LASER INTERFEROMETER GRAVITATIONAL WAVE OBSERVATORY

-LIGO-

CALIFORNIA INSTITUTE OF TECHNOLOGY

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Document Type	LIGO T070059-00-C **DRAFT**	March 20, 2007
AdvLigo Control and Data System Conceptual Design		
R. Bork		

Distribution of this draft:
This is an internal working note of the LIGO Laboratory

California Institute of Technology
LIGO Project – MS 18-33
Pasadena, CA 91125
Phone (626) 395-2129
Fax (626) 304-9834
E-mail: info@ligo.caltech.edu

Massachusetts Institute of Technology
LIGO Project – MS 20B-145
Cambridge, MA 01239
Phone (617) 253-4824
Fax (617) 253-7014
E-mail: info@ligo.mit.edu

www: http://www.ligo.caltech.edu/

LASER INTERFEROMETER GRAVITATIONAL WAVE OBSERVATORY 1
-LIGO- 1
CALIFORNIA INSTITUTE OF TECHNOLOGY 1
MASSACHUSETTS INSTITUTE OF TECHNOLOGY 1
1 Purpose 3
2 Scope 3
3 References 3
4 Overview 3
5 Supervisory Controls 5
5.1 Operator Stations 5
5.2 EPICS Gateway 5
5.3 File Servers 5
5.4 Application Servers 6
5.4.1 Script Server 6
5.4.2 Operations Electronic Logs 7
5.4.3 BURT, Conlog, CDS Status and Screen Snapshots 7
5.4.4 Projection Servers 7
5.4.5 Video Systems 7
5.4.6 Web Servers 8
5.4.7 NAT Router 8
5.4.8 Virtual Private Network (VPN) 8
6 CDS Computer Networks 8
7 Timing System 9
7.1 Network Time and Timing Clock Distribution 9
7.2 Atomic Clock Reference System 10
7.3 Subsystem FE Timing Diagnostics 10
8 Data Acquisition System (DAQS) 11
8.1 Realtime Front End (FE) Data Acquisition 12
8.1.1 Continuous Data Transmission Design 12
8.1.2 Local Data Storage Design 14
8.2 FrameBuilders 15
8.3 EPICS Data Collection Unit (EDCU) 16
8.4 NDS Gateway 16
9 CDS Subsystem Front End Controls 18
9.1 Computer, I/O buses and Signal Conditioning Hardware 18
9.1.1 FE Computers 19
9.1.2 I/O Chassis 19
9.1.3 I/O Modules 19
9.1.4 Anti-Aliasing (AA) and Anti-Imaging (AI) 20
9.2 DC Power Distribution 20
9.3 Electronics Construction and Testing 21
9.4 FE Computer Software 21
9.4.1 Operating System 21
9.4.2 FE Code Modules 21
9.4.3 CDS Realtime Code Generator (RCG) 23
10 Diagnostic Monitoring Tools (DMT) 25
10.1 DMT Computers 25
10.2 Frame Broadcaster 25
11 Enhanced Ligo (ELigo) Transition 25
11.1 Data Acquisition 25
11.1.1 Data Frames 25
11.1.2 Slow (EPICS) Data Acquisition 25
11.1.3 Fast Data Acquisition 25
11.1.4 NDSG 26
11.2 EPICS Gateway 26
11.3 If Available Additions 26

1 Purpose

The purpose of this document is to provide a conceptual design for the AdvLigo Control and Data System (CDS).

2 Scope

The primary scope of this document is the CDS infrastructure to be provided in AdvLigo. This includes such items as:

- Supervisory Controls ie those facilities used to integrate and support all of the various control subsystems and to provide operator interfaces into the system.
- Timing System
- Data Acquisition (DAQ) System
- Diagnostic Monitoring Tools (DMT)

The scope also includes that hardware and software which will be common to all subsystem controls which are to be provided as part of the CDS infrastructure.

3 References

A. Ligo T070056-00-C Advancd Ligo Control and Data System Infrastructure Requirements
B. Ligo T070057-00-C Advanced Ligo Subsystem Block Diagrams
C. Ligo T990110-B-D Calculating the Trend Data

4 Overview

This document provides a conceptual design intended to meet the CDS requirements as set forth in Reference A. It is based on technologies that are readily available today and have been, and continue to be, tested in prototype control systems at Lasti and the Caltech 40m lab. Computer technologies change fairly rapidly, therefore the exact hardware employed will possibly change over the next few years prior to first installation. However, the basic architectural philosophy should remain intact.

Much of the CDS hardware presently installed has been in Ligo for a number of years and is nearing a point in life where it is becoming obsolete and/or aged to a point where continued maintenance becomes a problem. Therefore, most of the CDS hardware will be replaced for AdvLigo. One system of note that is not presently planned for replacement are the Vacuum System Controls (VSC).

As for software, it is intended to reuse and continue to support as much of the present code as possible. There has been a large investment of operator and user training on present systems, and therefore it is not intended to make drastic changes in the look and feel of the various CDS tools.

The following sections will outline the design, primarily from a top level down to the individual subsystem controls. An overview block diagram is provided in the following figure. Additionally, two sets of block diagrams will be referenced to illustrate various design sections. Reference B contains diagrams of the individual control subsystems as a basis for the infrastructure required to support them.

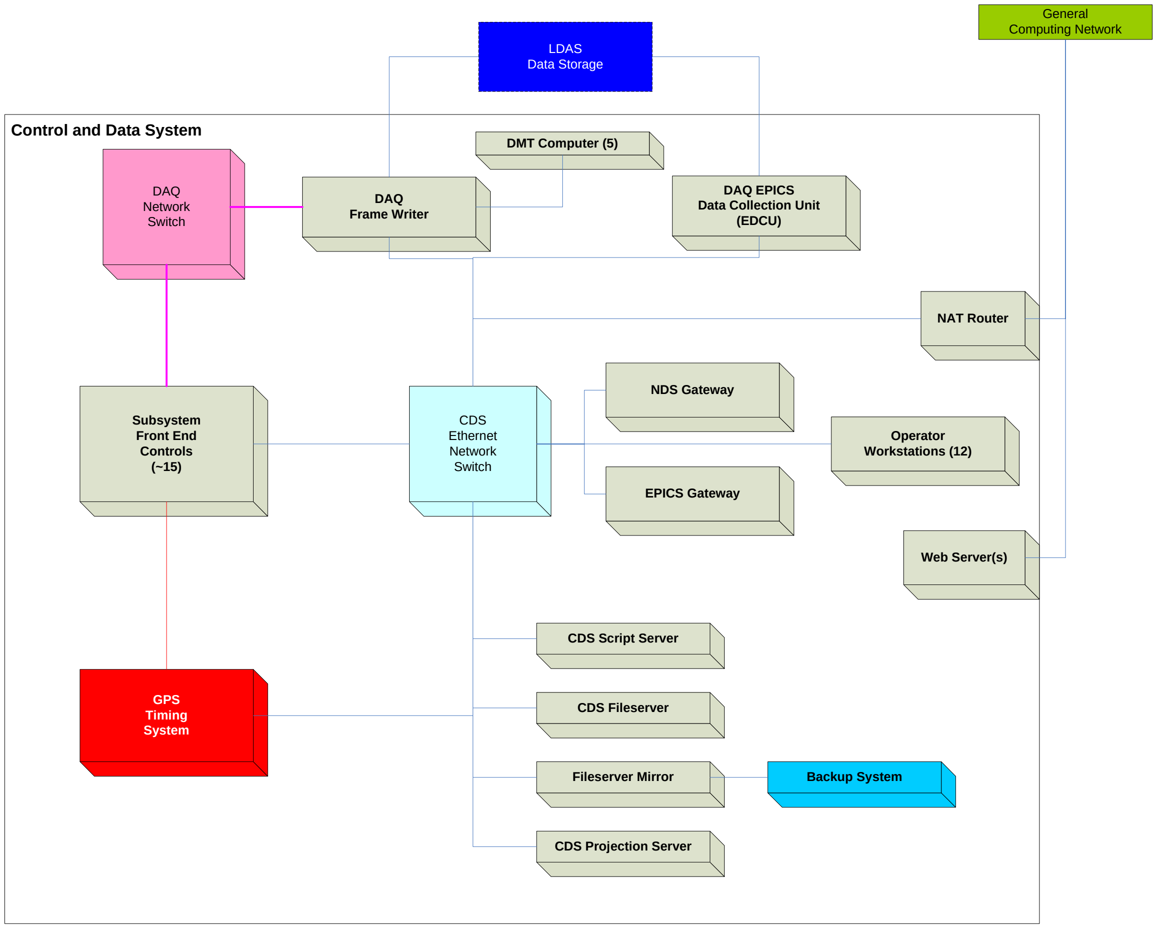

Figure 1: CDS Overview

5 Supervisory Controls

Supervisory controls are defined as those high level functions which provide:

- Operator interfaces into the CDS, including supporting computers and software.
- Networking services to bind all of the various CDS computers.
- Computer file services, for CDS control and monitoring application software, as well as for CDS users.
- Data connections between CDS and computer systems and users outside CDS.

5.1 Operator Stations

The CDS will provide computers and monitors to support twelve (12) operator stations in the control room at each site. The hardware is to employ the latest technology available at the time of purchase, with an operating system that is compatible with the AdvLigo software.

The primary operator interface software will continue to be those tools presently operating in Ligo. This includes:

- EPICS MEDM as the operator Graphical User Interface (GUI). This is the baseline design, however various other GUI applications will be reviewed during further design phases.
- EPICS StripTool for time plots of EPICS channels. Presently, this tool only starts plotting live data as a user selects a channel. This software will be enhanced to fill in the plot up to the present time with archived data from the DAQ system.
- EPICS Alarm Handler (ALH) will continue to provide operator alarms. This tool will be enhanced to provide multiple alarm tones to help determine the alarm conditions.
- Diagnostic Test Tool (DTT) will continue to be the primary tool for online analysis by operations staff. It will be improved, as per the requirements in Reference A, section 5.2.7.
- Dataviewer will continue to be supported for time domain plotting of CDS data.

