LIGO-T070284-00-Z

Recommendations Concerning LIGO Control Room Software Tools for Enhanced and Advanced LIGO

Duncan Brown[1], Shourov Chatterji[2,3], Justin Garofoli[4], Sergey Klimenko[5], Szabi Marka[6], Keith Riles[7], Vern Sandberg[4], Peter Shawhan[8], Daniel Sigg[4], Josh Smith[1], Patrick Sutton[9], Alan Weinstein[2] and John Zweizig[2]

[1]Syracuse University
[2]California Institute of Technology
[3]Istituto Nazionale di Fisica Nucleare
[4]LIGO Hanford Observatory
[5]University of Florida
[6]Columbia University
[7]University of Michigan
[8]University of Maryland
[9]Cardiff University

January 22, 2008

1 Executive Summary

This *ad hoc* committee of LIGO Lab / LSC scientists recommends that, in preparation for Enhanced and Advanced LIGO, Lab/LSC support for control room tool software be increased substantially. This support should be aimed to improve existing tools (including data access), to develop new tools, and to make those tools portable and widely accessible to the LSC. We urge a sharing of responsibility between the Lab and the rest of the LSC for these improvements. Specifically, we recommend that

1. The Lab hire a full-time programmer to assist John Zweizig in supporting, expanding, and distributing the Global Diagnostics System (GDS) software tree, including familiar tools like the data viewer, Diagnostic Test Tool (DTT), and Data Monitoring Tool (DMT) viewer;

2. The LSC (or Lab) appoint a scientist / programmer to develop and support new Matlab-based exploratory tools for the control room, leveraging a variety of "home-brew" tools already developed at the Observatories for commissioning and tools developed for offline data analysis.

3. The Lab and LSC jointly support a data access server at several archive locations that mimics the current network data server at the Observatories, but provides easy, transparent and secure access to archived data.

4. The LSC implement inspiral and burst search pipelines with latencies well under a minute, to permit issuing automated alerts to the astronomical community. Computational load demands a cluster-based solution for inspiral searches, but at least some burst searches could be carried out in the online DMT environment, permitting very low latencies.

2 Introduction

In response to a request from LSC Spokesperson David Reitze and LIGO Lab Deputy Director Albert Lazzarini, an *ad hoc* committee, broadly representative of LSC scientists who work in the Observatory control rooms (including Lab scientists), was formed to provide LSC guidance on control room software tools needed in the Advanced LIGO era. Our committee began by circulating comments among each other by e-mail on deficiencies in present tools and on wishes for the future, before meeting by teleconference for two hours on November 2, 2007. After further, iterative discussion by e-mail following the teleconference, we drafted preliminary recommendations in late November, the broadness of which demanded we solicit comments from not only the detector characerization community, but also the Data Analysis Software Working Group and the Inspiral and Burst Working Groups. We received many constructive suggestions from those groups and have modified the preliminary recommendations quite significantly to obtain the final recommendations presented here. While we do not claim to speak for those groups, our recommendations are strongly informed by their opinions.

Although our committee's original charge was to provide recommendations to Vern Sandberg and Rolf Bork on which tools should be enhanced or newly developed for the CDS & GDS "supervisory controls", it became apparent in our discussions that our recommendations needed to address broader issues that go beyond the CDS/GDS arena. Specifically, we concluded that LIGO Lab's Computing group would need to contribute significantly to implementing some recommendations, and that involving non-Lab LSC institutions in software development and maintenance would make these proposed changes more feasible by distributing the workload. In the following we suggest divisions of responsibility, attempting to be realistic about available resources and timescales. Hence we also provide prioritizations with respect to implementations need on the time scales of pre-Enhanced LIGO (commissioning), Enhanced LIGO, and Advanced LIGO.

