Project X Control System Requirements

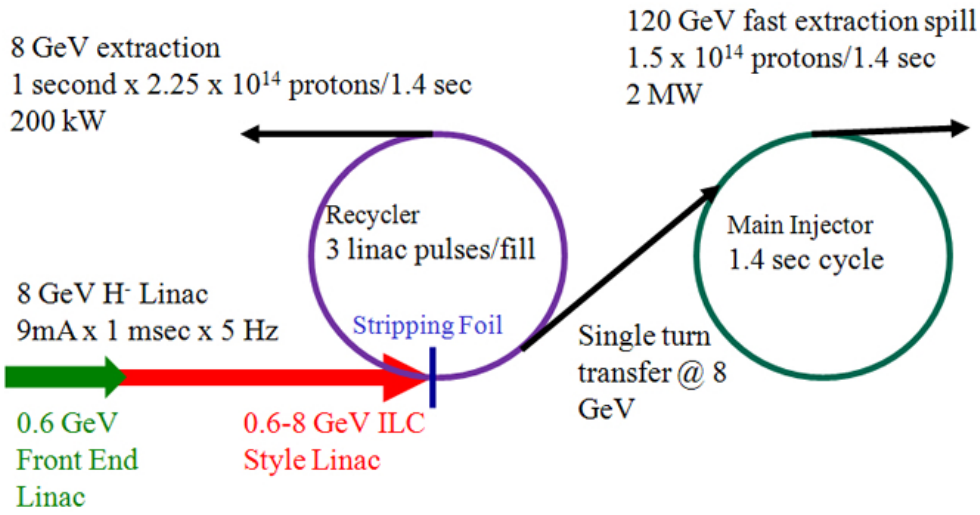
T.Bolshakov, C. Briegel, K. Cahill, L. Carmichael, D. Finstrom, S. Gysin, B. Hendricks
C.King, W. Kissel, S.Lackey, W. Marsh, R.Neswold, D. Nicklaus, J. Patrick, A. Petrov,
R.Rechenmacher, C. Schumann, J. Smedinghoff, D. Stenman, G. Vogel, A. Warner,
T. Zingelman

Table of Contents

Version	Date	Comments
2.1	2008-02-21	Updates to Controls in a Box
2.0	2008-02-01	Updates from internal review
1.0	2008-01-14	Document for internal review
0.3	2007-11-29	Fixed the section so the match the table of contents. Moved Macro Language and Synoptic display from 9 to 6.2
0.2	2007-11-18	Added references, merged with Andrey's outline for Central Services
0.1	2007-11-09	First draft into doc db

1 Introduction

Project X is a concept for an intense 8 GeV proton source that provides beam for the Fermilab Main Injector and an 8 GeV physics program. The source consists of an 8 GeV superconducting linac that injects into the Fermilab Recycler where multiple linac beam pulses are stripped and accumulated. The 8 GeV linac consists of a low energy front-end possibly based on superconducting technology and a high energy end composed of ILC-like cryomodules. The use of the Recycler reduces the required charge in the superconducting 8 GeV linac to match the charge per pulse of the ILC design; aligning Project X and ILC technologies. [1]



The control system for this accelerator (Control X) should be of modern design, use currently available high performance hardware and networks, and have a track record of success in the accelerator control business. To the extent possible, the equipment utilized should be readily available commodity equipment. Use of standards in equipment and software system will aid in development, diagnosis, and repair.

This document is an agreement between the users and the designers/developers of the functionality of the control system. While writing the document the authors, who are developers and users, are required to discuss and eventually agree on the functionality. This document deliberately avoids a specific implementation or design and focuses on the 'what' rather than 'how'.

The audience of the document, once it is completed, are the designers and developers of the control system. They will refer to the requirements to decide on a design that is most optimal for the requirements.

2 Base Requirements

In the following section are the very basic requirements driven by the specifics of Project X. These are meant to drive the other requirements, or considering this structure as a tree, the base requirements are the trunks from which smaller branches originate. For example, the scale of Controls X, will drive requirements for a middle tier, and a large network bandwidth. The intent is that one can always track a requirement to it's original base requirement.

2.1 Scale

The Tevatron complex control system currently controls about 200,000 devices[2]. In broad terms, Project X has a similar scale considering the linacs, the beam injection line, the main injector recycler, and target station. Each device can have up to five properties, which means Control X should be designed to control about one million properties.
A generous assumption for maximum load is 200 users accessing the control system simultaneously. The average load is probably about 50 users. From these estimates we derive the very basic scale requirement:

Controls X shall be able to support 200 users accessing 5000 properties each.

The large number of components and heavy traffic imply requirements for a middle tier to balance the load and consolidate the requests, a large network bandwidth. Any modern control system will assume that the user may or may not be on site, so the control system must support some remote access.

2.2 Availability

To maximize performance we have to maximize the time the beam is in the accelerator. And to maximize the time the beam is in the accelerator, the control system has to be very reliable, fault tolerant, and we have to design for a minimum mean time to repair (MTTR).

The uptime is often quoted in availability, for example 75% availability means, the accelerator is delivering beam 75% of the time. Scheduled maintenance shut downs are not included in the availability target. In the example above, the scheduled maintenance is not counted towards the 25% the accelerator is not available.

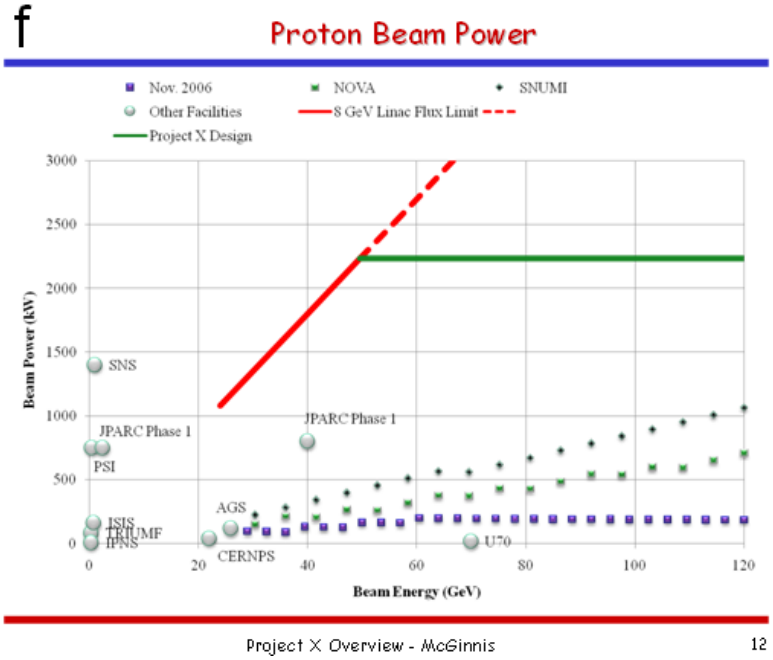
The availability is defined for the accelerator complex, and the control system is allowed to account for some fraction of that. A high availability requirement for Project X early on will ensure availability is considered at all stages of the design, as it can affect major design choices as well as the detailed design of each component. Software errors contribute up to 30% of failures and the control system has a great amount of software that could potentially contribute to failure. A detailed analysis of how control system availability relates to beam availability is complicated. Ideally, the control system should never fail[4]. The consequences of failure of a critical part

of the control system can be devastating, so the availability of Project X has to be considered in the design for Control X.
The ILC control system has a requirement of 2500-hr MTBF (mean time between failures and 5-hr MTTR (meant time to repair) and 15 hours downtime per year. [4].

Control X shall have no less than 2500-hr MTBF and no more than 5-hr MTTR.

2.3 Safety

The beam power of Project X is a magnitude of 10 times larger than that of the current operations.[3].



The current beam power is about 200 kW; Project X is targeted for 2 MW. At 2MW, an accident can cause serious damage to people and equipment. This drives the requirements of a stringent machine protection system (MPS), such as hardware and software interlocks, access control, and alarms.

Controls X shall have an extensive machine protection system, including hardware interlocks, software interlocks, access control, and alarms

With high beam power, accidents are not the only concern. Just routine losses can activate components so that they fail more often and become difficult to work on due to residual radioactivity. To prevent this beam trajectories must be well controlled. This will likely require the control system to do fast feedback.

Controls X shall have a fast (5 Hz.) feedback system to control the beam trajectory and thereby minimizing routine beam losses causing components to be activated and radioactive.

2.4 Legacy Constraints

At the time Project X begins operation, the Accelerator NUMI Upgrade will have been completed and the recycler, main injector, NUMI beam line, and 120 GeV fixed target lines operated for some years in that configuration. These elements will be controlled by an evolution of the current ACNET system. This includes field equipment, the timing system, front-end computers, services, and applications. While some changes will be needed in these accelerator components for Project X, the control system hardware and software represents a large investment that could be difficult to completely replace by the start of Project X operation. Hence the Project X control system must interoperate with the existing system to the extent that is necessary for seamless operation.

2.5 Summary of Base Requirements

Project X will have about 9 km of beam line and 1 Million device properties. It will have 10x more beam power, and it has some legacy constraints because it uses the Main Injector and Recycler.

From these constraints we derive the base requirements:

No.	Requirement	Source	Priority
CXR-10	The control system shall support 200 users accessing 5000 properties each.	S.Gysin 10-2007	Critical
CXR-20	The control system shall have no less than 2500-hr MTBF and no more than 5-hr MTTR.	S.Gysin, J.Patrick 10-2007	Expected
CXR-30	The control system shall have an extensive machine protection mechanism, including hardware interlocks, software interlocks, access control, and alarms.	S.Gysin 10-2007	Critical
CXR-40	The control system shall have a fast (5 Hz.) feedback system to control the beam trajectory and thereby minimizing routine beam losses causing components to be activated and radioactive.	J.Patrick 11-2007	Critical
CXR-41	The control system shall comply with the safety policy of the laboratory.	J.Patrick 1-2008	Critical

The control system for the linac and transfer line must satisfy the following requirements to meet the legacy constraints:

No.	Requirement	Source	Priority
CXR-50	Timing signals shall be provided in a format that can be accepted by legacy hardware.	J. Patrick 12-2007	Critical
CXR-60	Machine protection system inputs from legacy hardware shall be accepted.	J. Patrick 12-2007	Critical

CXR-70	It shall be possible to acquire data from legacy hardware into applications and into a common archive for proper correlation across the complex.	J. Patrick 12-2007	Critical
CXR-80	It shall be possible for applications in the legacy system to acquire data from new linac subsystems. It may not be necessary to support access to all devices and data acquisition protocols however.	J. Patrick 12-2007	Critical
CXR-90	The alarms service shall be able to receive alarms generated by the legacy system.	J. Patrick 12-2007	Critical
CXR-100	Applications that run on the legacy system shall be conveniently accessible to operators.	J. Patrick 12-2007	Critical
CXR-110	The control system shall adhere to lab wide security policy.	S.Gysin 2-2008	Expected

What follows are the requirements derived from these high level, and grouped into the three tiers of functionality. Low level i.e. the front-ends that interface directly to the instrument, central services i.e. software running on servers, and high level which is the software operators use. Additional sections that span all three layers are the machine protection system, software build system, hardware, and network.

3 Low-Level Systems

3.1 Timing System

Timing systems are critical to the ability of Project-X to coordinate beam acceleration and transfer between the various accelerators that will make up the complex. They are also essential to the ability of the control system to provide correlated data acquisition. In order to provide these timing capabilities, there will be two types of clock systems in Project-X.

The first will be a basic accelerator clock that provides high-level timing for the entire complex and is common to all machines. In the legacy systems this function is accomplished via the **TCLK** system. It is an 8-bit, 10 MHz, modified Manchester encoded serial transmission of clock events that provide basic accelerator timing information. As more timing functionality (16 bit events, event indexing, etc.) is necessary to facilitate the acquisition and correlation of machine data for the new accelerators of Project-X, a new clock system (herein referred to as **XCLK**) is required. As TCLK will continue to be generated to supply timing signals for the legacy systems, these two clocks systems will need to be strictly synchronized. It is expected that all new systems installed around the complex will make use of XCLK.

The second type of clock systems are machine-specific RF based timing systems (Beam Sync Clocks.) These systems will allow for the transmission of the individual machine's RF and beam synchronization markers to facilitate high precision (RF bucket level) timing for such things as instrumentation and kicker triggering. In the legacy systems this function is handled via the individual accelerator's Beam Sync Clocks (MIBS, RRBS, etc.). As with TCLK, the legacy beam sync clocks are 8-bit, modified Manchester encoded serial transmissions. However, their base frequencies are subharmonics of the machine's RF frequencies rather than the 10 MHz of TCLK. These clocks also transmit a low amplitude copy of the machine's RF. As beam in the new Linac of Project-X must be synchronized to the Recycler, the Recycler beam sync clock (**RRBS**) must be made available throughout the machine and to the source (the Chopper).

3.1.1 Basic Accelerator Clock (XCLK)

This section only covers requirements for XCLK.

No.	Requirement	Source	Priority
CXR-LL-10	Basic accelerator clock timing shall be sourced via a single Timeline Generator (TLG) and transmitted on optical fiber.	G.Vogel 12-2007	Expected
CXR-LL-20	TCLK events shall be encoded to occur synchronously on both TCLK and XCLK.	G.Vogel 1-2008	Critical
CXR-LL-30	XCLK shall run on a 1 GHz, or higher, carrier phase-locked to the TCLK's 10 MHz carrier.	G.Vogel 12-2007	Expected

No.	Requirement	Source	Priority
CXR-LL-40	16 bits of the XCLK data frame shall represent the XCLK clock event. The XCLK frame shall have an additional *n* bits for data payload.	G.Vogel 1-2008	Expected
CXR-LL-45	XCLK events outside the range $0000-$00FE shall generate a TCLK $FF event.	G.Vogel, J.Smedinghoff 1-2008	Expected
CXR-LL-50	The XCLK frame size shall not exceed 1.2uS.	G.Vogel 12-2007	Critical
CXR-LL-60	Events occurring on XCLK shall not affect the timing of events in the legacy system.	G.Vogel 1-2008	Critical
CXR-LL-70	The data in the event payload shall be self-describing.	C.Briegel 12-2007	Desired
CXR-LL-90	32 bits of the XCLK frame shall be reserved for a per-event counter (Event Index).	R.Rechenmacher 12-2207	Desired
CXR-LL-100	In order to provide redundancy needed to meet availability requirements, 2 fibers carrying XCLK shall run from repeater to repeater.	G.Vogel 1-2008	Expected
CXR-LL-110	The repeaters shall constantly monitor and compare the two transmissions and have auto-switchover if one carrier fails.	G.Vogel 12-2007	Expected
CXR-LL-120	The repeaters shall inhibit beam if total clock is lost.	G.Vogel 12-2007	Expected
CXR-LL-130	The hardware group shall provide hardware XCLK simulators for front-end developers.	G.Vogel 12-2007	Desired
CXR-LL-135	The hardware group shall provide a standard XCLK decoder design.	G.Vogel 1-2008	Expected
CXR-LL-140	Front-end software shall be able to simulate the clock system to allow development on machines that don't have access to XCLK signals.	R.Rechenmacher 12-2007	Desired

3.1.2 Beam Sync Clock

This section only covers requirements for the Beam Sync Clocks.

No.	Requirement	Source	Priority
CXR-LL-150	The Main Injector and Recycler shall continue to use the existing beam sync clocks.	G.Vogel 12-2007	Expected
CXR-LL-160	The Recycler Beam Sync Clock shall be made available to all 8 GeV line and Linac equipment locations.	G.Vogel 1-2008	Critical

3.2 Equipment Interface/Instrumentation

The Controls System needs to interface to a variety of equipment both purchased and designed in-house. These include instrumentation, vacuum, power supplies, water systems, etc. In order to ensure that the Controls System is reliable and easy to diagnose, a limited number of standard interfaces should be implemented and duplication should be avoided. The controls group will provide a standard equipment interface for other groups to incorporate into their equipment designs.

