LASER INTERFEROMETER GRAVITATIONAL WAVE OBSERVATORY

LIGO Laboratory / LIGO Scientific Collaboration

LIGO-T1000560-v1 *LIGO* September 25, 2010

CDS
Software Development Plan
FY 2011

R. Bork

Distribution of this document:
LIGO Scientific Collaboration

This is an internal working note
of the LIGO Laboratory.

California Institute of Technology
LIGO Project – MS 18-34
1200 E. California Blvd.
Pasadena, CA 91125
Phone (626) 395-2129
Fax (626) 304-9834
E-mail: info@ligo.caltech.edu

Massachusetts Institute of Technology
LIGO Project – NW22-295
185 Albany St
Cambridge, MA 02139
Phone (617) 253-4824
Fax (617) 253-7014
E-mail: info@ligo.mit.edu

LIGO Hanford Observatory
P.O. Box 1970
Mail Stop S9-02
Richland WA 99352
Phone 509-372-8106
Fax 509-372-8137

LIGO Livingston Observatory
P.O. Box 940
Livingston, LA 70754
Phone 225-686-3100
Fax 225-686-7189

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

Table of Contents

1 Introduction

The purpose of this document is to describe the status and plans for CDS software development, test and documentation over the next year, beginning September 2010. It is intended that this document be reviewed and updated, as necessary, on three to six month intervals.

2 Scope

The scope of CDS provided and maintained software includes:

- Computer operating systems for all CDS computers.
- Real-time Control Core Software
    - Tools used to develop real-time control applications, such as the CDS Real-time Code Generator (RCG).
    - Real-time "wrapper" software provided for each user application, referred to later as the Real-Time Sequencer (RTS) software.
    - Standard real-time code modules, such as IIR/FIR filter modules, etc.
    - I/O drivers.
    - Arbitrary Waveform Generator/Test Point Manager (AWGTPMAN), used to set real-time software test points and inject test waveforms.
    - Experimental Physics and Industrial Control System (EPICS) interface, used to communicate data between the CDS control network and the real-time applications.
    - Real-time data acquisition software modules compiled into all applications to extract defined DAQ and Test Point (TP) data from and inject AWG signals to all real-time applications.
    - Networking software to communicate DAQ/TP data from the real-time application to the DAQ network.
    -
- DAQ System Core Software, including:
    - Interface software to receive data from the various real-time systems, including 'fast' data channels from real-time applications and 'slow' EPICS data.
    - Data archival, employing the LSC standard FRAME data format.
        - Frame library software provided by LIGO Data Analysis System (LDAS) group and is not in scope.
    - Local data distribution to operator stations via the CDS networks. This include both 'live' data feeds and retrieval of data from archive.
        - Archive retrieval limited to data locally available on the LDAS disk system ie not from on site or off site tape systems.
    - Transmission of DAQ data to Diagnostic Monitoring Tool (DMT) computers.
- Operations support software, including:
    - A limited set of Global Diagnostic System (GDS) software:
        - Diagnostic Test Tool (DTT)
        - Foton, the IIR filter coefficient definition tool.
        - EZCA scripting tools
    - Dataviewer, used to display data from the DAQ system.

  - Limited set of EPICS MEDM displays, devoted primarily to providing CDS diagnostic information to operations staff.
  - Support of CDS interfaces to software provided by others, such as Matlab, EPICS, etc.

Specifically considered out of scope for CDS development are the following, other than to maintain current interfaces:

- EPICS Software: While EPICS will continue to be used for aLIGO, it is intended to use the EPICS packages as distributed by the EPICS collaboration.
- TDS: Scripting tool that is part of the GDS package.
- Guardian: Scripting tool being developed at MIT.
- mDv: Matlab scripting and display tool
- ligoDv: Matlab based dataviewer and analysis package
- NDS2: Security enhanced version of the CDS Network Data Server (NDS) used to access archived data from LDAS disk/tape systems.
- Diagnostic Monitoring Tools (DMT), other than to provide a Frame broadcast from the DAQ system to support the DMT interface software.

3 Overview

The remainder of this document describes the present status and continued development plan for the software listed in the scope section above. In addition, it also provides an overview of other CDS activities, including source code control, code distribution and code testing.

