NLC CONTROL SYSTEM REQUIREMENTS SUMMARY

I. INTRODUCTION

NLC COLLIDER OVERVIEW: The Next Linear Accelerator (NLC) is a very large Linear Accelerator with two separate Linacs facing each other for accelerating and Colliding Electrons and Positrons. These two 15 kilometer long Linacs have an additional 5km in between for the final focus structures and the two proposed particle Detectors.

This Linear Accelerator will be built for physics research at the TeV energy range, and will be built on much of the work done at the Stanford Linear Accelerator Center. This Electron-Positron Collider will be the next machine in a line of well-controlled and well-understood experimental environments in which new phenomena are presented for precision measurement and study. There are Design Reports available describing the technology, use, and construction of the NLC.

In addition to being large and spread out over substantial real estate, the Accelerator is large in terms of devices and instrumented data points, the required data acquisition bandwidth, and the level of functional reliability required to make it a viable physics research tool. In broad terms, there will be some 3 million instrumented data points, with each device and sensor performing as advertised for each pulse of the normal 120 Hz repetition rate for months at a time. System and Device reliability will be a major design factor for this research machine.

In addition to its two substantial linear accelerating Linacs, the NLC has a number of additional facilities. Notable among these, are the Damping Rings for both Electron and Positron Systems, the Positron Generation System with pre-damping ring, and the Beam Delivery and final focusing systems between the accelerators and the two proposed Particle Detectors.

Because of its large size and substantial cost, this project is envisioned as a multi-year, multi-country collaborative research venture. It is likely that the accelerator will be designed, built, operated, and maintained by an

International collaboration including many Universities and Laboratories around the world.

GENERAL REQUIREMENTS OF THE CONTROL SYSTEM: The Control System for this Accelerator 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. Due to the size and complexity of the whole Accelerator and control infrastructure, the Control System will need to have a very high availability (99.95) and a low Mean time to repair (MTTR). Use of standards in equipment and software systems will aid in development, diagnosis, and repair.

From an Architectural point of view, the Control System should provide good access security, layered and open software structures and protocols, ease of enhancement and maintainability, and some form of software and systems Support Collaboration. It is our specific intent to take an existing proven Control System, and extend its feature baseline to accommodate the larger physical plant and the few really new and unique requirements of the NLC. We believe that to do this requires a large software development team spread over many Laboratories and Universities with common contributing users. To develop a completely new Control System from scratch is not within the range of time and resources available for this project.

Due to the unique size, type, and operational parameters of the NLC, a number of the subsystems of the Accelerator Control System will be customized to support the specific attributes and requirements of this Accelerator. Independent of the SCADA system selected, there will certainly be custom software to support subsystems to measure Beam Position in the beam-pipe, Timing systems to coordinate and sequence the operation of the machine, and low level RF systems to accelerate the beams. In order to provide similar data to many application packages, there will need to be a high bandwidth synchronous data acquisition facility to collect beam parameters and server them out to many requesting applications for each beam pulse.

The Control System of choice must be structured so that these custom subsystems can be integrated with source code available for modification and maintenance. The Control System will be database driven so that additions or modifications to the hardware, software, or feedback configurations can be implemented quickly and easily. It is important to use as many hardware and software standards as possible, and commodity high reliability equipment will moderate the cost of acquisition, maintenance, and upgrade.

The SCADA component of the NLC Control System must be implemented and installed with its basic functionality prior to the commissioning of the first sections of the Accelerators. The Control System scale will have to expand incrementally as the hardware systems are installed and commissioned. Bandwidth and network performance will be the critical elements of the system scalability.

Most of the Application component must be complete and available prior to commissioning, with a few facilities scheduled for development through the time period when actually running the Accelerator gives evidence of new requirements. Due to the large volume of on-going software development, there must be a separate but smaller control system environment for testing software releases prior to installation on the Production Control System.

CONTROL SYSTEM BASICS: All Control Systems contain facilities for Data Acquisition and Set Point Control; collectively called a Supervisory Control and Data Acquisition System (SCADA). Generally this is configured as an API system plus a number of remote intelligent Input/Output device controllers (IOC). This system includes facilities to archive data, sound alarms when tolerances are exceeded, diagnostics to insure system integrity and summary display systems for effective status display. EPICS, a candidate for NLC use, is basically a SCADA system with the real-time facilities running in the IOC crates distributed throughout the accelerator housings. The NIF Control System has a more modern design but is not yet complete and established in the field (but elements may be purloined and applied to our needs).

Special purpose systems for Personnel and Equipment Protection are run in hardware and in parallel to the software based control system. These subsystems may use EPICS status and display facilities. These special purpose production systems are the Machine Protection System (MPS), Beam Containment Systems (BCS), and the totally separate hardware oriented Personnel Protection Systems (PPS).

Also down low in the infrastructure layers, are the very fast data acquisition and event logging systems. In the case of the NLC, we intend to collect beam related data in real-time and distribute it to competing applications from a server. This data set will have to be streamed out from the IOC's where it is acquired at the Real-Time Executive (RTE) task level, rather than by the normal SCADA processes. In this case, the facility will more closely resemble a Detector Control System than an Accelerator Control System.

On top of this Data Acquisition platform are a set of user Application Packages which perform analytical and automation functions, apply fast-feedback control and coordination loops, and measure accelerator performance and beam quality parameters. The SLC Control System has a number of these Application facilities, which will be implemented on top of the SCADA System in a context suitable to the NLC.

NLC CONTROL SYSTEM ARCHITECTURE: At the top of the functional Control System Architecture, are the operator consoles (OPI). These are nominally Sun workstations which provide the user the access and control of the systems. The OPI workstation is available to complete limited application processing for an individual console user.

At a middle level in the processing hierarchy there are a host of application and data servers, which accomplish the number crunching and data manipulation. These machines may be collections of stand-alone processors or in some cases, a larger multi-processor server “mainframe”. These servers provide the heavy duty processing for standard applications and display systems, as well as provide data availability without complicating network traffic.

At the foundation of the Control and Data Acquisition System, there are low level facilities, which actually accomplish the collection and transmission of the data and the management of set-point control. These facilities are distributed throughout the machine and are accomplished by Input-Output Controller (IOC) modules running the VxWorks Real-Time Executive from WindRiver. IOC’s use common executive and application software, but employ different database information to manage different devices and environments.

Above these IOC functional facilities, there are software drivers that bridge the functional facilities to the specific hardware implementations employed. These systems will be constructed in layers, such that new drivers for alternative hardware devices may be added or deleted from the system without having to re-write the functional software. The local database in the IOC will specify the numbers and types of modules interfaced by the specific IOC, and associate them with the named devices.

Also at the IOC driver level, are the network communication stacks to handle the fast synchronous data management, and the slower asynchronous status and periodic analog data. There will be requirements for Ethernet, field bus, and perhaps some custom high reliability self-correcting network protocols. These connections and their protocol requirements will be worked out in bottom-up analysis nearer to implementation time.

At the very lowest level, are the smart controllers, which actually operate complex physical devices. In most cases, these controllers will be located in or near the controlled device and will be connected to the IOC by a standard network or field bus connection. In general, the more capability pushed down into the device controller the better. As an example, BPM modules will have DSP processors embedded in the hardware which will facilitate the conversion of raw data into useful information, and perform a lot of data reductions for beam parameter measurements. Firmware for smart modules and controllers will be written in high level languages and managed as part of the software configuration system.

CONTROL ROOM FACILITIES: There will be a centralized Main Control Room from which the Accelerator is operated. From a Control System point of view; this facility will house the Control Room, Server Computer Rooms, and centralized Network facilities.

In addition, there will be subsidiary control rooms in the injector and damping ring areas for use in commissioning and troubleshooting. These facilities will not be built with any intent to include redundancy in their use profile.

Given the state of the art in wide area networking, and the distributed nature of the Control System, it will be possible to provide the capability to run the accelerator from minimal control facilities located at the Laboratories and Universities of the Research Collaborators. These facilities will require very tight network security, moderately high bandwidth, and reliable connectivity. It should be clear that remote users will require local staff to swap boards and performs hand-on maintenance activities.

GOALS AND STRATEGIES: It is our expectation that most if not all the Control System Application Software will be developed by a team of Software Engineers and Physicists from a number of Laboratories around the World. Software development tools, documentation, and facilities will be networked together such that this distributed team will be able to work together to develop, test, commission, and operate the Control System. Support packages (MATLAB/ORACLE/etc.), available on the Commercial market will be purchased rather than developed onsite.

While we recognize that Particle Accelerators are a somewhat unique target environment, we hope to use as much Off-the-Shelf software and as many commercially available software development tools as possible. Implicit in this goal is the recognition that we will have to make some adjustments in our desires and practices to accommodate the software obtained commercially. We will also have to use commercial packages with industry dominance and survivability or control copies of the source code.

Due to the size of the Accelerator and the period of time required to develop the necessary software, we expect to test and commission software throughout the construction cycle of the physical Laboratory facilities. Some software will have to be available very early in the construction cycle, and other systems can be developed as late as the early operating phases. Some software will never be completely done and will be modified and enhanced for the life of the Accelerator.

The size of the Accelerator and the proposed upgrade path require that the Control System scale rather cleanly to the size suggested by the increased energy proposals. Special care will be taken in the requirements, design, and implementation phases of the project to maintain scale flexibility.

ACCELERATOR LIFETIME EXPECTATIONS: It is reasonable to assume that the Construction Phase of the NLC project will take 5-7 years and that installed equipment will be tested, commissioned, and maintained over this period.

Once the facilities are built and into the Operating or Production phase, it is reasonable to assume that the Accelerator will be in operation for at least 10 years. This period could be considerably longer if the machine is upgraded, enhanced, or modified to accommodate follow-on Physics work. Thus the assumption is that the Control System will be in operation for at least 10 years and could still be in use after 15 years.

It is also reasonable to assume that the processor and networking capabilities will have to double every 3-4 years while technology changes every 2-3 years. The implications of these technology advances are somewhat more difficult to specify and even harder to predict the implications.

CONTROL SYSTEM ADAPTATION & EVOLUTION: Control Systems are notoriously difficult to replace once the Accelerator begins production operations. Due to the unique nature of the accelerator and its operational needs, millions of lines of code are developed which are not easily converted to run on newer computer hardware and use newer evolutionary languages and operating systems. Thus, it is important to develop the code so that new features can be added, and old systems reasonably supported over time, even though control system architectural models change and move forward every 8-10 years. Non-standard computer languages or language enhancements should not be used on any processing platforms.

Part of the flexibility of the Control System Architecture will come from software layering and networking schemes in a distributed environment designed to minimize the pain of long term maintenance and the addition of new facilities and equipment. Server systems will clearly buck this trend and need to be carefully considered.

Every effort will be made to use standard protocols and techniques in a structure where facilities can be changed out or substituted, and where maximum lifetime is obtained by using standards with long and successful lifetimes.