No.	Requirement	Source	Priority
CXR-LL-170	General purpose digitizing hardware shall allow all of its channels to be plotted simultaneously.	R.Neswold 12-2007	Expected
CXR-LL-180	The middleware shall provide a mechanism to redirect traffic to a back-up node in the event of a failure.	C.Briegel 1-2008	Desired
CXR-LL-190	The hardware group shall support and provide a set of general purpose preferred digitizers.	G.Vogel 12-2007	Expected
CXR-LL-200	Front-end hardware shall support, as a minimum, full-duplex, gigabit copper communications.	T.Zingelman 12-2007	Critical
CXR-LL-210	Front-end platforms shall support Ipv6 communications.	T.Zingelman 12-2007	Expected
CXR-LL-220	Front-end platforms shall support port scans gracefully.	T.Zingelman 1-2008	Expected
CXR-LL-230	Supported hardware and software shall be enumerated.	C.Briegel 12-2007	Expected
CXR-LL-240	A committee shall review all new support (requested or required) to aid in maintaining an appropriate set of solutions.	C.Briegel 12-2007	Expected

3.2.1 Site Facilities Integration

Various site facilities/utilities such as water, electrical power distribution & environmental management (Heating Ventilation Air Conditioning - HVAC) are indeed necessary for accelerator operation. Other systems such as electrical & radiation

safety systems are critical to the success of a physics program. Such areas may have evolved separately due to technological or political reasons. It is important to provide an integrated structure to the above, enhancing troubleshooting, trend analysis and cause & effect determination.

CXR-LL-245	The control system shall provide gateways for read access into mission critical safety, utility and environmental systems.	W. Kissel 1-2008	Desired

3.3 Development Environment

Due to technical and historical reasons, the front-end development environment is separate from ones used by other groups. We suggest a consolidation of toolsets is possible, so that working on different layers of the control system doesn't require a differing set of skills.

In addition to the requirements in this section, we would like to see the front-end development also follow the requirements expressed in the Software Development Environment section (10).

No.	Requirement	Source	Priority
CXR-LL-250	The development and build environments for front-end shall be the same as used by central and high-level application developers.	R.Neswold 12-2007	Expected
CXR-LL-260	Front-ends shall run on platforms that support memory protection.	R.Neswold 12-2007	Expected
CXR-LL-270	Front-end software shall use memory protection features in its design to improve reliability.	R.Neswold 12-2007	Expected
CXR-LL-280	Front-end systems shall run on platforms that support portable APIs (e.g. pthreads, unix system calls)	R.Neswold 12-2007	Desired
CXR-LL-290	The framework used to create front-end software shall also be used to create OACs.	R.Neswold 12-2007	Desired
CXR-LL-300	The front-end shall provide a user application framework for developing and deploying user code in the front-end.	C.Briegel 12-2007	Expected
CXR-LL-310	All front-ends shall have debugging facilities available to developers ("post-mortem" facilities, access to internal state, etc.)	R.Rechenmacher 12-2007	Expected

No.	Requirement	Source	Priority
CXR-LL-320	All front-ends shall have a common set of devices to monitor the front-end (CPU Temperature, system load, memory usage, version devices, etc.)	R.Neswold 12-2007	Expected
CXR-LL-325	All front-ends shall use the same software framework.	R.Neswold 1-2008	Expected

3.4 Data Acquisition/Setting

Fermilab has a distributed control system, meaning the data used by an application is generally acquired from a remote machine. A standardized network protocol is used to express how to request data and how the data is returned to the requestor. See also Central Services (4.2) and Network section (12).

No.	Requirement	Source	Priority
CXR-LL-330	One network protocol suite shall be used to for data acquisition from the front-ends.	R.Rechenmacher 1-2008	Expected
CXR-LL-340	A device shall be viewed as an object with many attributes and data types.	C.Briegel 12-2007	Expected
CXR-LL-350	Device data shall be acquired at any rate a user specifies. If the rate exceeds the capabilities of the device, data is returned at the device's maximum rate.	R.Neswold 12-2007	Expected
CXR-LL-360	Acquisition protocols shall provide a way to correlate the data.	R.Rechenmacher 1-2008	Expected
CXR-LL-370	The acquisition protocol shall support multicast requests to acquire data across multiple front-ends.	M.Sliczniak 1-2008	Expected
CXR-LL-380	Replies to a multicast request shall arrive before a deadline prior to the next cycle and the reply data must come from the current cycle.	M.Sliczniak 1-2008	Expected
CXR-LL-390	Data acquisition protocols shall include enough metadata information that each packet can be decoded in isolation.	R.Rechenmacher 12-2007	Expected
CXR-LL-400	All device data shall be described by a data definition derived from the front-end and consistent with the application environment.	C.Briegel 12-2007	Expected
CXR-LL-410	A time stamp of when the data was captured shall accompany all device data.	R.Neswold 12-2007	Expected

No.	Requirement	Source	Priority
CXR-LL-415	The acquisition protocol shall have a global event cycle count and a global error/status field.	R.Neswold 12-2007	Expected
CXR-LL-420	The latest device data shall be able to be retrieved (known as a "one-shot" request.)	C.Briegel 12-2007	Expected
CXR-LL-425	Data shall be retrieved at a periodic rate.	C.Briegel 12-2007	Expected
CXR-LL-430	"One-shot" and repetitive data shall be retrieved based on an XCLK event occurrence followed by an optional delay.	C.Briegel 12-2007	Expected
CXR-LL-440	"One-shot" and repetitive data shall be retrieved based on a state change.	C.Briegel 12-2007	Expected
CXR-LL-450	Data shall be retrieved based on an event with a specified event counter.	C.Briegel 12-2007	Expected
CXR-LL-460	Repetitive data may be filtered to return whenever the data changes by a delta (or range, tolerance, etc.)	C.Briegel 12-2007	Desired
CXR-LL-465	There shall be a minimal interval value in case the delta rarely occurs	M.Slicznia k 1-2008	Expected
CXR-LL-470	Front-ends shall be able to acquire data from each other, directly from the remote machine -- not through a consolidator.	R.Neswold 12-2007	Expected
CXR-LL-475	Front-ends shall be able to directly set other front ends.	R.Neswold 1-2008	Expected
CXR-LL-477	All settings shall be logged (note: add filtering)	R.Neswold 1-2008	Expected
CXR-LL-480	While device data can be returned either in raw or scaled values, all front-ends shall be capable of scaling the data for internal use. This includes data acquired from other nodes.	C.Briegel 12-2007	Expected
CXR-LL-490	Devices shall be able to handle rapid (~15Hz) settings.	C.Briegel 12-2007	Expected
CXR-LL-495	All settings get forwarded to an archive.	C.Briegel 1-2008	Expected
CXR-LL-500	The setting protocol shall allow optional reading of a parameter.	C.Briegel 12-2007	Desired
CXR-LL-510	All data on the network shall be in network byte order.	R.Neswold 12-2007	Desired

No.	Requirement	Source	Priority
CXR-LL-520	There shall be an acquisition protocol that returns data suitable for a real-time plot at a minimum of 1kHz.	C.Briegel 1-2008	Expected
CXR-LL-530	There shall be an acquisition protocol that returns an array of data samples on an event (i.e. a snapshot of a waveform.)	C.Briegel 1-2008	Expected
CXR-LL-540	The waveform snapshot shall also support collection on alarm activation.	C.Briegel 1-2008	Desired
CXR-LL-545	The front-end shall provide device metadata including , but not limited to, device type.	B.Hendricks 1-2008	Desired

3.4.1 Network Protocol Policies

As an evolution of our current control system, we envision the improved protocols allow "policies" to be enabled. For instance, we might decide that settings require authentication, or that devices can be temporarily owned by users.

No.	Requirement	Source	Priority
CXR-LL-550	"Policies" active in the control system shall be honored across all acquisition methods.	R.Rechenmacher 1-2008	Expected
CXR-LL-560	Device access shall have "ownership", and operators shall be able to query who is current "owner" of the device.	C.Briegel 12-2007	Expected
CXR-LL-570	There shall be a policy of access rights for users to restrict who is allowed to make changes, mainly for preventing interference in conflicting uses of the machine.	R.Rechenmacher 12-2007	Expected
CXR-LL-580	The setting protocol shall support optional transaction semantics across front-ends (i.e. settings can be queued for later commit or rollback.)	R.Neswold 12-2007	Desired

3.4.2 Alarm Support

Alarm reporting is a very important aspect of our current control system. A huge number of devices are scanned frequently to ensure they are operating within their constraints. This section of requirements covers alarm support, along with some improvements.

No.	Requirement	Source	Priority
CXR-LL-600	Front-ends shall scan devices periodically and compare the reading with alarm constraints. Readings that violate their constraint are said to be in alarm. Alarm reports are forwarded to a central alarm service, which reports the alarms to operators.	C.Briegel 12-2007	Expected
CXR-LL-610	There shall be a default alarm scan frequency, which can be overridden for each device. The overridden devices can be sampled on any acquisition event.	C.Briegel 12-2007	Expected
CXR-LL-620	There shall be a default alarm scan routine which, at a slow rate, checks to see if any devices have gone into alarm. This routine can be replaced on a device-by-device basis if a different strategy is required.	C.Briegel 12-2007	Expected
CXR-LL-625	There shall be support for multiple alarm blocks selected by event.	D.Nicklaus 1-2008	Expected
CXR-LL-640	Analog devices have alarm constraints represented by maximum and minimum values.	C.Briegel 12-2007	Expected
CXR-LL-650	Digital devices have alarm constraints represented by a bit-mask and pattern-match.	C.Briegel 12-2007	Expected
CXR-LL-660	A device shall be constrained by a user-defined constraint.	C.Briegel 12-2007	Desired
CXR-LL-670	All alarms shall have a consecutive alarm threshold.	C.Briegel 12-2007	Expected
CXR-LL-680	All alarms shall have the ability to pull a software abort.	C.Briegel 12-2007	Expected
CXR-LL-690	All alarms shall have the ability to set a device.	C.Briegel 12-2007	Expected
CXR-LL-700	All alarms shall have the ability to provide an unsolicited alarm notification to a specified set of servers.	C.Briegel 12-2007	Expected
CXR-LL-710	There shall be event alarms, which are notifications without a clear of the event.	C.Briegel 12-2007	Expected
CXR-LL-720	There shall be exception alarms, such as analog reading, digital status, and alarm setting alarms.	C.Briegel 12-2007	Expected
CXR-LL-730	One shall be able to individually bypass an alarm.	C.Briegel 12-2007	Expected
CXR-LL-740	Alarms shall be able to be grouped together and be bypassed as a group.	C.Briegel 12-2007	Expected

3.4.3 Reliability Requirements

The following requirements improve the reliability of the network protocol.

No.	Requirement	Source	Priority
CXR-LL-750	Network packets shall be acknowledged (or the reply itself be sent) within 100ms of reception.	R.Neswold 12-2007	Expected
CXR-LL-760	Long-term, slow frequency connections shall provide a "keep-alive" status to be able to detect broken or closed connections.	R.Neswold 12-2007	Expected
CXR-LL-770	The front-end infrastructure shall be able to report, within a few seconds, the flows/connections to front-ends that do not involve "standard" front-end interface ports.	R.Rechenmacher 12-2007	Desired

4 Central Services

A trivial control system can be built of just two parts: a set of low-level equipment front-ends and a set of high-level user applications. Both sides talk to each other through some common protocol, so that the front-ends' data is reflected in the applications, as well as the user's actions are reflected in the equipment.

The larger the system grows, the less this simple model can satisfy its demands:

1. There is no provision of data relevant to the entire system, such as a list of equipment.
2. The number of possible interconnections increases exponentially, causing addressing, performance, and licensing issues.
3. It becomes harder to maintain uniform implementations of the common communication protocol throughout the system, so applications may have to talk differently with different equipment.
4. It becomes harder to add new features into all the front-ends, such as authentication or archiving mechanisms.
5. It is difficult to implement a function that would involve a number of distributed front-ends at the same time, such as a transaction service.

In this document the term *Central Service* refers to a system supplying the common needs of both high-level applications and front-ends, and directly accessible through some sort of external calls. For each service we describe functional requirements, as well as essential features of its external interface. The latter should be applied to an API used by the clients, which include high-level user applications, front-ends, and other central services.

No.	Requirement	Source	Priority
CXR-CS-10	Each central service shall include a graphical user interface (GUI) for remote configuration and monitoring.	A. Petrov 12-2007	Expected
CXR-CS-20	Each central service shall report its internal status through the Data Acquisition Service [4.2].	A. Petrov 12-2007	Expected
CXR-CS-25	The central services shall be scalable downward to a portable minimalistic version capable to provide basic functionality for an autonomous control system.	A. Petrov 01-2008	Desired

4.1 Naming Service

Many classes of objects inside control systems are traditionally addressed by name. These can be front-ends, devices, events, datalogging channels, and error descriptors. Normally, the information about every entity is stored in a relational database, so it can be easily retrieved by applications. In practice, however, this can be affected by a number of issues:

- A limited amount of available database connections.
- The lack of direct connectivity due to security constraints.
- The need to know an exact structure of the database records for every class of objects, or an algorithm to extract the records from a legacy system.
- The data about a single class of objects can be stored in several separate databases.

A central Naming Service provides the clients with a set of characteristics for every named entry. For certain classes of objects, an implementation of a new data repository may not be necessary, because this information can already be retrieved from existing legacy systems. Regardless of how the data is actually stored, the Naming Service provides a uniform mechanism to lookup entries; browse, search, and modify the data.

No.	**Requirement**	**Source**	**Priority**
CXR-CS-30	The control system shall implement a central Naming Service providing arbitrary descriptive information about various classes of objects in the control system through a commonly accepted protocol.	A. Petrov 12-2007	Critical
CXR-CS-40	All entries shall be logically arranged in a tree.	A. Petrov 12-2007	Critical
CXR-CS-50	Each entry and intermediate node shall have a unique name.	A. Petrov 12-2007	Critical
CXR-CS-55	It shall be possible to create aliases (symbolic links) pointing to existing entries.	A. Petrov 01-2008	Expected
CXR-CS-60	Deleted entries shall be marked with a special attribute and preserved for future reference.	A. Petrov 01-2008	Critical
CXR-CS-70	For names of entries and intermediate nodes, the Naming Service shall mandate the character set, delimiter and escape characters, and a definition of uniqueness.	A. Petrov 12-2007	Critical
CXR-CS-80	At a minimum, each entry shall consist of a set of named characteristics.	A. Petrov 12-2007	Expected
CXR-CS-90	For characteristic names, the Naming Service shall mandate the character set.	A. Petrov 12-2007	Critical
CXR-CS-100	The characteristic values shall be type-safe.	A. Petrov 12-2007	Desired
CXR-CS-110	It shall be possible to specify a set of required and optional characteristics for different classes of objects.	A. Petrov 12-2007	Desired

CXR-CS-120	The Naming Service shall support, but not mandate, client authentication.	A. Petrov 12-2007	Critical
CXR-CS-130	It shall be possible to specify read and write permissions for a principal on a class of objects, and default permissions on all classes that are not the subject of explicit access control.	A. Petrov 12-2007	Expected
CXR-CS-140	Existing entries shall not be altered or reused in a way that invalidates the data stored in dataloggers and other long-term archives.	T. Bolshakov W. Marsh 01-2008	Expected
CXR-CS-150	The Naming Service shall mandate an entry serialization protocol.	A. Petrov 12-2007	Critical
CXR-CS-160	A client shall be able to retrieve any entry by name.	A. Petrov 12-2007	Critical
CXR-CS-170	A client shall be able to retrieve the set of children of any intermediate node.	A. Petrov 12-2007	Critical
CXR-CS-180	A client shall be able to search entries in a subtree by name, by presence of characteristics, and by characteristic values.	A. Petrov 12-2007	Critical
CXR-CS-190	A client shall be able to create new entries, modify characteristics of existing entries, and delete existing entries.	A. Petrov 12-2007	Critical

4.2 Data Acquisition Service

The purpose of a central Data Acquisition Service (DAQ) is to mediate in the real-time communication of data between the front-ends and the clients.

The front-ends provide data through individually addressable *devices*. Devices can be combined in groups to represent a machine, a subsystem, or a front-end. The *events* are generated by the control's timing infrastructure (XCLK, 3.1.1]), device state transitions, wall clock, and software. As the requirements do not prescribe any concrete model for devices and events, these terms here are provisional.

It is expected that the Data Acquisition Service may use different protocols to communicate with front-ends and with high-level applications, because the requirements (speed, security) on these levels may vary. However, it is desirable to use same data formats on both sides, so that the data samples [4.2.1] could come through the central layer transparently in either direction.