For each section, there is a brief description of recent progress and changes, along with a table of planned activities and staff assignments.

4 Primary Objectives

The primary goal for the year is to complete aLIGO CDS software development, test and documentation in time to support aLIGO commissioning. Toward this end, top level goals by quarter:

QTR 1: Complete development and test of real-time and DAQ core software and install at sites.

QTR 2: Switch development focus to operations support software.

QTR 3: Code cleanup and code reviews.

QTR 4: Operations Support Software and CDS software documentation complete.

During this time, there will also be additional demand in support of aLIGO installation and commissioning, along with continued support of various CDS installations, as noted in Section 12.

5 Staffing

The CDS software staff has not grown over the past year. However, with the science run in progress, the group has gained more availability of Barker and Thorne, which has been advantageous. The present plan is to off load most of Thorne's sysadmin responsibilities at LLO with a new hire, providing more availability for CDS software development work.

A list of CDS software staff and their assignments appear in the following table. To first order, we have tried to assign at least two people to any one major category to spread the knowledge base and provide internal code review.

Name	Primary Assignment	Secondary Assignment
R. Bork	Group Lead	Real-time software
A. Ivanov	DAQ Software	Real-time / Linux OS
M. Aronsson	RCG	Software Test
D. Barker	SysAdmin	Real-time software
J. Batch	GDS software	RCG
K. Thorne	SysAdmin	DAQ/GDS software
V. Sandberg	Software QA	

6 System Administration

6.1 Source Code Control

Over the past year, CDS software has been migrated from a Code Version System (CVS) repository to a Subversion (SVN) repository. This SVN repository is hosted at the LIGO Hanford Observatory (LHO) site.

The Global Diagnostic System (GDS) software repository, which includes software for which the CDS group now has responsibility, as described in the scope section above, has also been migrated from CVS to SVN. Along with the move, a fair amount of work went into ensuring that all CDS supported modules had proper build procedures to produce working executables for various platforms.

A new SVN has now also been created for the purpose of controlling software used within the CDS but is developed by other LIGO staff. A top level directory exists for each LIGO subsystem, with the following basic subdirectories:

- Matlab Simulink models, produced by control application developers to define real-time control applications using the CDS RCG.
- C code modules, developed by others, in support of real-time applications produced by the CDS RCG.
- Scripts, used in testing or operations.
- Channel Data, which provides a place to store DAQ channel configuration files and filter coefficient files.

Beyond these SVN repositories, various LIGO subsystems also maintain their own repositories, some of which contain software deployed in CDS. A study has begun on how best to link to these repositories and provide documentation.

Documentation on the CDS SVN repositories is contained in LIGO-T0900531.

SVN Plan

Description	Assigned to	Due Date
Locate CDS related software kept in non-CDS SVN repositories and document.	D. Barker	15 Nov 2010
Plan method to connect to CDS production system and document.	D. Barker	30 Nov 2010

6.2 Software Distribution

Beyond supporting the site interferometers, the CDS software is also used on various LIGO systems in support of research, prototype and subsystem test activities. Since there are many software applications and tools that run within CDS, a code distribution site and installation instructions are being developed. The intent is that this site contain tested and released software executables.

There are also a number of applications run in the iLIGO control rooms that was developed some time ago by both LIGO and non-LIGO staff. When deploying new systems, it has been found somewhat difficult to find all these components. These are being listed and located for placement on the distribution site as well.

Description	Assigned to	Due Date
Determine software packages to be included in the distribution area.	D.Barker K. Thorne	30 Oct 2010
Implement distribution site.	K. Thorne	1 Jan 2011

6.3 Software Installation and Configuration

The installation and configuration of CDS software in production systems is outlined in two documents:

1) LIGO-T1000248, which describes the CDS file structure.
2) LIGO-T1000379, which describes the standard setup and environment.

6.4 Software QA

6.4.1 Test Plans

An integrated system test plan is presently in development. This will include all test procedures that need to be run to validate CDS software, from individual code modules to fully integrated IFO distributed CDS and DAQ systems. The test plan is documented in LIGO-T1000561.

6.4.2 CDS Software Reviews

It is intended to have both internal and external software reviews.