In addition to these, two new software applications will be developed to aid in commissioning activities. There are:

- A time domain plotting tool that is faster than the present Dataviewer. This tool would essentially have a look and feel of a four channel digital oscilloscope, as prescribed in Reference A, section 5.2.8. Design details are TBD, but would be based on the fast plotting software developed during Ligo.
- A frequency domain waterfall plotting tool, as outlined in Reference A, section 5.2.9. The conceptual design involves taking a commercial product, such as Buadline, and modifying it for Ligo use.

5.2 EPICS Gateway

In the present Ligo CDS, all EPICS to operator station communications are handled directly by Channel Access (CA) clients running on the operator stations. This uses the CA broadcast methods to locate the desired data channels, then individual socket connections to all the computers having these channels. This puts an additional burden on realtime FE computers to maintain all of these data links, many of which are providing duplicate channel information requested by multiple machines.

To cut down on this FE loading, and to provide an added level of security, computers will be provided in AdvLigo CDS to act as EPICS Gateways. These will run a more or less standard version of the EPICS Gateway software, now used to relay data off site.

5.3 File Servers

The CDS will provide computers, disk storage media, and permanent archive backup facilities for the purposes of storing CDS application software and user data files. Architecturally, this is to be similar to the present system:

- A primary file server computer with local disk

- A separate computer and disk system to act as a mirror of the primary.
- A DVD, or later technology, backup system which provides archival data copies from the mirror file server.

5.4 Application Servers

The CDS will provide a number of computers for use as Application Servers. These are described in the following subsections.

5.4.1 Script Server

Various scripts, based on EPICS channel access, have been developed for Ligo to automate operator tasks, such as the automatic sequencing of systems to various states. These scripts are run on the operator workstation from which they were commanded to start.

For AdvLigo, these scripts will no longer run on operator stations, but rather a central computer dedicated to this purpose. An overview of this is shown in the following figure.

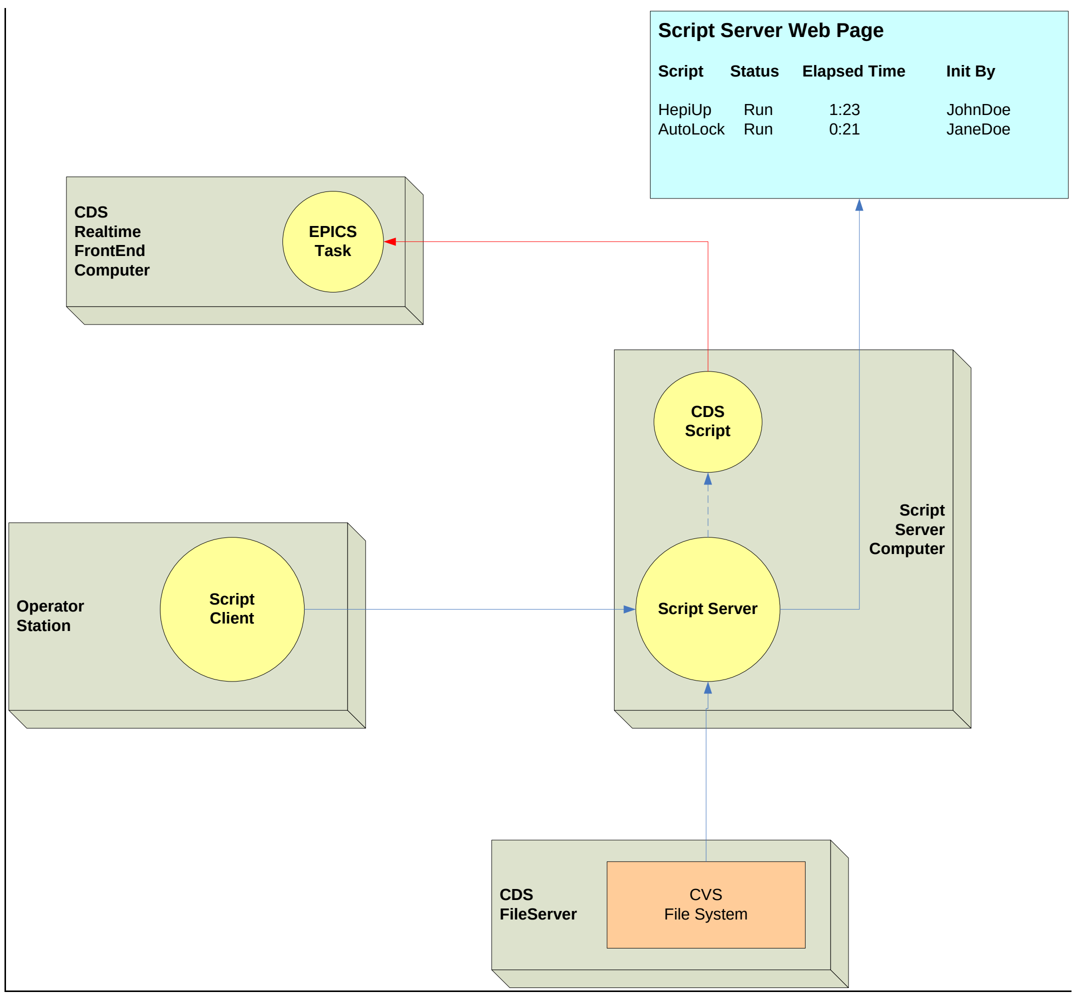

Figure 2: CDS Script Server

There are a couple of primary motivations to move to this central script server scheme:

- Presently, it is difficult to track which script is running where.
- Scripts are modified by CDS users without knowing who made the changes and why. In this scheme, only scripts entered into the CDS CVS repository would be allowed to run on the server.

The basic sequence of producing and running a script is as follows:

- A script is written and then checked into the CVS repository. The scripts may be written as a C shell script or done using Perl. The primary elements of EZCA and TDS, developed for Ligo, will continue to be available for script communications with EPICS.
- From any operator station, a script request is made, via a command line or operator screen, to start a script. This request is relayed to the script server.
- The script server loads the requested script from the CVS repository and initiates the task. The individual script then runs and communicates to CDS realtime FE computers via the CDS Ethernet and the FE EPICS task.
- Monitoring of scripts is provided by the server via a web page. This includes information on what scripts are running, their status (with a detailed status page available as well for each script), and who requested the script to run.

5.4.2 Operations Electronic Logs

The CDS will provide a computer for the operation and data storage associated with the present Ligo ilog software. The ilog software itself will be provided and maintained by non-CDS Ligo staff.

5.4.3 BURT, Conlog, CDS Status and Screen Snapshots

Presently, CDS supports a number of software applications to support operations, which run on various computers. Among these are:

- EPICS BackUp and Restore Tool (BURT). This tool is used to save operator settings once per hour, or on demand. These data files may then be used at a later time to restore EPICS settings.
- Conlog, a tool which also monitors EPICS channels and records changes.
- A CDS status system, which is a collection of EPICS channels which are used to monitor if CDS control settings are within prescribed bounds.
- Screen snapshots. Most EPICS screens are built with a user push button, which takes a screen snapshot and save it as a jpeg file for later comparison of settings.

All of these tools provide similar functionality, but all produce their own data files. They all also have various limitations. Therefore, for AdvLigo, there will be a move to combine these features into a new tool. The design for this software tool is TBD. However, a key element to this is the new DAQ design (see DAQ section) which will continually record all EPICS channels at a 1Hz rate. This will allow such features as the ability to restore EPICS settings to any time (not just the present one hour frame of autoBURT) by simply requesting EPICS channel values from the DAQ for a specific time, creating a temporary BURT snapshot file and reloading them using the BURT restore function.

5.4.4 Projection Servers

Ligo employs a number of computers and video projectors to provide large data displays on the walls of the control room. For AdvLigo, the CDS will provide replacement projectors, new large LCD displays, or similar units, based on the latest technologies, to continue to provide this capability.

There is not an intention for AdvLigo CDS funds to provide new computers to drive these display systems. Rather, as existing CDS computers are replaced, such as operator workstations, these computers would be moved to update the present projection system computers.

5.4.5 Video Systems

Other than the projection systems described above, there is no intention of providing new video cameras or switches in the AdvLigo CDS. Existing systems would be updated, as necessary, using CDS maintenance funds. Any new camera systems needed for AdvLigo would be provided as part of the Interferometer Sensing and Control (ISC) system.

5.4.6 Web Servers

The AdvLigo CDS will continue to provide computers for use as CDS web page and wiki page servers. These computers will connect to the site General Computing (GC) network and provide Read Only information. These machines will not provide any log in capabilities from the GC network.

5.4.7 NAT Router

For the purposes of remote log in to CDS computers from the outside, CDS will provide a NAT router, or similar technology, for connection to the GC network. The security will be in accordance with Ligo policies and the CDS security plan TBD documentation.

5.4.8 Virtual Private Network (VPN)

For the interconnection of CDS networks at the two sites, plus connection of remote control rooms and specific users to CDS networks, VPN equipment will be provided. Design details are TBD.

6 CDS Computer Networks

The CDS infrastructure will essentially provide two private networks for the interconnection of all CDS computers. A third network system is also to be provided as part of the data acquisition system (see section 8).

The first is a standard Ethernet configuration for the communication of EPICS traffic to/from realtime FE computers and supervisory controls and general computing types of data traffic, such as file transfers, NTP services, etc. In concept, the design for AdvLigo is to essentially replace the existing networking hardware, including the Ethernet switches, with the latest available technology.

