10.1 BCD Proposal: Controls Standard Architecture

1. Overview

The system topology is assumed to be two parallel tunnels with a central control room near the interaction point.

The baseline configuration (BC) shall use as a reference design existing packaging standards such as VME and VXI, and be similar to the model envisaged for the NLC and Tesla as well as modern machines such as LHC. The software standard will be a 3-tier architecture with established frameworks at each tier. This approach would minimize development effort.

However an alternative configuration (AC) is under consideration to develop a new architecture and packaging standard for the ILC, driven by the need for High Availability (HA) design of both hardware and software. This requires R&D evaluation of the technical and operational benefits of a significant HA investment to enhance the capabilities of both hardware and the 3-tier software at every level. HA systems use Intelligent Platform Management diagnostics and control which can also be extended to other electronics systems, including power electronics.

The BC can draw cost models from the NLC and TESLA models as well as newer machines. The AC requires additional R&D to evaluate and converge on a new incremental cost model with enhanced HA architectural features.

2. Baseline Configuration

a. Description

The baseline design envisions a dual star network model for controls emanating from a central modular computer cluster (Figure 1).

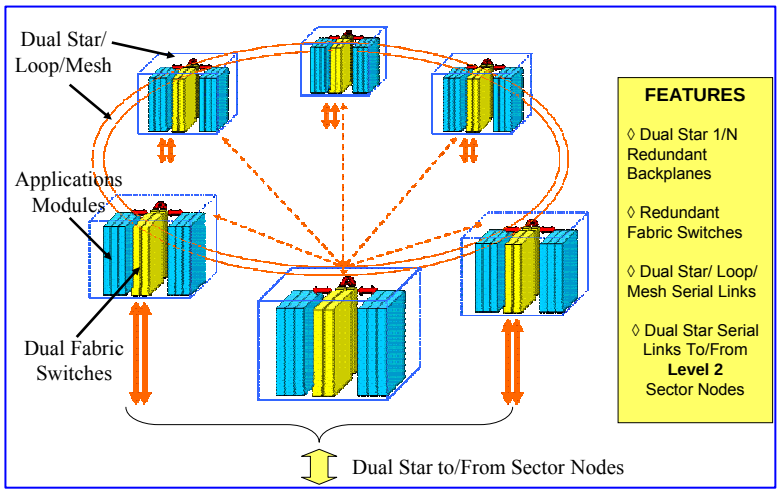

Figure 1. Control Room Cluster & Dual Serial Networks

Dual star data links provide branch control to all sector nodes of the various machines (Figure 2).

10.1 BCD Proposal: Controls Standard Architecture

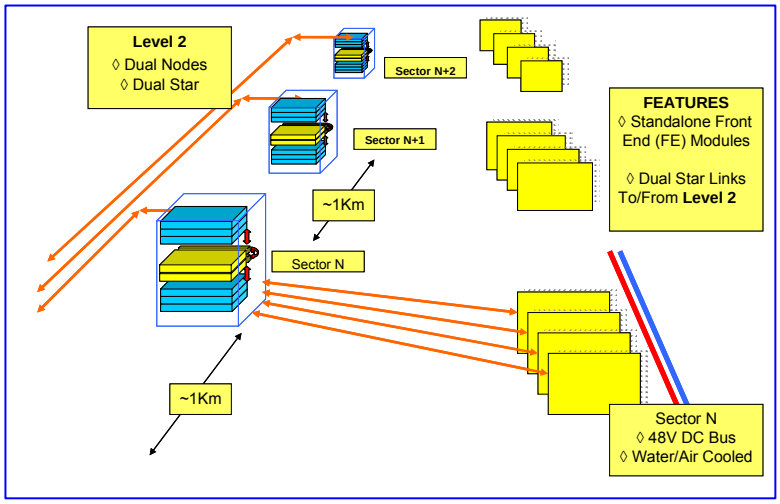

Figure 2. Dual Links, Sector Node Processors, Front End Modules

The baseline software architecture utilizes a standard 3-tier approach: client tier, services tier, and real-time tier. This approach provides separation of concerns, re-use, load management, change management, and many other benefits. A significant portion of the logic that traditionally used to reside in the client tier is now provided as a service for use by many clients. Services provide a means to coordinate the activities of many applications, and also serve to integrate real-time and relational database data into a seamless API.