Internal to the CDS group, the staff has been assigned to projects as a two person team, with one assigned as the primary developer and the second as the primary reviewer/tester. Part of the intent here is to have at least two people with expertise in all of the CDS software.

In addition, weekly CDS software staff meetings are held with varying agendas, including code design, code walkthroughs, bug list updates, etc.

6.4.3 Change Control

Once software is reviewed and declared complete, any further changes to the software will require a change request and review process. To support this, the following documentation needs to be written.

Description	Assigned to	Due Date
Produce change request procedure document	V. Sandberg	July 2011
Produce standard CDS software change request form	V. Sandberg	July 2011

6.4.3.1 Code Bug Reporting

The Bugzilla software bug and tracking system is used for reporting CDS software problems. Once a month, at a CDS software staff meeting, this bug list is reviewed to ensure problems are being addressed. This bug tracking system is described in LIGO-T1000496.

7 Operating Systems

Three operating systems are presently supported for use in aLIGO:

- CentOs and ubuntu Linux : Operator stations and non-realtime controls (in down select process)
- Commercial Real-time Linux / gentoo Linux with CDS real-time patch: Real-time control computers
- Solaris: Operator stations and DAQ Frame writer computers.

The primary activity planned in this area is to move away from the commercial real-time operating system to a General Public License (GPL) Linux. The primary motivation is cost and complicated licensing involved with the commercial system.

There has been good progress in this area over the past month, with systems now running in test both on campus at Caltech and in the lab at AEI Hannover, Germany. To get the real-time performance required for CDS, a Linux kernel patch was developed. This patch provides the capability to lock specific CPU cores, on multi-core CPU platforms, to CDS defined real-time tasks. This patch removes the core from the standard Linux scheduler and interrupt tables, thereby allowing CDS code to run deterministically without intervention from the operating system. This patch was recently applied to the latest Linux kernel release (2.16.34) and reviewed by two Linux consortium members at a meeting at AEI. One member is an expert on kernel device drivers and the other has a lead role in the PREEMPT real-time extensions to Linux. A similar patch is also being worked by the PREEMPT group, which they call cpu isolation. This is in response to numerous requests from the High Performance Computing (HPC) computing community, which also desires this core locking feature. Once the PREEMPT group has completed their implementation, they would like to send it to us for test with our software.

Work has also been done to allow CDS real-time computers to boot Linux across the network from a single boot server. This has several advantages:

- No disk drives in real-time machines ie equipment cost savings
- Operating system (OS) updates, patches and backups only need to be done on one system, thereby reducing maintenance time.
- All computers are known to have the exact same OS

This remote boot capability is now functional in the development lab and will be moved to larger test platforms soon. In initial tests, Linux OS boot time is now 8 seconds from start to login prompt.

Description	Assigned to	Due Date
Produce a standard Linux OS, with patch, for use in CDS systems.	A. Ivanov	1 Oct 2010
Install Linux OS on LHO test system for integrated testing	A. Ivanov	1 Oct 2010
Document Linux patch, along with instruction set for building new kernels.	A. Ivanov	1 Jan 2011
Document disk-less boot procedures	Ivanov / Thorne	1 Jan 201

8 Real-time control computer software

Presently, the latest version of deployed real-time and DAQ software is tagged version 1.9 in the CDS SVN repository. The plan calls for a new major release, V2.0, to be installed on the LHO DAQ test system and H2 aLIGO system within the next month. It is being tagged as a major release, as it will not be completely backward compatible with any V1 releases. Among other things, this release is compatible with the new GPL Linux OS. Further details and plans are discussed in the following sections.

8.1 Real-time Code Generator (RCG)

Primary work on the RCG over the recent past has been in the addition of new parts and the modification of Makefiles and build routines to support the production installation code directory structure. The RCG is described in LIGO T-080135. This user's guide is presently in the process of being updated to match the RCG V2.0.

New additions to RCG being deployed with V2.0:

- New Inter-process communication (IPC) parts: Modified to allow automatic generation of IPC lookup tables needed to define communication paths between tasks within the same computer and/or across CDS real-time networks.
- Modified ADC part: ADC part used Matlab Bus Creator part to define an ADC. This needed to be changed to allow next item.
- Matlab Bus Creators and Selectors: Allows user to define signal buses, primarily to provide cleaner code diagrams.
- Matlab Signal Tags: Allow connection from one point to another in a RCG model without having to draw/connect lines. Again, this is provided to make model cleaner.