No.	**Requirement**	**Source**	**Priority**
CXR-CS-200	The control system shall implement a central Data Acquisition Service (DAQ).	A. Petrov 12-2007	Critical

CXR-CS-210	The DAQ shall provide data through a single Data Acquisition API (DAQ API). The DAQ API shall specify a device model, an event model, and data abstractions.	A. Petrov 12-2007	Critical
CXR-CS-220	Each device in the control system shall be uniquely identified by its name.*	A. Petrov 12-2007	Critical
CXR-CS-230	The definition of devices shall be provided through the Naming Service.	A. Petrov 01-2008	Expected
CXR-CS-240	It shall be possible to express each event as a text string containing the event type, a unique event identifier within the type, as a set of parameters.	A. Petrov 12-2007	Critical
CXR-CS-250	The definition of XCLK events and software events shall be provided through the Naming Service.	A. Petrov 01-2008	Expected
CXR-CS-260	A client shall be able to acquire data from a device in one blocking operation.	A. Petrov 12-2007	Critical
CXR-CS-270	A client shall be able to acquire data from a device on an event in one blocking operation. If the event did not occur in a certain time frame, the call shall abort.	A. Petrov 12-2007	Expected
CXR-CS-280	A client shall be able to monitor a group of devices. A change of any device in the group shall cause a callback containing the new data.	A. Petrov 12-2007	Expected
CXR-CS-290	A client shall be able to monitor a group of devices on an event. The event shall cause a callback containing a snapshot of all the devices' data.	A. Petrov 12-2007	Critical
CXR-CS-300	A client shall be able to limit the maximum rate of the data returned through callbacks.	A. Petrov 01-2008	Expected
CXR-CS-310	A client shall be able to set data to a device in one blocking operation.	A. Petrov 12-2007	Critical
CXR-CS-320	A client shall be able to generate software events.	A. Petrov 01-2008	Critical
CXR-CS-330	A client shall be able to monitor a group of events. Each event occurrence shall cause a callback	A. Petrov 01-2008	Expected

* A device may also have one or multiple aliases, per CXR-CS-55.

	containing the event identifier and a timestamp.		
CXR-CS-340	A client shall be able to set data synchronously to a group of devices using distributed transactions [CXR-LL-580].	A. Petrov 01-2008	Desired

4.2.1 Data Abstractions

In this document, an elementary quantum of data handled by the Data Acquisition Service is called *a sample*. Each sample represents an atomic reading from (or setting to) a front-end. At runtime, samples exist as programmatic objects or structures specified in DAQ API. Whereas a particular format of these artefacts is not important for the requirements, it is essential that every sample can be rendered in three tangible forms:

- **Visual Representation,** a text string or an image suitable for human users.
- **Transport Format,** used to send the sample over network.
- **Persistent Format,** used to store the sample for a prolonged time.

The visualization function is specific to every class of objects. It needs to provide, perhaps, only a one-way transformation. The serialization mechanisms for transport and persistent formats shall be defined in the scope of the entire system, and have to work in both directions.

A sample consists of three parts: *a timestamp*, *a payload*, and *a status.* The Data Acquisition Service may parse and use the sample's timestamp and status. The payload is always application-specific and needs to be passed between front-ends and user applications unchanged.

The sample's payload is a collection of values and attributes. Each value represents the result of an actual measurement (for example, an ADC output) or a calculation, which can change over time. Multiple values in a single sample are allowed because some phenomena or processes have to be observed simultaneously at several points. The attributes provide descriptive information either about individual values (for example, a units text) or about the entire sample. Properties of the sample's data source, such as a device name, should not be included. The attributes are close to being constant, but they may occasionally change if the system is reconfigured. This fact must be considered in the design of the transport and persistent formats. For example, when samples are sent over network, some static information can be omitted to save bandwidth, providing that later on this data will be restored by other means. But when samples are archived, all their attributes shall be retained.

The DAQ API can provide multiple implementations of generic data types used as a payload, including a simple container holding one attributed value, an array of multiple attributed values, and more complex hierarchical formations. Also, the system can offer more specific structures for certain groups of applications, such as Wire Scanners. The clients must be able to properly interpret all applicable data types.

The sample's timestamp identifies a time when the measurement of the sample occurred. This can include the absolute calendar time, as well as time relative to certain system events (e.g., beam cycle).

The sample's status field [CXR-LL-415] indicates whether the reading operation was successful. It may also include additional information about conditions of the data acquisition (e.g., if the device has not responded properly) useful for a client .

For efficiency or logical harmony the control system may combine individual samples into *time sequential groups (TSG)*. Each TSG represents a temporal sequence of data as a whole. For example, a fast transient process recorded by a digital oscilloscope can be received on the client as a single TSG, rather than a long stream of individual readings. The TSG itself does not hold a timestamp and a status field.

No.	Requirement	Source	Priority
CXR-CS-400	Each distinct data sample handled by the Data Acquisition Service shall include a timestamp, a payload, and a status field.	A. Petrov, R. Neswold 01-2008	Critical
CXR-CS-410	The timestamp shall include absolute Coordinated Universal Time (UTC) of the sample.	A. Petrov 01-2008	Critical
CXR-CS-415	It shall be possible to include additional references in the timestamp, expressed as an amount of time passed since a certain instance of an event (using per-event counters [CXR-LL-90]).	A. Petrov 01-2008	Desired
CXR-CS-420	The DAQ shall be able to acquire samples in time sequential groups (TSG).	A. Petrov, B. Hendricks 01-2008	Critical
CXR-CS-430	A client shall be able to limit the maximum number of samples returned from a device in one call.	A. Petrov 01-2008	Expected
CXR-CS-440	The DAQ shall mandate serialization protocols for DAQ samples and TSG sent over network.	A. Petrov 12-2007	Critical
CXR-CS-450	The DAQ shall mandate serialization protocols for DAQ samples and TSG stored for prolonged time.	A. Petrov 12-2007	Critical
CXR-CS-460	The DAQ sample's payload shall be a collection of values and attributes. The values represent the results of actual measurements. The attributes contain descriptive information about individual values and the entire sample.	A. Petrov 12-2007	Expected
CXR-CS-470	The shall be a description of each device's data structures.	A. Petrov B. Hendricks 01-2008	Desired
CXR-CS-490	The DAQ shall not interpret or modify samples' payloads.*	A. Petrov 12-2007	Expected
CXR-CS-500	The DAQ API shall provide	A. Petrov	Critical

* Note that the Data Logging Service may interpret and alter the payload when the data is returned to clients; see [4.4.1].

	implementations of generic data types.	12-2007	
CXR-CS-510	The DAQ API shall provide implementations of data types, specific to common domains of applications.	A. Petrov 12-2007	Desired
CXR-CS-520	For each data type, the DAQ API shall support a mechanism of conversion to a human-readable format.	A. Petrov 12-2007	Critical
CXR-CS-530	The DAQ API shall explicitly define the use of special values, such as *Null*, *Infinity*, and *NaN* in samples, timestamps, payloads, and status fields.	A. Petrov 01-2008	Expected

4.2.2 Acquisition of Alarms

Alarms are a special kind of data generated by the front-ends. Each alarm indicates an abnormal condition in the equipment, for example when device data is out of band. The Data Acquisition Service shall acquire alarms from the front-ends in the same way it acquires and processes any other data, with a few exceptions stated below.

No.	**Requirement**	**Source**	**Priority**
CXR-CS-540	The DAQ API shall provide a data abstraction for alarms. At a minimum, an alarm abstraction shall include an alarm state indicator and a bypass state indicator.	A. Petrov 12-2007	Critical
CXR-CS-550	Each alarm in the control system shall be uniquely identified by its name.	A. Petrov 12-2007	Critical
CXR-CS-560	The definition of alarms shall be provided through the Naming Service.	A. Petrov 01-2008	Expected
CXR-CS-570	A client shall be able to change the bypass state of an alarm device in one blocking operation [CXR-LL-730].	A. Petrov 12-2007	Critical
CXR-CS-580	Changes of the bypass state of an alarm device shall not trigger the monitors set by clients.	A. Petrov 12-2007	Critical
CXR-CS-590	If an alarm device is "bypassed", changes of its alarm state shall not trigger the monitors set by clients.	A. Petrov 12-2007	Critical

4.2.3 Front-End Façade

It is hard to achieve a complete unification of low-level equipment in a real-world control system. Most likely, the entire collection of front-ends will not fully support one common set of functions and will not be able to speak exactly one communication protocol. The higher level components have to consider the equipment diversity and address it accordingly. As it would be too complex to design and too burdensome to maintain a large number of end-user applications able to talk individually to every piece of equipment, the front-end façade service takes over the responsibility to cast all front-end communications into one unified form.

No.	Requirement	Source	Priority
CXR-CS-600	The Data Acquisition Service shall support multiple types of front-ends and multiple protocols.	A. Petrov 12-2007	Critical
CXR-CS-610	The Data Acquisition Service shall support pluggable adapters to particular front-end types, linked dynamically at runtime.	A. Petrov 12-2007	Expected
CXR-CS-620	The semantic of each DAQ API call shall be constant, unrelated to the type of front-end targeted.	A. Petrov 12-2007	Critical
CXR-CS-630	The Data Acquisition Service shall emulate the features of DAQ API missing from particular front-end implementations.	A. Petrov 12-2007	Critical

4.2.4 Data Redirection

Normally, a control system operates with devices connected to the real equipment front-ends. The function of data redirection replaces, transparently to clients, the entire set of "real" devices with a programmatically created one. There is a number of useful models that can be employed in data redirection, for example:

- **Setting Mirror** —a snapshot of data represented in a memory model, that can be updated without addressing actual equipment.
- **Retrospection** —a playback of data from an archive.
- **Physical Model** —emulates behaviour of the machine according to some theoretical principles.

No.	Requirement	Source	Priority
CXR-CS-650	The Data Acquisition Service shall support data redirection.	A. Petrov 12-2007	Critical
CXR-CS-660	The Data Acquisition Service shall support pluggable implementations of redirection models, linked dynamically at runtime.	A. Petrov 12-2007	Expected
CXR-CS-670	The semantic of each DAQ API call shall be constant, unrelated to the	A. Petrov 12-2007	Critical

	presence and the model of redirection.		
CXR-CS-680	The common GUI shall include a visual indicator of the data redirection status.	A. Petrov 12-2007	Expected
CXR-CS-690	It shall be possible to operate data redirection from both the application and the central service.	A. Petrov 12-2007	Desired

4.2.5 Interoperability

Several issues arise from the fact that the entire Data Acquisition Service may consist of multiple programming instances distributed throughout the control system.

No.	Requirement	Source	Priority
CXR-CS-700	Remote links inside the DAQ shall be transparent. The control system shall attempt to recover lost connections and restore internal states of the peers.	A. Petrov 12-2007	Critical
CXR-CS-710	The DAQ shall automatically adjust to the actual number of running instances.	A. Petrov 12-2007	Critical
CXR-CS-720	It shall be possible to reconfigure the control system without a restart.	A. Petrov 12-2007	Expected
CXR-CS-730	Each DAQ instance shall provide an equal set of functions for the clients, including access to the entire set of devices.	A. Petrov 12-2007	Expected
CXR-CS-740	A client shall not be required to choose which DAQ instance to connect.	A. Petrov 12-2007	Critical
CXR-CS-750	The control system shall remain operational if multiple nodes fail.	A. Petrov 12-2007	Expected

4.2.6 Data Consolidation

The volume of user requests to the low-level equipment in a control system is unpredictable. In different cycles some data becomes more important for the machine operation than others, thus attracting more traffic to the front-ends providing it. As some front-ends (or the network they are connected to) may not be capable of sustaining the high load, they can start returning data with delays or deny the service for some clients altogether. Data Consolidation shields the equipment from the excessive traffic by acquiring data from each front-end and caching it internally. All requests from the user applications and other central services are managed by DAQ data pools and do not address front-ends directly.

No.	Requirement	Source	Priority

CXR-CS-760	The Data Acquisition Service shall support data consolidation.	A. Petrov 12-2007	Critical
CXR-CS-770	The consolidation assignments among multiple DAQ nodes shall be defined dynamically.	A. Petrov 12-2007	Critical

4.2.7 Resource Locking

No.	Requirement	Source	Priority
CXR-CS-800	A client shall be able to set a lock on any device or a group of devices in the control system, in order to prevent settings from other clients.	A. Petrov 01-2008	Expected
CXR-CS-810	Each lock shall be associated with an authenticated client's principal.	A. Petrov 01-2008	Expected
CXR-CS-820	It shall be possible to specify an event (including a wall clock event), which will automatically remove the lock.	A. Petrov 01-2008	Desired
CXR-CS-830	A client shall be able to remove his own locks.	A. Petrov 01-2008	Expected
CXR-CS-840	A client with certain access privileges shall be able to remove any lock in the control system.	A. Petrov 01-2008	Expected
CXR-CS-850	A client shall be able to check whether a device or a group of devices is locked, and who owns the locks.	A. Petrov 01-2008	Expected
CXR-CS-860	If settings to a device are prohibited due to a lock, the client shall receive a clear error message referring to the lock owner.	A. Petrov 01-2008	Expected

4.2.8 Access Control and Audit

No.	Requirement	Source	Priority
CXR-CS-900	The Data Acquisition Service shall provide access control and support, but not mandate, client authentication.	A. Petrov 12-2007	Critical
CXR-CS-910	It shall be possible to specify read, set, and lock permissions for a principal on a single device, and on a group of devices. It shall be possible to specify default read, set, and lock permissions for a principal on all devices that are not the subject to explicit access control.	A. Petrov 12-2007	Expected
CXR-CS-915	It shall be possible to specify a permission for a principal to remove	A. Petrov 01-2008	Expected

	any lock in the control system.		
CXR-CS-920	It shall be possible to specify permissions for a principal to generate an event, or a group of events. It shall be possible to specify default event permissions for a principal on all events that are not the subject to explicit access control.	A. Petrov 01-2008	Expected
CXR-CS-930	The Data Acquisition Service shall include a mechanism to disable settings from a client (*a setting lock*). It shall be possible to operate this mechanism from both an application and a central service.	A. Petrov 12-2007	Critical
CXR-CS-940	The common GUI shall include a visual indicator of the setting lock status.	A. Petrov 12-2007	Expected
CXR-CS-950	The Data Acquisition Service shall keep a log of connected clients.	A. Petrov 12-2007	Critical
CXR-CS-970	The Data Acquisition Service shall be able to identify potential misuse of the system by the clients, and keep a log of these events.	A. Petrov 12-2007	Expected

Note: Settings log requirement a given in CXR-LL-477.

4.3 Alarm Management Service

Alarm Management Service provides additional functions for aggregation and distribution of alarms in the control system.

