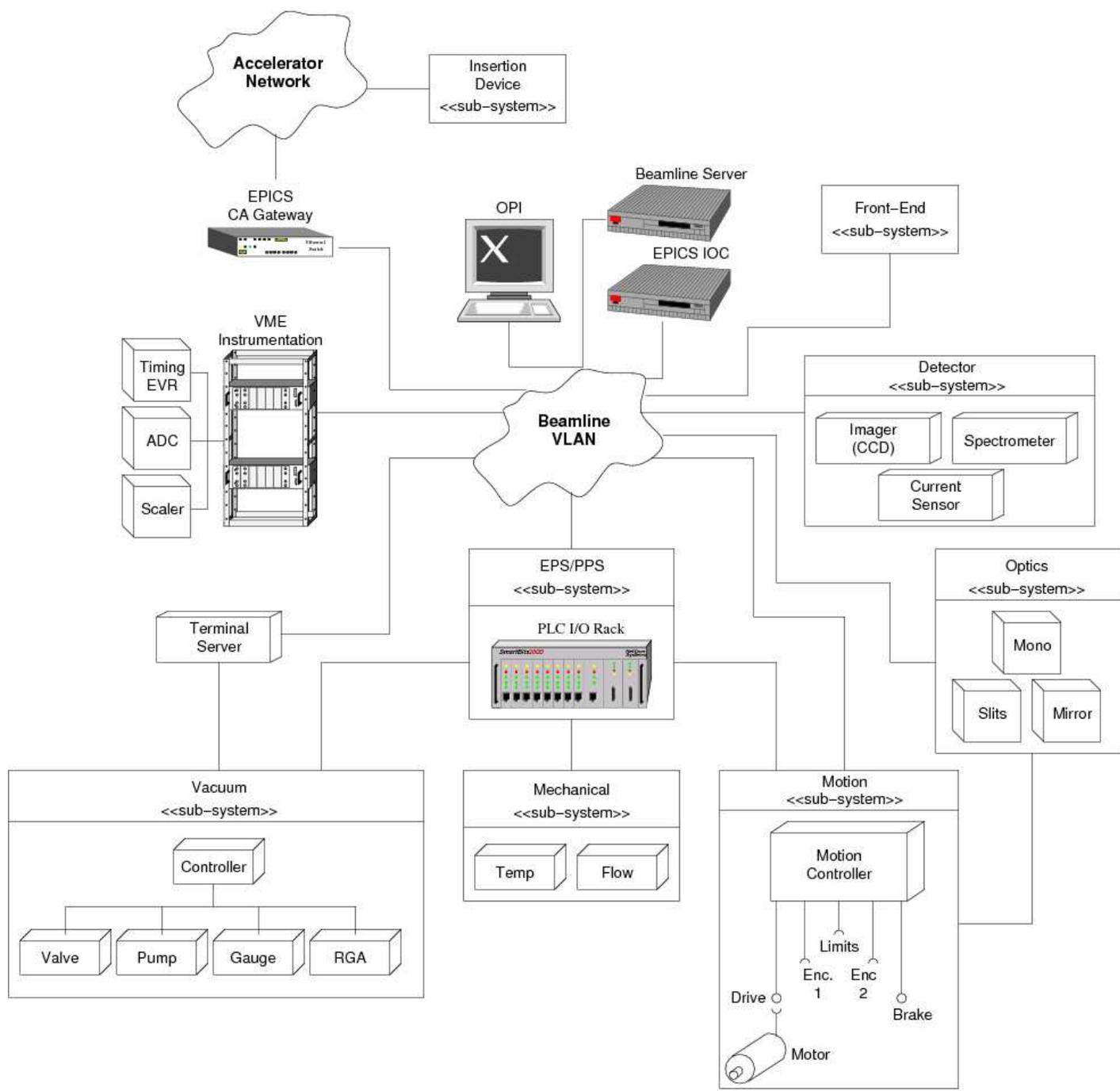


Figure 2: Beamline sub-system layout.

5.3.1 Integration with Monochromators

Insertion devices require access by both accelerator operators and beamline users, yet they are integrated into and considered a component of the accelerator systems. Control system access by beamline staff and users shall be via a Channel Access Gateway server. Indeed, all accelerator process variables shall be accessed from the beamlines in this way.