EXTERNAL NETWORKED INTERFACES: The very size of the NLC project requires that the operation and maintenance of the facility be distributed over a number of related control systems. It is likely, for example, that in addition to the Accelerator Control System, that there will be control systems to monitor the health and performance of the infrastructure subsystems such as electricity, cooling water, compressed air and gases, air-conditioning, cryogenics, fire, and physical security.

All of these systems will need to be networked together so that trouble shooting and fault diagnosis efforts are supported by all of the applicable systems as needed. There is a role here for an Enterprise Database from which device characteristics, parameters, and configurations readily available during fault analysis.

DATA MANAGEMENT & ARCHIVING: The size and complexity of the Accelerator suggests that anticipating the data collection requirements of fault diagnosis will be difficult. We propose therefore to include in the low level data acquisition capability of the SCADA, the ability to collect and archive huge data sets on a pulse-by-pulse basis. This mass data store will also reduce the contention for data among subsystems, but will require the ability to retain rather large data sets with fast response on the most recent data.

Data will be acquired on a pulse-by-pulse basis, passed to database servers, and eventually transported to tape for archiving. Data will be available on-line for a reasonable period, and then will be transported to bulk storage on tape or CD for archiving purposes. Archived data will be stored in off-site vaults as well as multiple locations on-site. This data may be compressed to control the scale of the storage challenge.

OPERATIONAL AND SYSTEMS SECURITY: System and Operator security are important in this machine to allow access only to qualified Operators and users either on-site or at remote control facilities. Effective security can only be achieved early at design period and must extend across the functionality of machine control as well as the computers and networks in the infrastructure.

Provision will be made for user authentication and encryption of data and control actions. Further security will be implemented by password access to Control System subsystem functionality as well as for system configuration and interlock databases.

II. GENERALIZED REQUIREMENTS:

LOW LEVEL FUNCTIONALITY:

Network & Protocol Infrastructure: Network traffic breaks down into distinct categories; at the highest level there are packets moving synchronously with the beam pulses, and there are asynchronous packets providing slower status and configuration information. These data flows break down into large data sets moving at high rates and slow rates, while other data sets are small in size and may be moving at fast or slow rates. The network bandwidth is the sum of these categories, analyzed for peak and average conditions.

1. **Networks Bandwidth** (Structure/Bus Types/Bandwidth)

2. **Device Boot/Reboot Capability**

Purpose:
For maintenance and diagnostic purposes, all smart control devices will accept a downloaded image for system, database, and configuration. This includes network and fieldbus devices, but not firmware for programmable logic chips (Xilinx). All similar devices will use the same firmware image with only configuration or database differences.

Use:
There will be a provision for retaining these images locally to facilitate reboot delay time (ROM, flash, not disk). All remote controller devices will have remote reset capability in their command strings as well as via some hardware network.

Firmware and database download must be fast for major control nodes, will not be a regular event (average one node, once daily), and must be secure and device specific. Checksums and image version must be readable by device ID to confirm proper loading and configuration. Hardware board type and revision data will be retrievable. Version data will be readable as system and configuration. (PPS devices will have more elaborate version control and verification processes.)

It must be possible to upload a ram memory usage so that states may be captured for analysis. This facility must be a function of the executive so that some process survives a fatal error and can upload the image.

Control nodes must be able to reload to the device set points in place prior to the reboot or reload, or reset to established configurations.

Procedure:
Streamlined procedures will be required for booting all such devices concurrently (i.e. return from power loss or Database upgrade). Ability to download a specific device with a standard or custom version system. Ability to select between ROM version locally (for speed) or downloaded version. Ability to perform soft reset and hard reset remotely if needed.

Dynamic compatibility checks to warn users that firmware versions may not be compatible or to load known compatible images.

Concepts:
Boot node with production, development, diagnostic, or special. Boot images as all, or data and config only. There should be a forced broadcast download for a ‘boot all’, that bypasses the normal request for image booting scheme.

It might be useful to provide a process for automatically uploading an image and then loading a known good image (last prod or other named image) for continuation. Facilities for returning messages or data to software engineering, while IOC is operational (various modes).

User Interface:
Provision to identify currently running version, select standard or custom versions for download, and ability to use ROM data for all or part of the download to save boot time. Ability to verify device and image downloaded as well as authorization for the download selected.

Facilities for developing and selecting from lists of devices to be downloaded, and lists of which firmware versions to load to sets of devices. Such lists should be stored and called like configurations.

Performance requirements:
Provision to reload system, data, and configuration in any individual node in less than 10 seconds, and all devices concurrently in 10 minutes.

System Impact: possible effects on others – speed - contention
Verification of correct images to expected IOCs.

Protocols:
Standard IP booting protocols should be considered, but performance may require low overhead custom protocols. Booting images to fieldbus devices will require its own in-house low-level transport protocol.

Services Used:
Some services have to work when the ‘IOC system’ is down. Services to up and download images may need to reside in

reset enabled PROM. Some totally separate reboot or watchdog system may have to be provided.

Taxonomy:

How initiated	manual with configuration help & verification
Frequency	not often – one IOC per day or less
Read/write	both
Shareable input	yes
Running state	normal running/diagnostic ok with normal running
Latency requirement	human
120 Hz	no (data and/or control)
Sync time	time stamp in msec range
Missing Data	intolerable (error detect, retry)
Geography (input)	local / global / none (offline)
Geography (output)	local / global / none (offline)

3. Device Firmware Download

Purpose: (very short top-level description of the facility) This facility will be responsible for rebooting and downloading of binary images to NLC embedded processors (includes all smart controllers and devices). Images for programmable logic should also be downloadable and verifiable across the network.

Use: (when, what conditions, how often, machine state) Booting is a users-initiated process, done on demand. It should support both cold reboots and incremental loading. Cold reboots are certainly incompatible with regular running: incremental loading is likely to be similar but perhaps less impactive. Reboot occurs relatively infrequently in a mature production system, but typical often in a development or debug environment.

Procedure: (major procedural steps)
Authorized Operators would select a target device, which would display a list of appropriate boot images from a version control and compatibility facility. The selection would be down loaded with a confirmation of correct transport. A verification step would read back the device code version fields to make sure that the right code got to the right device. Each step would be logged.

User Interface: (minimal description of I/O interface)
Users must be able to remotely initiate a reboot of a processor set. The user must be able to compose a list of images and target processors, with the download resulting from the union of the lists. These lists should have some persistence in order to facilitate use of pre-existing palettes of such lists. Facilities should exist to check the firmware versions and code date before and after the download.

Performance requirement: (precision, latency, response time, reliability)
There will be a large number of CPU's in the NLC, with at least 1000 IOC nodes and an unknown number of other embedded systems, many of which will need to be rebootable.

It must be possible for the operator to specify at the time of invocation the set of images to be downloaded and the targets of this download. Once should be able to incrementally load and unload images. This not only avoids the time involved in a reboot, more importantly, it preserves the processor state, which is especially important while debugging.

A method whereby an individual processor may request a particular file based on its identity from the downloader should be provided, although it is not anticipated that this will be widely used in a production system since it would seriously impact efficiency.

In order to avoid development code from sneaking into production, the specification of the production code to be downloaded must be kept separate from the development environment.

While not a strict requirement, given the size and the geographical distribution of the system, one should consider whether processors should be booked or reserved for use. This is not meant to be a security issue, but merely a method to avoid two or more developers from inadvertently using the same processor.

Response Time:
The entire NLC Control System should boot quickly on a human timescale. The pull method employed by most real-time operating systems heavily taxes both the file servers and the physical download medium; in such a large system it may be incompatible with acceptable response times.

These performance requirements will likely be met only by resorting to a multicast system. This takes advantage of the fact that the same set of code goes to multiple processors most of the time. (It is anticipated that most code is not targeted for individual processors, but rather is tailored to the processor when it arrives by parameters stored locally on that processor.

Reliability:
Reliability comes in two flavors. The first is ensuring that the downloader reliably moves the specified set of images into the specified set of processors. The second flavor, is to verify that the collection of code both within a processor and across processors is compatible. This implies some sort of dynamic version control and verification. Static version control is typically implemented through a software release system, while necessary, is inadequate in a dynamic environment (i.e.,

having the right images on disk does not guarantee that the right images are in the processors). While the downloader cannot perform this function without help, it can act as the implementing agent by providing the ability to query a set of processors for the identity, both by name and version, of it resident software. It should be the responsibility of another facility to digest the result of such a query.

Cold Reboot:
A reliable method of initiating a cold boot must be provided. This cannot be an inband software solution since the boot must work independent of the state of the processor. The granularity of such a reset signal is not specified. While providing the ability to reboot an individual processor would be nice, it may not be necessary. However, given the number of processors, the system must provide the ability to reset groups of processors. In order to reduce confusion, the specification of the collection of processors to be rebooted (reset?) should be exactly the same as when specifying a collection of processors to download.

System Impact: possible effects on others

Services Used: ‘toolbox’ – requirements on low-level utilities

Taxonomy: quick lookup table of usage parameters

How initiated	manual/free-running/both/remote/scheduled
Frequency	continuous / as needed / every N minutes
Read/write	read / ephemeral write / write
Shareable input	yes / no
Running state	normal running / diagnostic incompatible with normal running
Latency requirement	N pulse / human / none
120 Hz	yes / no (data and/or control)
Sync time	pulse / longer time / doesn’t matter
Missing Data	intolerable / can manage / can’t retry
Geography (input)	local / global / none (offline)
Geography (output)	local / global / none (offline)

4. NLC Control System Time Serving

Purpose:
There is a system facility required to synchronize real-time systems, workstations, and servers to a common local clock time for time stamp purposes. Time coordinated systems include all the smart micro/IOC type devices, real-time systems, and servers, but not individual device controllers and field bus systems.

This facility should use standard protocols and transport to the extent possible (i.e. something like NTP). The master time for the NLC will likely be obtained from Public Primary Servers, which are operated at

Stratum One and directly synchronized to Coordinated Universal Time via GPS. Backup GPS facilities onsite are a possibility.

This time and synchronization process should not be confused with the real-time fiducial signal on the Main Drive Line for beam time coordination, which is part of the Timing System.

Use: (when, what conditions, how often, machine state)
This facility would be used to maintain synchronization of systems at th 1 to 10 msec level, in order to maintain parity of ‘system clock’ increments which are on the order of 10 msec. Such parity would allow system driven error and archive logs to be compared and/or integrated.

Timing modules will be smart devices, which will have substantial functionality in the module. There will be substantial firmware in the module, with detail analysis and design yet to be accomplished.

Detail Accelerator coordinated heart beats would be based on the beam pulse ID. All beam-time devices would ‘time stamp’ events with the beam pulse ID, bucket number (where applicable), and the local device time stamp (if available).

Procedure: (coarse, implementation-independent list of major steps)
Low level smart beam control devices would report beam-code ID to the lo distributed control node which would append its network time stamp and forward the message to the logger process. The logger would sequence th messages by time of arrival and retain the beam code and local timestamp for reference.