The second is a deterministic, realtime network for communications of data between realtime processes on CDS FE computers. Unlike the present Ligo system of a Reflected Memory (RFM) serial network, this is to be a star fabric configuration network.

In prototype systems to date, this has been a proprietary network with <10usec latency in computer to computer communications. Multiple computers have been connected through a network switch during these tests. During preliminary design phase prototyping, it is intended to expand testing to Infiniband, which is available from multiple vendors.

One new feature of these networks that is being used is the remote direct memory capability. Using this, the sending computer can directly write into the memory of another computer. The target computer does not have to listen for a message, then copy it off from the Network Interface Card (NIC) to its computer memory space. This is all handled by the network and NIC.

7 Timing System

The CDS infrastructure will provide two timing systems, one based on GPS, the other, used as a checking system, base on an atomic clock.\

7.1 Network Time and Timing Clock Distribution

The CDS will continue to provide a timing system for synchronous realtime operations, based on the Global Positioning System (GPS). The key components are shown in Figure 3.

The design for this system is based on the timing system installed at LHO during the past year. It consists of:

- A GPS receiver. This would be a single unit at the corner station as opposed to the present one GPS receiver per building. This unit provides time to all CDS computers using Network Time Protocol (NTP) on the CDS Ethernet. It also provides a GPS one PPS output for connection to the Ligo Master Timing Unit (MTU).
- MTU. This uses the 1PPS input from GPS as a synchronization pulse. Locked to this, is a 38MHz clock, which is output, via fiber optic links, to all of the Slave Timing Units (STU). There are a total of 8 outputs from an MTU, both a transmit and receive. The receive link is a return from the STU used by the MTU to verify timing accuracy.
- STU: At various locations, near the CDS FE I/O chassis, there will be STU. These units receive clocking signals from the MTU, and provide synchronous, differential TTL clock outputs to synchronize the various CDS I/O modules. The individual output clock frequencies are adjustable, in 2^n steps, from 1Hz to 4MHz via internal jumpers.

Further details of the MTU and STU are provided in Ligo document TBD.

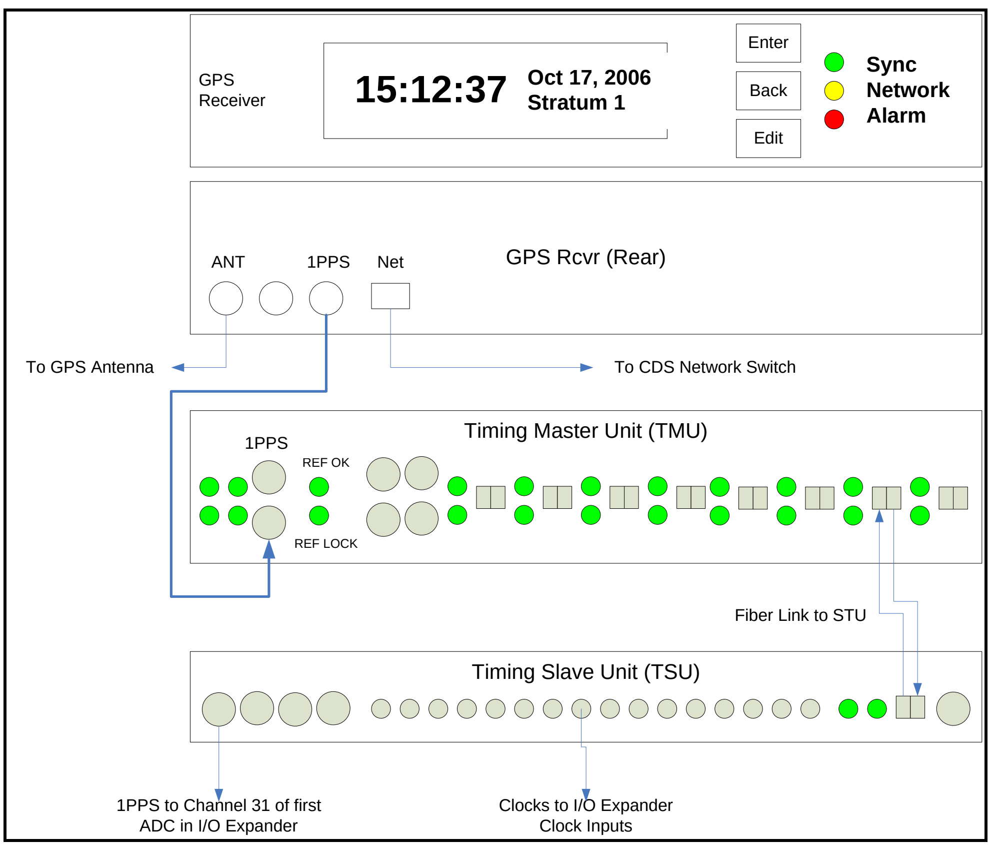

Figure 3: Timing Distribution System

7.2 Atomic Clock Reference System

The CDS infrastructure will provide an Atomic clock system as a reference with which to compare the accuracy of the GPS timing system. This system will be nearly identical to the present Ligo system. However, since that system may no longer be operational, due to aging, by the time of AdvLigo installation, it may need to be replaced.

7.3 Subsystem FE Timing Diagnostics

As part of the timing system, diagnostics will be provided to determine the accuracy of the timing of various, specified subsystem controllers. The design of this diagnostic is TBD.

8 Data Acquisition System (DAQS)

A key element of the CDS is data acquisition. This system must provide:

- The capability to acquire and store user defined fast channels (>=16Hz) and all EPICS data channels (at 1Hz)
- The ability to select and acquire test point data on demand.
- The ability to select and inject excitation data on demand.
- The capability to distribute any acquired data to other CDS systems via the CDS networks..

A basic block diagram of the AdvLigo DAQS architecture is shown in Figure 4.

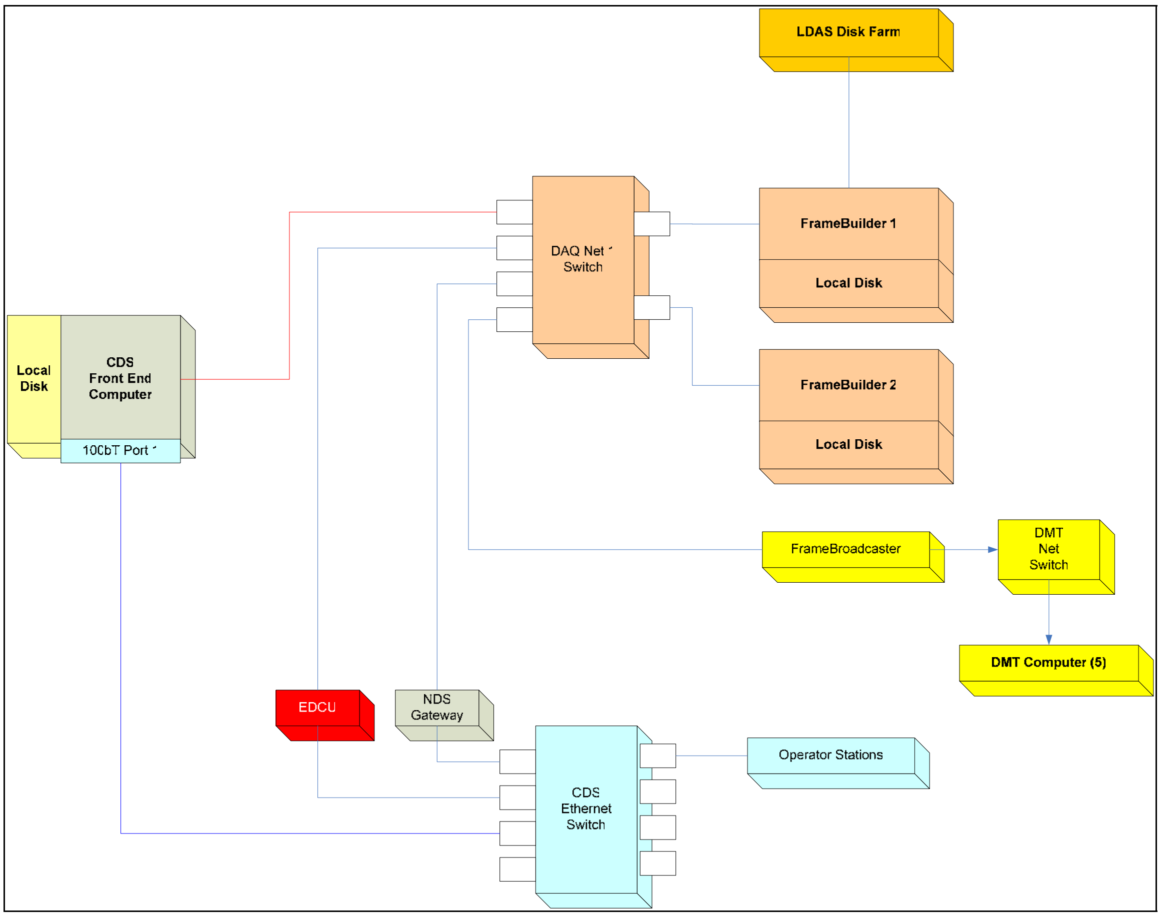

Figure 4: AdvLigo DAQ Architecture

In the present Ligo DAQS, all realtime FE computers transmit their data acquisition channels to central DAQS computers (FrameBuilders) via a realtime, proprietary RFM network. All EPICS data is acquired from the CDS Ethernet by these central computers using EPICS Channel Access (ECA). These computers, known as FrameBuilders, then produce a single Ligo standard data Frame per site. This data is stored in a single 32 second Frame file, both locally and to LDAS disk drives. One of the two site FrameBuilders also runs the FrameBroadcaster software, which broadcasts data frames to the DMT computers via a private network.