Process variable access by proxy in this manner imposes constraints on the timeliness with which values may be retrieved. However, the timing constraints may be of similar magnitude to those present over Channel Access in the absence of a Gateway server. It shall be important to characterize these delays, as they will dictate the timeliness with which an ID may be synchronized with its associated monochromator, when using Channel Access as the sole means of disseminating ID gap/phase information.

As an alternative to a software means of coordinating ID and monochromator motion, at least one motion controller vendor (Delta Tau) provides a hardware mechanism for distributing motion system information amongst physically separated units. Using the Motion And Control Ring Optical (MACRO) system, ID gap/phase position may be transmitted over optical fiber or copper at servo rates (2-20 kHz).

Additional details on ID-monochromator synchronization may be found in an associated design note [2].

5.4 Front Ends

Beamline Front Ends are those components between photon source at the storage ring and vacuum gate valve outside the ratchet wall. The current standard Front End design for NSLS-II contains eight axes of motion: a four-axis slit, and a pair of x-ray BPMs with two axes each. The controls requirements for the front ends are defined in the Front Ends PDR .

The photon safety shutters (PSS) at the ratchet wall are a critical element of the beamline personnel protection system and will be interlocked with optical enclosure access points downstream of each Frontend, as well as accelerator state. The beamline control system needs to know the status of the front end shutters, and also needs the ability to request the shutters to close or prevent them from opening. Refer to FE PDR for details .

5.5 Vacuum

To the greatest extent possible and practical, vacuum components utilized in the accelerator systems shall be used for beamline vacuum systems. This includes valves, pumps, gauges, associated controllers, PLCs, IOCs, and EPICS software.

Many vacuum pump, RGA, and gauge controllers are controllable via serial protocols (RS-232/422/485). To permit control system software access, these and *all* other serial protocol devices shall interface with terminal servers, such as the Moxa NPort. Note that the high-voltage operation of certain vacuum gauges may require the use of terminal servers with appropriate electrical isolation on their communication ports.

5.6 EPS/PPS

Equipment and personnel protection systems are typically monitored by programmable logic controllers (PLC). This includes systems like optical enclosure interlocks to prevent personnel access during safety-

critical conditions, and beamline vacuum interlocks which can actuate valves to isolate problematic sections of beam transport line.

The reaction time of the beamline EPS components should be similar to that of the accelerator's EPS: no more than 20 ms from fault-recognition to the initiation of mitigating action.

The NSLS-II Controls group has a preference for and recommends the use of Allen-Bradley and Compact-Logix PLC and Compact-I/O equipment. However, this preference is subject to equipment adherence to safety approval rating when utilized in a PPS capacity.

The photon safety shutter (PSS) control shall be accessible from the beamline control system. This will maximize possibilities for automation of the shutter control within the experimental control strategy.

The EPS functionality for beamlines will be documented in the Controls PDR for each beamline. These documents are currently being developed. Typical EPS functions will include:

- monitoring high heat load component temperatures
- monitoring coolant flow to components
- monitoring vacuum status in the beamline
- taking protective action as required - closing photon/safety shutters, closing vacuum valves
- reporting status of beamline components to the control system

5.7 Motion

Many elements of a typical beamline have an associated motion aspect to them: mirror mounts, slit apertures, endstation goniometers and diffractometers, etc. Thus, motorization is one of the largest and most critical sub-systems on a beamline.

5.7.1 Generic Motion System

For the purposes of this discussion, consider a Motion System to be those components interconnected as illustrated in Figure 3. The major system components are:

- **IOC:** Input/Output Controller. This is a computing system hosting one or more EPICS applications, at least one of which exposes a Channel Access (CA) interface to the Motor Controller, `MotorControlApp`. The CA interface permits control of the motion system regardless of the operator's location (subject to proper operator permissions).
- **Operator Interface:** Often abbreviated as OPI, this is the software interface for exercising control over the motion system. Setting and monitoring kinetic parameters (position, velocity, etc.) of the motion system are performed via this interface. Monitoring of sensor values associated with the motion system will also be available through the OPI. Control System Studio and associated tools such as BOY (Best OPI Yet) will be used for the beamlines OPI.