The beam-code broadcast could supply both beam pulse ID and clock time t wide range of smart devices.

User Interface: (minimal description of I/O interface)
The user needs a convenient way to scan and localize logged data that reasonably bounds the events of interest, which comes from the logger ti and date stamp (in increments of that servers time clock). Further analysis and association will be based on the beam pulse and distributed device time stamp.

Search facilities will be needed that can scan for devices, node names, error message types, and event context. Considerable through needs to g into the scan and analysis features provided.

Control panels to check node time and to set device or controller time clocks will be needed. Facilities to scan time for agreement across the accelerator networks will be required. The NLC synchronization subnets are large and intricate, with many opportunities from misconfiguration and network problems. Tools for monitoring and debugging will be needed for identifying and fixing problems. Facilities will be required to manage the network time facilities, sources, and timeservers.

Data to be collected based on Universal time (UTC) and translated by the API from archived data to deal with standard or daylight savings time.

Performance requirement:(precision, latency, response time, reliability)

Performance will be a tradeoff between the accuracy required and the lev of network traffic involved in obtaining it. Other tradeoffs include th quality of the crystals in remote equipment and their relative time stability over time and environment.

Multiple redundant timeservers onsite, and diverse network paths around lab will be appropriate to achieving the high accuracy and reliability anticipated.

It would be useful to maintain a few millisecond tolerance over several hours to minimize network traffic in support of time synchronization.

Certain authentication will be needed to protect the time service and servers from protocol attacks and denial of service problems.

System Impact: (possible effects on others)
For the time synchronization facility to be wrong for more than a beam pulse period could lead to problems resolving failures or unusual events Synchronization to 5 milliseconds or better should not be a major proble better than 1 msec is another story.

Services Used: ‘toolbox’ – requirements on low-level utilities
There will be network services to implement the protocol (standard and/o other), and system services in an executive to manage the timer systems the processor. There is no proposal for time services in systems too sm to support a real-time executive.

Taxonomy: quick lookup table of usage parameters

How initiated	manual configuration/ scheduled operation
Frequency	scheduled periodically as needed
Read/write	read for test/ write for setting
Shareable input	shared source/ all data time-stamped
Running state	continuous/ nonimpactive test & set
	API Daylight Savings & Leap Year adjusts
Latency requirement	Small number of milliseconds
120 Hz	Beam-pulse & bucket data included
Sync time	Periodic to within N milliseconds
Missing Data	can manage / can retry
Geography (input)	Global
Geography (output)	Local

Field Bus Utilization: The Fieldbus picture is Application dependent, and generally includes small data sets transmitted rapidly for MPS (i.e. 1394), or more slowly for devices like Power Supply Controllers (i.e. CAN). It is likely that more than one fieldbus will be utilized, although maintenance gets more complicated with each additional supported bus. Current technology provides many possibilities, most of which may not be applicable at implementation time (the R&D on this subject is being delayed deliberately).

Programmable Logic Controllers will be introduced in many areas of the Accelerator, and these devices are rich in proprietary fieldbuses and protocols. An effort should be made to minimize the number of PLC vendors and network solutions site-wide.

In the cases where proprietary designs are implemented on-site, the choices should be limited to those which provide chips to support the physical layers and protocols. These network solutions should be built on daughter cards for maintenance ease, and flexibility of following technology and performance improvements. Test equipment and bus analyzers for these networks are necessary.

In general it is useful if the bus is fiber and copper friendly, to support long haul communications, provide good electrical isolation, and apply copper in high radiation areas. Fieldbus length parameters have to exceed half of NLC sector spans by 20% for reliability.

Fieldbus utilization

Purpose: (very short top-level description of the facility)
Provides point-to-point network connectivity on links where packets are small and simplicity is a virtue. Some bus choices are fast and some are slow.

Use: (when, what conditions, how often, machine state)
Fieldbus connections will be used between data concentration devices and a host of smart modules. These modules may be located anywhere in the machine, and the total node count will be very large. Most implementations will involve small packets sent periodically, although, some fieldbus connections must have low latency for MPS sensors.

Procedure: (coarse, implementation-independent list of procedural steps)
Messages containing data and functional commands are routed to/from appropriate modules. Packet acknowledgments are required.

User Interface:(minimal description of I/O interface)
These are machine interfaces at low levels in the control system. Implementations will be in silicon, on standardized daughter boards.

Performance Requirements:(precision, latency, response time)
Most links are slow, but MPS sensors will require fast response times.

Reliability: (Data quality, uptime, failure implications)
Reliability has to be excellent, and repairs through daughter card substitution must be quick and easy, with minimal network addressing complications. Diagnostic tools and network analysis and test boxes will be required.

System Impact: (possible effects on others)
System impact is high as many of these devices will be part of protection systems. Redundancy and self correction protocols are not required.

Services Used: (reqs on low-level infrastructure & utilities)
Working at very low levels in control system. Requires message services, protocol management, and traffic routing, and message complete checking.

Taxonomy:(quick lookup table of usage parameters)

How initiated	full time element of infrastructure
Frequency	continuous, and at least beam pulse rate
Read/write	read & write
Shareable input	yes
Shareable output	yes
Running state	normal running with periodic diagnostic packets
Latency requirement	at least pulse rate
120 Hz	yes (data and control)
Sync time	pulse
Missing Data	intolerable / can’t retry
Geography (input)	local
Geography (output)	local / global

PLC Proprietary Bus utilization

Purpose: (very short top-level description of the facility)
Provides point-to-point and loop network connectivity on links between PLC devices.

Use: (when, what conditions, how often, machine state)
These fieldbus connections will be used between data concentration PLC’s and higher level supervisory PLC modules/boxes. These modules may be located anywhere in the machine, although for the most part they are embedded in systems for interlock management and PPS.

Procedure: (coarse, implementation-independent list of procedural steps)
Messages containing data and functional commands are routed to/from appropriate modules. Packet acknowledgments are required. Only PPS implementations will be redundant.

User Interface:(minimal description of I/O interface)
These are machine interfaces at low levels in the control system. Implementations will be supplied by one/few vendors, to minimize the spares and support logistics. All of these devices have minimal user interfaces at the physical level and remote user and development support at the PC level.

Performance Requirements:(precision, latency, response time)
Most links are slow, but MPS sensors will require fast response times. Vendors will have to supply a range of devices in performance and cost.

Reliability: (Data quality, uptime, failure implications)

Reliability has to be excellent, and repairs through daughter card or module substitution must be quick and easy, with minimal network addressing complications. Diagnostic tools and network analysis and test boxes will be required. Vendor software must be able to report low level errors at higher levels so that problems can be picked up early.

System Impact: (possible effects on others)
System impact is high as many of these devices will be part of protection or interlock systems. Redundancy and self-correction protocols are not required in most instances. In most cases these physical layers and protocols are vendor specific and not industry standards, making diagnosis and repair more difficult.

Services Used: (reqs on low-level infrastructure & utilities)
Working at very low levels in control system. Requires message services, protocol management, and traffic routing, and message complete checking.

Taxonomy:(quick lookup table of usage parameters)

How initiated	full time element of infrastructure
Frequency	continuous, and at least beam pulse rate
Read/write	read & write
Shareable input	yes
Shareable output	yes
Running state	normal running with periodic diagnostic packets
Latency requirement	at least pulse rate
120 Hz	yes (data and control)
Sync time	pulse
Missing Data	intolerable / can’t retry
Geography (input)	local
Geography (output)	local / global

LOW LEVEL UTILITY INFRASTRUCTURE

Data Concentration scheme
Messaging system (text & codes)
Error log (exe alarm/page person/run code)
Event Logging (client time stamp/tag driven?/msg distributio
Buffered Acquisition

Fast streamed data acquisition (mass store)

WEB as a key technology
Security

Crate Status/poll

SCADA Level Capability

1.Contention Handler

Purpose:
Handle competing demands on Control System devices and services (magnets,klystrons,etc.) and pseudo-devices (machine parameters), efficiently as possible while attending to priorities. The NLC is bigge than the SLC and Control System functions must be significantly more automated. The ad-hoc mechanisms used to avoid conflicts, which were barely adequate for SLC will certainly not do for NLC.

Users:
Procedures initiated interactively (from Correlation Facility), individu device control, etc.). Regularly scheduled and automated procedures. Continuous processes like feedbacks and watchdogs.

Conflict Situations:

* Multiple interactive users writing to devices or pseudo devices (machine parameters)
* Continuous users and one-shot interactive users
* Writing to a device (pseudo-device) may affect downstream info
* Multiple readers of by-request data
* Operator versus automatic process

Implications for Functionality:

- Need write locks, perhaps with associated queues for devices and pseudo-
- devices. (At what granularity? – release problems)

* Need a low-overhead way to tell continuous users to hold of temporaril
* Need read-share for significant quantities of 120 Hz data

Digital Status Acquisition
Analog Data Acquisition
Parameter Scanners (beam scan/wire scan/laser scanner)
Magnet Mover Control
Magnet Power Supply Control
Synchronized Device Control (magnets/movers/klystrons)
Save and Restore IOC
Configuration Control & Restore
Virtual Area Designation (DGRP)

Accelerator Specific Subsystems: There are a number of Subsystem Devices or Systems which must be operated or supervised by the Accelerator Control System. They are discrete and separate systems in their own right, and as custom installations, must be interfaced to the Control System. These include the following:

1. Low Level RF Control: The Low Level RF System provides the accelerating potential to accelerate the beam through a structure. The RF system is capable of providing RF power safely and in conformance with the needs of the accelerator. The control system provides facilities to monitor the behavior and parameters of the subsystem, means to accomplish adjustments and calibrations, and provides first order diagnostics and error condition management and archiving.

There are Structure BPM devices included in the acceleration structures to measure phases and beam positioning within the structure. The BPM and other sensors reside in the structure and are serviced by electronics in adjacent crates, which use fiber and Ethernet to transmit performance data back to the RF system controllers upstairs in the gallery. In this case the Control facilities include only the fibers between the elements of the RF subsystem, and crate monitoring facilities.

The Control System must communicate with these RF systems via Ethernet (over fiber) for user control purposes. There will be a set of operator console screens and functions to operate and diagnose these RF systems.

It would be useful to provide some fast error logging and data element archiving that could be used for diagnostic purposes. Some number of parameters per device will be archived as a matter of normal operation.

Modulators used to power these RF stations will also require interfacing to smart controllers for operation and diagnostic purposes. In addition, there will be latched status points for monitoring the performance and interlocks of these modulators (temperature, water flow, etc).

2. Timing System Control & Monitoring: The Timing System provides the heartbeat for the coordination and timing of the Accelerator. The NLC Master Oscillator provides a time and phase reference for the Accelerator, which is distributed over a phase stabilized fiber network to each sector with ___ pico-second accuracy. This signal carries a fiducial, which allows down-counters to trigger beam events for the hardware.