No.	**Requirement**	**Source**	**Priority**
CXR-CS-999	The control system shall implement a central Alarm Management Service.	A. Petrov 01-2008	Critical
CXR-CS-1000	The Alarm Management Service shall implement a mechanism which combines multiple front-end alarms in a single aggregated alarm, using boolean expressions.	A. Petrov 01-2008	Critical
CXR-CS-1010	DAQ API shall operate with aggregated alarms in the same way it works with front-end alarms.	A. Petrov 01-2008	Expected
CXR-CS-1020	The Alarm Management Service shall provide a GUI for configuration of aggregated alarms.	A. Petrov 01-2008	Critical
CXR-CS-1030	The Alarm Management Service shall provide a mechanism for notifying users about alarms via	A. Petrov 01-2008	Expected

	email.		
CXR-CS-1040	The Alarm Management Service shall support integration with notification mechanisms other than email.	A. Petrov 01-2008	Desired
CXR-CS-1050	The Alarm Management Service shall implement access control and mandate client authentication.	A. Petrov 01-2008	Critical
CXR-CS-1052	The Alarm Management Service shall keep a log of alarm.	A. Petrov 01-2008	Expected
CXR-CS-1054	For redundancy, the Alarm Management Service shall run in parallel on several nodes.	K. Krause 01-2008	Critical
CXR-CS-1056	The Alarm Management Service shall remain operation if multiple nodes fail.	A. Petrov 01-2008	Expected
CXR-CS-1058	The front-ends shall be able to push alarm to the Alarm Management Service.	J. Patrick 01-2008	Expected

4.4 Data Logging Service

No.	**Requirement**	**Source**	**Priority**
CXR-CS-1060	The control system shall implement a central Data Logging Service, which includes a number of multi-channel dataloggers.	A. Petrov 01-2008	Critical
CXR-CS-1070	Each datalogging channel shall store data from a single device, acquired on instances of a single event.	A. Petrov 01-2008	Critical
CXR-CS-1080	The configuration of dataloggers and channels shall be provided through the Naming Service.	A. Petrov 01-2008	Expected
CXR-CS-1090	Each datalogger shall be uniquely identified by its name.	A. Petrov 01-2008	Critical
CXR-CS-1100	Each channel within a datalogger shall be uniquely identified by the device and the event.	A. Petrov 01-2008	Critical
CXR-CS-1110	It shall be possible to assign an optional description and alias to a channel. Each alias shall be unique within the current datalogger.	A. Petrov 01-2008	Expected
CXR-CS-1130	The Data Logging Service shall support *direct* dataloggers,	A. Petrov 01-2008	Critical

	acquiring data from the Data Acquisition Service.		
CXR-CS-1140	The Data Logging Service shall support *backup* dataloggers, acquiring data from other datalogging channels.	A. Petrov 01-2008	Critical
CXR-CS-1150	The Data Logging Service shall support *client* dataloggers, accepting data from arbitrary external processes.	A. Petrov 01-2008	Critical
CXR-CS-1160	The Data Logging Service shall mandate authorization of processes submitting data to the client dataloggers.	A. Petrov 01-2008	Expected
CXR-CS-1170	The Data Logging Service shall log device access errors in the same way it logs successful samples.	J. Patrick 01-2008	Expected
CXR-CS-1180	The Data Logging Service shall provide data through a single Data Logging API (DL API). DL API shall reuse the device model, the event model, and the data abstractions from DAQ API.	A. Petrov 01-2008	Critical
CXR-CS-1190	A client shall be able to acquire data from a datalogging channel in one blocking operation.	A. Petrov 01-2008	Critical
CXR-CS-1200	A client shall be able to acquire data from a datalogging channel through asynchronous callbacks.	T. Bolshakov 01-2008	Expected
CXR-CS-1210	The Data Logging Service shall output data in TSG*: ● If the channel acquires TSGs, it shall return the same TSGs to clients. ● If the channel acquires discrete samples, it shall pack them into TSGs.	A. Petrov 01-2008	Expected
CXR-CS-1220	A client shall be able to limit the maximum number of samples returned from a channel in one call.	A. Petrov 01-2008	Expected
CXR-CS-1230	A client shall be able to start/stop data acquisition in direct and backup dataloggers, and enable/disable client dataloggers.	A. Petrov 01-2008	Critical
CXR-CS-1240	For a given device, a client shall be able to obtain the list of associated	A. Petrov 01-2008	Critical

* Time sequential group, see [4.2.1].

	direct and backup datalogging channels.		
CXR-CS-1250	For a given datalogging channel, a client shall be able to obtain the associated device and event, and a timestamp of the oldest data sample.	A. Petrov 01-2008	Critical
CXR-CS-1260	The Data Logging Service shall implement access control and support, but not mandate, client authentication.	A. Petrov 01-2008	Critical
CXR-CS-1270	It shall be possible to specify reading and operational (start/stop) permissions on a datalogger, and default permissions an all dataloggers that are not the subject of explicit access control.	A. Petrov 01-2008	Expected

4.4.1 Transformations of Archived Data

No.	**Requirement**	**Source**	**Priority**
CXR-CS-1280	The data logging system shall not alter the data stored in the dataloggers.	A. Petrov 01-2008	Expected
CXR-CS-1290	The data logging system shall provide a mechanism to correct data returned from the dataloggers, in order to offset past measurement inaccuracies and replace obsolete data formats.	K. Cahill 01-2008	Critical
CXR-CS-1300	The data logging system shall be capable to store a historical list of correction functions for each device.	T. Bolshakov 01-2008	Critical
CXR-CS-1310	A client shall be able to disable the correction function.	T. Bolshakov 01-2008	Critical
CXR-CS-1320	A client shall be able to specify a filter to be applied to the data acquired from a dataloggers. (For example, if a datalogger stores arrays of data or some complex data structures, a client may want to extract only one element rather than the entire structure).	T. Bolshakov 01-2008	Desired

4.5 Hierarchical Logging Service

The Hierarchical Logging Service (also known as Sequenced Data Acquisition, SDA) is a multi-stage event-driven datalogging system, which records only the data that are essential on a particular stages of machine operation. A generalized model of hierarchical datalogging is shown on Fig. 1.

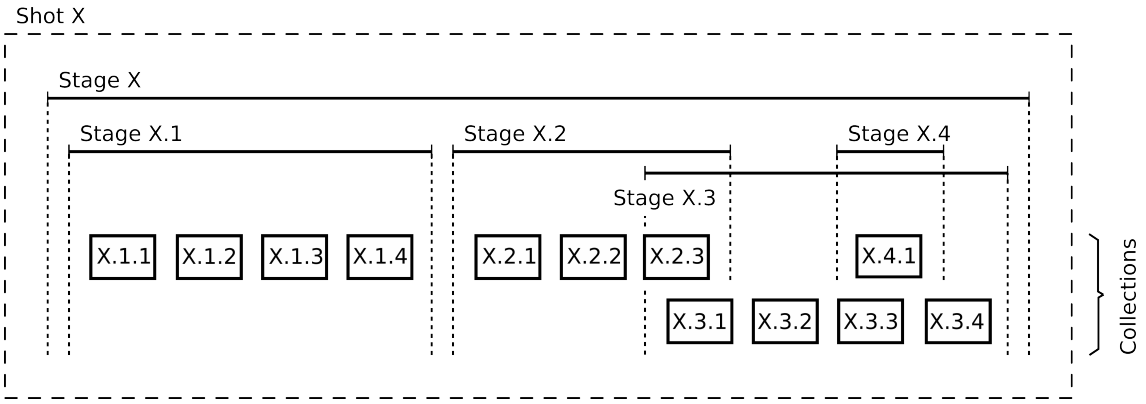


Fig. 1: Model of Hierarchical Datalogging

The hierarchical datalogging operates with *stages*. Each stage is associated with a number of events. The events specify conditions upon which a particular stage starts and finishes. The stages are arranged in a trees. When a stage starts, it enables all stages on the next level, so they can start too when appropriate events will occur. When a stage finishes, it disables all the underlying stages, possibly terminating those that are still running. Stages on one level can run in parallel.

The stages on the bottom of the hierarchy are called *collections*. Each collection is an equivalent of a multi-channel datalogger [4.4]. Yet, the collections usually store much less data and may have different implementations.

The definition of every single top-level stage, along with all stages and collections lying underneath it, constitute a *shot*. When the top-level stage of a shot is triggered, the system starts moving through a sequence of events, and can ultimately reach collections that will record the necessary data. All stage instances (i.e., stages that have already "happened" in the past) and collection instances are individually addressable from the top to the bottom of the hierarchies.

This requirements do not constrain an implementation of hierarchical dataloggers by prescribing one invariable model. It is assumed that the control system may have more than one type of hierarchical logging service, which utilize different mechanisms of storing and accessing the data, and different state transition rules. In particular, there may be a separate way of dealing with small frequent shots (also known as *pulses*).

No.	Requirement	Source	Priority
CXR-CS-1330	The system shall implement a central Hierarchical Logging Service.	A. Petrov 01-2008	Critical
CXR-CS-1340	Each shot shall consist of a tree, which includes stages and collections.	T. Bolshakov 01-2008	Expected

CXR-CS-1350	The number of level for each shot shall be constant.	T. Bolshakov 01-2008	Expected
CXR-CS-1360	The stages and collections shall be started and stopped on events. If a stage is running, it enables all stages and collections underneath.	T. Bolshakov 01-2008	Critical
CXR-CS-1370	Each collection shall be a functional equivalent of a multi-channel datalogger [4.4].	A. Petrov 01-2008	Expected
CXR-CS-1380	Each stage and collection shall have an index, unique on the current level within the shot.	T. Bolshakov 01-2008	Critical
CXR-CS-1390	It shall be possible to assign an optional description and alias to each stage or collection. The alias shall be unique on the current level within the shot.	T. Bolshakov 01-2008	Expected
CXR-CS-1400	The definition of shots and a list of available collections shall be provided through the Naming Service.	A. Petrov 01-2008	Expected
CXR-CS-1410	The Hierarchical Logging Service shall implement access control and support, but not mandate, client authentication.	A. Petrov 01-2008	Critical
CXR-CS-1420	It shall be possible to specify permissions to reconfigure shots, permissions to read data from collections that belong to a particular shot, and default reading permissions on all shots that are not the subject of explicit access control.	A. Petrov 01-2008	Expected

4.6 Postmortem Logging Service

In this section, the term *postmortem logging* refers to the acquisition of data immediately preceding a failure of the system in a large scale (e.g., accelerator quench). It does not cover various specialized postmortem tools, such as operating systems' core dumps or front-end debugging facilities.

No.	**Requirement**	**Source**	**Priority**
CXR-CS-1430	The control system shall implement a central Postmortem Logging Service, which orchestrate the collection and distribution of postmortem data.	A. Petrov 01-2008	Critical

CXR-CS-1435	The Postmortem Logging Service shall use a "snapshot on event" function [CXR-LL-530] on front-ends. (A snapshots may include the data before the event, or the data before and after the event).	A. Petrov C. Briegel 01-2008	Expected
CXR-CS-1440	The Postmortem Logging Service shall emulate the "snapshot on event" function, if it is missing in a front-end.	A. Petrov 01-2008	Expected
CXR-CS-1445	The Postmortem Logging Service shall be responsible for the snapshot configuration in front-ends.	A. Petrov 01-2008	Expected
CXR-CS-1450	Upon a critical event, the Postmortem Logging Service shall collect data from front-ends and store it in a central datalogger.	K. Cahill 01-2008	Critical
CXR-CS-1460	Upon collection of postmortem data, the Postmortem Logging Service shall trigger an event to notify the clients.	K. Cahill 01-2008	Critical
CXR-CS-1470	Each postmortem logging channel shall be uniquely identified by the associated device.	A. Petrov 01-2008	Critical
CXR-CS-1480	The configuration of each channel shall be provided through the Naming Service.	A. Petrov 01-2008	Expected
CXR-CS-1490	A client shall be able to read postmortem data via DL API [4.4].	A. Petrov 01-2008	Expected

4.7 Save And Restore Service

No.	Requirement	Source	Priority
CXR-CS-1500	The control system shall implement a central Save And Restore Service.	A. Petrov 01-2008	Critical
CXR-CS-1510	Each Save And Restore archive shall store snapshots of data samples acquired synchronously from a group of devices.	A. Petrov 01-2008	Critical
CXR-CS-1520	The configuration of archives and the list of available snapshots shall be provided through the Naming Service.	A. Petrov 01-2008	Expected
CXR-CS-1530	Each archive shall be uniquely identified by its name.	A. Petrov 01-2008	Critical
CXR-CS-1540	Each snapshot within an archive shall	A. Petrov	Critical

	be uniquely identified by the timestamp.	01-2008	
CXR-CS-1550	It shall be possible to assign an optional description and alias to a snapshot. The alias shall be unique within the current archive.	A. Petrov 01-2008	Expected
CXR-CS-1560	A client shall be able to obtain the list of available snapshots in an archive.	A. Petrov 01-2008	Critical
CXR-CS-1570	A client shall be able to obtain an individual snapshot from an archive.	A. Petrov 01-2008	Critical
CXR-CS-1580	A client shall be able to save a new snapshot into an archive.	A. Petrov 01-2008	Critical
CXR-CS-1590	A client shall be able to restore a previous state of the system by using an archived snapshot.	A. Petrov 01-2008	Critical
CXR-CS-1600	A client shall be able to modify data in the existing snapshots.	K. Cahill 01-2008	Expected
CXR-CS-1610	The modified snapshots shall be clearly marked with a special attribute.	K. Cahill 01-2008	Critical
CXR-CS-1620	The Save And Restore service shall be able to take snapshots automatically on events.	A. Petrov 01-2008	Critical
CXR-CS-1630	The Save And Restore Service shall implement access control and support, but not mandate, client authentication.	A. Petrov 01-2008	Critical
CXR-CS-1640	It shall be possible to specify reading, saving (“make a new snapshot”), restoring (“restore data from a snapshot”) and writing (“modify a snapshot”) permissions on an archive, and default permissions an all archives that are not the subject of explicit access control.	A. Petrov 01-2008	Expected

5 Application Infrastructure

This section discusses different types of application used in a control system. It also gives requirements on the operational environment, data protocols, and internal functionality common to all applications.

5.1 Types of Applications

There are three types of applications in the control system:

1. Low-Level Applications (described in [3]), which acquire data from physical equipment, front-ends, or other external sources. This section includes one kind of low-level applications, the **Open Access Clients (OAC),** also known as *software front-ends*.

2. High-Level Applications (described in [6]), which provide human-readable data to the end users. The two types of high-level applications are:

- **Custom Applications,** programs designed specifically for use in the current system.
- **Third-Party Applications,** general-purpose programs that integrate with the rest of the system through well-defined protocols or data formats. This includes web browsers, MATLAB, JAS, and other tools routinely used to visualize and process data.

3. Middleware, which mediates the communication between the low-level and the high-level applications and provides additional services to both. The middleware includes:

- **Central Services** (discussed in section 4).
- **Daemons**, periodically or continuously running routines.
- **Server-Side Parts** of multi-tier high-level applications.
- **Web Applications,** providing data via HTTP.*

The applications may run on three types of computer nodes:

- **Client Nodes,** users' PCs, laptops, etc.; as well as shared consoles in control rooms.
  The client nodes can be used to execute only high-level applications.
- **Central Nodes,** servers that are administered and controlled by the Controls Department and accessible by the users via remote tools. The central nodes can be used to execute high-level applications, the middleware, and Open Access Clients. As all server-side programming components are headless†, they shall run under the control of a specialized application environment (an application server), rather than a human being.

* A Web Application is a server-side component for a web browser, but as this technology becomes ubiquitous, Web Applications are viewed as an individual class of applications.

† Without a graphical user interface.

- **Front-End Nodes,** servers controlled by the Control Department and generally not accessible by the users. The front-end nodes can be used to execute only low-level applications. Specific requirements are given in section 3.

No.	Requirement	Source	Priority
CXR-AI-10	The control system shall provide a technical specification for the hard- and software environment sufficient to run high-level applications.	A. Petrov 01-2008	Critical
CXR-AI-20	The control system shall enable high-level applications on the central nodes.	A. Petrov 01-2008	Expected
CXR-AI-25	For high-level applications running on the central nodes, the control system shall provide a means to display their GUI on the client nodes.	A. Petrov 01-2008	Expected
CXR-AI-30	The control system shall include a set of central nodes capable of running high-level applications.	A. Petrov 01-2008	Expected
CXR-AI-35	The control system shall enable a subset of high-level applications on the client nodes.	A. Petrov 01-2008	Expected
CXR-AI-40	The control system shall provide a tool for the users to select and launch high-level applications (see also CXR-HL-201).	A. Petrov 01-2008	Critical
CXR-AI-50	The use of third-party high-level applications shall be approved by a board of experts. Unapproved programs shall not be supported.	C. Schumann 01-2008	Critical
CXR-AI-60	The control system shall provide an application server for the server-side programming components.	A. Petrov 01-2008	Critical
CXR-AI-70	The application server shall be capable of executing Central Services.	A. Petrov 01-2008	Critical
CXR-AI-80	The application server shall be capable of executing Open Access Clients.	A. Petrov 01-2008	Critical
CXR-AI-90	The application server shall be capable of executing deamons, which are continuously or periodically running headless routines.	A. Petrov 01-2008	Expected
CXR-AI-100	The application server shall be capable of executing server-side parts of multi-tier high level	A. Petrov 01-2008	Expected

	applications.		
CXR-AI-110	The application server shall be capable of executing web applications.	A. Petrov 01-2008	Expected
CXR-AI-120	It shall be possible to start, stop, and reconfigure individual components inside the application server without affecting others.	A. Petrov 01-2008	Expected
CXR-AI-130	The application server shall be able to recover the states of all the components upon restart or after spontaneous failures.	A. Petrov 01-2008	Critical
CXR-AI-140	The control system shall support individual configuration of application servers for each central node.	A. Petrov 01-2008	Critical
CXR-AI-150	The application server shall report its internal state through the Data Acquisition Service [4.2].	A. Petrov 01-2008	Expected
CXR-AI-160	The application server shall provide a mechanism to collect and keep application log files.	A. Petrov 01-2008	Expected

5.2 Application Protocols

The application protocols are used by high-level applications to communicate with the middleware and low-level components. They are also used by the server-side components to communicate with each other. The right choice of application protocols can facilitate the development of high-level applications and improve integration with third-party software.

The requirements in this section use a terminology from the *OSI Reference Model*, described at http://en.wikipedia.org/wiki/osi_model.