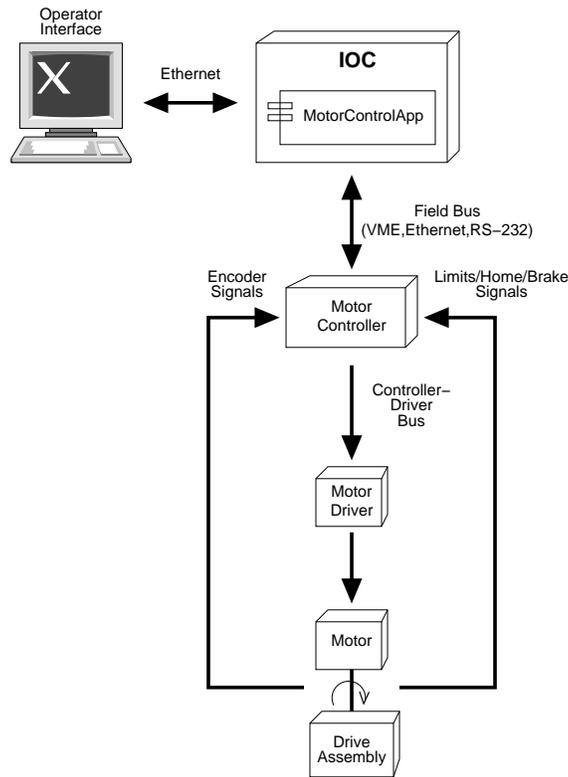


Figure 3: Motion control component template.

- **Motor Controller:** The embedded system responsible for controlling one or more axes of a Motion System. The motor controller accepts and processes limit, HOME, and encoder inputs, executes commands from the IOC, as well as providing the drive signals to one or more motors and possibly drive brakes. The controller is interfaced to the IOC via a *field-bus*, which is commonly a VME backplane, Ethernet, or RS-232 connection. Inputs from an encoder may be in the form of TTL quadrature signals, a synchronous serial interface (SSI), or another serial protocol (eg: Heidenhain enDat protocol).
- **Motor Driver:** Also known as an *amplifier*, the motor driver converts step and direction, or low-voltage analogue signals from the motor controller into appropriate current signals to power the motor. The driver's current and voltage ranges, as well as signal outputs, must be properly matched to the requirements of the motor utilized on that axis.
- **Motor:** May be a stepper, servo, or piezo. The choice of motor must be made to suit the mechanical requirements of the problem; full consideration of resolution, range, torque, velocity, and acceleration needs will lead to proper motor selection.
- **Drive Assembly:** The drive assembly includes the gearing, drive-screws, and the load to be moved.
- **Encoder:** These devices measure, either absolutely or relatively, a linear or angular position. The

encoder characteristics must be carefully chosen to meet a motion system’s requirements for positional precision and reproducibility: closed-loop control depends on the accuracy and bandwidth of its feedback sensor(s). Encoder location within the drive assembly is also important. For example, it is ineffective to sense position at a point in the drive assembly prior to the location where backlash occurs.

- **Limit Switches:** Also known as over-travel, or end-of-travel switches. These are typically normally closed (NC) switches mounted on a drive assembly and interfaced with a motor controller (or similar PLC device) such that when engaged, action is taken by a controller to prohibit further motion along that direction.
- **Brake:** If it is desirable to supplement the static holding capacity of motor, a braking system will be installed, which is actuated by the motor controller (or similar PLC device). This permits de-energizing of the motor during periods of immobility where that action may result in unintended motion (example: a vertically-driven axis).

5.7.2 Calibration

A typical beamline will have on the order of 100 axes of motion. Given this volume, a comprehensive calibration system becomes a practical necessity. Those axes equipped with absolute encoders will retain their positions across power cycles, whereas those systems with incremental (relative) encoders, or those without encoders, will not. Calibration routines for the latter systems typically involve moving each system axis in turn until a known position is reached (i.e. the “Home” position), as indicated by the activation of a precision switch. At that point the axis may be assigned an absolute position in engineering units.

Motor positions shall be maintained in at least two coordinate systems. Where a motor driven system is fitted with encoders, a third *Encoder coordinate system* shall be required [6]. The *Dial coordinate system* shall be set to correspond to the physical dial settings of the instrument where applicable and have the same engineering units as the *User coordinate system*. The User coordinate system shall be related to the Dial coordinates by a simple linear relation, determined by a calibration procedure for each motor. Dial, User and Encoder coordinates shall be related by the following equations:

$$\begin{aligned}
 Dial &= motorSteps \times motorResolution \\
 Encoder &= (encoderPulses \times encoderResolution) + encoderOffset \\
 User &= (Sign \times Dial) + userOffset
 \end{aligned}$$

Where motor-steps is the motor step count reported by the motor controller and encoder-steps is the encoder step count reported by the encoder hardware.