Phase stability is enhanced by burying the fiber cables in low temperature-drift trenches, and uses active temperature feedback to complete the stabilization. A short section of fiber resides in a temperature-controlled oven-refrigerator, with feedback to compensate the temperature changes in the longer section of the buried cable. Electronics measure the round trip delay in the full fiber run and signal the temperature feedbacks on a course scale and modulate the laser frequency on a much finer scale to achieve fast precise control of the apparent length of the fiber.

There is a fiducial generator to superimpose a negative fiducial on the _____ MHz waveform. This trigger starts counters, which have been preset in each beam-line device to mark the time of beam passage. The timing interface counter is a standard counter daughter card which can be mounted as a mezzanine on a variety of beamline modules. Presets for the counter originate in the Master Pattern Generator, and are passed to the counter via the IOC on the fast synchronous networks. Each counter has several counts for different modes, which are selected from memory for each machine pulse and can be reloaded on a pulse-by-pulse basis if needed.

The timing electronics will be monitored and controlled by an internal smart controller. This controller will likely by a PC chipset, with provisions for new firmware to be downloaded across the network. The processor will monitor the performance of the timing electronics, signal fault conditions, and provide status and analog values (10s of bytes every pulse as required). Signals to be monitored and archived will include voltages, amplitudes, phases, and delay intervals. The likely bandwidth approximates 10 kilobytes per pulse for some 200 master or slave timing chassis.

Firmware will enable monitoring of device performance and signal any degradation in signal level, phase stability, or timing. Capability for the processor to command compensations in the electronics, based on some short term buffering of parameters for sampled analysis will be included in the design. Failures will be reported with diagnostic messages to describe the nature of the failure. Firmware development will be substantial.

The Control System must be able to download a firmware image to flash memory in the controller (one of several available versions), and collect beam pulse marked status and parameter data. The Control System will also have to download parameters for operating points and fault thresholds on an infrequent and logged basis. The device controller will message fault conditions via the Control System for logging and action by Control Room Operators and maintenance personnel.

A GPS system will tie the timing system to a universal standard. This signal will be used to synchronize a Rubidium oscillator, which will be the actual time reference.

Timing System details can be found on the Controls WEB pages www-project.slac.stanford.edu/lc/local/notes/timing/

Component Device Parameters:

Stable Fiber Optic Transmitter:

Function-provides phase & fiducial transmission over long fiber links
provides feedback for link stabilization

Internal Processing-real-time active feedback, status, and control

Data rate-8x10E6 16 bit samples/sec for 2 channels
~8x16 bit, 100 hz ADC + ~4x16bit, 100 Hz DAC

Level 1 processing-8x10E6 32 bit address adds/sec (Xilinx?)

Level 2 processing-10E4 flops feedback @ 120 Hz rate

Status & Control-~8 analog status/software download/1 mb data upload
internal processor/firmware/ 2+ mb memory

Special-fast connection to MPS

Format-100 units of VXI module or dedicated box

Location-central campus

Fiber Optic Sector Receiver:

Function-receives optical signal on long fiber links
Converts to RF and local optical distribution
Internal processing- no processing but some monitoring
Status & Control-~8 analog status
Special-fast connection to MPS
Format-100 units in dedicated rack mounted box, fieldbus connections
Location-2 in each sector alcove

Phase Control Unit:

Function-receives optical signal on redundant long fiber links
selects operating channel
provides narrow band VCXO for MPS
Internal processing-10 kHz PID loop, status & control @ 120 Hz
Status & Control-10 kb data buffer upload, software download
Special-fast connection to MPS
Format-rack mounted box upstairs, modules downstairs
Location-20 per sector upstairs (1000 total)
100/sector downstairs (4000 total)

Trigger Module:

Function-PDU style device, signals from phase control unit
Internal processing-pattern broadcast recognition & verification
large memory for stored patterns
performs pattern consistency check
loads timing registers based on patterns
Status & Control-monitors pattern broadcast
Provides diagnostics, status & control capability
Special-fast read/write connection to MPS

Beam Position Monitoring (linac/rings/interaction area)

Dedicated System Level Controllers: While most IOC devices use a standard oftware set, and use database configuration information to customize their functionality to their specific role, there are some custom standalone

Master Pattern Generator:

The Master Pattern Generator (MPG) is an embedded processor device, which is used to control the beam patterns in the Accelerator(s). There will be two inter-connected processors, one for each linac. The MPG will schedule beam production pulse-by-pulse (electron and positron), schedule beam-synchronized data acquisition, and assist in Machine Protection through protection device control. While the MPG schedules events, the precise timing for these events is derived from the timing system and its PDU capability.

Purpose:
Schedule and synchronize pulse-by-pulse events within the Accelerator. Assist in startup sequencing, emergency shutdown procedures, and machine protection functions.

Use:
The MPG runs at all times the Accelerator is functional in order to coordinate pulsed operations, and schedule machine states and modes. To reduce complexity, one MPG will operate the electron linac and anther coordinated MPG will operate the positron linac. The MPG executes startup and shutdown sequences in order to protect the accelerator from errant beams or unscheduled beam acceleration.

Procedure: coarse, list of major procedural steps
On each 480th of a second, the MPG issues a command string broadcast across the fast synchronous network to configure IOC scheduling, event configuration, and beam related tasks to be accomplished. These events are executed based on the Main Timing System fiducials.

The MPG posts events and errors to the central logging server, provides displays of current and pending events, and queues actions and tasks in buffers for event reconstruction. Events are based on Beam Group instructions presented to the machine in an orderly scripted form.

The MPG will schedule beam based device alignment, orbit adjustments, and energy management operations.

The MPG receives Machine Protection signals from abort systems and protection interlocks in order to reschedule beam production and activate protection dumpers as required.

Concepts:
The MPG should understand areas of the machine capable of independent operation, recognize machine development and production configurations, and manage beam intensity, beam emittance, and beam pulse bucket structure. There is no concept of rate-limit in the NLC. Are time slots still a factor?

User Interface:
A scripting language is required to clearly describe machine sequences, limit conditions, coordinated events, and device actions to be scheduled. User must be able to generate a script (online or offline), load it in and play it back in a simulated mode, and then present it for execution once ready and tested. Authorized user identification and actions will be logged.

Displays will be necessary to view past, current, and future beam codes. Buffers retaining pas events and command strings must be storable to file or analyzed online. Buffers will be storable with beam pulse identification and coordinated time stamps.

Automated response signals and protection system event signals will be logged to the logging server system.

Performance requirements: precision, latency, response time, reliability

The MPG must operate at low utilization levels of memory and cpu utilization 9average and peak) such that device performance does not impact proper scheduling of events and sequences. MPG devices must be very reliable and quickly diagnosable.

System Impact:

As the big show coordinator, the MPG effects everything. It should be designed for clarity and user friendly operation and debug.

Services Used:

Event logging, archiving, fast network communications

Taxonomy: quick lookup table of usage parameters

How initiated	manual setup and start, continually running
Frequency	continuous
Read/write	read / write
Shareable input	yes
Broadcast output	all beam coordinated devices
Running state	normal running
Latency requirement	beam pulse / human
120 Hz operation	yes
Sync time	pulse
Missing Data	intolerable
Geography (input)	local / global
Geography (output)	global

Gun Controllers: ?

Positron Target Controller: ? Hardware

Damping Ring Energy Feed-forward: ? Hardware

GPIB Instrumentation: There will be many instruments built for VXI crates or stand alone boxes which communicate via GPIB networks. Instruments most likely to appear are oscilloscopes, random pattern generators, interferometers, signal and network analyzers, frequency counters, voltmeters, and pulse generators. These are network cables workable over 10 of feet and serviced by Ethernet to GPIB converter boxes.

Purpose: very short top-level description of the facility
High performance instrumentation distributed along the beamlines. Remotely monitored, controlled, and programmed.

Use: when, what conditions, how often, machine state
Remotely controlled equipment monitoring beam conditions for all states of the accelerator. These units are located in distant or difficult to reach areas where remote control is essential. Physical conditions are often difficult.

Procedure: list of major procedural steps
Operators use vendor provided user interfaces to control the facilities of the remote device. Messages transport command and configuration information to the device.

User Interface: minimal description of I/O interface
Special x-window displays are used which allow front panel emulation of the instrument controls, or dedicated software drivers to setup and configure the instrument. Software systems generate the signals to run the boxes and transport the message packets over the Ethernet. X-window facilities are common and getting better and faster.

Performance requirements: precision, latency, response time, Reliability

These are real-time and near real-time devices used as operational and diagnostic instrumentation. In many cases these devices will be required for beam operation rather than just monitoring, and as such require high reliability. These devices may have to supply data at 120 Hz or better. Response time will be fast and latencies short.

System Impact: possible effects on others
Most GPIB instruments are near real-time and not essential for beam operations. Other devices like function generators are production devices that would stop the accelerator. Control bandwidth is low, but data bandwidth might be high.

Services Used: ‘toolbox’ – requirements on low-level utilities

At a minimum, there will be configuration and setup facilities for these instruments. In addition, there will be requirements to archive data from these devices and log error messages and conditions. Some of these devices may have to be knobable.

Taxonomy: quick lookup table of usage parameters

How initiated	manual/remote setup and configuration
Frequency	continuous / as needed / beam rate
Read/write	read / write
Shareable input	no
Sharable output	yes

Running state	normal running / diagnostic
Latency requirement	N pulse / human
120 Hz	yes data / no on control
Sync time	pulse / longer time
Missing Data	can manage / retry
Geography (input)	local / global
Geography (output)	local / global

Smart Commercial Instrumentation: Complex instrumentation purchased as stand alone boxes, or as devices supported by PC computing devices. Many of these boxes will use standard Ethernet network facilities, and some provide x-windowing capabilities. Synchrotron light monitors, interferometers, spectrum analyzers, and residual gas monitors are included. Some of these devices communicate via Ethernet connections.

Purpose: very short top-level description of the facility
High performance instrumentation distributed along the beamlines. Remotely monitored, controlled, and programmed.

Use: when, what conditions, how often, machine state
Remotely controlled equipment monitoring beam conditions for all states of the accelerator. These units are located in distant or difficult to reach areas where remote control is essential.

Procedure: list of major procedural steps
Operators use vendor provided user interfaces to control the facilities of the remote device. Messages transport command and configuration information to the device.

User Interface: minimal description of I/O interface
Special x-window displays are used which allow front panel emulation of the instrument controls, or dedicated software drivers to setup and configure the instrument. These devices are not easily connected to the control system, and generally use proprietary software. X-window facilities are common and getting better and faster.

Some of these devices may be operated by a remote terminal package which allows one PC (in the Control Room) to operate another remote PC as if the operator were actually sitting at that PC’s local keyboard.

Performance requirements: precision, latency, response time, Reliability
These are real-time and near real-time devices used as operational and diagnostic instrumentation. In a few cases these devices will be required for beam operation rather than just monitoring, and as such require high reliability. These devices may have to supply data at 120 Hz or better. Response time will be fast and latencies short.