No.	**Requirement**	**Source**	**Priority**
CXR-AI-170	The application protocols shall be platform and language independent.	A. Petrov 01-2008	Critical
CXR-AI-180	The control system shall specify the protocols that will be supported.	A. Petrov 01-2008	Expected
CXR-AI-190	Each application protocol shall be implemented with a binary presentation layer.	A. Petrov 01-2008	Desired
CXR-AI-200	Each application protocol shall be implemented with a human-readable presentation layer.	A. Petrov 01-2008	Desired
CXR-AI-210	All presentation-layer data formats	A. Petrov	Expected

	shall be explicitly specified.	01-2008	
CXR-AI-230	The inbound connections shall be accepted on well-defined ports.	A. Petrov 01-2008	Expected

5.3 General-Purpose Database

No.	Requirement	Source	Priority
CXR-AI-900	The control system shall provide a general-purpose application database, accessible from, at least, the central nodes.	A. Petrov 01-2008	Critical
CXR-AI-910	The control system shall implement database connection pools. All applications shall obtain database connections from the pools.	A. Petrov 01-2008	Expected
CXR-AI-920	The database shall support, but not mandate, client authentication.	A. Petrov 01-2008	Critical
CXR-AI-930	The database shall mandate authorization of clients modifying the data.	A. Petrov 01-2008	Critical
CXR-AI-940	Common dictionaries of objects shall use the Naming Service [4.1], rather than the application database.	A. Petrov 01-2008	Expected
CXR-AI-950	High-level applications shall not utilize direct database access from the client nodes. To access the database, the client-side applications shall use a two-tier approach, when the DB connection is opened in the middle tier.	A. Petrov 01-2008	Expected
CXR-AI-960	All central services and high-level applications shall be designed in a way that makes them reasonably independent from any particular database implementation (e.g., by using Service Provider Interfaces).	A. Petrov 01-2008	Desired
CXR-AI-970	The control system shall keep a log of application database queries.	K. Cahill 01-2008	Expected

5.4 Security

No.	Requirement	Source	Priority
CXR-AI-240	The control system shall have a written security policy.	A. Petrov 01-2008	Critical

CXR-AI-250	The control system shall implement a central Identity Database, which registers credentials of the principals (users).	A. Petrov 01-2008	Critical
CXR-AI-260	At a minimum, each credential shall include a unique name of the principal, a principal type, an optional description, and a set of assigned roles (named permissions).	A. Petrov 01-2008	Expected
CXR-AI-270	The data in the Identity Database (list of user and roles) shall be available through the Naming Service [4.1].	A. Petrov 01-2008	Expected
CXR-AI-280	The Identity database shall include a graphical user interface (GUI) for remote configuration and monitoring.	A. Petrov 01-2008	Expected
CXR-AI-290	The control system shall support, but not mandate, strong authentication of user principals that satisfies the requirements of the lab security policy.	A. Petrov 01-2008	Critical
CXR-AI-300	The control system shall support a mechanism of Single Sign-On for user principals.	A. Petrov 01-2008	Expected
CXR-AI-310	The control system shall support principals assigned to computer nodes and programming processes, and provide appropriate mechanisms of their authentication.	A. Petrov 01-2008	Expected
CXR-AI-320	The control system shall provide a mechanism of default authentication on shared consoles.	A. Petrov 01-2008	Critical
CXR-AI-325	A client program shall be able to identify itself during authentication (e.g., by providing a high-level application name, OAC name, etc). This information shall be used only in addition to a trusted form of authentication.	T. Zingelman 01-2008	Expected
CXR-AI-330	The control system shall keep a log of all authentication attempts.	A. Petrov 01-2008	Expected
CXR-AI-340	The application protocols [5.2] shall incorporate, in the transport layer, a cryptographic protocol providing client authentication and protection against message tampering.	A. Petrov 01-2008	Desired

CXR-AI-350	Authorization decisions shall be made by the server-side components or front-ends based on roles assigned to the current principal.	A. Petrov 01-2008	Expected
CXR-AI-360	A middleware component shall be able to delegate client credentials to another middleware or low-level component.	A. Petrov 01-2008	Desired

5.5 Application Framework

No.	Requirement	Source	Priority
CXR-AI-370	The control system shall provide a programming framework for standard high-level applications.	A. Petrov 01-2008	Expected
CXR-AI-380	The application framework shall provide a graphical user interface with a common look-and-feel [CXR-HL-100].	A. Petrov 01-2008	Expected
CXR-AI-390	The application framework shall provide an API to the Central Services.	A. Petrov 01-2008	Expected
CXR-AI-400	The application framework shall implement the GUI features required by the Central Services [CXR-CS-680, CXR-CS-690, CXR-CS-940].	A. Petrov 01-2008	Expected
CXR-AI-410	The application framework shall incorporate an API to the application database [5.3], and a database connection pool.	A. Petrov 01-2008	Expected
CXR-AI-420	The application framework shall test the availability of required external resources (Central Services, application database, and server-side components). If a resource is not reachable, the framework shall notify the user and disable the entire application.	A. Petrov 01-2008	Expected
CXR-AI-430	The application framework shall send email notifications to application subscribers should the program terminate abnormally.	C. Schumann 01-2008	Expected
CXR-AI-440	Each application shall have a link to an online help page.	A. Petrov 01-2008	Expected

6 High Level Applications

The high level application programs for Project X will have to serve a diverse community of users. These users have different needs and they can be divided into roughly four groups. Machine operators have the primary responsibility for operating the accelerator. They monitor and control the accelerator complex around the clock. They need both an intuitive interface to deal with the myriad of systems that they encounter as well as more low level, efficient interfaces for systems that they are proficient with. The machine specialists are experts in a particular portion of the complex. They carry out studies to make improvements to their area, and they are often called upon to deal with difficult operational situations. They need flexible, easily configured application environments to carry out their studies, and they have similar needs to the machine operators when they are dealing with operational problems. Equipment specialists are responsible for particular pieces of equipment and need applications which can access detailed diagnostics information about their equipment, and they need reasonable access to general facilities such as datalogging to monitor the performance of the equipment that they are responsible for. The final group of users are the experimenters themselves. They need intuitive applications to monitor accelerator information that may impact their experiment. They may also need some access to control parameters as well. To be successful, the high level application programs must meet all of these various needs.

The requirements in this document are not intended to apply to existing legacy applications. They apply to new custom applications created for Project X.

6.1 User Interface

It is important to provide a user interface that is intuitive and more importantly consistent. It should provide a wide variety of information about a given machine parameter with a minimum of mouse clicks.

No.	Requirement	Source	Priority
CXR-HL-100	All applications shall have a common look and feel. [CXR-AI-380]	W. Kissel 12-2007	Expected
CXR-HL-110	The user interface shall support transferring data between applications in a standard and consistent way, including drag and drop, and copy and paste.	W. Kissel 12-2007	Expected
CXR-HL-120	The user interface shall provide a standard way to select devices (see Naming Service section 4.1)	W. Kissel 12-2007	Expected
CXR-HL-130	The user interface shall support searching for devices by name, front-end node, device type, and any other device database attributes.	B. Hendricks 12-2007	Expected

CXR-HL-140	The user interface shall support launching real time parameter plots for selected devices. [CXR-LL-520]	W. Kissel 12-2007	Expected
CXR-HL-150	The user interface shall support launching datalogger (archived data) plots for selected devices.	W. Kissel 12-2007	Expected
CXR-HL-160	The user interface shall support accessing database information for selected devices.	W. Kissel 12-2007	Expected
CXR-HL-170	The user interface shall support exporting data for selected devices to electronic log books.	W. Kissel 12-2007	Expected
CXR-HL-180	The user interface shall support launching applications and opening them with any selected devices.	B. Hendricks 12-2007	Expected
CXR-HL-190	The user interface shall support accessing control system error help.	B. Hendricks 12-2007	Expected
CXR-HL-195	The user interface shall support accessing information about selected front-end nodes.	B. Hendricks 12-2007	Expected

6.2 Applications

The user applications for Project X should support the needs of both new employees and visiting workers from other institutions who need a humanly understandable view of the accelerator complex. They should also support the needs of veteran power users who want to get their work done as quickly and efficiently as possible.

It is also desirable that applications not duplicate one another in functionality. If functionality needed by one application is fulfilled by another, the first program should call the other one to achieve the desired functionality.

The requirements below list only fundamental attributes of each application. Before implementation, a detailed requirement for each application should be created.

No.	**Requirement**	**Source**	**Priority**
CXR-HL-200	There shall be a synoptic display application which will allow users to investigate the accelerator in a graphical way.	B. Hendricks 12-2007	Critical
CXR-HL-210	The synoptic display application shall support the launching of other applications.	B. Hendricks 12-2007	Critical
CXR-HL-220	The synoptic display application shall be editable by end users employing standard drag and drop features.	B. Hendricks 12-2007	Expected

CXR-HL-230	There shall be a master menu application which will support launching any other application program.	B. Hendricks 12-2007	Critical
CXR-HL-240	There shall be a parameter plotting application which will support high frequency plotting of data from the accelerator.	B. Hendricks 12-2007	Critical
CXR-HL-250	The plot application shall support zooming functionality.	B. Hendricks 12-2007	Expected
CXR-HL-260	The plot application shall support the plotting of multiple devices simultaneously.	B. Hendricks 12-2007	Critical
CXR-HL-270	The plot application should support the saving and restoring of plot setup files.	B. Hendricks 12-2007	Expected
CXR-HL-280	The plot application shall support a real time strip chart mode.	B. Hendricks 12-2007	Critical
CXR-HL-290	The plot application shall support plots triggered by machine events.	B. Hendricks 12-2007	Critical
CXR-HL-300	There shall be a sequencer application which will support programmable, sequential execution of accelerator operations.	B. Hendricks 12-2007	Critical
CXR-HL-310	The sequencer application sequences shall be programmable by end users perhaps with some restrictions.	B. Hendricks 12-2007	Critical
CXR-HL-320	The sequencer application shall be aware of machine events.	B. Hendricks 12-2007	Critical
CXR-HL-330	The sequencer application shall support individual command execution as well as complete sequences.	B. Hendricks 12-2007	Critical
CXR-HL-340	The sequencer application shall maintain an operational log to allow users to evaluate past execution.	B. Hendricks 12-2007	Critical
CXR-HL-350	There shall be an alarm display application.	B. Hendricks 12-2007	Critical
CXR-HL-360	The alarm display shall support user configuration of which devices are displayed as well as where they appear on the display.	B. Hendricks 12-2007	Expected

CXR-HL-370	The alarm display shall support displaying alarms chronologically.	B. Hendricks 12-2007	Critical
CXR-HL-380	The alarm display shall support displaying alarms by priority.	B. Hendricks 12-2007	Critical
CXR-HL-390	The alarm display shall support the generation of sounds when an alarm is received.	B. Hendricks 12-2007	Critical
CXR-HL-400	The alarm display shall support an option to display all alarms including ones that are not presently mapped.	B. Hendricks 12-2007	Critical
CXR-HL-401	The alarm display shall support displays of current readings and alarm block information.	B.Hendicks 12-2007	Critical
CXR-HL-402	The alarm display shall support modifying alarm block information including disabling alarms and changing limits and masks.	B.Hendricks 12-2007	Critical
CXR-HL-410	There shall be a datalogger display application.	B. Hendricks 12-2007	Critical
CXR-HL-420	The datalogger display application shall support plotting multiple parameters on the same grid.	B. Hendricks 12-2007	Critical
CXR-HL-430	The datalogger display application shall support selection of archival data event parameters.	B. Hendricks 12-2007	Critical
CXR-HL-440	The datalogger display application shall support the exporting of data in ASCII and popular spreadsheet compatible formats.	B. Hendricks 12-2007	Critical
CXR-HL-450	The datalogger display application shall support saving and restoring of recently used plot setups as well as user configured saved setups.	B. Hendricks 12-2007	Expected
CXR-HL-460	There shall be a device configuration database viewer and editor application.	B. Hendricks 12-2007	Critical
CXR-HL-470	There shall be a parameter list application.	B. Hendricks 12-2007	Critical
CXR-HL-480	The parameter list application shall support reading parameter values.	B. Hendricks 12-2007	Critical

CXR-HL-490	The parameter list application shall support setting parameter values.	B. Hendricks 12-2007	Critical
CXR-HL-500	The parameter list application shall be end user configurable.	B. Hendricks 12-2007	Critical
CXR-HL-510	There shall be a parameter save, restore, and compare application.	B. Hendricks 12-2007	Critical
CXR-HL-520	The parameter save, restore, and compare application shall support displaying save file values.	B. Hendricks 12-2007	Critical
CXR-HL-530	The parameter save, restore, and compare application shall support restoring values from a save file.	B. Hendricks 12-2007	Critical
CXR-HL-540	The parameter save, restore, and compare application shall support configuring and making save files.	B. Hendricks 12-2007	Critical
CXR-HL-550	There shall be an overall site status display application.	B. Hendricks 12-2007	Critical
CXR-HL-560	The overall site status display application shall display important parameters and the state of all portions of the accelerator complex.	B. Hendricks 12-2007	Critical
CXR-HL-570	The overall site status display application shall display important machine event information.	B. Hendricks 12-2007	Critical
CXR-HL-580	The overall site status display application shall support the display of operator entered messages.	B. Hendricks 12-2007	Critical
CXR-HL-590	There shall be an application to set the gradients and phases of the RF cavities for the linac.	B. Hendricks 12-2007	Critical
CXR-HL-600	There shall be an application to steer the beam in the linac.	B. Hendricks 12-2007	Critical
CXR-HL-610	There shall be an application to align the linac beam along the magnetic axis.	B. Hendricks 12-2007	Critical
CXR-HL-620	There shall be an application to configure and monitor the machine protection system.	B. Hendricks 12-2007	Critical
CXR-HL-630	There shall be an application to analyze machine protection system trips.	B. Hendricks 12-2007	Critical

CXR-HL-640	There shall be an application to monitor and control the cryogenic systems for the linac.	B. Hendricks 12-2007	Critical
CXR-HL-650	There shall be an application to monitor and control the high level RF systems for the linac.	B. Hendricks 12-2007	Critical
CXR-HL-660	There shall be an application to monitor and configure central services.	B. Hendricks 12-2007	Critical
CXR-HL-670	There shall be an electronic log application which can be used by the operations staff as well as individual machine departments. It shall be web based.	B. Hendricks 12-2007	Critical
CXR-HL-680	There shall be an application to monitor and control the beam position monitor (BPM) system.	B. Hendricks 12-2007	Critical
CXR-HL-690	There shall be an application to monitor and control the beam loss monitor (BLM) system.	B. Hendricks 12-2007	Critical
CXR-HL-700	There shall be an application to monitor and control the vacuum system.	B. Hendricks 12-2007	Critical
CXR-HL-710	There shall be an application that supports the display of correlated parameters from a given beam pulse.	B. Hendricks 12-2007	Expected
CXR-HL-720	There shall be an application to monitor and control beam-based feedback.	J. Patrick 1-2008	Expected
CXR-HL-730	There shall be an application to display hierarchical archived data (SDA)	B. Hendricks 1-2008	Expected
CXR-HL-740	There shall be a parameter scan application.	B. Webber 1-2008	Expected

6.3 Nonstandard Applications

There must be tools for nonprogrammers or those who need to develop something quickly to work around or to study a short term problem. In general, there should be high level tools to empower end users to create application programs. This allows them to create programs on their own schedule. It also allows them to get the exact functionality and interface that they desire.

No.	Requirement	Source	Priority
CXR-HL-800	There shall be a user configurable, drag and drop synoptic display editor.	B. Hendricks 12-2007	Expected

CXR-HL-810	There shall be a scripting language which supports generic access to accelerator parameter values as well as other control system parameters.	B. Hendricks 12-2007	Expected
CXR-HL-820	There shall be support for selected third party mathematical applications to acquire control system data.	B. Hendricks 1-2008	Expected

7 Controls In A Box

As previously stated, the Control System needs to scale to a large user community and a million properties. There is also a need for the control system to be available on a small scale for one user or a small group of users to test a small number of instruments in a relatively informal setting. We refer to this small control system as Controls in a Box.

There are several important advantages to Controls in a Box, the most obvious is that a single user can control an instrument without the overhead of the large and complex control system designed to operate an accelerator complex consisting of 9 km of instrumentation and stringent safety and availability requirements. Controls in a Box has different requirements, it does not have to meet many of the Operation Controls base requirements (safety, availability, legacy constraints) and hence can focus on the needs of the individual such as ease of use, wide and general distribution, and installation.

Operation Controls also benefits by separating the testing of individual instruments from the operation of the accelerator complex. It lessens the load on all resources and avoids interference with operations.