In the AdvLigo DAQS conceptual design, there are to be some key changes in the DAQS system. These key elements are:

- Rather than a single full data Frame per site, there would now be two types of data frames stored per IFO:
  - Fast data frame, which contains synchronous data at 16Hz to 32768Hz. This new system would include support for synchronous acquisition of EPICS channels at 16Hz.
  - Slow data frame, which stores all EPICS channels at 1Hz.
- Trend data frames would continue to be written, as they are now, except there would be trends for fast data and trends for slow data. Trend data will be written in accordance with Reference C.
- Each CDS FE control computer will write its own EPICS data file. This will later be copied to and post-processed by the FrameBuilders, as described later.
- To support legacy EPICS control systems, such as vacuum controls, an EPICS Data Collection Unit (EDCU) computer will be provided for each IFO.
- An NDS Gateway (NDSG) computer will be provided on each IFO to handle data distribution to CDS supported software tools, such as DTT, Dataviewer, etc...This NDSG will also have a read-only mount point to the archive to distribute stored data.
- A separate computer may be provided to transmit data frames to the DMT computers via a private network. This is further described in Section 10.

8.1 Realtime Front End (FE) Data Acquisition

All fast, synchronous data originates in the various CDS realtime FE computers. For AdvLigo, fast data also includes the ability to acquire EPICS channels synchronously at 16Hz as part of the fast data stream, in addition to the usual 1Hz asynchronous mode.

To acquire and store this data, two concepts are to be pursued during the preliminary design phase. What makes these concepts possible is a key change in the FE computers of AdvLigo. This change, further described in section 9, is a move toward multi-processor computers for use as FE controllers. This allows FE controllers to run multiple support tasks, such as DAQ tasks, without interfering with its realtime control tasks.

A second important change is the move away from a single GDS TPM/AWG per IFO. In the new design, there is a TPM/AWG running on the FE computer to support each FE computer realtime task. Again, this is further described in section 9.

One concept involves having the FE computers continue to send their data over a realtime network to the FrameBuilders, which then produce the Frame files. This concept is similar to the present DAQS, but uses a different network topology than RFM.

The second concept is to have each FE computer write its own data to local disk. A task on the FrameBuilders then comes along in some TBD period of time, copies all of the FE local files to its file system, performs diagnostics, and writes a concatenated frame file to its local disk and LDAS disk.

It should be noted that, using either design option, each FE computer is now capable of operating as a stand-alone system. That is to say each FE computer is capable of running its control algorithms and running the complete DAQS software suite, including FrameBuilder software, locally. This allows test systems to be built using a single computer providing all of the CDS features. This is as opposed to the present Ligo system, which requires at least one FE computer, one computer to run GDS TPM/AWG and one to operate as a FrameBuilder.

8.1.1 Continuous Data Transmission Design

The following diagram depicts the software and architecture involved with acquiring data from FE computers, where FE computers continuously send data to central FrameBuilders. This is similar to the present Ligo DAQS model, and it is the design that has been used to date in CDS prototyping activities at Lasti and the 40m lab.

This design uses a realtime (deterministic) network to send data from all FE computers. This is the realtime network described earlier in Section 5.

The following paragraphs describe the basic data acquisition process.

1) FE computers read a list of signals to be sent to FrameBuilders for data acquisition. This is done by having an EPICS sequencer read a file, via the CDS network, and place the channel list and rates into the FE computer shared memory. This is, in turn, read into the DAQ/GDS task on the realtime CPU on startup and whenever a new configuration request arrives via EPICS.

2) The FrameBuilders read in a Master list of DAQ channels and all available GDS signals. They then develop a data frame, in local memory, which gets saved to disk at a prescribed rate (presently, every 32 seconds).

3) FE computers use Remote Direct Memory Access (RDMA) to send data to the FrameBuilders. In this scheme, the FE computer uses the network to write its data directly into the processor memory space of the FrameBuilder ie the FrameBuilder does not have to move the data from its NIC to its local memory. Therefore, on startup, the FE DAQ/GDS task sends a message to the FrameBuilder requesting a local memory area to transmit data to. Upon reply, the DAQ/GDS task completes its initialization and waits to be called by the realtime FE synchronization software.

4) The DAQ/GDS task maintains two data sets in memory, each sized to contain 1/16 second of data. It uses these buffers in a ping-pong fashion, transmitting data from one while filling the other with new data. The FE synchronization software task calls the DAQ/GDS task on every cycle. At this time, this task reads data from the local memory, performs any necessary decimation and/or data formatting, and writes the data into one of its buffers. At a rate of 256Hz rate, this task transmits data from its second buffer to the FrameBuilders via the realtime network.. This rate was chosen somewhat arbitrarily to balance the network load.

5) At a 16Hz rate, the FE DAQ/GDS software transmits a ‘Data Ready’ message to the FrameBuilders. This message includes the channel names and rates as data indexes for the FrameBuilders. The FrameBuilders then move this data into the Frame

6) At a 1Hz rate, the FE DAQ/GDS software checks for new GDS TP selections, as placed in shared memory by the local Test Point Manager (TPM). If TP are selected, their data is sent at 256Hz, at the end of the DAQ channel data stream. Likewise, if AWG channels are selected, the DAQ/GDS relays the AWG signal to the FE realtime code at the sample rate of the FE system and also sends the AWG channel data to the FB at 256Hz at the end of TP data. At the 16Hz messaging time, the TP/EXC channel names and rates are added to the DAQ message.

7) The FrameBuilder computers operate essentially as they do now in the Ligo DAQS. They build the data frames, and store data locally and to the LDAS disk system. They also run a Network Data Server (NDS) task to deliver data, on request, to other CDS computers.

8) A FrameBroadcaster runs on one of the FrameBuilder computers and broadcasts data Frames to the DMT machines via a separate network.

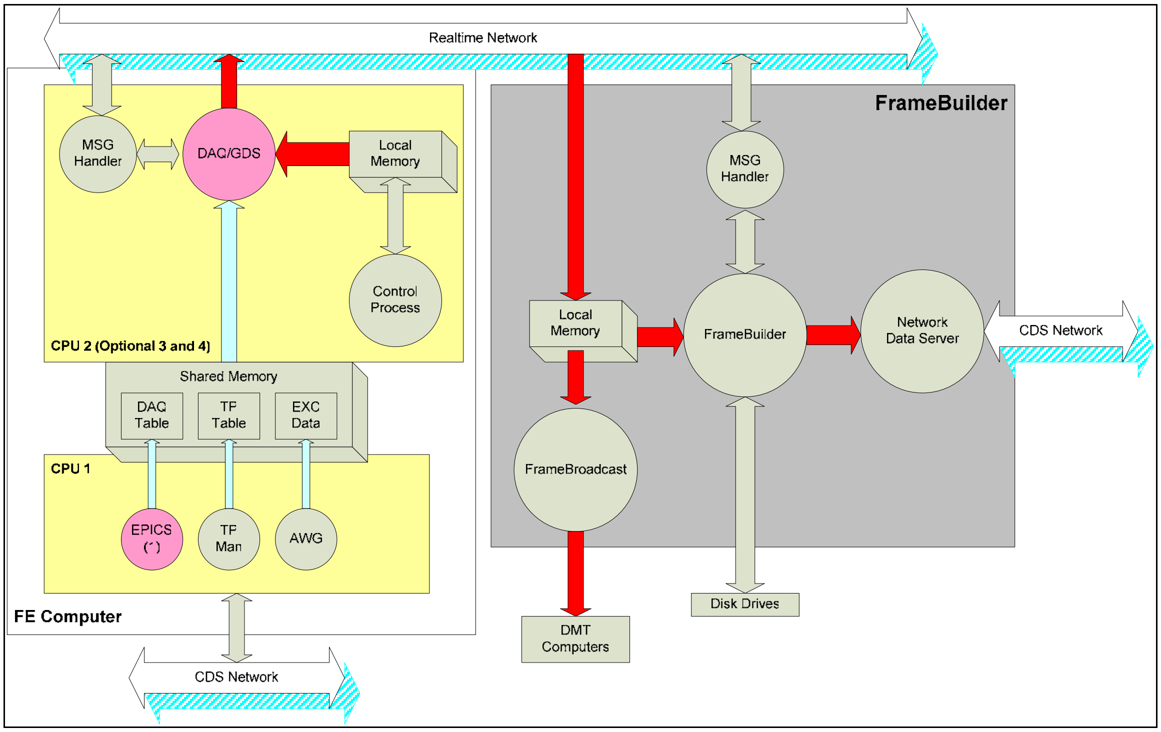

Figure 5: FE Software w/continuous data transmission

Though not shown in the above figure, the scheme for storing EPICS data is to be the same as described in the following design option.

8.1.2 Local Data Storage Design

A second design option is to have each FE computer store both its EPICS data and fast data to its local disk drives. The software modules required to make this change are shown in Figure 6.

The key changes in software include:

- The present FE DAQ/GDS process would be modified to write data to local shared memory instead of a realtime network.
- A slightly modified version of the present FrameBuilder code would run on the FE computer. This is a present capability of the CDS prototypes at Lasti and elsewhere. This code would acquire the requested fast data channels from the FE computer shared memory, format it and store it to local disk. The list of channels to be acquired, their rates and formats, would continue to be provided in a CDS .ini file.
- A new EPICS Data Collection Unit (EDCU) software module would be built to acquire and save all FE EPICS channels at a 1Hz rate. The list of all channels would be provided automatically by the CDS Realtime Code Generator (RCG), described in section 9 below. The actual EPICS data is already provided in shared memory by the realtime FE tasks.