System Impact: possible effects on others

These devices are networked on individual network cables (fibers) where there are no determinism issues and the required bandwidth is guaranteed. Control bandwidth is low, but data bandwidth might be high.

Services Used: ‘toolbox’ – requirements on low-level utilities

Most of these devices are not connected to the control system due to the lack of networking facilities on the vendor side of the Ethernet card. Some of these devices are capable of transmitting data packets which can be collected by network workstations for archiving purposes.

It is very important to understand what these boxes are intended to do and what kind of vendor networking and remote capabilities are available. In a very few cases, there are control system drivers available from vendors. Consultation with Control Software Engineering is important early in the game for these devices.

Taxonomy: quick lookup table of usage parameters

How initiated	manual/remote setup and configuration
Frequency	continuous / as needed / beam rate
Read/write	read / write
Shareable input	no
Sharable output	yes
Running state	normal running / diagnostic
Latency requirement	N pulse / human
120 Hz	yes data / no on control
Sync time	pulse / longer time
Missing Data	can manage / retry
Geography (input)	local / global
Geography (output)	local / global

Unusual devices ?

Feedback Systems

Non-linear Feedback systems
Beam-based Feedback
Parameter Control (PID)
Software based Fast Feedback/Slow Feedback
File driven position & angle feedback control

Detector Data Links:

There are a few Accelerator or Detector systems, which need to pass data back and forth from the Control System. In most cases, this data traffic is small, but a majority of the data is beam pulse related. In the case of the mass data streaming of raw data out of the accelerator, the data rate is beam rate and the data volume is large (large has yet to be defined clearly).

Detector data passing: This is low volume data passed to the detector database, which parameterizes the beam conditions and parameters for each beam pulse. This data is written to tape along with the data collected by the detector.

IP Beam Quality data and optimization displays: This is data collected by the detector, which can be used by the Accelerator Operators to tune the accelerator for optimal beam quality. It is generally on the order of hundreds of data elements at beam rate.

Remote data analysis or event reconstruction: This is data which is generally extracted from the database and archives for off-line analysis at some remote (non accelerator site computer) location. The data is moved infrequently but may be large in volume. There are no requirements to move this data at critical times during the day.

Mass Streamed Accelerator Data: This is parametric and status data being extracted from many/all devices on a per beam pulse basis. This data may be archived, or may be distributed to many real-time devices, functions, or displays.

MIDDLEWARE

Definition (as applies to NLC)

Application Development

Methodologies
Languages
Platforms
Performance Requirements
Real-time capabilities
Database
Tools

Services

Lifecycle management
Persistence
Directory & naming
Events
Time and timing
Transaction processing
Security

Device Abstraction

Contention control & prioritization
Distributed objects management
Device name & location database
Standard devices & drivers

Server Facilities & Connectivity

Database Servers (RT database/BPM database)
WEB servers and archives
Analytical Engine Platforms
Domain Name Serving

Accelerator Parameter archives
Proxies
Secure Gateways
Network access
Mass Archive Storage
Master Tool/Compiler Server

Application Level Infrastructure

1. NLC Requirements: Correlation Facility (JRB)

Purpose:
Flexible acquisition, analysis and display of (finite-length) sequences of correlated data from different sources. Must also allow for correlated control of devices and pseudo-devices, such as feedback loops (in what follows ‘device is usually shorthand for ‘device or pseudo-device’).

Use:
The facility will be used frequently, both in an interactive mode and in canned procedures. Requests to the Correlation Facility may be disruptive of normal running, depending on details of the control component (if there is one) of the request; however, many typical uses will be compatible with normal running.

Procedure:
Acquire a sequence of data points for a collection of specified devices approximately concurrently; meanwhile optionally stepping one or more controlled devices. Time may also be uses as a step variable.

If there are control devices, for each acquisition point the facility will wait until the new value is reached (or settle time has elapsed?) before acquiring data from read-only devices. Control devices are returned to their initial states at the end of the sequence.

User Interface:
User (human being, script, other code) specifies data to be acquired (i.e., sources), time structure of request, and devices to be controlled along with a sequence of set points and, if applicable, settle times for each controlled device.

Acquired data are available to calling code and to the interactive user for analysis (fitting, filtering, transforms,...) and display. Standard fits (e.g., polynomial, trigonometric) and plots (histograms, scatter plots) will be readily available to the interactive user. The programming interface will include the ability to supply custom analysis and plot routines

Interface Issues:
Contention control will be necessary for some uses of the Facility, depending on devices involved, and impact on beam.

Require some degree of uniformity of interface of devices to be read and controlled so that the Facility may handle them in a generic fashion.

Controlled devices should provide some form of read-back or other acknowledgement that they have reached the requested set-point (or are not going to).

Services Used:
Plotting (scatter plots, histograms, ...), fitting (polynomials, trig functions, ...) filtering and analysis. That is, these are probably not built into the Correlation Facility itself, at least not in a tightly coupled way, but are available as separate packages.

Access to all devices and pseudo-devices for which one might wish to gather data or issue control operations in a correlated manner.

Taxonomy:

How initiated	manually; perhaps automatic-cyclic for some appl.
Frequency	as needed (probably often)
Read/write	read and often ephemeral write
Shareable input	often yes
Running state	often but not always compatible with normal running;

	Depends on application
Latency requirement	human interactive time
120 hertz	may read 120 hertz data
Sync time	typically fraction of a second
Missing data	can tolerate (but caller may have to take some action, such as retry)
Geography (input)	may be global
Geography (control)	may be global

Plotting and Graphics Facility

User Interface

Scripts & Script Driven Applications

Macro Facility

Mathematical Analysis Tools (MATLAB)

Alarm Handler (Reporting/Priority/Analysis)

History & Archiving Facility

Linac/Ring Modeling

Linac Energy Management

Purpose:
Adjust the strength of the Linac quadrupoles to match the current energy profile.

Use:
This procedure is done for initial setup of the main and injector linacs using a pilot beam, and then is executed at Operator request during normal operation. In the main linacs, the energy is held fixed by a feedback at each diagnostic region and the energy is calculated separately for each section of the linac (from a modeling package).

Procedure:
Calculate the expected energy at each quadrupole base on the phase and amplitude of the upstream RF structures (data from the RF subsystem). Any overall phase controls (i.e. phase ramp) must be included. Compare the total calculated energy gain for the region with the setpoint of the downstream feedback and calculate an overall correction factor to be applied (fudge factor). Scale the energy for each quadrupole by a fraction of the correction factor proportional to distance down the beam line. This applied the correction gradually over the full region. Calculate the desire quadrupole strength based on the model and the current energy. Implement the new quadrupole values. Feedbacks should in general remain on while these adjustments are being made.

Analysis:
Calculate linac energy profile and determine new quadrupole settings. This procedure may always be used at Operator request with the results reviewed before implementing. If sufficiently robust, it may later want to be run autonomously without Operator intervention, with OC and errors flagged to displays and logs.

Taxonomy:
Single user, used for setup and intermittently during normal operations. Procedure should execute as efficiently as possible. Changes device setting and updates the machine database. Must detect bad or missing data and reject sample, flag error.(keep going?)

User interface:
User selects range of devices, or regions, etc
User sees graphic display of energy profile and proposed changes.
Records approval and executes changes.

Services used:
* Access to readings of all relevant phase and amplitude information, klystrons and others
* Access to feedback setpoints and current actuator values
* Access to model value for quadrupole strengths
* Changes magnet strengths, updates database
* Graphic display of values versus position

Optical Modeling & Tool Packages (MAD/DIMAD/LIAR)

Optical Modeling & Modeling Tools: Optical modeling and simulation tools are necessary offline tools for the development of the Accelerator Optical Decks. Optical deck development is being done with the MAD package, which has been updated substantially from the CERN version (unfortunately, this package is written in Fortran, and will either have to be integrated into C++ based systems, or rewritten in C++). Ray tracing simulations are being done with LIAR. Space charge analysis in the injector is using PARMELA.

Purpose:
Provide tools for the development of the Accelerator Optics decks.

Use:
Optical development and simulation tools are needed in the off-line development phase of the project, and some set of these tools will have to be implemented in real-time form for Operating the Accelerator. The on-line version will be used to check the accelerator optics and change energy profiles.

Procedure:
Investigate implications of various optical configurations, including chromatic corrections.

User Interface:
For the time being a standalone package using optics decks will be run on individual workstations.

Performance requirements:
Application has to perform analysis at human speed based on rather static data and configurations. Processing power is mostly computational.

System Impact:
Off-line analysis use has almost no impact on others. Data files may be exchanged. Recent high level of enhancements may lead to debug problems.

Services Used:
Currently no Control System interface at all. On-line version will require access to model configuration, magnet settings for quadrupole and sextupole strengths, and energy profiles.

Taxonomy:
Single user performing efficient analysis on single user data sets

How initiated	manual/remote/on demand
Frequency	as needed during design
Read/write	read / write
Shareable input	no
Running state	offline standalone
Latency requirement	human
120 Hz	no
Sync time	doesn’t matter
Missing Data	correctable
Geography (input)	local
Geography (output)	local

Ring BPM systems
Linac BPM systems

Beam Management Applications

1. Beam Based Alignment of Quadrupoles and Sextupoles (NP)

Purpose:
Measure the offset between the magnetic center of each quad or sextupole with respect to the readout center of the BPM captured in it.

Use:

This procedure is done infrequently, on a time scale of weeks/months. It is a setup procedure not run during normal operation. It would use a pilot beam, a 1 bunch train at 120 Hz, which would vary from low intensity, high emittance to nominal parameters.

Procedure:
For a range of devices, step through each device one at a time, acquiring data at different magnet strengths. Take a reference orbit. Change strength of magnet and allow a settling time. Read ~ 40 BPM’s for ~ 200 pulses. Fit the difference orbit for each pulse for incoming orbit; reject errant pulses, fit for the kick at the magnet. After looping over ~ 5 steps, (mini) standardize the magnet and move to the next. Feedbacks downstream of data acquisition devices would remain on. Procedure should handle feedback within the range automatically.

Analysis:
Fit the average kick at each step to find the magnet center/BPM offset Linear for quads, quadratic for sextupoles. Update the BPM offset in the database. Data should be saved for later examination. For commissioning of the accelerator (and software) one will want to review all data before implementing. Later the analysis and database updating should be completely automated, with QC and errors flagged. Data quality may be better if offsets are implemented as they are determined.

User Interface:
User selects range of devices, number of steps, step size, settle time, etc. User sees a summary table of measured offsets but may examine data for individual devices and display fitted orbits for individual pulses.

Performance Requirements:
The large number of devices involved requires a fully automated procedure optimized for speed. Typical resolution required is 1.2 um in the main linac, 20 um in the damping rings. Simulation facilities are needed.