Every axis shall have an associated ‘homing’ procedure, whether implemented in the motor controller or in EPICS. Stateful calibration information shall be maintained for each axis as well, including at a minimum calibrated/not-calibrated status and time of last calibration. Also, motor positions and settings shall be persistent across software and hardware restarts.

5.7.3 Motor Controller

With the volume of axes expected, a standardized solution for motion control is required. The specifications for this controller are formally detailed elsewhere [4]. In overview, the specification calls for a 3U rack-mount unit with integrated low-voltage drives capable of controlling 8 axes.

The motor controller shall support the following functions and ability to set the following parameters for each axis:

- Make moves specified in relative and absolute distances
- Make a move at a constant velocity
- Trapezoidal and S-curve velocity profiles
- Coordinated motions of complex multi-axis systems (example: hexapod platforms)
- Abort a move
- Adjust maximum velocity, acceleration, and *jerk* for a move
- Setting of upper and lower *soft* limits
- Calibration, or *homing* procedures against positive or negative limits or reference signals
- Selection of homing algorithm
- Adjustment of closed-loop gain parameters (example: PID and velocity feed-forward gains)
- Amplifier maximum and minimum (holding current) output values
- Adjustment of the number of microsteps per full step

The motor controller shall include the ability to read back the following parameters for each axis over a range of 10-100 Hz:

- Current motor position, encoder, and step counts
- Current motor commanded position
- Following error when operating in closed-loop
- Current motor velocity and acceleration, in step and encoder counts
- Hard and soft limit, and reference signal status
- Amplifier status and current output value

The following capabilities are also required:

- Synchronized motion initiation across multiple axes, triggered by software command or by external signal

- Complex motion where one or more axes follow a trajectory specified as a table of positions and times (PVT mode)
- Output-on-position, where an output signal is generated upon an encoder reaching predefined, tabulated positions. The output signals shall be TTL logic compatible.
- Position capture, where encoder positions are tabulated upon the reception of hardware or software input triggers. The hardware input signals shall be TTL logic compatible.
- User programmable complex kinematics permitting multiple axes to be realized as one or more logical, or *virtual*, axes
- Lookup-table compensation or linearization of encoder values
- Readout of all controller internal status and current settings of variables
- The ability, at the hardware level, to coordinate motion between physically distributed controllers. For example, distributing at servo update rates (2-10 kHz), an encoder position from Controller A to Controller B, which may be located 10's of meters away from Controller A. This would permit Controller B to tightly coordinate the motion of its axes with those of Controller A.
- Software to permit the tuning and analysis of controller-specific motion and configuration.

5.8 Optics

In order to focus and direct photons onto target, beamlines employ multiple optical elements along their length. These include mirrors, monochromators, slits, and attenuators. While x-ray beam attenuators are typically pneumatically actuated, most other optical elements are driven by stepper or piezo motors.

To the greatest extent practical, use shall be made of the *optics* module distributed with the *synApps* collection of EPICS-based synchrotron applications. This module includes support for optical tables, monochromators, mirrors, slits, attenuators, and diffractometers.

Many optical components will be purchased as turn-key systems from vendors. To ease integration into beamline control systems, optical component sub-systems must conform to standards established for use on the beamlines (eg: motion controllers and cabling). Given that vendors may be unfamiliar with these standards, they must be approached early in the design phase and offered assistance and incentive to encourage adoption. This may take the form of equipment lending and assistance with configuration and testing.

6 Data Acquisition

Providing an effective and efficient data acquisition system for the beamline users is an essential part of the beamline control system development. The initial six beamlines will have flexible experimental arrangements, and so will require the experimental control system to be equally flexible. Specific data acquisition requirements for each beamline are being gathered and will be documented in each individual beamline Controls PDR.

6.1 Detectors

Existing and anticipated detectors will substantially increase the data rates and volumes at each beamline. Suitable infrastructure will be required to support these detectors. These requirements will be identified in each individual beamline Controls PDR.

6.2 Scanning

Most beamline experiments are moving towards performing scanning on-the-fly. This significantly reduces the overheads in collecting a data set, but places additional requirements on the data acquisition hardware, motion control hardware and experimental control system. The motion controller specification defines capabilities that will allow the standard NSLS2 motion controller to integrate well into on-the-fly data acquisition systems. Specific motion controller capabilities to support on-the-fly scanning include:

- position capture on external trigger
- generation of trigger at defined points in trajectory
- output of motor position information to external data acquisition hardware

Existing scanning software (e.g. EPICS sscan record, SPEC scripted scans) do not support the full range of anticipated experimental data acquisition functionality. The NSLS2 Controls Group plans to design and implement a data acquisition architecture that will support the anticipated data acquisition needs of the NSLS2 beamline user community. This project will collaborate with other facilities to ensure the solution is sufficiently flexible to be adopted outside NSLS2. It will also build on existing developments in high-level physics applications at NSLS2 where appropriate.