Control in a Box is an excellent way to introduce and familiarize users with the control system, without them having to use the Operations Control resources such as servers and central services. Assuming Controls in a Box is available as a download from the Web, one can install it on the client node (laptop) and learn it by locally controlling an instrument and having complete control over the configuration and resources.

Any modern accelerator will be built in collaboration with national if not international research labs. The off-site labs require remote access and a somewhat autonomous control system for local testing and development. Hardware designers will write controls software to operate their hardware during test and development. If the testing software is written for a different control system than the one ultimately used for its operation, it needs to be ported to the Operation Controls. If off-site hardware designers use Controls in a Box, the integration is straight forward and no porting is necessary. This not only saves the time it takes to rewrite the software, but it also eases communication by using the same terms and language when integrating new instrumentation.

In the requirements below, 'Operation Controls' refers to the large control system used to control the accelerator complex. 'Controls in a Box' refers to the local, small version of the control system.

No.	Requirement	Source	Priority
CXR-CB-10	The control system shall be available as a small scale version (Controls in a Box) intended for local and autonomous use.	S.Gysin 01-2008	Expected
CXR-CB-30	Controls in a Box shall provide locally programmable clock-simulating events.	S.Gysin 01-2008	Expected
CXR-CB-50	Controls in a Box shall include a local	S.Gysin,	Expected

	and autonomous version of selected Central Services. See section 4 requirement CRX-CS-25.	A.Petrov 01-2008	
CXR-CB-60	The Central Services for Control in a Box shall at a minimum include the Naming Service and the Data Acquisition Service. See section 4 for definitions	S.Gysin 01-2008	Expected
CXR-CB-80	Controls in a Box shall be able to store data locally.	S.Gysin 01-2008	Expected
CXR-CB-100	Controls in a Box shall be free of dependencies on proprietary software.	J. Patrick 01-2008	Expected
CXR-CB-110	Controls in a Box shall be available and simple to down load from the Web.	S.Gysin 01-2008	Desired
CXR-CB-120	Controls in a Box shall be Open Source product.	A. Petrov 01-2008	Desired
CXR-CB-130	Controls in a Box shall have an install script	S.Gysin 01-2008	Desired
CXR-CB-130	Controls in a Box shall be easy to configure by offering the user a configuration wizard or a text file to edit.	J. Patrick 01-2008	Desired
CXR-CB-140	Controls in a Box shall be supported on Linux Operating Systems	S.Gysin 01-2008	Expected
CXR-CB-145	Controls in a Box shall be supported on Windows Operating Systems	S.Gysin 01-2008	Desired
CXR-CB-150	Controls in a Box shall provide a user interface to read and set values	S.Gysin 01-2008	Desired
CXR-CB-160	Controls in a Box shall provide a user interface to export data.	S.Gysin 01-2008	Desired

8 Beam-Based Feedback

In order to keep beam parameters such as position, energy, and intensity stable through the linac and transfer line, and minimize losses, fast feedback loops will be needed. This section addresses requirements for a feedback infrastructure that works across major (5Hz) accelerator pulses. It is assumed that intra-pulse feedback/feed-forward will be handled by low level hardware. Inter-pulse feedback requires obtaining data from sensors such as beam position monitors, performing calculations, and writing settings to such devices as corrector magnets prior to the next major pulse. The following are requirements for the feedback infrastructure.

No.	Requirement	Source	Priority
CXR-BF-100	There should be a standard infrastructure that supports creating and running feedback loops.	J. Patrick 01-2008	Critical
CXR-BF-110	Data acquisition, calculation, and setting shall all be accomplished between major pulses.	J. Patrick 12-2007	Critical
CXR- BF-120	It shall be possible to involve multiple front-ends in a feedback loop.	J. Patrick 12-2007	Critical
CXR- BF-130	It shall be possible to conveniently enable and disable loops.	J. Patrick 12-2007	Critical
CXR- BF-140	It shall be possible to put a lock on a disabled loop to prevent it from being re-enabled by another application or operator.	J. Patrick 01-2008	Critical
CXR- BF-150	It shall be possible to easily mask sensors without rebuilding or reloading code.	J. Patrick 12-2007	Critical
CXR- BF-160	It shall be possible to set limits on sensor readings used in calculations.	J. Patrick 12-2007	Critical
CXR- BF-170	It shall be possible to set limits on absolute and relative changes between major pulses.	J. Patrick 12-2007	Desirable
CXR- BF-180	There should be sufficient network bandwidth so that readings are not lost or delayed. Either the network must have sufficient intrinsic bandwidth, or should have Quality of Service capability.	J. Patrick 01-2008	Critical
CXR- BF-190	All sensor devices available for use in calculations shall be available to the general control system.	J. Patrick 12-2007	Critical
CXR- BF-200	Calculated results shall be available to the general control system.	J. Patrick 12-2007	Critical

CXR- BF-210	Settings and the readings used to compute them shall be logged. This information shall include which if any sensors were disabled or excluded from the calculation due to error status or data outside of limits. To reduce the logged data to a manageable volume, it would be acceptable to filter the data.	J. Patrick 12-2007	Critical
CXR- BF-220	A counter indicating the number of cycles performed shall be available to indicate the loop is functioning even if settings are not changing.	J. Patrick 12-2007	Critical

9 Machine Protection System

The 8 GeV superconducting linac under consideration for Project X will have the potential to produce beam pulses capable of damaging the acceleration structures, the beam line vacuum chambers, and machine components in the event of an aberrant accelerator pulse. As a result, a Machine Protection System (MPS) must be considered to mitigate such damage to the system. The MPS should be considered to be the collection of all devices involved in the monitoring and safe delivery of beam to its final destination and not limited to any particular subsystem or diagnostic device. The MPS protects the machine only and is not a personnel protection system.

To deal with different machine settings and operational scenarios, a number of beam modes should be defined by the path that the beam must take and the pulse intensity. This section outlines the controls requirements for a protection system. It does not describe the various modes of operation that will be a requisite part of the protection scheme.

Multiple time scales of interest to the protection system design dictate the required reaction times for various parts of the MPS system. The slow part of the protection scheme will need to have a reaction time of 200 ms determined by the maximum pulse rate of the machine (5 Hz). Fast signals that require switching off the beam immediately within a bunch train must be processed rapidly enough to terminate beam in mid-pulse depending upon the distance from the chopper. Such signals may include RF signals and beam loss.

Several other subsystems will play a critical role with the MPS, namely RF monitoring and quench protection monitoring. Signals derived from both the low level and in some cases higher level RF systems can potentially be used as precursors to failures whose results should be available via the controls system. Time stamped diagnosis of this type of data is desirable**.**

The assumption is that the existing machine protection systems, machine specific beam permit/abort systems, and the Beam Switch Sum Box (BSSB) will be used for legacy portions of the accelerator complex used in Project X.

9.1 General Machine Protection System Requirements

The machine protection system shall have several components including machine and beam-line specific permit systems along with software and hardware switches that inhibit or allow beam to the various destinations. In addition, it is expected that the system will have inputs from the safety system. The MPS requires a control interface that allows for system configuration, system status displays and tools for monitoring and diagnoses.

Since the machine protection system must prevent damage to accelerator structures in the event of a mis-steered beam pulse or a component mishap, the MPS system will therefore control the startup procedure and monitor operations to detect and take immediate action if fault conditions occur. A set of accelerator parameters and settings must therefore be defined which allows for the safe transport of the beam.

This set of parameters will also characterize the maximum allowed difference between pulses. Unless safe settings are detected and subsequent beam pulses lie within the boundaries defined for operation, no beam beyond a critical current can be injected.

No.	Requirement	Source	Priority
CXR-MP-100	The machine protection system shall pre-establish safe conditions for startup based on the integrity of all preset conditions including steering magnets, beam position monitors, and RF power and phase.	A. Warner 1-2008	Critical
CXR-MP-110	The machine protection system shall transition to a partial length beam pulse, then to the full repetition rate of 5 Hz, and finally to the full length beam pulse while not exceeding the pre-programmed inter-pulse difference (PPID).	A. Warner 1-2008	Critical
CXR-MP-120	The machine protection system shall actively compare each new beam pulse with its predecessor at the 5 Hz machine rate to check that operational conditions are met.	A. Warner 1-2008	Critical
CXR-MP-125	If the conditions are violated, extraction shall be inhibited or the beam shall be diverted to a beam dump before entering the next acceleration stage.	A. Warner 1-2008	Critical
CXR-MP-130	The machine protection system shall monitor steady state machine operations and inhibit injection of pulses as appropriate if other faults suddenly occur. Such faults will include vacuum, magnet power supplies, excessive background, over temperature and of course beam orbit or feedback system faults.	A. Warner 1-2008	Critical
CXR-MP-140	The machine protection system shall be sensitive to machine operation modes.	A. Warner 1-2008	Critical
CXR-MP-150	The machine protection system shall make use of the Project X timing system, XCLK.	G. Vogel 1-2008	Critical
CXR-MP-160	The machine protection system	A. Warner	Critical

	shall monitor the status of the beam permit systems of the legacy machines.	1-2008	
CXR-MP-170	The machine protection system shall archive timestamped values when a failure event occurs.	A. Warner 1-2008	Critical
CXR-MP-180	There shall be an application program to analyze a failure event to determine the cause.	A. Warner 1-2008	Critical
CXR-MP-190	The machine protection system shall maintain alarm as well as inhibit ranges for input parameters.	A. Warner 1-2008	Expected
CXR-MP-200	The machine protection system shall post alarms for input parameters which are out of tolerance.	A. Warner 1-2008	Expected
CXR-MP-210	The machine protection system must be expandable in terms of numbers of inputs.	A. Warner 1-2008	Critical
CXR-MP-220	The machine protection system shall monitor long term equipment failure information and generate alarms warning about potential failures.	A. Warner 1-2008	Expected
CXR-MP-230	The machine protection system shall be designed with redundant paths for detecting and responding to failures.	A. Warner 1-2008	Desired
CXR-MP-240	The machine protection system shall make use of combinations of XCLK events, beam permits, and beam switches to define operational modes.	G. Vogel 1-2008	Expected
CXR-MP-250	The machine protection system shall be designed with both mechanical and software beam switches to prevent beam from being accelerated to send to an area not intended to receive beam.	G. Vogel 1-2008	Expected

9.2 Beam Permit

A beam permit system is a system that generates a signal indicating that the machine or beam line covered by the beam permit is ready to receive beam. A typical beam permit at Fermilab consists of a signal source and a standalone hardware path for the signal with local inputs that can inhibit the signal from reaching its final destinations, usually the BSSB and the beam dump or transfer device of the next most upstream machine. In this way dropping a beam permit will both inhibit beam from being transported into the non-ready area and prevent the next beam pulse from ever leaving the source by inhibiting the chopper.

The linac beam permit system must protect the machine from damage due to beam. It should be able to dump beam in the machine as well as inhibiting future pulses until the initial problem is addressed.

No.	Requirement	Source	Priority
CXR-MP-300	The beam permit shall be on a separate hardware loop from the control and safety systems.	G. Vogel 1-2008	Critical
CXR-MP-310	The beam permit shall have both software and hardware inputs.	G. Vogel 1-2008	Critical
CXR-MP-320	The status of all beam permit inputs shall be readable and displayed via an application program.	G. Vogel 1-2008	Critical
CXR-MP-330	The control system shall be capable of resetting the beam permit.	G. Vogel 1-2008	Critical
CXR-MP-340	All beam permit inputs shall be configurable and maskable.	G. Vogel 1-2008	Critical
CXR-MP-350	The control system shall support limiting who can configure given inputs to prevent critical inputs from being improperly masked.	B. Hendricks 1-2008	Expected
CXR-MP-360	The status of the beam permit shall be passed to the next upstream system to prevent beam from entering an inhibited region.	G. Vogel 1-2008	Critical

10Software Development Environment

This section will categorize software into two types of software. These two types are applications and libraries. These two types are defined as follows:

- Applications …
  - have a "main" routine and are therefore capable of being run by a user given the proper run-time environment, and
  - are not to directly contain code that is useful to other applications.
- Libraries …
  - do not have a "main" routine, and
  - contain code that is useful to other applications.

The term library is used despite that term being more closely aligned with some program languages (C/C++) than others (Java) because of a lack of a term that does not associate with any particular language. For Java one can think of a "library" is being implemented as a "main"-less jar file share by multiple applications. The term library was preferred over "main"-less jar file because the term library while not used by Java is used by languages other than (C/C++), e.g., FORTRAN, and is therefore the more general term.

10.1 Production Applications and Libraries

A goal of the software development environment (SDE) is to allow any software upon which the accelerator complex becomes operationally dependent to be reasonably maintained even after such software's original author is no longer available. Another goal of the SDE is to allow software to be developed upon which the accelerator is not operationally dependent with less administrative burden than for software upon which the accelerator complex is dependent. To support these twin goals all software is categorized as either production or non-production. Only production code should be used to operate the accelerator complex. Additional requirements are enforced on production software to insure its long-term maintainability.

The production vs. non-production distinction is applicable to both applications and libraries.

No.	Requirement	Source	Priority
CXR-SD-100	The SDE shall make a distinction between production and non-production applications/libraries.	C.Schumann 1-2008	Expected
CXR-SD-120	For all production applications/libraries the SDE shall allow individuals to subscribe to code change email notifications.	C.Schumann 1-2008	Expected
CXR-SD-130	For all production applications/libraries the SDE shall maintain meta-information about the program, e.g., the developer currently responsible for the	C.Schumann 1-2008	Expected

	program		
CXR-SD-140	For all production applications/libraries the SDE shall provide a deployment mechanism by which the application can be made available to all control system users.	C.Schumann 1-2008	Expected
CXR-SD-150	For all production applications/libraries the SDE shall maintain the applications source code in a version control system.	C.Schumann 1-2008	Expected
CXR-SD-170	For all production applications/libraries The SDE shall insure that the application links only to production libraries. (3rd party libraries installed on the systems by the system admins are considered production libraries.)	C.Schumann 1-2008	Expected
CXR-SD-180	There shall be a review process for the adoption of new third party libraries.	C.Schumann 1-2008	Expected
CXR-SD-190	The SDE shall provide maintenance of application source code and front-end code in the same system.	C.Schumann 1-2008	Expected

10.2 Non-Production Applications and Libraries

No.	Requirement	Source	Priority
CXR-SD-200	The SDE shall allow non-production applications.	C.Schumann 1-2008	Expected
CXR-SD-210	The SDE shall allow non-production applications to link to 3rd party non-production libraries for testing purposes.	C.Schumann 1-2008	Expected
CXR-SD-220	The SDE shall prevent non-production applications from being deployed through the deployment mechanism reserved for production applications.	C.Schumann 1-2008	Expected
CXR-SD-230	The SDE shall provide a testing environment for non-production applications.	C.Schumann 1-2008	Expected
CXR-SD-240	The SDE shall make it apparent to application users whether or not they are using a production or non-production application.	C.Schumann 1-2008	Expected

10.3 Modular Code Development

No.	Requirement	Source	Priority

CXR-SD-300	The SDE shall distinguish between libraries and applications.	C.Schumann 1-2008	Expected
CXR-SD-310	The SDE shall allow libraries to use other libraries.	C.Schumann 1-2008	Expected
CXR-SD-320	The SDE shall allow applications to use libraries.	C.Schumann 1-2008	Expected
CXR-SD-330	The SDE shall prevent applications from using other application code directly. (This requirement is present to make autonomous deployment of a single application tractable.)	C.Schumann 1-2008	Expected

10.4 Ease of Use

No.	Requirement	Source	Priority
CXR-SD-400	Any system on which the SDE is made available shall be equivalent to any other, e.g., the availability of libraries should be identical.	C.Schumann 1-2008	Expected
CXR-SD-410	The SDE environment shall automate the build process, i.e., it should be capable of building an application given only a collection of source code and a modest amount of meta-information about the application without any build system configuration in the form of makefiles or build.xml. The meta-information mentioned above would include, e.g., any libraries that the application requires.	C.Schumann 1-2008	Expected

10.5 Integrated Development Environment (IDE)

The SDE shall provide support for the recommended IDE:

No.	Requirement	Source	Priority
CXR-SD-500	The SDE shall recommend, but not mandate, an Integrated Development Environment (IDE).	C.Schumann 1-2008	Expected
CXR-SD-510	The SDE shall provide plug-ins, if needed/useful, for integration with other subcomponents of the SDE, e.g., the version control system	C.Schumann 1-2008	Expected
CXR-SD-520	The SDE shall provide site-specific documentation about the IDEs use	C.Schumann 1-2008	Expected
CXR-SD-530	The IDE shall support any language supported by the SDE	C.Schumann 1-2008	Expected
CXR-SD-540	The SDE source code control shall only be for IDE independent files.	S. Gysin 1-2008	Expected

10.6 Debugging Tools

No.	Requirement	Source	Priority
CXR-SD-600	The SDE shall provide profiler(s) for supported language.	C.Schumann 1-2008	Expected
CXR-SD-610	The SDE shall provide heap/memory leak debugging tools.	C.Schumann 1-2008	Expected
CXR-SD-630	The SDE shall provide (an) appropriate source level debugger(s).	C.Schumann 1-2008	Expected

10.7 SDE Deployment

No.	Requirement	Source	Priority
CXR-SD-700	A subset of the SDE shall be available on the users' desktops.	R.Rechenmacher 1-2008	Expected
CXR-SD-710	There shall be a specification about how the client nodes, i.e., users' desktops, must be configured in order to run the SDE.	A.Petrov 1-2008	Expected
CXR-SD-720	There shall be a mechanism for deploying changes to the SDE.	C.Schumann 1-2008	Expected

10.8 Testing Environment

A completely inclusive virtual machine/complex testing environment shall not be provided. This is a concession to the difficulty of providing such a environment, which would to be truly complete would have to provide a simulation of the real machine physics, the real database(s), the real file system, etc. However, some testing environment will be provided. Its capabilities are specified in the following requirements.