Services Utilized:
120 Hz DAQ of ~40 BPMs for ~ 200 pulses synchronized with
magnet
strength changes.
BPM difference orbit fit including kick at magnet
Magnet control – setting & standardize (mini standardize)
Model matrices
Plotting and fitting of linear and quadratic functions
Data filtering
Automated handling of feedbacks
Database updating and data archiving
Contention resolution
Simulation facility

Taxonomy:

Initiated	manually
Frequency	as needed

Running state	diagnostic incompatible with normal operation
Read/write	ephemeral write & write
Latency	none
120 Hz	data and control
Shareable	input maybe
Sync time	few pulses
Missing data	can manage or retry
Geography	regional
# of users	varies, max of 4?
input data	>200K bytes

2. Linac Autosteering with movers (NP)

Purpose:
Establish and maintain the beam orbit centered in magnets and structures. Each quadrupole and each girder of structures are equipped with movers, which are used to align the device on the beam trajectory.

Use:
This procedure is done for initial setup of the main Linacs using a pilot beam and then runs continuously during normal operation. This spec describes steering of the main Linacs but the same algorithm would be used to steer other beam lines equipped with movers, e.g. the damping rings, bunch compressors, prelinacs, etc. These lines might be steered on demand rather than continuously. A complete pass of steering the main linacs is expected to take about 30 min.

Procedure:
Starting at the beginning of the linac, align the first n (~50) magnets, iterating as needed, then move downstream n/2 magnets and repeat until reaching the end of the beamline.

Read all QBPMs in range plus enough upstream BPMs to establish the incoming trajectory (10-20). May need averaging of orbits for several pulses. Fit incoming trajectory and reject bad pulses.

Calculate mover motions required to align the quadrupoles and implement.

Read all structure BPMs, calculate girder moves, and implement. Iterate until converged, then move downstream n/2 magnets. Feedbacks should in general remain on. Procedure should handle feedback within the range automatically

User interface:
User selects range of devices, algorithm, etc.
User sees corrected orbit, graphic display of changes.

Performance requirements:
The function must execute as efficiently as possible. It must automatically detect and filter bad data. Mover motions must be

synchronized. Simulated data must be available for system testing.

System impact:
Because it runs continuously, this function must gracefully coordinate control with other applications.

Services Used:
Continuous operation during normal physics running. Both main Linacs and several other regions would typically be being steered at the same time.
Intermittent Data acq of ~100 QBPMs and ~ 200 structure BPMs for a few pulses.
Model matrices
Data filtering
Automated handling/interface to feedbacks
Contention resolution
Simulation facilities

Taxonomy:

Initiated	manually and free running
Frequency	continuous
Running state	normal operation
Read/write	write
Latency	few pulses
120 Hz	data
Shareable input	yes, BPMs must also go to other users
Sync time	data - same pulse, control - few pulses
Missing data	can manage or retry
Geography	local/regional
Input data	>?? K bytes

3. Beam Parameter Control with Fast Feedback (LJH)

Use:
Feedback is a fully automated system, which runs continuously.

Procedure:
A variety of feedback systems will be available, with differing algorithms and specifications. Because a large number of systems must be supported, the software must be generalized and database-driven. Here is a proposed procedure for the linac steering feedback system: Each of the two Linacs will contain 5 feedback loops, each with around 30 beam position monitors (BPMs) and 8 correctors. (For a single loop, these devices are distributed along the accelerator and not clumped in a single location.) The feedback algorithm will be designed in advance using the accelerator model, and matrices will be periodically calculated offline using a math package and stored in the online database. After the beam passes, the BPM system calculates position measurements for a single selected bunch, or for the average of a number of bunches. Each feedback calculates intermediate beam states for its region and sends information to all downstream loops. The downstream loops combine the beam information from all upstream loops and their own measurements to determine the amount of a disturbance,

which must be corrected by this loop. Actuator settings are calculated and implemented.

Performance Requirements:
Typical feedback response time should be 2 120-Hz machine pulses. This includes time for measurement digitization, any system latencies, network traffic, calculation time, and time for corrector power supplies to change and affect the beam. Since corrector changes are likely to require a full 120-Hz interval, only a single 120 Hz period can be allocated for the other functions. Timing variability should be limited to 1/360 second. Because of the large number of feedback loops and their importance to operations, the system must remain functional without requiring frequent human intervention. For instance, a power supply driver which frequently freezes up and requires user reset would be unacceptable for feedback use. System designers should plan to accommodate periodic, automated control even for slow devices for which feedback control is not initially anticipated.

System Impact:
Feedback must coordinate with other applications using devices in the same region. This includes pausing feedback for another activity, sharing of data, and synchronization of control requests. All feedback data must be shareable with other users.

Control System Services Utilized:

- 120 Hz data from BPMs, charge monitors, and other pulse-Synchronized devices
- 120 Hz control of magnets, RF, timing devices
- 120 Hz distribution of feedback state data to other systems
- Synchronized control of slow devices
- Orbit plotting facility for BPM data
- Database access for matrices, parameters
- Archiving of data from ~few K pulses + long term archiving
- Error logging and metering
- Contention resolution
- Simulation facilities

Taxonomy:

Initiated	manually or automatically.
Frequency	continuous.
Running state	compatible with normal operation.
Read/write	write.
Latency	2 pulses.
120 Hz	data and control.
Shareable input	yes.
Sync time	1 pulse.
Missing data	can tolerate some broken BPMs.
Geography	regional/global.
# of users	varies, single feedback instances by region.
Input data	lots

Measurements:

Supported measurements must include beam position monitors, charge and intensity measurements, and a variety of ADC devices (analog to digital converters), which provide general pulse-synchronized measurements. In addition, the system is capable of reading back other analog signals such as magnet voltages. Data rates must be configurable, and allow selection of a subset of pulses, synchronized over multiple regions of the machine. The system must be capable of sharing measurement data with users, so that it is not interrupted frequently. Feedback should not require dedicated measurement devices, but must be capable of utilizing the standard control system devices.

4. Beam Parameter Control with Non-linear Feedbacks:

For the most part, SLC fast feedback systems are linear controls, which are applied to non-linear situations, where the problems can be linearized over a narrow range or the errors of linear treatment are not large. In some cases, this may not work adequately for the NLC.

This item is included as a place holder until the need for non-linear facilities is better understood and reasonable requirements generated.

Purpose: very short top-level description of the facility
Provide fast software oriented feedbacks controlling and regulating non-linear parameters of the accelerator.

Use: when, what conditions, how often, machine state
Precise control of non-linear machine parameters at real-time rates. These feedbacks would be standardized if possible, and would be part of production control and operation.

Procedure: coarse list of major procedural steps
Continuously running feedbacks, automatically started and operated, operator configurable, and setup so that they can share correctors and other control devices.

User Interface: minimal description of I/O interface
Operable from workstation consoles, with relatively simple facilities to turn thing on and off, change actuator amplitude, reconfigure input sensors, etc.

Performance requirements: precision, latency, response time, reliability
These will be high performance production equipment, which must be able to cope with non-functional sensor input. Monitors will be needed to track performance and correction quality. These systems are of the general type you would like to set and forget.

System Impact: possible effects on others
Non-functional feedbacks will hurt operations, while misbehavior due to errors in sensor data could substantially degrade the performance of the accelerator.

Services Used: ‘toolbox’ – requirements on low-level utilities
At a minimum, fast sensor data (BPM, toroids, cavities) must be provided on a pulse basis, event and error logging, and archiving will be required.

Taxonomy: quick lookup table of usage parameters

How initiated	manually or automatically, but monitored
Frequency	continuous
Running state	normal operation
Read/write	write
Latency	few pulses.
120 Hz	data and control.
Shareable input	yes
Sync time	few pulses
Missing data	must tolerate some broken BPM or bad data
Geography	regional / global.
Bandwidth	high

5. Automated Damping Ring Orbit Control

6. Automated Beam Initiation

Purpose:
Beam startup requires threading beam through many kilometers of beampipe without hitting any structures or causing beam energy deposition damage. Automation of this process would save lots of time for rather ordinary startups where known good orbits can be replicated and the beam softened in a high emittance state.

Use:
For startup, an automated process could blow up the beam cross section, and feed it through the machine from stopper to stopper with known good orbits and configuration settings. This process would be used after device or beam trips anywhere in the machine, and would occur multiple times per day.

Procedure:
Under the control of the MPS, a process would restore a known good orbit from Configs and sequence through the machine from stopper to stopper until the beam successfully reached the interaction area or tuning dumps. Once threaded, the beam emittance would be automatically minimized with standard tuning processes. Operator optimized configurations would allow the machine optimization to be improved to the last best configuration automatically.

User Interface:
Lots of user interface required to choose areas of the machine, select specific configurations from archives or recent operations, display orbit offsets by area, study high beam loss

areas, and sequentially reduce the beam emittance for normal operation and tuning.

Performance requirements:
Move beam rather quickly through known good areas, introducing operator assistance when required, and returning predictably to known good orbits and conditions existing prior to the previous shutdown. This system must be quick, reliable, and safe from a machine protection point of view.

System Impact:
This is a rather fundamental system for startup and the impact on machine operation if an application failure causes beam damage to accelerator structures could be substantial. Coordination of Machine Protection systems, beam sequencing through machine areas, orbit localization from BPM systems, abort system protections, and beam emittance control are substantial challenges to implementation.

Services Used: lots of fast data acquisition support.

Taxonomy: quick lookup table of usage parameters

How initiated	manual/free-running/both/remote/scheduled
Frequency	continuous / as needed / every N minutes
Read/write	read / ephemeral write / write
Shareable input	yes / no
Running state	normal running / diagnostic incompatible with normal running
Latency requirement	N pulse / human / none
120 Hz	yes / no (data and/or control)
Sync time	pulse / longer time / doesn't matter
Missing Data	intolerable / can manage / can't retry
Geography (input)	local / global / none (offline)
Geography (output)	local / global / none (offline)

7. Dispersion Correction
8. Emittance Measurement & Correction

Other Subsystems

8. Detector Interface
9. Beam Performance Tuning/Optimization

120 hertz Correlation of Acquisition/Control (JRB)

Most of the proceeding applies to pulsed data/control virtually unchanged. In what follows remarks specific to pulsed data and control will be in Italics.

Purpose:
Flexible acquisition, analysis and display of (finite-length) sequences of correlated data from different sources. Must also

allow for correlated control of devices and pseudo-devices. All device reads must be able to provide readback labeled with a pulse ID and may have to do the readback on pre-selected pulses indicated by some synchronized signal (and similarly for writing to control devices).

Use:
Same as Correlation Plots. There is a potential for producing large amounts of network traffic and poor response due to serialization if underlying facilities don't provide efficient and low latency support.

Procedure:
Acquire a sequence of data points for a collection of specified pulse-aware devices concurrently (on the same pulse and pulse ID), meanwhile optionally stepping one or more control devices. May also want to acquire data for non-pulse aware devices at approximately the same time (move correlated?).