Each FE computer is to run its own Network Data Server (NDS) task. This is the same code that presently runs on the DAQS FrameBuilders to send selected data to various CDS software tools. In this case, however, the NDS is tasked only in providing ‘live’ data ie it will not accept requests for data on its local disks.

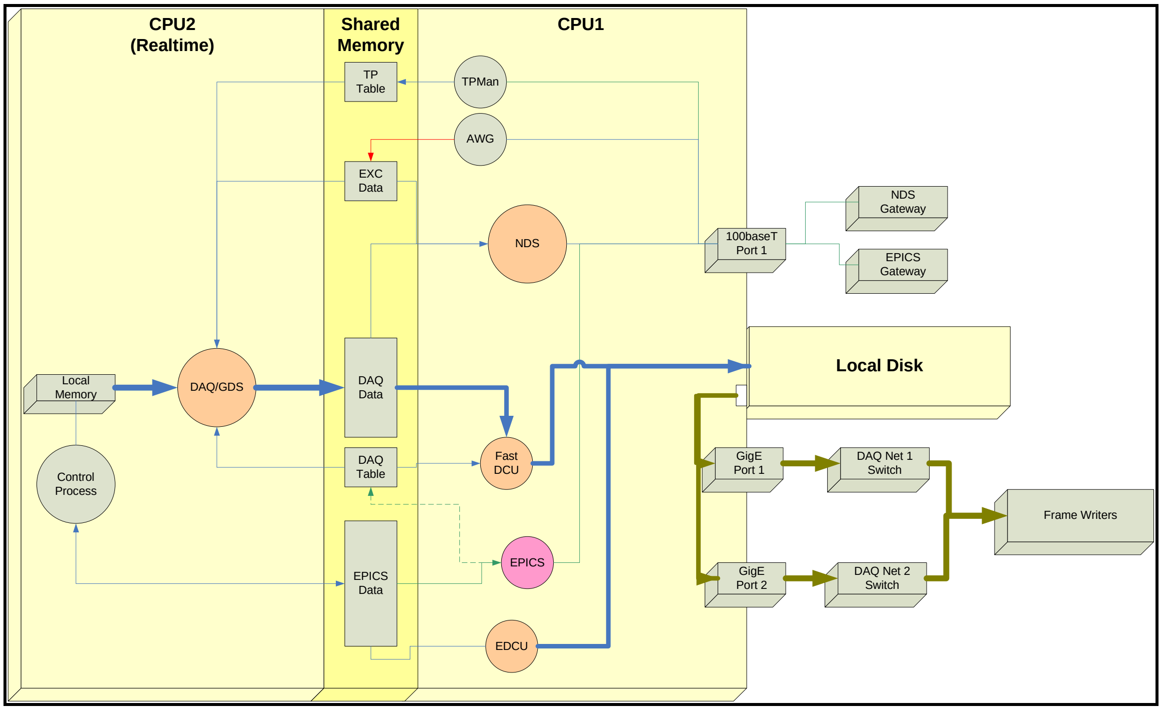

Figure 6: CDS Front End Data Acquisition

8.2 FrameBuilders

The design of the FrameBuilder software will be dependent on the FE computer design option chosen during the following design phases.

If the continuous data stream option is decided upon, the FrameBuilders continue to operate in much the same manner as in the present Ligo design. This was briefly described in the section 8.1.1 above.

With the option of each FE storing its data locally, the FrameBuilder design involves major changes:

- The FrameBuilders NFS mount each FE computer DAQS disk drive and copy their DAQS files to local disk at a periodic rate (FE computers could write data files every 32 seconds, as in present DAQS?). This would not require the use of a realtime network, as in first option. Rather, one of the FE computers standard GigE Ethernet ports would be used.
- The FrameBuilders would run diagnostics on the various FE files to verify that data is correct, such as Frame CRC checks.
- The FrameBuilders would combine the fast data frames from the various FE files into a single Frame file per IFO. The period of time covered by this data Frame could be longer than the individual FE frames.. .
- NDS would no longer run on the FrameBuilders. Live data would now be provided, via the NDSG, by an NDS running on each FE computer. Stored data would be served directly by the NDSG via a DAQS storage mount point.
- EPICS data files are retrieved from each FE computer and compiled in the same fashion as the fast data files. This concept applies to both this DAQS option, as well as the continuous data stream option.

With either option, the FrameBuilders will now also provide data compression using a TBD algorithm.

8.3 EPICS Data Collection Unit (EDCU)

New CDS FE computers will acquire and store their own EPICS channels. However, some legacy systems, such as vacuum controls, will remain which require EPICS channel archival. To support these systems, a separate EDCU will be provided.

The EDCU will essentially employ the software tools developed for new FE systems. It will not have its own hardware I/O, but rather will get data from the CDS Ethernet.

8.4 NDS Gateway

In the present DAQS, NDS software runs on the FrameBuilders and provides data, on request, to various operations software. In the AdvLigo design, there are numerous NDS running, on all FE computers for live data and the FW for archived data. To support this, a new NDS Gateway computer and software is proposed for AdvLigo. This NDSG will only respond to internal CDS computer requests for data and from selected user computers via a VPN connection. All other data requests from GC machines will be directed to the site Secure Network Data Server (SNDS) on a GC machine that has the LDAS archive mounted in Read Only mode. How this gateway fits into the new architecture and its connections are shown in Figure 7 below.

The basic operational sequence of the gateway is as follows:

- On startup, the NDSG requests connections to all NDS and a channel list from each, which it combines into a lookup table.
- A new feature of NDS is to be a secure log in feature (SNDS). Prior to responding to any data requests, it will be necessary for the user to provide identification and password authentication.
- The NDSG then processes data requests:
  - Channel list. Client software may request the list of available channels, grouped as fast channels, EPICS (slow) channels, and DMT channels. As part of the reply, data info is also provided in the form of sample rate, data type, archived or not, etc.
  - Archived data. The client software sends a request with a list of desired data channels and the desired time period. The NDSG, in turn, sends a request for this data from the FW NDS. If the requested data no longer resides on LDAS disk drives, the request will be forwarded to the SNDS on the GC data archive computer.
  - Live Data: Live data requests are relayed to the appropriate FE NDS. Responses from the various NDS are combined and formed into a single data response by the NDSG to the requesting client.

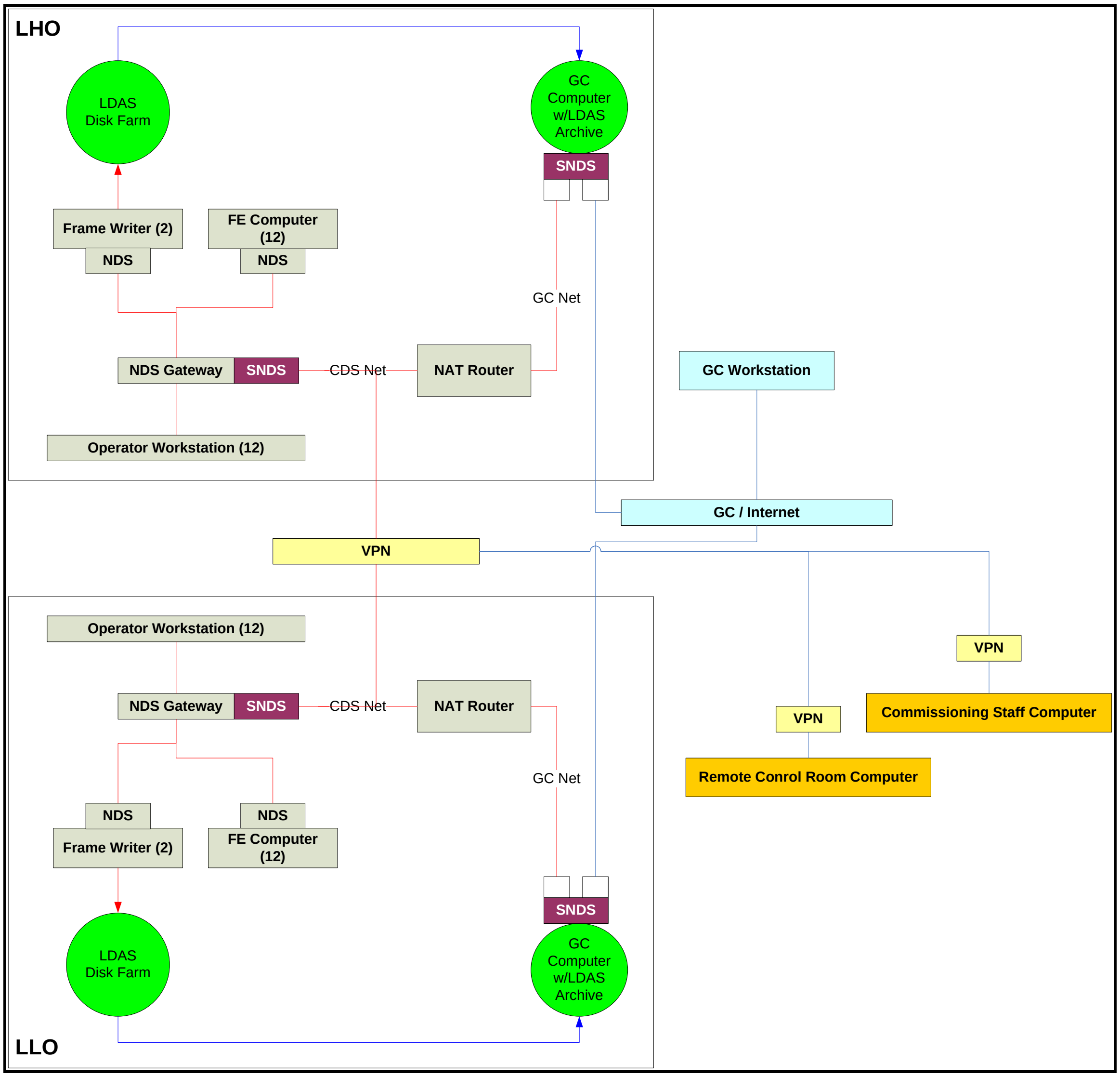

Figure 7: NDS Gateway Connections

9 CDS Subsystem Front End Controls

It is not within the scope of the CDS infrastructure to provide the computers, application software, I/O interfaces and signal conditioning required for the operation of individual subsystems. However, it is within the CDS infrastructure scope to specify the common subsystem hardware and software components such that the end result is a fully integrated system. As a reference, block diagrams of what the AdvLigo subsystem controls would look like, using the concepts described here, are shown in Reference B.