6.3 Scripting interface

In order to provide experimental users with the flexibility to customise the operation of their experiment, a scripting interface will be provided at each beamline. It is expected that the supported scripting language will be Python, given its widespread use within the scientific community. There are existing interfaces to the underlying EPICS communications protocol for Python.

A number of the beamlines have user communities that are familiar with the use of SPEC for beamline control. The scripting interface provided at NSLS2 will need to provide at least some aspects of the SPEC functionality. This will include the ability to define operations in reciprocal space, and to implement scanning operations. Both of these functions are (at least partially) implemented in the GDA scripting environment. This should reduce the development effort required to incorporate these functions into the NSLS2 scripting environment.

6.4 Data acquisition applications

6.4.1 Generic Data Acquisition (GDA)

Originally developed at Daresbury Laboratory, the Generic Data Acquisition software, or GDA, is now maintained by the Diamond Light Source (DLS) in the UK. This software package features a client/server

architecture, scripting, and data analysis and plotting capabilities. GDA is build upon the Eclipse rich-client platform (RCP), and therefore depends heavily upon Java technology. Graphical clients communicate with one or more server-side applications using CORBA. The client-side scripting capability is provided using Jython, which permits inter-mixing of Java and Python objects from a Python interpreter command-line. Ad-hoc scanning is available from the Jython terminal including support for one and two-dimensional scans as well operations defined in reciprocal space (i.e. hkl-scans).

A preliminary study of GDA (version 7.14) indicates that it is a promising, if very complex piece of software. For instance, GDA is structurally comprised of approximately 2300 Java files, 700 Python files, 100 Java jar-files (dependencies), and 200 XML files. Despite its structural complexity, GDA may be tailored for use at a specific beamline simply by providing a declarative, XML description of the EPICS process variables which are required for data acquisition and experiment control (motors, detectors, etc). In some cases, experiment sophistication may dictate the development of additional software classes and further extensions to GDA's basic functionality.

Further review of GDA, both through visits to Diamond and a GDA workshop hosted at BNL, suggests that it is not a suitable application for adoption. The software architecture is inflexible, and it does not clearly define useful data types for beamline control and data acquisition. Development of customised user interfaces requires specialised software engineering resources, which will not be available at each beamline. This limits the ability of the beamline staff to configure GDA for their own requirements.

6.4.2 SPEC

The data acquisition and analysis software package known as spec is a commercial offering from Certified Scientific Software (<http://www.certif.com>). This software is widely used in the synchrotron community, and it features direct control of motion and detector hardware, a C-like syntax and interpretive scripting, many built-in and user-contributed scanning algorithms, as well as real-time and offline data plotting.

Concerns have been raised regarding the long-term stability and support of spec given that the company is a relatively small enterprise with a closed-source policy of software distribution. It is also a concern that there appears to be little in the way of spec development: most of the software development is comprised of adding hardware support and incorporating user 'macro' functions.

A number of beamlines have stated that Spec functionality will be required by their users. Alternatives to Spec that offer the same or similar functionality will be investigated.

6.4.3 BLISS

The BLISS package (<http://www.esrf.eu/UsersAndScience/Experiments/TBS/BLISS/BLISS-Applications/>) developed at ESRF has been proposed as a possible framework for beamline data acquisition. This will be investigated further. One key issue with BLISS is how tightly it is integrated to the underlying TANGO control system.

6.5 Remote access

Many beamlines will require remote access to both experiment control and experimental data. The requirements for this will be defined during the design phase for each beamline. Common solutions will be implemented where appropriate.

7 Interfaces to Accelerator Systems

7.1 Network

Interfacing between the accelerator network and that of individual beamline VLANs will be an EPICS Gateway server. This will serve as a proxy to control system network traffic, caching accelerator process variables for use by beamline IOCs. This has the net effect that a heavily subscribed accelerator process variable (ex: storage ring current or lifetime) may be obtained from a beamline's EPICS Gateway by its client applications, rather than directly from the IOC hosting that PV. This reduces accelerator network and IOC resource loading. The EPICS gateway also controls whether accelerator PVs can be written to from the beamline or are read-only.