After getting RCG 2.0 out to test, a couple of other items need priority attention:

- Clear compilation error messages from RCG. At present, when RCG models fail to compile due to an error in the model, the error messages are not sufficiently clear and/or non-existent, making it difficult for the user to find and fix the problem.
- Ability for users to define decimation and up sampling filters used in the real-time code in reading data from ADC and writing data to DAC modules.
- Add EPICS record types in support of Guardian scripts (being developed by MIT staff)

Description	Assigned to	Due Date
Complete check in of RCG V2.0 parts library	M. Aronsson	1 Oct 2010
Add user defined decimation filtering	M. Aronsson	15 Nov 2010
Add EPICS record types for Guardian	M. Aronsson	15 Oct 2010
Compile error message improvement	M. Aronsson	30 Oct 2010

8.2 Real-time "Wrapper" Software

The RCG produces real-time C source code from the user defined RCG model. This C source, along with any other user provided C source defined in the model, is compiled into a single, in-lined executable along with CDS provided real-time core software. The major components of this core are:

- Real-time Sequencer (RTS), which controls timing and interfaces. This software is described in LIGO T-0900607, with requirements documented in LIGO-T0900603.
- Filter module software. This code provides a standard method for performing IIR and FIR filtering. This code is documented in LIGO-T0900606. This software is essentially

unchanged from the iLIGO/eLIGO implementation. However, there is a new filter module implementation using Bi-Quad Form (BQF) IIR filtering, which may be implemented through a compile option. This code has only undergone minimal testing, but is advertised to reduce the digital computation noise.

- DAQ/GDS interface software: This software provides a standard interface between the real-time application and the DAQ system, along with injection of test signals and transmission of test signal data to the DAQ.
- Real-time inter-process communications (IPC) software: A standard method to communicate synchronous data between real-time applications, documented in LIGO-T1000587, via:
    - Computer shared memory
    - iLIGO type reflected memory (communications between end and corner station computers only)
    - PCIe network reflected memory between co-located computers (up 6 meters via cable, 100 meters over fiber)

Note that, for aLIGO, only real-time IPC traffic will exist on the reflected memory networks ie will no longer carry DAQ, GDS or EPICS data traffic.

8.2.1 Real-time Sequencer (RTS)

The requirements for the RTS are defined in LIGO T0900603 and design in LIGO T0900607. Primary recent activity on this code has been:

- Modifications necessary to use GPL Linux
- The addition of code necessary to interface with the PCIe real-time control network.
- Compile options to run as either an I/O processor (IOP) or "slave" user application process.

The IOP is new, first released in V1.9. The IOP software has several primary functions:

- Provide startup synchronization for all I/O devices and real-time applications.
- Provide ADC data to all real-time applications and write all DAC data from real-time applications.
- Map and provide I/O address pointers for all other PCIe devices, such as binary I/O cards, reflected memory modules/networks, etc.

For aLIGO systems, a single IOP runs on each real-time control computer. It is set to automatically start after system boot. All other "slave" user real-time applications connect and synchronize with the IOP on startup.

Code development for the RTS software is nearing completion. After integrated testing with V2.0 and completion of documentation, this code should be ready for code review.

Description	Assigned to	Due Date
Complete check in process for V2.0	A. Ivanov	1 Oct 2010
Complete Integrated Test with V2.0 at LHO	D. Barker	1 Nov 2010

8.2.2 Real-time DAQ and GDS Interface Software

A single function call is used in all CDS real-time code to provide an interface to DAQ and GDS. This code is documented in LIGO T0900638. The primary changes to this function have been to shift reading and writing from/to iLIGO reflected memory to local computer shared memory to interface to the awgtpman and DAQ network drivers. This software is essentially complete, pending full integrated system testing.

8.2.3 Real-time Communications Software

For aLIGO, a standard communication method has been developed to handle all real-time IPC, regardless if between two tasks on the same machine or across several computers. This software is documented in LIGO T1000587. This software is complete, pending final test and review.