There is strong agreement among committee members that some infrastructure is better suited to Lab support than to most non-Lab LSC institutions, because of greater personnel continuity, especially when looking ahead to the Advanced LIGO era. There is less unanimity on responsibility for implementation of new algorithms. The recommendations provided here aim in part at making it easier for LSC scientists to develop, implement and maintain responsibility for new diagnostic tools. An alternative minority view on the committee advocates a NASA-style model in which all implemented software is written and supported by the Lab. In this model the Lab collaborates with creative LSC scientists who develop new algorithms in non-production environments. This model would require the Lab to hire further full-time programmers and would also likely require close LSC supervision to ensure timely deployments of new tools. Setting aside whether resources would exist for such a model, there is some concern on the committee about the the wisdom of separating development from deployment, despite the increased ease of development.

Based on past experience, there is excellent reason to believe that non-Lab LSC institutions can contribute quite substantially to establishing and maintaining software and computer infrastructure relevant to the recommendations discussed below.

The recommendations here are focused primarily on software related to the LIGO control rooms and to LIGO data analysis. But it is important to remember that our solutions, especially those related to data access, transfer, and analysis, should be coordinated with GEO and Virgo plans. A recurring theme in the following is the need to establish clean interfaces and protocols among LIGO programs. The same principle holds, of course, with respect to GEO and Virgo.

3 Detailed Recommendations

We provide here a set of 11 detailed recommendations. For each recommendation we suggest a division of responsibility, the time scale on which the recommendation should be implemented, along with background information.

1. **Portability of existing tools** – Steps need to be taken to improve dramatically the portability of the primary control room GUI tools: data viewer, DMT viewer, and Diagnostic Test Tool. (We did not address EPICS controls.)
   Responsibility: Primary = Lab, secondary = LSC
   Time scale: Enhanced LIGO

   Background: The primary set of GUI tools now used by LSC scientists in the control room are the data viewer, DMT viewer, and Diagnostic Test Tool (DTT). These programs were all developed originally for Solaris, and in principle, are linux-compatible. In practice, LSC institutions have had great trouble installing them at home, and many have not bothered to try. None of these programs is available for Windows or Macintosh. Establishing a robust linux package containing all three of these tools, one that is downloadable and immediately usable by LSC scientists, should be a top priority. We note that moving toward the use of linux workstations in the Observatory control rooms would greatly facilitate compatibility of control room and offline analysis tools.

We also believe that Windows and Macintosh installations would prove valuable to the LSC, and we encourage their development, but we recognize the likely difficulty in porting some tools to Windows. We suggest an exploratory effort, but do not assign it a high priority, given the ongoing development of new Windows-compatible Matlab tools discussed below. We also note that the ligo_viewer program (TCL/TK-based) already provides some of the functionality of the data viewer and runs under unix, linux and Windows. It is a good example of a "home-brew" program developed by the user community to get around Solaris-centric software.

We urge that a full-time programmer be hired to assist John Zweizig in establishing and supporting this portability, including standard releases of packages for reference platforms. Assistance from the LSC at large is likely to be helpful; the Data Analysis Software Working Group (DASWG) has demonstrated well the effectiveness of collaborative effort on portability.

2. **Improvement of existing tools** – The primary GUI tools listed above need significant improvement in light of S5 and commissioning experience. In addition, improved support is needed for carrying out blind injections.
   Responsibility: Primary = Lab, secondary = LSC
   Time scale: Advanced LIGO

   Background: Complete wish lists for improvement of these tools would be quite lengthy, but below are lists of issues we consider most important, excluding data access issues discussed in item 3) further below.