User Interface:
Setup is as above except user must be able to specify conditions under which a pulse may be used for acquisition/control. Also any specifications having to do with time – i.e. settle time – would have to be handled differently. All other remarks above apply as well to correlation of 120 Hz data and control.

Performance Requirements:
The Facility must be extremely broad and flexible, encompassing essentially all forms of read-only pulsed data available to the Control System (including detector data) and pulsed control operations.

Plots and analysis: remarks as above.
Response time: remarks as above.
Guarantee of appropriate form/degree of synchronization. For pulsed data, this means all returned data must be labeled with a pulse id and is synchronized to a pulse. It must be possible to specify the kind of pulse on which a control operation or readout is to occur.
5. to 9. As above.

Interface Issues: As above. Also add
Because of the potential for generating requests for large volumes of read-only data, and particularly since it is likely that this data will be in demand by other applications, there must be a way to share it

Services Used: As above. Also,
The primary mission of this facility -- to provide for synchronized acquisition and control at 120 hertz -- cannot be accomplished without help from lower layers of the Control System. It depends on the existence of a utility to provide broadcast signals or in some other way to synchronize read or write operations on an arbitrary collection of far-flung pulsed devices.

Taxonomy:

How initiated	manually; perhaps also automatic-cyclic for some applications
Frequency	as needed (probably often)
Read/write	read and often ephemeral write
Shareable input	often yes
Running state	often but not always compatible with normal running; depends on application.
Latency requirement	can tolerate human interactive time wait before start
120 hertz	reads 120 hertz data; may write to devices at 120 hertz
Sync time	one pulse
Missing data	tolerates (but caller may have to take some action, such as retry)
Geography (input)	may be global
Geography (control)	may be global

PROTECTION SYSTEMS: There are several significant protection systems among the Accelerator facilities, including Personnel Protection Systems (PPS), Machine Protection Systems (MPS), and Beam Containment Systems (BCS). Some of these systems are stand alone hardware systems (PPS & BCS), and others are loosely integrated into the control system (MPS). Many of these facilities will use redundant systems, and all signaling will be high true so that failed components will be evident from their signal absence.

The Personal Protection System will be comprised of a series of Programmable Logic Controllers (commercial operating systems and lab custom firmware) networked together with proprietary buses to supervisory units and networked to consoles and programming stations in the Control and Maintenance Labs. These devices will use dedicated fiber based network equipment.

Beam Containment Systems are dedicated hardware systems used to insure that beam does not escape the accelerator structures and are only monitored by the control system. Beam control is exercised via control of the gun, stoppers, and beam accelerating subsystems. Additional redundancy capabilities are built into the MPG controlling dumps and stoppers.

Machine Protection Systems are integrated into the control system because these systems control the turn-on sequencing of beam and beam intensity. On the protection control side, they are hardware structured to kill or limit the beams. On the sequencing side, these systems are networked to the Master Pattern Generator (MPG) and the monitoring elements of the control system.

Facilities will be necessary to monitor the radiation dose at multiple selected locations within the accelerator structure to record radiation doses for device and equipment monitoring. Radiation profiles developed and archived will provide preventative maintenance insight for equipment living in the tunnels.

More information is available on the NLC Protection Systems on WEB page:www.project.slac.stanford.edu/lc/local/notes/protection-systems.

Equipment Protection, Integrity, and Reliability:

Personal Protection System (PPS)

The Personal Protection System is intended to protect Personnel from harm and injury. The most common hazards are beam radiation, residual radiation, exposed electrical circuits, and laser emissions. All of these hazards are turned off automatically (except residual radiation which is allowed to decay before access) as access doors are opened, and held off until the accelerator housing is evacuated and the machine safely re-started.

The PPS system will be implemented as hardware stand alone system to reduce the chance of failures in software systems. The system runs stand-alone so that outages of the Control System will not compromise the integrity of the protection envelope. For reliability reasons, the system will be implemented around networked programmable logic controllers using ROM configurations.

Purpose:
Safety systems intended to turn off hazardous systems prior to humans coming in contact with any potential hazards.

Use:
These systems operate independently from the control system, are implemented as hardware systems (but may use smart control elements like PLCs), and require high reliability and periodic performance and integrity verification.

User Interface:

The Control System has status knowledge of the PPS system, but no control, supervisory, or shutdown functions. There will be one or more Operator Interface Consoles for exercising, verifying, and diagnosing the PPS systems. There will be one or more PPS Engineering Consoles where detailed configuration and engineering issues can be examined (but not modified on the fly).

There will be a limited subsystem of the PPS Entry Module and production PLC systems for software development and fault analysis. It will be possible to download development firmware to PLC units in this environment.

Performance Requirements:
Reliability is the major issue. Security and configuration control are other important issues.

System Impacts:
Nothing operates unless the PPS is in a secure mode and all systems are locked up properly. This control function is enforced by hardware systems throughout the Accelerator.

Services Used:
No Control System services are used to operate the PPS in any mode. Status reading digital acquisition is used to obtain status information, which is displayed on System Status Monitors.

Taxonomy:

How Initiated	Manual initiation followed by continuous operation.
Frequency	Periodic Verification and continuous operation
Sharable input	Some status PPS inputs shared with Control System
Run State	
120 Hz	At all times in all modes everywhere in the machine
Missing Data	Accelerator shutdown
Input Geo	
Output Geo	

Machine Protection System (MPS)

Purpose:
The MPS system protects the accelerator hardware from damage due to errant beam pulses. A single full intensity beam pulse downstream of the damping rings will destroy any material it intercepts. The MPS protects against single pulse failures as well as integrated damage from thermal stress and radiation.

MPS also controls a programmed ramp sequence to return the NLC to full beam intensity after an interruption.

Use:

MPS is active at all times when there is beam and must be fully automated. It acquires data at full 120 Hz beam rate from distributed sources and provides diagnostic info to users. Configuration of the MPS system is an expert function done under tight administrative control.

Performance requirements:
MPS is an essential system with the highest priority. It has the tightest time requirements of any software in the NLC and must operate at full beam rate at all times. CPU and network conflicts with other subsystems may not be allowed to adversely impact its performance. The system must be highly reliable and failsafe, but designed to avoid spurious trips. A "heartbeat" from the MPS system in every region of the machine is needed; if the MPS does not respond, the machine must be shut off.

Procedure:
The MPS has several distinct functions with different time scales and latency requirements. There are effectively two separate MPS systems for the electron and positron halves of the machine.

The Beam Ramp system operates at 120 Hz. After an MPS trip, it generates a low charge, high emittance 1 bunch beam to verify system integrity. The beam is restored to nominal rate, intensity, emittance and # of bunches through a programmed sequence of steps using the permissive from MAID and other MPS to determine that the beam quality is adequate to proceed.

User interface:
Users must be able to monitor all aspects of MPS functions, including display of BPM data used by MAID on pulses preceding a trip. User must be able to bypass malfunctioning devices with password control. Users and invasive application software must be able to request any of the ramp up beam configurations with less than nominal parameters.

Services Used:
120 Hz BPM data.
Pre-beam 120 Hz klystron and kicker status.
Orbit plotting facility for archived BPM data
Contention resolution
Simulation facilities

Misc. comments:
A global control system watchdog in each sector trips the beam if the controls are not alive. This has an implication on accelerator downtime, since rebooting any microcomputer would take away the beam.

Where beam position monitor data is used (MAID), each region will have specified criteria indicating what fraction of data may be bad or missing. In the case of unusual expert BPM functions, which may not be compatible with MPS, the contention resolution system must insure that the allowed fraction of missing measurements is not exceeded.

Taxonomy:

Initiated	manually and free running
Frequency	continuous
Running state	normal operation
Read/write	write
Latency	few pulses
120 Hz	data
Shareable input	yes, BPMs must also go to other users
Sync time	data - same pulse, control - few pulses
Missing data	can manage or retry
Geography	local/regional
Input data	>?? K bytes

Beam Containment System (BCS)

Purpose:
The beam containment system monitors for radiation or particle beam leaving the accelerator beam-pipes and dumps.

Use:
These systems operated at all times that the accelerator is operating in all machine states. The system may be separable into segments depending upon the components of the accelerator that may be operating at any given time.

Procedure:
Special verifiable hardware systems control permissive signals to allow or disallow beams to be produced, injected, or transported in the accelerator.

User Interface:
These are mostly hardware systems, which report system status, and provide information on elements, which are faulted or inactive. Some of these systems have displays useful for accelerator diagnostics, beam aborts, or beam loss.

Performance Requirements:
BCS equipment must run at all times that the accelerator is operating with 100% reliability. Faults in beam containment and monitoring systems must fault to a machine off and safe condition. All control signals must be high safe so that power failures or transmission line failures will fault the machine off.

BCS equipment must be verifiable on a periodic basis from established procedures and specifications, and must have multiple system shutdown paths.

Elements:
Hardware devices designed to detect (and localize) beam spray radiation, radiation levels, dump burn through monitors, and toroid current monitors to detect unexpected beam loss.

System Impacts:
Failures of the monitoring system will bring the machine down and interrupt operations. Failures in machine safety and

integrity could lead to harm to users and damage to accelerator systems.

Services Used:
BCS monitoring derives information from Digital Status and Analog data acquisition systems in the Control System. Otherwise, the system depends upon the integrity of monitoring devices and its hardware interconnects (isolated from networks and software systems).

Taxonomy:

How Initiated	Automatic/by-passable/precursor to operations
Frequency	Runs continually
Sharable input	status used as diagnostic
Run State	normal running/compatible running and diagnostic
120 Hz	yes
Missing Data	intolerable/ by-passable
Input Geo	never offline
Output Geo	offline in bypass mode only

Radiation level reporting & archiving

Purpose:
A facility for monitoring the radiation levels in various parts of the tunnel system, or associated with specific beam-line devices, to measure integrated radiation dosage as an aid to device reliability and lifetime estimates.

Use:
Radiation monitors in the tunnel could integrate radiation during operation and forward the data to an Enterprise database would accumulate does by device and provide estimates for device lifetime and preemptive replacement.

Procedure:
The system would employ radiation detection devices attached to beamline equipment (i.e. thermo luminescent detectors – TLDs), or levels in the tunnels from ion chambers. An application would match the radiation area with the devices that live there and accumulate the dosages.

User Interface:
This would be a database application providing data on accumulated dose and expected dose at device failure for beamline devices. Such facilities could produce graphical and tabular displays of equipment requiring inspection or replacement.

Facilities would be needed to load estimates of radiation failure dose and schedule inspection or replacement. Provision should be provided to database radiation exposure and failure date to build database experience if calculated data is unavailable or inaccurate.

Performance requirements:

Relatively low priority systems to accumulate data on beamline devices and signal preemptive maintenance actions or accumulate data for improved failure estimates over time.

System Impact:
Preemptive replacement might have a large impact on beamline up time.

Services Used:
Graphical user interface, statistical package, and Enterprise database facilities.