9.1 Computer, I/O buses and Signal Conditioning Hardware

An overview of the hardware involved in subsystem FE controls and their interconnection is shown in Figure 8.

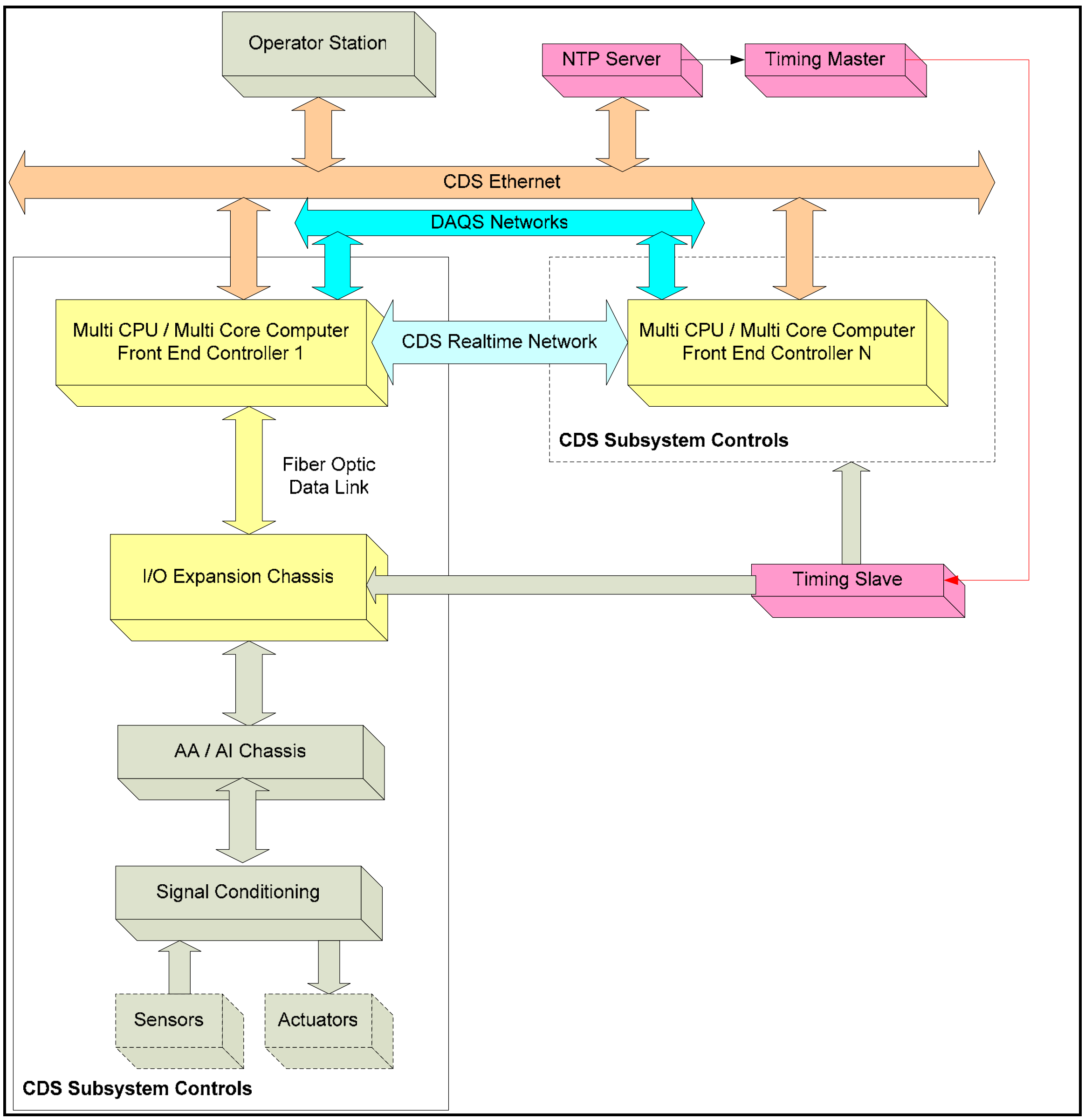

Figure 8: CDS FE Controls and Connections

9.1.1 FE Computers

FE computers are to be multi-core, multi-processor systems (present prototypes have two processors, each with two cores). These computers are to be located in the MSR, or other remote location, away from the subsystem I/O modules and signal conditioning units.

These computers will be specified to have a minimum of three Ethernet ports, preferably supporting GigE or higher rates. Two will be used for connection to the DAQS networks and one to the CDS Ethernet. A realtime NIC will be provided as part of the CDS infrastructure to support deterministic communications between FE computers.

Computer technologies will undoubtedly change over the next few years, which may affect the final design. However, the present design calls for a multi Opteron processor server running a realtime Linux operating system. These systems have been run in prototypes and are planned for use in Enhanced Ligo for the Output Mode Cleaner (OMC) controls.

9.1.2 I/O Chassis

Analog to Digital (ADC), Digital to Analog (DAC) and binary I/O modules are to be housed in a separate Digital I/O chassis. Presently, two design options are being pursued. The first design is based on the custom integration of commercially available components. A second design, based on a totally custom implementation, is being prototyped by LHO staff members. Only the baseline commercial unit is described here. The custom design will be detailed prior to the CDS preliminary design review.

The commercial component design is shown in Figure 9. It consists of a 4U high, rack mount chassis, with six PCI-X slots for the insertion of ADC, DAC and binary I/O modules. This bus contains a PCI-X to PCI Express (PCIe) bridge for data communications to the FE computer via a fiber optic link and a PCIe module installed in the FE computer. The fiber optic link runs at 2.5Gbit/sec, full duplex (single PCIe lane). Via this bridging and fiber optic link, it appears to the FE computer, its operating system, and CDS applications, as if the PCI-X I/O modules reside inside the FE computer. Latencies associated by having this link have been measured to be on the order of 2usec in prototype systems. It is envisioned in the future, that I/O modules meeting Ligo specifications will also be available with a PCIe interface. The PCI-X bus in the I/O chassis would then be replaced with an available PCIe bus. Among other things, this would afford more I/O slots per chassis (some commercially available units have as many as 19 slots).

For the connection of field cabling and timing clocks from the CDS timing system, custom interface modules have been designed for installation at the rear panel of the I/O chassis. All field connections are made at the back panel via SCSI cables. The interface modules then route the incoming signals to the particular connector and cabling required by the commercial I/O module. The timing signals come into a separate back panel module via two pin LEMO connections. These signals are then routed internally to the various I/O interface modules.

The present I/O chassis also contains its own DC power supply. In a final design, this supply will likely be removed and the unit modified to accept DC power from the AdvLigo DC power distribution system.

9.1.3 I/O Modules

The CDS will support a limited number of 16 bit (or greater), simultaneous sampling ADC and DAC modules. The module selection will be based on manufacturer’s specifications and subsequent in-house testing against Ligo specifications.

All ADC modules will be clocked at 64KSamples/second. The FE computers will provide software decimation filtering to the sample rate required by the individual FE controller, typically 16384Hz or 2048Hz.