No.	Requirement	Source	Priority
CXR-SD-810	To facilitate (some) testing and perhaps learning/understand/embracing of the control system, a scaled down server and client environment shall be provide and maintained to allow for basic testing of newly developed client and server applications, respectively.	R.Rechenmacher 1-2008	Expected

10.9 Diagnostics for Development and Deployment

No.	Requirement	Source	Priority
CXR-SD-900	The SDE shall automatically provide embedded version information in each build of an application or library.	C.Schumann 1-2008	Expected
CXR-SD-910	The SDE shall provide a log which includes version information.	R.Rechenmacher 1-2008	Expected

10.10 Version Control

No.	Requirement	Source	Priority
CXR-SD-1000	The SDE shall require a comment for any code change.	C.Schumann 1-2008	Expected
CXR-SD-1010	The SDE shall record the programmer for any code change.	C.Schumann 1-2008	Expected
CXR-SD-1020	The SDE shall support renaming files and directories.	C.Schumann 1-2008	Desired
CXR-SD-2030	The SDE shall support branching and should do this in a way that does not interfere with users who do not wish to use it.	C.Schumann 1-2008	Desired

10.11 Collaborative Development

The SDE will support collaborative development. This breaks down into two requirements which work together as a pair to insure a well-ordered collaborative process.

No.	Requirement	Source	Priority
CXR-SD-1100	The SDE shall support distribution of source code to collaborators.	C.Schumann 1-2008	Expected
CXR-SD-1110	The SDE shall support integration of source code changes from collaborators back into production systems.	C.Schumann 1-2008	Expected
CXR-SD-1120	The SDE shall track the use of distributed licensed third party code.	A.Petrov 1-2008	Expected

10.12 Issue Tracking

No.	Requirement	Source	Priority
CXR-SD-1200	There shall be an issue tracking system.	C.Schumann 1-2008	Expected
CXR-SD-1210	The issue tracking database shall support the ability to assign an issue to any developer.	C.Schumann 1-2008	Expected

CXR-SD-1220	The issue tracking database shall send notification when to a developer when an issue is assigned to him/her.	C.Schumann 1-2008	Expected
CXR-SD-1230	There shall be a method available for correlating code changes to the issue tracking database.	C.Schumann 1-2008	Desired

10.13 Language Support

The best language for the task should be used.

No.	Requirement	Source	Priority
CXR-SD-1300	The SDE shall support C/C++ and Java.	C.Schumann 1-2008	Expected
CXR-SD-1310	The SDE shall be extended to support other languages as approved by a review process.	C.Schumann 1-2008	Expected

10.14 Documentation

No.	Requirement	Source	Priority
CXR-SD-1400	The SDE shall support inline API documentation which is to be published to the web.	C.Schumann 1-2008	Expected

10.15 Software Quality and Process

No.	Requirement	Source	Priority
CXR-SD-1510	Critical software shall be developed by a team or peer approach.	R.Neswold 1-2008	Desired
CXR-SD-1520	The SDE shall provide a tool for viewing code in a standard format.	B.Hendricks 1-2008	Desired

11 Hardware/Operating Systems

11.1 Hardware

The hardware requirements cover all three tiers of functionality as described in Section 5: front-end nodes, central nodes and client nodes.

No.	Requirement	Source	Priority
CXR-HO-100	Commercially available hardware based on open standards (commodity hardware) shall be used whenever possible.	D.Finstrom, J. Smedinghoff 12-2007	Desired

11.2 Hardware Requirements for Low Level

Hardware requirements for the low level are covered in Section 3.2.

11.3 Hardware Requirements for Central Nodes

The central node hardware should be chosen to minimize the operational impact of hardware failure.

No.	Requirement	Source	Priority
CXR-HO-200	Central nodes shall use hardware that is rack mountable.	D.Finstrom, J. Smedinghoff 12-2007	Expected
CXR-HO-210	Central nodes shall support full-duplex gigabit Ethernet over copper at a minimum.	D.Finstrom, J. Smedinghoff 12-2007	Expected
CXR-HO-220	Central nodes shall undergo a burn-in process to minimize infant mortality failures.	D.Finstrom, J. Smedinghoff 12-2007	Desired
CXR-HO-230	Central nodes shall be installed in a computer room with controlled access, sufficient power (both conventional and protected), and environmental controls (CRX-NW-490, CRX-NW-500, and CRX-NW-510).	D.Finstrom, J. Smedinghoff 12-2007	Expected
CXR-HO-240	Central nodes shall have a hardware support call list.	D.Finstrom, J. Smedinghoff 1-2008	Expected
CXR-HO-250	Central nodes shall have a hardware support contract or adequate spares and a failover plan.	D.Finstrom, J. Smedinghoff 12-2007	Critical
CXR-HO-260	Central data storage and databases	D.Finstrom, J.	Expected

	shall use hot swappable, redundant and scalable disk arrays.	Smedinghoff 12-2007	
CXR-HO-270	Critical (required for control system availability) central nodes shall have redundant power supplies.	D.Finstrom, J. Smedinghoff 12-2007	Expected
CXR-HO-280	A file backup system shall be provided for use by central nodes.	D.Finstrom, J. Smedinghoff 12-2007	Critical
CXR-HO-290	All control system data from central nodes shall be backed up.	D.Finstrom, J. Smedinghoff 1-2008	Critical
CXR-HO-295	Failed critical central hardware shall utilize automatic failover.	D.Finstrom, J. Smedinghoff 1-2008	Desired

11.4 Hardware Requirements for the Client Nodes

Client node hardware is the hardware executing only high-level applications and is determined by the user. If a client node is unable to execute high-level applications, they will be able to run the application from a central node [CXR-AI-30]. Control rooms contain client nodes with expanded display capabilities.

No.	Requirement	Source	Priority
CXR-HO-300	Control room client nodes shall be capable of running multiple screen displays.	D.Finstrom, J. Smedinghoff 12-2007	Expected
CXR-HO-310	Control room client nodes shall have a hardware knob.	D.Finstrom, J. Smedinghoff 1-2008	Expected

11.5 Operating Systems

Operating system requirements cover for all three tiers of functionality: low level, central nodes and client nodes.

11.6 Operating Systems for Low Level

No.	Requirement	Source	Priority
CXR-HO-400	The operating systems for low level nodes shall comply with the Fermilab Strong Authentication Policy.	D.Finstrom, J. Smedinghoff 12-2007	Desired
CXR-HO-410	As described by the Fermilab Policy on Computing, the operating systems for	D.Finstrom, J.	Desired

	low level nodes shall be reasonably recent and supported versions for which a Fermilab security configuration baseline exists.	Smedinghoff 12-2007	
CXR-HO-420	The operating systems for low level nodes shall support remote monitoring of system information (CXR-LL-320).	D.Finstrom, J. Smedinghoff 1-2008	Desired
CXR-HO-430	The operating systems for low level nodes shall support portable APIs such as pthreads (CXR-LL-280).	D.Finstrom, J. Smedinghoff 1-2008	Desired

11.7 Operating Systems for Central Nodes

No.	Requirement	Source	Priority
CXR-HO-500	The operating systems for central nodes shall comply with the Fermilab Strong Authentication Policy.	D.Finstrom, J. Smedinghoff 12-2007	Expected
CXR-HO-510	As described by the Fermilab Policy on Computing, the operating systems for central nodes shall be reasonably recent and supported versions for which a Fermilab security configuration baseline exists.	D.Finstrom, J. Smedinghoff 12-2007	Expected
CXR-HO-520	The operating systems for central nodes shall support remote monitoring of system information.	D.Finstrom, J. Smedinghoff 1-2008	Desired
CXR-HO-530	The operating systems for central nodes shall support portable APIs such as pthreads.	D.Finstrom, J. Smedinghoff 1-2008	Desired
CXR-HO-540	On central nodes, automated operating system installation tools such as Kick start shall be used.	D.Finstrom, J. Smedinghoff 1-2008	Desired
CXR-HO-550	Scientific Linux Fermi shall be the preferred operating system for central nodes.	D.Finstrom, J. Smedinghoff 12-2007	Desired
CXR-HO-560	The operating systems for central nodes shall support random number generation.	D.Finstrom, J. Smedinghoff 1-2008	Desired

11.8 Operating Systems for Client Nodes

No.	Requirement	Source	Priority
CXR-HO-600	The operating systems for client	D.Finstrom, J.	Expected

	nodes shall comply with the Fermilab Strong Authentication Policy.	Smedinghoff 12-2007	
CXR-HO-610	As described by the Fermilab Policy on Computing, the operating systems for client nodes shall be reasonably recent and supported versions for which a Fermilab security configuration baseline exists.	D.Finstrom, J. Smedinghoff 12-2007	Expected
CXR-HO-620	Clients running Linux shall use Scientific Linux Fermi.	D.Finstrom, J. Smedinghoff 12-2007	Desired
CXR-HO-630	The operating systems for client nodes shall support X Windows.	D.Finstrom, J. Smedinghoff 12-2007	Critical

12Networks

12.1 Project X Network Overview

The Network shall be designed to support four distinct networks including separate networks for the commissioning and operation of the accelerator. The design should consist of a Controls network, DMZ network (Demarcation Zone), Development network and General network. The design shall supply secure network access for specific individuals via authenticated gateways in the DMZ network to the Controls network. This would include and encourage AAA access (authentication, authorization and accounting) for users in other Fermi divisions and collaborators at sites throughout the internet. The Development network shall allow for testing an implementation of new devices, architecture, etc without interfering with either the Controls or site network. The General network shall accommodate desktops and laptops with authentication and authorization as prescribed by site policy. Each of these networks shall allow for necessary bandwidth, network protocols, VLANs and subnets, secure authentication and authorization, ACLs, firewalls, redundancy, cable plant, QoS, IPv6 and future technology changes.

No.	Requirement	Source	Priority
CXR-NW-10	The Network shall provide and support a separate dedicated Controls network for accelerator controls	D. Stenman 1-2008	Critical
CXR-NW-20	The Network shall provide a DMZ network for authenticated gateway access to Controls Network	D. Stenman 1-2008	Expected
CXR-NW-30	The Network shall provide a Development network that isolates development activities but allows access to site and internet services, subject to network policies	D. Stenman 1-2008	Expected
CXR-NW-40	The Network shall support a General network for desktop, laptop and domain services	D. Stenman 1-2008	Expected
CXR-NW-45	The Network shall support AAA access for intra Division personnel and collaborators throughout the internet.	D. Stenman 1-2008	Expected

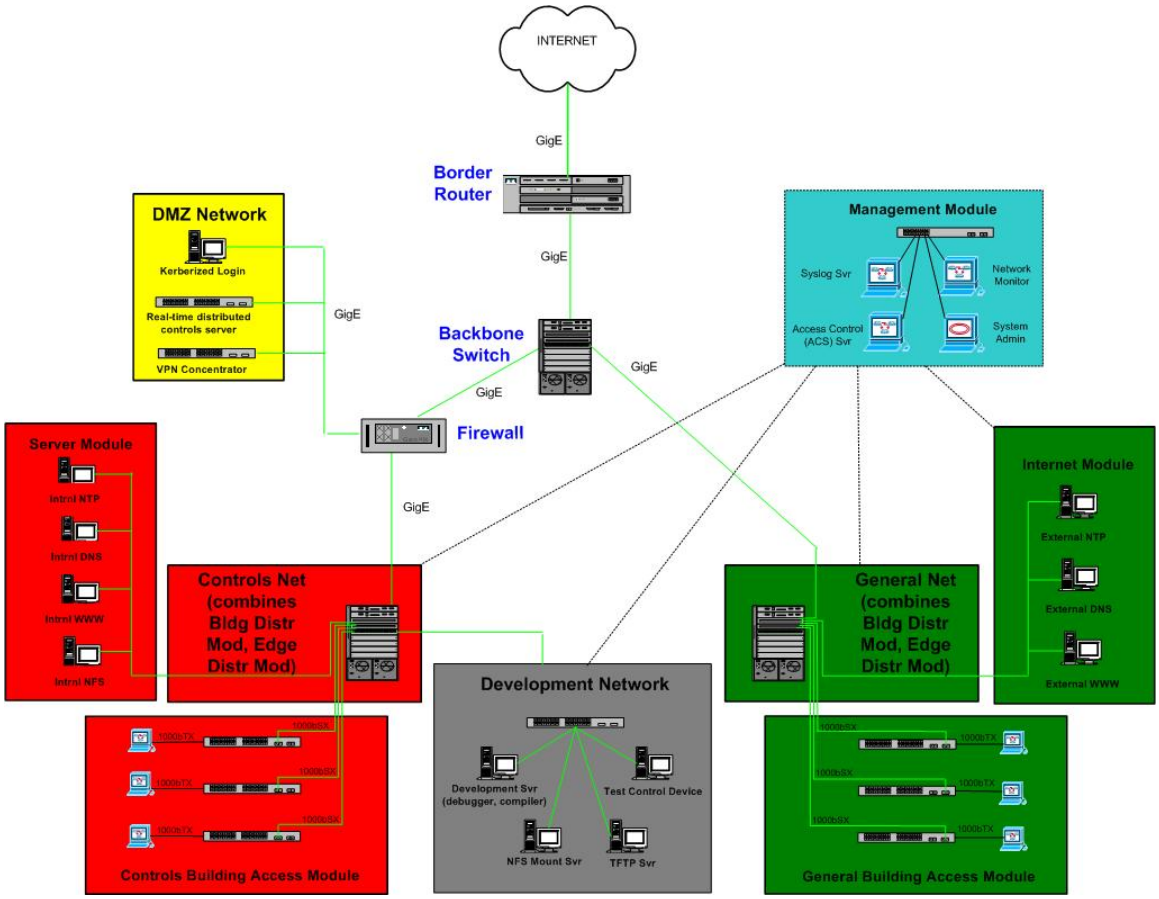


Figure 1. The Four network modules

12.2 The Controls Network

The Controls network shall be physically separate from the DMZ, Developmental and General networks via dedicated infrastructure using both fiber and copper cable plant along with separate distribution and access network switches.

- The Controls network shall have a Firewall for select traffic with default deny inbound and outbound except for selected services.
- Authenticated access to the Controls network will be possible via gateway devices in the DMZ, including VPN (network based) and bastion hosts (login based).
- Critical network services, such as NTP, DNS, KDC and W2k Domain controllers shall have instances located in the Controls network to maintain usability when isolated from the other networks.
- A single point of disconnection shall be provided to enable total isolation of the Controls network from external network disturbances.

The Controls network shall be populated by network devices assigned static IP addresses in the Class B space of 131.225.x.x

- The third number in the dotted decimal notation shall be provided by Computing Division as requested by the AD Network Group, such as 131.225.146.0 to 131.225.146.255)
- The third and fourth set of numbers in the dotted decimal notation shall then be assigned by the AD Controls group. The third set of addresses shall refer to subnets having related functionality or geography.
- AD Controls shall have an option to specify strategic address allocation for network diagnostic equipment, routers and switches
- All Controls network addresses shall be in one CIDR block (able to be fully specified in the format 131.225.x.y/z where 0<=x<=255, 0<=y<=255 and 16<z<32.).