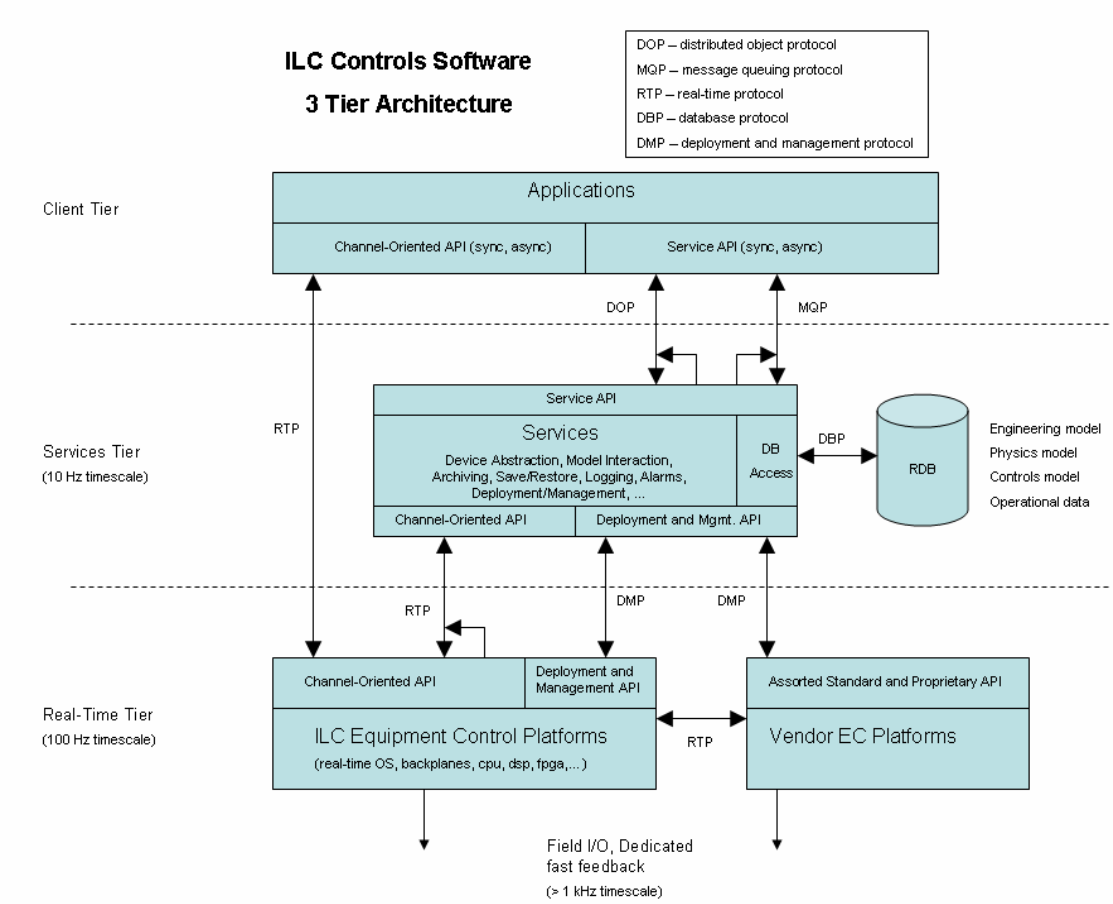

Figure 3. Software Architecture

10.1 BCD Proposal: Controls Standard Architecture

The performance requirements within and between the three tiers varies considerably, and therefore different protocols are required at different levels. See Figure 3 for a summary of the tiers, protocols, and relative performance timescales. The RTP (Real Time Protocol) provides a high-performance, narrow interface to channel (process variable) oriented data. The DOP (Distributed Object Protocol) provides a slower, wide interface to the services tier. The integral use of a relational database for maintaining an engineering model, physics model, controls model, and operational data should be noted.

b. Supporting Documentation

[1] Johnson, A., Clausen M., "High Availability Architecture for ILC", October 2005. http://docdb.fnal.gov/ILC/DocDB/0001/000106/001/10.1.R.1%20HA%20Software%20%20Architecture%20for%20ILC.doc
[2] Saunders, C., Kraimer M., "ILC Controls Software Architecture", October 2005. http://docdb.fnal.gov/ILC/DocDB/0001/000105/001/10.1.R.2%20Controls%20Software%20Architecture.doc
[3] Larsen, R. "Controls and Instrumentation Standard Architecture Options", July 2005. http://docdb.fnal.gov/ILC/DocDB/0001/000104/001/10.1.R.3%20Controls%20HA%20Standard%20Architecture.doc

c. Cost Estimation

The NLC and TESLA cost estimates for controls can serve as a starting cost model. Costs or cost estimates of recent control systems of SNS, LHC and other newer accelerators will serve as useful reference points. An inter-regional collaboration cost team will be established to agree on the technical cost model, while actual estimates will follow a set of GDE cost rules yet to be defined. The goal is to define the technical model sufficiently to perform a preliminary bottom-up cost estimate, and compare this result parametrically with other machines. Cost development can progress alongside development of the technical model.