8.3 Real-time Support Software

8.3.1 AWGTPMAN

There are three primary changes to awgtpman from iLIGO to aLIGO:

- One awgtpman per real-time application. In iLIGO, there was a single awgtpman for each interferometer.
- Awgtpman task now resides on the same computer as the real-time application, communicating via shared memory. In iLIGO, communications was via a reflected memory network.
- Awgtpman is synchronized by the real-time IOP. In iLIGO, awgtpman was synchronized by a separate IRIG-B time code receiver module.

Other than these interface changes, no other modifications have been made. This software development is complete, pending review.

8.3.2 EPICS Sequencer

In iLIGO, communication between real-time tasks and EPICS was done via a reflected memory network, with the EPICS interface running on a separate computer. For aLIGO, this EPICS to real-time interface software has been moved to the same computer as the real-time application that it supports. Communication is done via local shared memory.

Only one additional change is planned. This interface task presently runs asynchronously at approximately 5Hz. This will be modified to run synchronously with the real-time task at either 8 or 16Hz.

Description	Assigned to	Due Date
Synchronize EPICS interface to real-time	A. Ivanov	1 Jan 2011

8.3.3 DAQ Network Driver

In iLIGO, all DAQ data was sent from the real-time control system to the DAQ system via a reflected memory network. For aLIGO, this data is transmitted via gigabit Ethernet. This resulted in two primary code changes:

- Change network interface software.
- Move network communication out of the real-time task into a separate executable, with communication between the two via shared memory.
    - Network connects/reconnects and diagnostics lead to inconsistencies in real-time task run times.
    - By using shared memory, same real-time code can be used either with the new network task, in a distributed system, or with DAQ software running directly on the real-time computer (stand-alone CDS computers used in support of subsystem testing).

Software development is complete, however documentation and review is still required.

9 DAQ System Software

The core DAQ software is essentially the same as that presently running at the sites as part of eLIGO and documented in LIGO-T0900636. The primary changes have been to the communications interface. In eLIGO, Myrinet and Infiniband are being used. For aLIGO, this will shift to only use Ethernet for all connections. At present, there are only a couple of tasks planned, as shown in the following table. However, there are a few key items that still require investigation, which could add to the list:

- Possible upgrade of CDS Network Data Server (NDS) to incorporate some features added by John Zweizig in the development of NDS2 (primarily developed for data retrieval from off site archives with added security features).
- EPICS data handling:
    - Can the DAQ system handle the much larger EPICS data load (>75000 channels anticipated) for aLIGO To the extent possible, this is to be tested soon using the LHO DAQ test system and computers coming in for aLIGO controls.
    - While the system will still need to support EPICS data via EPICS channel access over Ethernet for legacy iLIGO systems, should new systems send their data directly along with the 'fast' data?
    - How best to archive this data. The LSC Frame format requires about 100bytes of descriptor data for each channel being recorded. EPICS channels are collected at a maximum rate of 16Hz (typically less), so the overhead data comprises a fair fraction of the channel data at the present Frame time period of 32 seconds per file.
- Port of Framewriter software to Solaris. In support of early commissioning, it is presently planned to run in the same configuration as for eLIGO ie Framewriter runs on a Linux machine, which writes data via an NFS mount to a Solaris machine, which, in turn uses QFS, required to write data to the LDAS disk system. While this two step process is not the most desirable, it has worked well in eLIGO and there is a fair amount of uncertainty in the future direction of Solaris under the new ownership (Oracle).

Description	Assigned to	Due Date
EPICS data load testing	D. Barker	1 Dec 2010
Rework of EPICS data collection (if necessary, once above tests are performed.	A. Ivanov	1 March 2011
Port DAQ Framewriter to Solaris (?)	A. Ivanov	15 Jun 2011

10 GDS Software

For aLIGO, CDS has taken on the additional task of maintaining and upgrading certain Global Diagnostic System (GDS) software, limited to:

- Diagnostic Test Tool (DTT) package
- Foton
- Awgstream

The primary activities in this area over the past year have been:

- Foton enhancements and bug fixes, among them
    - File locking, to prevent overwriting by other Foton sessions
    - Copy/paste feature
    - Import feature to allow importing of Matlab generated filter coefficients
    - Allow gain only filters to be defined
- GUI tool fixes
    - Fix plotting to scale initial display of graphs to fit in the display window.
    - Fix color selection menu to display colors and allow selection of colors for graphs.
    - Fix printing of plots to fit the paper properly and eliminate extra blank pages.
    - Fix Save As... dialog box to check the proper directory for an existing file of the same name.
    - Implement file extension checking in Save As... dialog box for printing of plots.