Taxonomy: quick lookup table of usage parameters

How initiated	manual/free-running/both/remote/scheduled
Frequency	continuous / as needed / every N minutes
Read/write	read / ephemeral write / write
Shareable input	yes / no
Running state	normal running / diagnostic incompatible with normal running
Latency requirement	N pulse / human / none
120 Hz	yes / no (data and/or control)
Sync time	pulse / longer time / doesn’t matter
Missing Data	intolerable / can manage / can’t retry
Geography (input)	local / global / none (offline)
Geography (output)	local / global / none (offline)

SYSTEM & SOFTWARE RELIABILITY

Control System Reliability

- System Integration & Testing
- Hardware Reliability Analysis by conventional means
- Software Reliability Analysis by unknown means
- Fault tolerance
- Element redundancy
- Error Recovery (check sums/hamming codes/CRC)
- Fault Recovery (failover/??)

Control System Dynamic Reliability

- Network Performance & Slowdown
- Beam Position Monitor access from MPS
- Explicit Bypass and Bypass Management

DIAGNOSTIC FACILITIES

Network Infrastructure

- Network Performance Monitoring
- Network Design Capability

Control System Based Local Diagnostics

- Self-test features in modules
- CAMCOM type stuff
- High level diagnostic code
- Communications test & verification software

Control System Remote Diagnostics

- Export of Control Console Capability
- Security of Remote Connections

Field Test Equipment: There are a lot of ways to conduct calibration and diagnostic procedures, most of which should be supported for both local and remote troubleshooting or systems test and evaluation.

Portable test systems based on PC laptop or palm devices would make good sense to take advantage of the commodity costs of these devices. These devices might be supported by a few custom test boxes or fixtures used locally to attach to key devices.

Some locations within the machine housings will require some form of local control systems consoles. These devices will require network connections and may have to perform in a high electrical noise or strong magnetic field environment. There are a number of wired and wireless techniques for connecting these devices to the network.

Remote Vendor Test & Certification Systems: A number of Lab designed or commercially purchased devices will be manufactured in large quantities by external vendors. Large numbers of devices will necessitate either test equipment on the supplies production floor or testing devices and fixtures at the Laboratory site. This equipment set will run the range from network and timing system interface daughter cards to electronic modules, to major devices such as klystrons, magnets, and smart power supplies.

It would seem sensible for these systems to collect and transport test and calibration data back to the Lab on an individual serial number basis for inclusion in the Enterprise database.

These systems should be smart devices and either be sub-elements of the control system or use the same technology as the control systems and electronics for the Accelerator. The technology utilized should match that of the control system, and hardware should be the same as is used at the Laboratory.

DISTRIBUTED SOFTWARE DEVELOPMENT ENVIRONMENT

Tools and Infrastructure

Distributed Version Control Tool
Distributed Requirements Tool
Automated System Build Tools

Centralized Resources: The plan here is to put commonly used tools and data in an archive on a server available to all software collaborators world wide.

There will be a server with commercial Rational software to archive released software, firmware, programmable chip configurations, and PLC software and firmware. The structure of the archive will allow control of released versions of compatible elements into a specific system, and releases running at various periods and consoles.

In addition, there will be a tool server, which will provide access to the commercial tools being used by the collaboration at any given time. There will be provision for moving forward or backward among releases of this standard tool environment. Tools expected to be available, are compilers, debuggers, editors, performance monitors, browsers, and a common set of cross development tools and executives for embedded applications.

It is not yet clear what administrative support applications might be available from the server that might not be available on workstations or admin NT stations across the collaboration.

Master Source Store(software/firmware/hw config ROM)

Master Tool Server (compilers/cross tools/etc)

Software Test Facility: An extensive software test facility will be available at the Laboratory site. Less extensive test facilities may be developed at distributed sites to support specific software development in progress at that site. These environments should be accessible across the network in a secure mode.

In general the facility will provide hardware representatives of all equipment utilized at the Lab, and test boxes to allow control of inputs to these modules or devices, or indicator panels for hardware device outputs. In some cases these input/output boxes will have to be smart boxes in the sense of device or subsystem simulators in order to allow for realistic testing and evaluation of the software/firmware.

There will be a need for smart distributed test instrumentation to develop control system drivers and evaluate network and bus performance. In some cases these devices will reside in the test facility and in other cases, expensive equipment might migrate out into the Accelerator once the drivers are tested and verified. Included in this set of equipment are GPIB and Ethernet instruments (voltmeters, function generators, spectrum analyzers, interferometers, etc.).

Firmware development for embedded devices and controllers will require in-circuit emulator equipment in addition to test panels and I/O simulators. These will likely be PC based systems plus processor pods.

Long term performance analysis and diagnostic instruments and systems will be required in the test facility and the control room. Devices in this class are communication network and protocol analyzers. Facilities to allow remote keyboard access to distributed PC systems provided as controllers for commercial equipment will be necessary.

Representative Hardware with I/O simulation
System Simulators
Test & Measurement Instruments
In-circuit Emulators
Protocol, Bus & Network Analyzers

Standards & Documentation: Documentation will be developed and made available to Accelerator Users and Software Developers which describe the way software is developed, integrated into the production system, and maintained over its useful lifecycle. Standards will be available for languages currently in use or expected to be used.

Organizations providing software or firmware to the Accelerator program are responsible for maintaining that software over its lifecycle and will be required to provide and maintain extensive help files and training videos describing what the system does, how to use its features, and what its error messages mean.

The following manuals are representative of the kind of documentation that will be available:

Requirements & Design Reviews
Basic Users Guide
Principals of Operation
Programmers Guide
Embedded Programmers Guide
Code Review Guidelines
Hardware Manual
Software Test Procedures

Integration & Test: Control System software and Application code are continuously upgraded and enhanced while the Accelerator is operational. Experience suggests that new software can be released into the system during operation if the testing is done thoroughly and carefully.

In order to test unit and system level code without accidentally impairing the Accelerator Control environment, there will be a separate but identical control system running on a more modest scale for testing and evaluation. This system will support all of the accelerator hardware and be the basis for the test facility described elsewhere.

There will be facilities and procedures for using the test control system environment (hereafter the development system) to create for testing a representative environment. Automated procedures will facilitate the configuration, loading, and testing of software and firmware (download capability and remote network debugger).

There will be facilities on the Accelerator Control System (hereafter Production System), to allow a control console to use developmental level (not released to production) software for final test and system performance measurements not obtainable on the development system.

Automated tools will be available for inserting updated software into the base releases for the control system, the applications set, and the embedded firmware. An easy back-out procedure will be provided for situations where major unexpected problems develop and it becomes prudent to back up one revision level.

The entire set of Control System and Application code will be rebuilt weekly to ensure software integrity and compatibility. No elements of the control system will be built from binary inclusions.

NETWORKS & PROTOCOLS

General Networks: All of the general purpose and control networks will be fast switched Ethernet facilities. Some networks will be implemented as isolated or isolated non-routed networks both for security and for segregation of high-speed synchronous and low speed asynchronous traffic.

Networks will operate in distinct levels. There will be a switched backbone network for very high-speed traffic among major nodes. There will be isolated high-speed networks for streaming data out of DAQ modules and nodes and into database servers. There will be networks for high speed but general-purpose traffic, and there will be low priority administrative networks. These facilities will be crafted out of quality of service priorities, virtual private networks, and fully isolated subnets based on the technologies available at the time.

There will be networks implemented as field buses for low speed accelerator devices. Non of these networks will use proprietary signaling or protocols.

System Download & Boot: It is expected that there will be in excess of 1000 IOC and front end processor modules in the Control System. All of these IOC's will use a standard software package with database images to define device, object, and configuration information. The IOC code (executive and base software), and database image will be downloaded to the IOC cpu. This download will be executed in parallel to keep the boot time reasonable.

It is likely that a flash version of the base system and executive will be maintained in the IOC or front end module, and that the only image routinely downloaded would be the database configuration image. The system code would be downloaded when changes or enhancements are made.

All of these IOC and front-end processors will use standard fast Ethernet links employing TCP/IP. Boot functions will be provided by a protocol such as TFTP.

There will be a large number of smart devices and controllers in the machine, all of which need to be able to download firmware images, either by network or fieldbus.

There should be a provision for using a remote debugger over this same bus or network connection into the embedded device. All smart devices should have a kernel monitor which can be accessed by a laptop/palm device through a local serial port.

Network Security provisions: Software infrastructure tools will be utilized to access the Control System environment from the Lab and from remote facilities, which provide for user authentication and end-to-end data and message encryption. These facilities will be coordinated with site and control system firewall systems.

In situations where staff wish to access Control System facilities while not physically at the Laboratory, they will use secure encrypted connection tools (~SSH) and/or the equivalent of one-time password keys.

DCXP will be supported on the Control sub-networks, however, only pre-registered machines will be able to log-on. Registrants will be screened for operating system compatibility, security patch enhancement levels, and potential virus contamination. DOE utilization guidelines will be posted and enforced on all control system nodes.

Special Purpose Protocols: There are requirements for several special purpose network protocols:

NTP: The Network Time Protocol will be utilized to coordinate the date and time of day among processing nodes. This facility will maintain coherence in the millisecond range. Real-time devices will use an internal crystal clock and be coordinated from a time server to better than a millisecond. Time will be based on the GPS system.

TFTP: Operating images will be downloadable over the network using TFTP or equivalent.

DCXP:

Wireless Network Ports: Diagnostic and data-logging equipment used in the tunnels will require wireless network connections.

Telecommunication & Video: There will be a need for telephones in the tunnels and access portals, as well as video systems for surveillance of Personnel entry modules. All of these facilities will be digital and use the infrastructure networks.

--- Spence Clark, Editor 4-21-2000

APPENDIX A: INSTRUCTIONS & FORMATS FOR ADDING MATERIAL

It is the express intent of the Software Engineering Team, to make the development and evolution of the Control System Requirements a collaborative and supportive effort. We will negotiate and accept all contributed goals, strategies, and functionality suggestions for inclusion in the Control System Requirements Document, or in an appendix for materials that don't exactly belong in the Requirements but should not be lost in the collection process.

Short-form NLC Application Requirements format (Nan 11-10-99):

Purpose: very short top-level description of the facility

Use: when, what conditions, how often, machine state

Procedure: coarse, implementation-independent list of major procedural steps

User Interface: minimal description of I/O interface

Performance requirements: precision, latency, response time, reliability

System Impact: possible effects on others

Services Used: 'toolbox' – requirements on low-level utilities

Taxonomy: quick lookup table of usage parameters

How initiated	manual/free-running/both/remote/scheduled
Frequency	continuous / as needed / every N minutes

Read/write	read / ephemeral write / write
Shareable input	yes / no
Running state	normal running / diagnostic incompatible with normal running
Latency requirement	N pulse / human / none
120 Hz	yes / no (data and/or control)
Sync time	pulse / longer time / doesn’t matter
Missing Data	intolerable / can manage / can’t retry
Geography (input)	local / global / none (offline)
Geography (output)	local / global / none (offline)

-- N. Phinney Jan 2000