3. Alternative HA Configuration

a. Description

Software and hardware engineering have to work hand in hand to achieve the required availability for the ILC. Depending on the time span of the breakdown of an individual component until the machine will go down we have to distinguish several different areas, which result in different approaches for mean time to automatically replace (MTTAR) (not repair) a component. These areas reflect the timescales of machine operation (from smallest to largest): bunch-by-bunch, macro pulse, process controls, static controls. The high availability solutions for each of these areas is potentially very different. For controls and instrumentation, the Advanced Computing and Telecommunications Architecture (ATCA) is a strong candidate for a base platform. ATCA is an open standard platform designed by a strong industry consortium called PICMG. The features include dual redundant communications processors and links, intelligent crate (shelf) diagnostics and management, and hot-swap-capable modules and sub-modules. See Figure 4 for an example of ATCA applied to an integrated beam instrumentation package.

10.1 BCD Proposal: Controls Standard Architecture

b. Benefits

Current accelerators (other than light sources) have up-times ranging from 40-70%. The ILC will be ten times the size of any comparable existing linear machine so availability becomes a dominant issue. Recent machine availability simulations presented at Snowmass 2005 show the difficulty of reaching 80% even assuming much improved reliability of all subsystems.

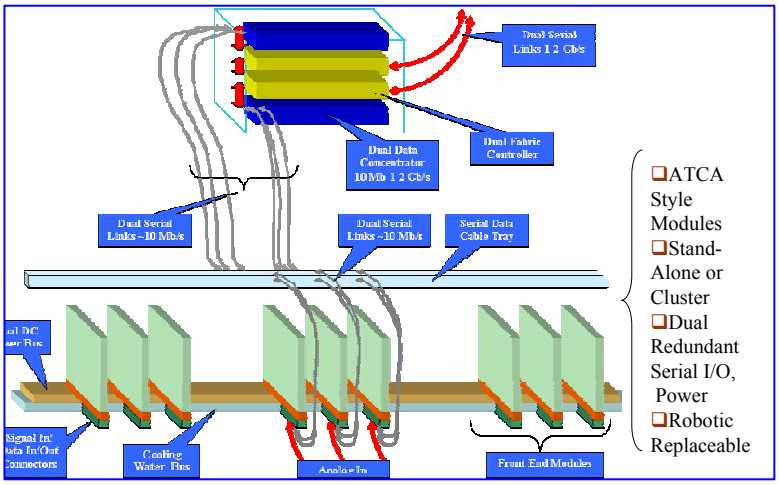

Figure 4. Beam Instrumentation Packaging Concept

Because of the recognized difficulty, it is essential that the ILC controls, instrumentation and electronics systems, including power electronics, be designed with High Availability (HA) architectures. In addition, HA controls design with orders of magnitude increased reliability will make errant beam problems due to controls failures far less frequent and easier to diagnose, and may ease the design complexity of machine protection systems. HA architectures incorporate intelligent management features to negotiate around faults in the system. Fault management typically involves five stages:

1. Detection - the fault is found
2. Diagnosis - the cause of the fault is determined
3. Isolation - the rest of the system is protected from the fault
4. Recovery - the system is adjusted or restarted so it functions properly
5. Repair - faulty system components are replaced.

Operating, middleware and applications layers all are impacted by HA design. In the ultimate, critical applications can be designed with fault-avoidance auto-failover features through a combination of software design, hardware diagnostics and hot-swap serviceability. In principle, controls and instrumentation systems can be made very robust against single-point failures in communications or logic operations that typically plague large complex experimental accelerators and detectors.

c. Required Research and Development

1. Evaluate existing HA architecture standards for applicability to all ILC electronics systems, subsystems, hardware, and software.
2. Evaluate incremental cost models for HA systems

10.1 BCD Proposal: Controls Standard Architecture

3. Demonstrate dual redundant backbone system.
4. Demonstrate proof-of-principle conceptual designs of typical applications.
5. Adopt hardware, software standards for ILC controls down to instrumentation subsystem interfaces.