Planned activities are listed in the following table.

Description	Assigned to	Due Date
Complete present list of GDS fixes/modifications	J. Batch	1 Nov 2010

11 Operations Support Software

Given staffing limitations and the focus on real-time and DAQ software over the past year, there has only been minimal effort on the operations support software side. For initial aLIGO installation and commissioning, it is planned that iLIGO operator systems software will be used as is. As staff moves away from real-time and DAQ software, it is intended that more focus will be placed on this software, starting in early calendar 2011.

While there is a list of requests for enhancements to iLIGO operator software, the primary new software being planned is a GUI based oscilloscope package. Investigation of what outside packages might be available and feasibility of connecting with DAQ NDS is in progress. At the moment, this is deemed the highest priority in this software category.

Only near term activities are listed in the following table, which are primarily design studies. Once these are complete, the list will be expanded to include development activities.

Description	Assigned to	Due Date
Investigate Oscilloscope Software	K. Thorne	15 Jan 2011
Investigate interface to Baudline Software	K. Thorne	15 Jan 2011
Determine Conlog suitability/options for aLIGO	D. Barker	15 Jan 2011
Determine BURT suitability/options for aLIGO	D. Barker	15 Jan 2011
Build EPICS Gateway between operations net and real-time computers Verify routing of GDS RPC calls in this configuration	Thorne	15 Nov 2010

12 Installed Systems Support

Over the course of the next year, CDS equipment and software will be installed at both sites. In addition, there are a number of CDS systems already installed to support various subsystem tests. The latter range from stand-alone test support systems to larger distributed systems. All of this will require some effort from the CDS software staff for installation and maintenance.

A list of CDS installed base systems, not including site main IFO controls, is provided in the following table.

Description	Status
Caltech 40m lab	Upgrade from iLIGO to aLIGO CDS in progress, with bulk of work being done by 40m lab staff.
MIT LASTI	Upgrade requested from eLIGO to aLIGO hardware/software to support aLIGO application development and cavity locking.

AEI Hannover	aLIGO PSL test system needs upgrade; desirable to do this soon, as they are taking final sets of data and testing prior to first delivery in early 2011. However, this would require a loan of an aLIGO I/O chassis. aLIGO CDS software also running at GEO (squeezing experiment and output mode cleaner controls) and soon at the AEI 10m test facility.
Stanford	SEI test facility – needs update to CDS V2.0 software.
LLO	DAQ Test system. SUS and SEI stand-alone test systems (1 SUS install remaining)
LHO	DAQ Test system. SUS and SEI stand-alone test systems (1 SUS install remaining)
Japan	Two development systems – no present support
Australia	Development system at ANU – no present support

13 Documentation

By the end of FY11, it is intended that the software documentation listed below be completed. In addition, a list of software test procedures to be written can be found in the CDS Software Test Plan LIGO-T1000561.

LIGO DOC Number	Description	Assigned to	Status
T0900603	Real-time Control Software Requirements	Bork	Waiting Review
T0900607	Real-time Sequencer Software Design	Bork	Needs update
T0900638	Real-time Data Acquisition Software Design	Bork	Internal Review
T1000587	Real-time Communication Software Design	Bork	10 Oct 2010
T1000561	CDS Software Test Plan	Bork	In Progress
T080315	RCG Application Developer's Guide	Bork / Aronsson	Needs update to Version 2.0
T080316	CDS Software SysAdmin Guide	Barker / Thorne	Requires major updates to V2.0
T0900531	CDS SVN Repositories	Barker	
TBD	Overview of CDS Timing and Data Flow	Bork	

TBD	DAQ Data Concentrator and Framebuilder Software Design	Thorne / Ivanov	
T1000379	CDS Environment Configuration Scripts	Thorne	In progress
T1000248	aLIGO CDS File System Directories	Thorne	In progress
T0900636	aLIGO CDS Frame Builder and Network Data Server	Ivanov	