No.	Requirement	Source	Priority
CXR-NW-50	The Controls network shall be physical separate with a dedicated cable plant and network devices	D. Stenman 1-2008	Critical
CXR-NW-60	The Controls network shall have a firewall with default deny inbound and outbound except for select services	D. Stenman 1-2008	Critical
CXR-NW-70	Authenticated access to the Controls network shall be via gateway devices in the DMZ network, including both a VPN and kerberized bastion hosts.	D. Stenman 1-2008	Critical
CXR-NW-80	The Controls network shall be physically isolated by way of a single point of disconnect.	D. Stenman 1-2008	Critical
CXR-NW-90	Static IP addresses shall be assigned to devices in subnets related to geography and functionality.	D. Stenman 1-2008	Expected
CXR-NW-100	Critical network services such as NTP, DNS, KDC and W2K Domain controllers shall have instances located in the Controls network	T. Zingelman 1-2008	Expected
CXR-NW-110	All devices connected to the Controls network will have a registered system administrator responsible for them	D. Stenman 1-2008	Expected
CXR-NW-120	All Controls network addresses shall be in one CIDR block (able to be fully specified in the format 131.225.x.y/z)	T. Zingelman 1-2008	Desired

12.3 The DMZ Network

The DMZ network shall provide the method to access the Controls network.

- There shall be a Kerberized bastion host and a VPN server in the DMZ that provide account authentication and allow login or network access for individuals according to Lab strong authentication policy.
- There shall be a real-time distributed controls system gateway server in the DMZ network that will provide software running outside the Controls subnets with access to specific control and monitor points within the Controls network.
- The real-time distributed controls system server shall recognize a subset of Lab User IDs from specific hosts and authorize them remote program access to the Controls network.

No.	Requirement	Source	Priority
CXR-NW-122	Only those devices in the DMZ network shall have access to the Controls network.	D.Stenman 1-2008	Critical
CXR-NW-124	There shall be a Kerberos and VPN server in the DMZ network for user authentication.	D.Stenman 1-2008	Critical
CXR-NW-126	There shall be a real-time distributed controls system gateway server in the DMZ network.	D.Stenman 1-2008	Expected
CXR-NW-128	The real-time distributed controls system server shall recognize a subset of Lab User IDs from specific hosts.	D.Stenman 1-2008	Expected

12.4 The Development Network

The Network shall support a Development network to allow for engineering and testing of components to be used in the Controls network.

- The Development network shall isolate development activities from the other subnets, therefore not interfere with normal activities on the Controls network or AD and site-wide Fermi networks.
- Network devices in the Development network shall be assigned static IP addresses and the Mac address, System Administrator, Primary User and location registered accord to Lab policy.
- The Development network shall be accessible through secure channels to network devices, desktop and laptop computers used by the AD Controls Group, while also allowing for access to resources on the Fermi network and the internet.
  The AD Controls Group shall be able to add a reasonable number of net devices to the Development network, which will not be accessible from the internet except through secure channel or have access to any service outside the Fermi network.
- The Development network shall contain a development server, accessible from all devices on the development subnet.
- The Development server shall provide TFTP service (not visible outside the development network), NFS mount service, development tools (debuggers, compilers, etc), and any other software necessary for the Controls Group.

No.	Requirement	Source	Priority

CXR-NW-130	The Development network shall isolate activities from other subnets	D. Stenman 1-2008	Expected
CXR-NW-140	All Development network nodes shall have static assigned IP addresses, and be registered according to Lab policy	D. Stenman 1-2008	Expected
CXR-NW-150	The Development network shall be accessible through secure channel from Controls Group computers.	D. Stenman 1-2008	Expected
CXR-NW-160	The Development network shall have a development server, accessible via secure channel by Controls Group and offering development services.	D. Stenman 1-2008	Expected

12.5 The General Network

The General network shall offer secure access for development, commissioning and general computing services, following Lab strong authentication policy. The General network includes Static and DHCP network addresses, WiFi access, Printing services, domain login, personal home directory storage, WWW access and email. VoIP and Video data streaming should also be included using QoS to insure network priority.

- General network shall have access to Fermi network and internet.
- Static IP addresses shall be assigned desktops, printers and servers.
- DHCP addressing shall be available for laptops.
- The General network shall have support for QoS.

No.	Requirement	Source	Priority
CXR-NW-170	Fermilab strong authentication policy shall apply to all network attached devices in the General network	D. Stenman 1-2008	Critical
CXR-NW-180	Static IP addresses shall be assigned to desktops, printers and servers in the General network	D. Stenman 1-2008	Expected
CXR-NW-190	DHCP addressing shall be available for laptops in the General network.	D. Stenman 1-2008	Expected
CXR-NW-200	The General network shall have support for QoS.	D. Stenman 1-2008	Desired

12.6 Acceptable Failure Rate and Impact

Redundancy or immediate failover of key core and backbone network resources shall be present in The Network, such as central router, firewall, DNS, NTP, Kerberos. On-site spares shall be present which involves manual replacement of access network equipment. Maintain of these live spares shall consist of all network devices, including backbone, distributed and access switches/routers.

No.	Requirement	Source	Priority
CXR-NW-210	Redundant firewall, border router, DNS server, NTP server, Kerberos server and W2K Domain server shall	D. Stenman 1-2008	Critical

	be provided.		
CXR-NW-220	7/24 hardware/software maintenance shall be provided for critical core routers and backbone switches	D. Stenman 1-2008	Critical
CXR-NW-230	On-site hot spares shall be available to replace distribution and access network devices.	D. Stenman 1-2008	Critical

12.7 Monitoring and response

All network devices and services shall be monitored via a central management station. A Management VLAN shall be configured for The Network to enable secure login, software upgrade, traffic monitoring and logging of network appliances. Allow for automated notification of network degradation. Network problems shall be coordinated with the MCR and a designated call list shall be provided for off-hour response. Search for node location per switch port and visibility of network port statistics shall be provided for users.

No.	Requirement	Source	Priority
CXR-NW-240	Network devices shall be managed from a central server through a dedicated VLAN	D. Stenman 1-2008	Critical
CXR-NW-250	There shall be an automated notification of network degradation	D. Stenman 1-2008	Expected
CXR-NW-260	Network problems shall be coordinated with the MCR and a designated call list for off-hour response	D. Stenman 1-2008	Expected
CXR-NW-265	Search for node location and port statistics within switched network shall be provided for users	D. Stenman 1-2008	Expected

12.8 Physical Layout and Network Model

The Controls network shall be completely self-contained. Specifically, it shall be able to perform all control system functions and provide all services without a physical connection to any other network.

Infrastructure shall follow the TIA/EIA-568 standard hierarchical cable system architecture using single mode fiber for backbone and distribution layer, and copper for access of most attached devices. Single mode fiber allows for 1 Gigabit (1000BASE-X) or 10 Gigabit uplinks (10GBASE) while copper accommodates common 1 Gigabit interface (1000BASE-T). Required is a comprehensive calculation of the number of network attached devices, bandwidth/timing needs, distances from service buildings, tunnels, Main Control Room, Data Center, and any specific protocol requirements or priorities. 1U switches shall be used for the access layer and network chassis for the distribution layer. Physical separation of the four networks dictates individual switches and fiber. Design shall resist the flat design model. Maintain a hierarchical design that allows for isolation of critical systems,

network/computing security, distribution of higher or lower capacity sub-networks, independent subsystem commissioning and network protocol or traffic configuration.

The Controls network shall be centered in the AD Computer Room and connect all areas of the AD accelerator facility.

- All Controls network nodes shall have the ability to communicate with any other node in the Controls network.
- Each device shall have a primary Ethernet connection on a specific subnet.
- Some devices may desire a secondary Ethernet connection on a different subnet from their primary one.
- No wireless access devices or nodes in the Controls network
- Terminal service and power management shall also be available on dedicated subnets of the Controls network.

There shall be physical connection network taps located at specific points in the accelerator area and wireless access points temporarily available.

- Only roll-around computers and laptops registered and/or configured by the AD network group shall have access in the accelerator area.
- Wireless access points pre-configured by the AD network group shall be provided exclusively in the General net to allow restricted wireless access in the accelerator area.

There shall be uniform speed and duplex for connections within and between all networks.

- All network devices in the Controls and DMZ networks shall have the ability to communicate at speeds up to 1000 Mbps, full duplex, preferably on copper media except where distance or RF interference are an issue.
- All network devices in the Development network shall have the ability to communicate at speeds up to 1000 Mbps, preferably on copper media except where distance or RF interference are an issue.
- Connections to the Controls network from the DMZ shall have the ability to communicate at speeds up to 1000 Mbps.

Subnet Requirements; there shall be a set of VLANs (virtual networks or broadcast domains) available on the Controls network to provide an organized structure and isolate specific functions which have bandwidth or timing requirements.

- All subsystems shall have the ability to belong to a separate dedicated VLAN/subnet or broadcast domain.
- There shall be a Network Management VLANs for dedicated network services.

No.	Requirement	Source	Priority
CXR-NW-270	The Network cable system shall be of TIA/EIA-568 Hierarchical design using single mode fiber or better for the backbone and Category 6e or better for access attached nodes	D. Stenman 1-2008	Critical
CXR-NW-280	The cable plant shall have individual cable and network devices for the Controls network to ensure isolation from DMZ, Development and General networks	D. Stenman 1-2008	Critical
CXR-NW-290	The Network shall have physical	D. Stenman	Expected

	connection network taps located at specific points in the accelerator area.	1-2008	
CXR-NW-295	There shall be no wireless access devices or nodes in the Controls network	D. Stenman 1-2008	Expected
CXR-NW-300	The General Network shall have wireless access points temporarily available for the accelerator tunnels and areas in addition to the permanent access points	D. Stenman 1-2008	Expected
CXR-NW-310	All subsystems shall have the ability to belong to a separate dedicated VLAN/subnet or broadcast domain	D. Stenman 1-2008	Expected
CXR-NW-320	All Controls network nodes shall have the ability to communicate with each other	D. Stenman 1-2008	Expected
CXR-NW-330	The only 'portable' devices which shall be connected to the Controls network are those that have been securely configured by the network group	D. Stenman 1-2008	Critical
CXR-NW-340	All network devices shall be capable of 1000 Mbps, full duplex communication on all ports	D. Stenman 1-2008	Critical
CXR-NW-350	All network devices shall be capable of IPv6 operation	T. Zingelman 1-2008	Expected
CXR-NW-360	All network edge connections shall be via copper	D. Stenman 1-2008	Expected
CXR-NW-370	All network distribution connections shall be via fiber	D. Stenman 1-2008	Expected
CXR-NW-380	There shall be Network Management VLANs	D. Stenman 1-2008	Expected

12.9 Network Security

The Controls network shall be fire walled with default deny inbound and outbound except for selected services.

- VPN, real-time distributed controls system gateway and Bastion Hosts shall be in the DMZ network to allow authenticated inbound traffic through the firewall.
- Controls VPN shall have authentication time limited network access.
- Bastion Hosts shall have Kerberized login as per Fermi strong authentication policy.
- Bastion Hosts shall have time limited logins.
- Bastion Hosts shall have SSH port forwarding allowed.
- Bastion Hosts shall have NFS mounts of selected inside disk to allow kerberized FTP access (FTP shall be blocked at Firewall).

There shall be ACLs on particular subnets and/or VLANs to further limit access for sensitive devices and ones that cannot conform to all provisions of strong authentication.

Network devices shall reside on secure Network Management VLANs and require either authentication login using Kerberos or individual Radius user accounts.

There shall be an emergency disconnect at division border to ensure Accelerator operation regardless of site or internet traffic storms.

Cyber protections shall provide security without disruption to accelerator operations.

- Patching and anti-virus services shall be available for OSs and Applications of all network attached devices, including network routers, switches and access points.
- Scanning of The Network for standard network services/ports, adherence to critical system patches and anti-virus shall be tolerated by all network attached nodes.

An inventory of all systems on the network shall be maintained, to include OS, hardware, MAC address, IP address, sysadmin and primary user

- Any changes in registered IP and MAC address shall have automated notification.

No.	Requirement	Source	Priority
CXR-NW-400	There shall be Access Control Lists on subnets and VLANs as required to provide isolation in addition to the firewall.	D. Stenman T. Zingelman 1-2008	Expected
CXR-NW-410	VPN, a real-time distributed controls system gateway and Bastion host shall have time limited login.	D. Stenman T. Zingelman 1-2008	Expected
CXR-NW-420	There shall be an emergency disconnect at division border.	D. Stenman T. Zingelman 1-2008	Expected
CXR-NW-430	Cyber protections shall provide security scanning and patching without disruption to accelerator operations.	D. Stenman T. Zingelman 1-2008	Expected
CXR-NW-440	An inventory of all systems on The Network shall be maintained, to include OS, hardware, MAC address, IP address, sysadmin and primary user.	D. Stenman T. Zingelman 1-2008	Critical
CXR-NW-450	Any changes in registered IP and MAC address shall have automated notification.	D. Stenman T. Zingelman 1-2008	Critical
CXR-NW-460	Network devices shall reside within Management VLANs with secure authentication.	D. Stenman T. Zingelman 1-2008	Critical

12.10 Remote Network Monitoring

Network devices shall implement RMON, a flow-based monitoring and analyzing client/server model. The RMON2 agents will include MIB legacy groups of RMON1 (such as host statistics, top users, alarms, SNMP traps, etc) and the extended groups of RMON2.

- Network Statistics
- Alarms
- Hosts Statistics and top Hosts
- Network-Layer Host
- Network-Layer Matrix
- Application-Layer Host
- Application-Layer Matrix
- Probe configuration
- Protocol Directory and Distribution

No.	Requirement	Source	Priority
CXR-NW-470	RMON shall be implemented to monitor and analyze Ethernet packets	D. Stenman 1-2008	Desired
CXR-NW-480	RMON2 shall be included with its extended MIB groups to add support for network and application layer	D. Stenman 1-2008	Desired

12.11 Data Center

The AD Computer room, XGC-108, shall expand to include additional network equipment, Controls servers, Front Ends, Database servers, and file servers.

Available conventional power shall be expanded to include additional central service nodes, Controls database servers, Front Ends, etc. Amount of expansion depends on watts per unit device.

Available UPS power shall be expanded to include additional central service nodes, Controls database servers, Front Ends, etc. Amount of expansion depends on watts per unit device.

Air flow and temperature and humidity control shall be increased according to level of heat load per square foot within Computer Room.

No.	Requirement	Source	Priority
CXR-NW-490	Conventional power in the computer room shall be expanded according to increase of watts per unit device	D. Stenman 1-2008	Critical
CXR-NW-500	UPS power in the computer room shall be expanded according to	D. Stenman 1-2008	Critical

	increase of watts per unit device		
CXR-NW-510	Air flow and temperature/humidity controls in the computer room shall be expanded to accommodate increase of heat load per square foot within Computer Room.	D. Stenman 1-2008	Critical

13References

[1] Project X web site http://projectx.fnal.gov/
[2] Patrick, Jim "ACNET Control System Overview" Beams Document 1762, February 17, 2005, https://beamdocs.fnal.gov/AD-private/DocDB/ShowDocument?docid=1762
[3] McGinnis, Dave "Project X Overview", October 2007
[4] Carwardine, John et. al., "ILC Reference Design Report", "Requirements and Technical Challenges" (3.12.2) , January 2007, https://docdb.fnal.gov/ILC-private/DocDB/ShowDocument?docid=371
[5] Hendrickson, Linda, "Feedback Systems for ILC", http://docdb.fnal.gov/ILC-public/DocDB/ShowDocument?docid=166
[6] Phinney, Non "Feedback Studies for ILC", http://docdb.fnal.gov/ILC-public/DocDB/ShowDocument?docid=12
[7] Larson, Ray "Diagnostic Interlock Layer", http://docdb.fnal.gov/ILC-public/DocDB/ShowDocument?docid=49
[8] Allen, C.K. et. al. "A Novel Online Simulator for Applications Requiring a Model Reference", http://www-linac.kek.jp/seminar/2005/Allen-NovelSim-Icalepcs2003.pdf
[9] Galambos, J. et. al, "Application Programming", http://neutrons.ornl.gov/APGroup/appProg/xal/talks/ASAC_March04.pdf
[10]Church, Mike et. al, "ILCTA/NML Controls Requirements", http://docdb.fnal.gov/ILC-public/DocDB/ShowDocument?docid=325