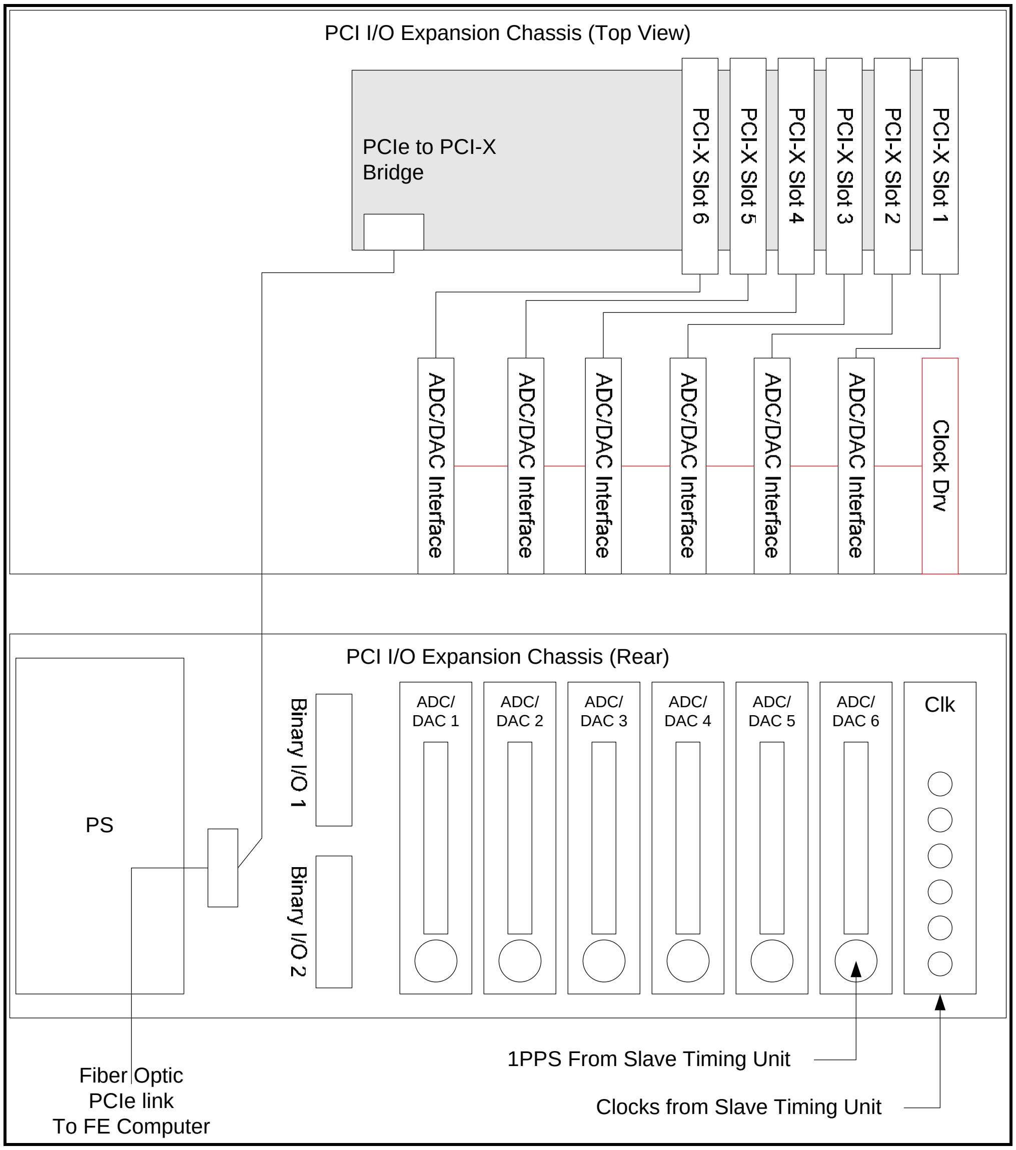

Figure 9: PCI I/O Expansion Chassis

9.1.4 Anti-Aliasing (AA) and Anti-Imaging (AI)

The CDS will provide a standard design for AA and AI chassis to support all CDS subsystems, with cutoff frequencies appropriate for a 64KSample/second data rate. Any further AA/AI and signal conditioning requirements for a particular subsystem are to be designed into its custom signal conditioning electronics.

9.2 DC Power Distribution

At the present time, there is a prototyping activity in progress to distribute DC power to all of the CDS electronics. This will be more fully documented, with test results, prior to the CDS preliminary design review. Basically, this system involves:

- Bulk DC power supplies, mounted in a location remote from the CDS electronics, to reduce the introduction EMI from these supplies.
- The bulk supplies provide 48VDC to each CDS electronics rack. This is done via a system of conduits or other method with appropriate shielding.

- Power regulators are provided at each rack to step down and regulate voltages required by the CDS electronics. These units also provide filtering to keep the noise below specified limits.

9.3 Electronics Construction and Testing

To the extent possible, all CDS electronic components will incorporate design standards. These standards are specified in separate documentation. Among the present standards for AdvLigo designs are:

- Equipment chassis enclosure, including power conditioning. A custom design has been developed during prototyping activities for 1U and 2U (or more) high 19” rack mount enclosures. Chassis of this design are readily available from a commercial vendor.
- DC power connections. All CDS chassis now have a standard power connection and dual power breaker as part of the design.

All CDS electronic chassis or testable subassemblies will be provided with the following set of basic documentation:

- Schematics and files necessary to construct the unit.
- Test plan.
- Traveler. For AdvLigo, a prototype system is being developed to provide electronic travelers, instead of the present paper system.
- Quick Start Guide. This is a short document which describes the basic features of the unit.

To the extent possible, all testing of CDS produced electronics will be out sourced for fabrication and testing to commercial vendors.

For system integration and testing, a plan is being developed to have a CDS rack of equipment as an assembly and test unit. This plan would involve:

- Sending all tested CDS electronics rack chassis to the sites.
- Providing site staff with rack assembly drawings and instructions.
- The site staff then pulls the prescribed electronic chassis from the stock of tested units and assembles them into the rack, per the instructions and using standard installation guidelines.
- The rack is now tested as a complete assembly, using the provided rack system test procedure.

To aid in this test process and after installation testing, new CDS designs include test units to be delivered with each system. These test boxes typically emulate a sensor and/or actuator to help verify that all up stream CDS electronics are functioning correctly prior to connection to the actual actuator/sensor hardware.

9.4 FE Computer Software

The CDS infrastructure will provide a common set of software modules for use in all FE systems, as described in the following subsections.

9.4.1 Operating System

As part of the CDS infrastructure, a realtime Linux operating system will be provided for all FE computers.

9.4.2 FE Code Modules

The following figure depicts the code modules to be provided for each FE computer as part of the CDS infrastructure development. (DAQ components were previously shown and described in Section 8).

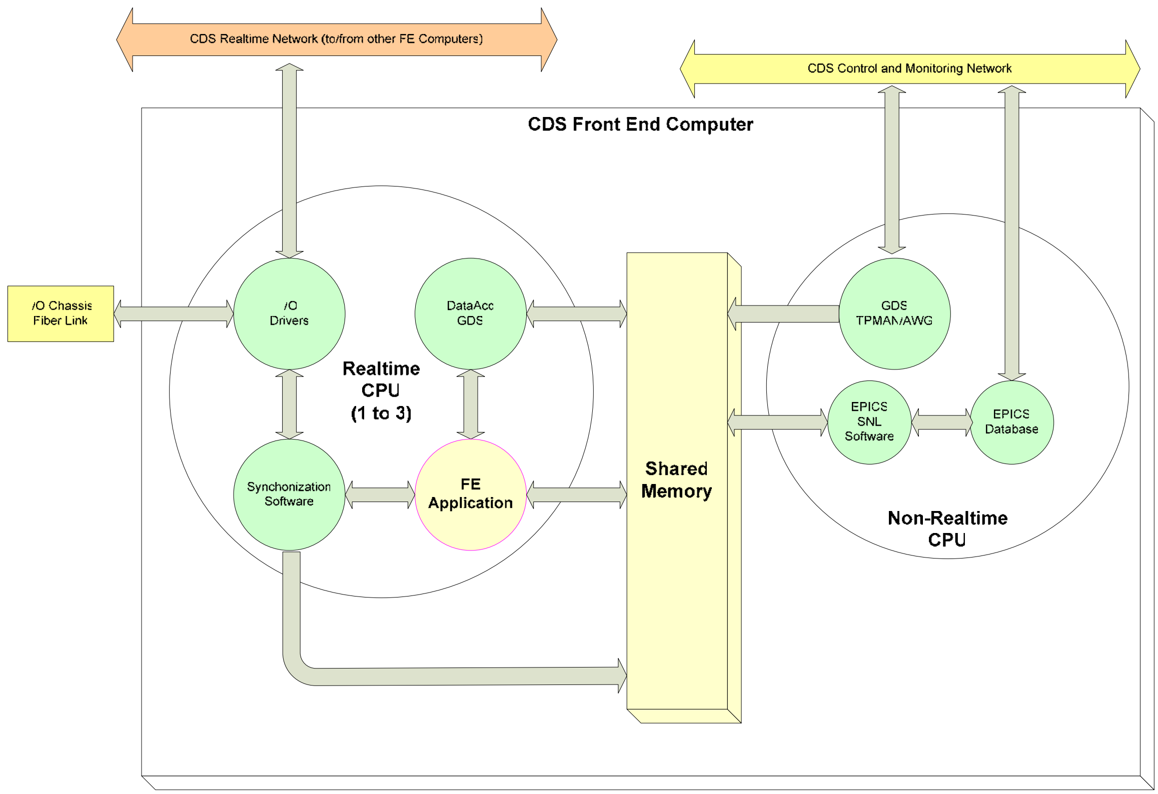

Figure 10: FE Controls Software Modules

9.4.2.1 Realtime Code Modules

For AdvLigo controls, it is intended that only the FE application specific to each subsystem controller be unique. All other supporting realtime software is to be provided and maintained as part of the CDS infrastructure. These supporting modules exist in three CDS software files and are inline compiled as part of each FE application into a single executable file. These common components are:

- I/O Drivers. A set a software modules are included to provide data communications between the realtime application and all supported I/O modules and the CDS realtime network.
- DAQ/GDS Software. Code is provided which handles all FE data acquisition and connections to the Global Diagnostics System (GDS) TPMAN and AWG. This code communicates data via the CPU shared memory or via the CDS realtime network, dependent on the DAQS design option chosen (as described in Section 8).
- Synchronization Software. This code module provides for system initialization and proper sequencing of all the code components, including the FE specific application. This code is slaved to the ADC modules to operate synchronously with the desired sample rate. This code also provides various system diagnostics, including ADC/DAC overflows, timing information, and FE status information.

Once compiled with the FE application, this code is run on and locked into one of three available realtime processors on the FE computer.

9.4.2.2 Non-Realtime Code Modules

The first CPU of FE computers will run common Linux tasks and non-realtime critical CDS software modules. These code modules include:

- EPICS Database. An EPICS standard database will be generated for on run on each FE computer.

- EPICS sequencer. A code module, based on EPICS State Notation Language (SNL), will run on each FE computer. Its primary purpose is to relay data between the realtime software, via shared memory, into the EPICS database. Once in the database, this data is available, via standard EPICS CA, to the CDS Ethernet. A secondary purpose for this software is to load the DAQS .ini configuration file into the shared memory for use by the realtime DAQ/GDS code module.
- GDS TPMAN. In the Ligo CDS, a single TPMAN was provided to support all of the FE computers associated with one interferometer. In the new design, a TPMAN runs locally on each FE computer.
- GDS AWG. As with TPMAN, the new design includes a separate AWG on each FE computer.