7.2 Timing

The NSLS-II timing system consists of a centrally located event generator (EVG) distributing event-based information synchronously to event receivers (EVR) located at each beamline. The inputs to the EVG include a 500 MHz master-oscillator and a GPS-based, absolute-time signal. This system will distribute beam-based timing information such as fill-pattern and bunch-timing, facility time-stamps, as well as sending event-codes and data to EVR modules, all with 8 ns to 400 ps resolution and 25 ps to 5 ps jitter (peak-to-peak). EVR modules have several digital output channels that may be activated upon the recognition of event-code reception, and thereby serve as triggers to beamline electronics.

Several requirements have emerged from beamline science staff:

1. Master-oscillator and timing system signals are available *at all times*. This will permit experiment configuration and debugging in the absence of beam.
2. Stored-beam parameters be made available over the timing system. This might include attributes such as beam-stability metrics, average beam-current, fill pattern or bunch-timing properties.
3. Injection-blanking signals. This signal indicates injection progress, from injection to storage ring until the instabilities have damped out, and serves as a binary indicator of optimal conditions for data acquisition programs that are sensitive to electron beam instabilities.
4. Triggers to indicate specific buckets on each storage ring revolution.

7.3 EPS

Each beamline will need to interface with accelerator equipment and personnel protection systems in order to be able to react accordingly to system state and alarm conditions. For example, front-end system vacuum trips, or beam-loss monitor and x-ray BPM information needs to be communicated between accelerator controls and those on the affected beamline.

Some beamlines will have a fast vacuum sensor installed in the First Optics Enclosure. This will be connected to the fast vacuum valve controller to close the fast vacuum valve in the front end in the event of a rapid beamline vacuum failure.

7.4 PPS

Each beamline will require the ability to close and open the front end photon and safety shutters. Each beamline will also require information about the status of those shutters. These functions have been incorporated into the machine PPS design.

7.5 Diagnostics

At least two of the initial six beamlines (CSX and CHX) have specified a requirement of obtaining storage ring current at rates on the order of 10 kHz. This rate coincides with the maximum sampling rate outlined during the preliminary design of the accelerator diagnostic control system [7]. Thus, this requirement does not seem possible to satisfy using information from storage ring pulsed-current-transformer (PCT).

In order to disseminate information from the accelerator diagnostics at such rates in a deterministic manner, the Cell controller nearest to the interested beamline may need to be utilized [5]. While it is not possible to distribute information from the PCT at 1-10 kHz over Channel Access, it should be possible to use the intensities of those e-BPMs nearest the interested beamline(s) to achieve similar measurements. Correlation with storage ring current values shall then provide calibration in absolute terms.

8 Beamline configuration database

IRMIS will be used to store beamline component configuration information. This will include optical components such as slits, mirrors and monochromators as well as individual devices such as motors, encoders, limit switches and vacuum pumps and gauges. IRMIS will store the housing, power and controls hierarchy for the beamline components. It will also be used to store configuration files for configurable devices.

References

- [1] A. Broadbent. Beamline systems electrical & control interfacing standard: NSLS-II-SPC-0001-wiring. Technical report, NSLS-II, 2008.
- [2] D. Chabot. Design note: Insertion device control. Technical report, NSLS-II, 2010.
- [3] D. Chabot. Insertion device control: Preliminary design report. Technical report, NSLS-II, 2010.
- [4] D. Chabot. Technical Specification for the NSLS-II Motion Controllers. Technical report, NSLS-II, 2010.
- [5] Joe DeLong. Cell controller design document. Technical report, NSLS-II, [http://groups.nsls2.bnl.gov/acceleratorsystems/controls/Shared Documents/Cell Controller/Cell controller design document.docx](http://groups.nsls2.bnl.gov/acceleratorsystems/controls/Shared%20Documents/Cell%20Controller/Cell%20controller%20design%20document.docx), 2010.
- [6] M. Fodje. CMCF software requirements specification. Technical report, Candian Light Source Inc., 2008.
- [7] Yong Hu. Preliminary design of beam diagnostics control system. Technical report, BNL, 2010.
- [8] Erik Johnson. NSLS-II nomenclature standard. Technical report, NSLS-II, <http://groups.nsls2.bnl.gov/acceleratorsystems/interface/SharedGlobal>
- [9] M. McKibben. Control system common specification. Technical report, Candian Light Source Inc., 2009.