9.4.2.3 Multiple FE controls on a single FE computer

The model described above is for a single FE application. However, the software presently supports up to three separate FE control applications per FE computer, using three of the CPUs locked to realtime. Each application can have its own realtime application and supporting DAQS/GDS and EPICS tasks. Each FE task may also run at its own sample rate. So, for example, CPU1 could run an LSC task at 16KHz, CPU2 an ASC task at 2KHz, and CPU3 a fast loop at 64KHz. Each task may be totaling independent, with the ability to stop/start any of the three applications without affecting the others. It is also possible to communicate between the tasks if data from one is required by another.

9.4.3 CDS Realtime Code Generator (RCG)

To aid in the development and documentation of specific FE application software, the AdvLigo CDS design includes an automated RCG. A block diagram describing the code generation process is shown in Figure 11. This RCG has been used to develop all of the application code for all Lasti and 40m prototypes over the past year.

The Matlab Simulink graphical editor is used as the application definition tool. CDS provides a .mdl parts library of all parts supported by the RCG. The user drags and drops these basic blocks into their own application and makes all of the appropriate linkages. Among the parts supported are:

- CDS standard filter modules
- Matrix parts, with user defined number of inputs and outputs
- Limited Simulink standard parts, such as math functions.
- Communication parts, which allow communications between realtime computers via the CDS realtime network.

It is intended that CDS staff continue to add parts to the library as user needs arise. An example of an application built using this Simulink interface can be found at .

Once the user has finished the graphical layout of the application, a Matlab .mdl file is saved. This file is then moved into the CDS defined directory and entered into the CDS CVS repository. At this point, the application is ready to be built.

To now build the application, a ‘make applicationName’ command is entered on the command line at the top level CDS directory. This results in the following sequence, shown in Figure 11.

- A Perl script (feCodeGen.pl) is invoked. This script reads in data from the application .mdl file and parses all of the parts in the file, as well as all of the links between these parts. It then generates its own parts list in the sequence that the parts are to be computed, based on the .mdl file part links.
- This Perl script then generates five new files:
    - A header file, which describes the data variables that are to be placed into the shared memory between the realtime process and the EPICS process for communications between the two.
    - Makefiles for both the realtime code and the EPICS codes.
    - C code file, which will be used to generate the realtime application.
    - A text file, which will be used by another Perl script to generate the EPICS software.

- The realtime compiler is now invoked. The compile step inlines the user application with the supporting libraries provided by CDS. The end result is a realtime executable file.
- A second Perl script is invoked (fmSeq.pl) which produces:
  - EPICS SNL code, which is then compiled. The resulting executable code is used to move data between the realtime application and the EPICS database, via shared memory.
  - An EPICS database (.db) file.
  - A number of basic MEDM operator displays. These may be used, as is, for initial testing and/or used to copy and paste parts into a user defined MEDM screen.
  - An autoBURT file, for use by the EPICS BURT tools.
  - The header for a filter coefficient file, for later use by foton to define filters.
  - A GDS .par file, which describes all of the GDS test points available in this application.
  - A skeleton .ini file for use in data acquisition.
  - A file for use by the EDCU task to acquire all EPICS channels as part of the data acquisition process.
  - A startup file, which is run from the command line and starts up all of the code.

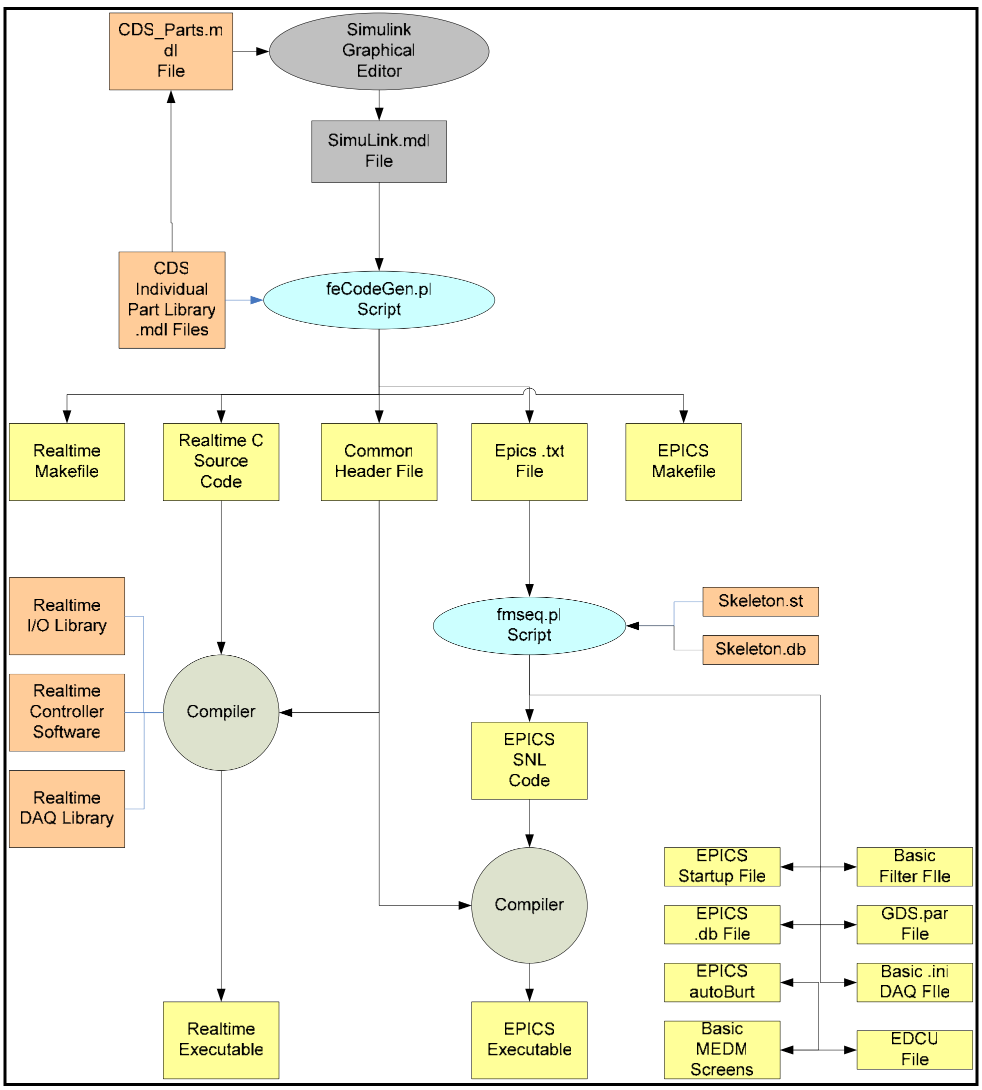

Figure 11: RCG Block Diagram

10 Diagnostic Monitoring Tools (DMT)

A number of computers for each interferometer are provided to run diagnostics in support of interferometer operations and commissioning. These are designated at DMT machines.

10.1 DMT Computers

The CDS infrastructure will provide five computers per interferometer for use as DMT machines. The specifications for these computers are TBD.

Most of the software that runs on these machines is provided by Ligo Science Collaboration (LSC) members. However, the CDS infrastructure will continue to support the interface software for these computers that provides a connection to the DAQS.

10.2 Frame Broadcaster

The present requirements still call for the broadcasting of data, in the form of Frames, to the DMT computers. This is presently done by running a FrameBroadcaster task on one of the FrameBuilder computers. This Frame contains all of the site data, as well as any selected TP data.

In the AdvLigo design, the FrameBroadcaster is to be a separate computer. It will broadcast a custom Frame, based on its own DAQS .ini file. The exact mechanism of how the FrameBroadcaster get its data is still TBD.

11 Enhanced Ligo (ELigo) Transition

It will be necessary to add new control and monitoring systems for the Output Mode Cleaner (OMC), and the seismic and suspension systems that support it, in late 2007 as part of the ELigo upgrades. These new systems are being designed using the concepts presented in this document. However, these new components must integrate with the existing CDS. This primarily involves integrating the new and existing DAQS.

11.1 Data Acquisition

11.1.1 Data Frames

Prior to the next science run, at a time to be coordinated with LDAS, it is planned to move to the new scheme of writing one fast data frame and one slow data frame per IFO.

11.1.2 Slow (EPICS) Data Acquisition

For the acquisition of slow data, this will require the installation of a new EDCU per IFO to acquire data from existing Ligo systems. It will also require new software on the FrameBuilders to gather up the data files from the EDCU and new control systems and write a combined slow data file.

11.1.3 Fast Data Acquisition

The first option is to have new FE computers use the continuous data transmission option (described previously) over the new realtime network. The FrameBuilders would continue to receive data via the existing RFM networks and from this new network. They would then merge the data into the single, 32 second frame file.

If each FE computer writes its own fast data frame locally, then:

- The existing FrameBuilders continue to received data from existing CDS computers and compile a fast data frame.

- A new task, either on the existing FrameBuilders or a new machine, gathers up the local frame files from the FrameBuilder and the new systems and compiles a single IFO fast data frame.

Essentially, in this latter scheme, the present FrameBuilders act as just another FE computer. Instead of receiving data via local memory, they get data via the existing RFM networks.

11.1.4 NDSG

Whichever DAQS scheme is used, the new NDSG will be required for operation of ELigo.

11.2 EPICS Gateway

It is planned to install new internal EPICS gateways within the CDS at both sites. Among other things, this should help reduce the loading on existing EPICS systems due to multiple data sockets involved with data requests.

11.3 If Available Additions

Some new concepts of AdvLigo CDS, such as script servers, will be introduced during the commissioning period for ELigo if available in time.