   - Data viewer:
     - Simple arithmetic on single channels (e.g., offset/mean subtraction)
     - Combining together of channels (linear & non-linear)
     - Filtering (as in the DTT Foton package)
     - Fourier analysis in real-time
     - Real-time time/frequency map displays (comparable in speed to the commercial linux-based baudline program) - It may

make more sense to make this a separate stand-alone program - see below.
  - Graphical output to files could be streamlined
- DMT viewer:
  - Need better robustness of communication between the DMT viewer and the monitor engines running in background
  - Need ability to look back in time to previous conditions (e.g., using DMT minute-trend equivalents to real-time objects)
  - The web version of the viewer is relatively primitive and should be upgraded to mimic more features of the control-room version (good candidate project for an LSC institution to take on)
  - Graphical output to file could be streamlined
- Diagnostic Test Tool:
  - Need more mathematical manipulation of spectra
  - Faster startup would be desirable
  - Graphical output to file could be streamlined
- Blind injections
  A more reliable scheme for blind injections is desirable, with the true excitation channel information preserved but encrypted, to avoid reliance on the present honor system.

3. **Improvement of data access** – Data access throughout the collaboration needs to be made more transparent and uniform. The tools used for data display should ideally be the same, whether one sits in a control room, logs into an LSC cluster, or sits in one's office. This is an area where close coordination and cooperation between the CDS and Computing groups will be needed. The existing Network Data Server (NDS) provides a working model at the Observatories, which we believe could be adapted to this purpose at all LSC data archives. We urge that its support be greatly increased. Particular care should be taken to define interfaces between clients and servers cleanly, to facilitate client development on multiple platforms and in different code environments.
   Responsibility: Primary = Lab, secondary = LSC
   Time scale: Enhanced LIGO

Background: Right now data access & display capabilities vary greatly throughout the LSC and even within a single observatory site. If one wants to look at real-time data or very recently collected data (still in disk buffer), the control room tools discussed above are most convenient – if one is physically sitting in the control room. With the portability improvements discussed above, it might no longer matter much where one is sitting, but it would still matter whether one wanted very recent data or old data. Eliminating (or at least mitigating) the obstacles in accessing and graphing data & data products from the past would improve productivity dramatically. One would no longer have to switch to more cumbersome tools when looking at old data.

We would like to see a unified interface for data retrieval and graphing, where the physical location of the data is irrelevant to the user, at least for the hundreds of channels in the level-1 RDS. If disk storage is cheap enough in the Advanced LIGO era to permit disk storage of all raw data in the Caltech archive, then the RDS condition could be relaxed too. In short, when one starts the data viewer or DTT in the control room (or in one's office, or on a Windows laptop at the airport), one should be able to see DARM_ERR from a moment ago, from one hour ago, or from three months ago, with the same ease. Providing online brief documentation (1-2 sentences per channel) would also be desirable.

A related issue is "publication" of data via metadata. Online cluster-based searches based on reading frame files would be made more reliable if metadata about these files were recorded when the files are written to disk, as opposed to being "discovered" via the appearance of the files. Close coordination among the CDS and Computing Groups, as well as DASWG, is important here.

At the same time, we suggest that real-time searches reduce dependence upon publishing. The DMT approach of examing all data as it comes, and using state vector channels to determine science mode, seems better suited to robust real-time analysis. Building a "relay" of the present frame broadcaster, used for online DMT programs, to extend to the observatory cluster merits serious consideration.

4. **Improvement and development of Matlab data analysis tools** – New Matlab-based tools need to be developed to facilitate commissioning and exploratory analysis, with an emphasis on flexibility and user-

control, using present "home brew" tools like mDv as a start. These tools should be developed on an expedited basis to assist in pre-S6 commissioning. Coordination with the existing MatApps software project under DASWG is highly desirable.
Responsibility: Primary = LSC, secondary = Lab
Time scale: Pre-Enhanced LIGO **- Urgent**

Background: While the GUI tools discussed above generally do well what they do, and their standardized interfaces have been well thought out for easy use, they do not allow users to "get under the hood". One is limited in the manipulations one can carry out on data products, and typically, if post-processing is needed, the user must export the data to file and read it into a program like Matlab. This cumbersomeness has led to the development of home-brew programs at the observatories and the 40 Meter. These include the Matlab data access client written by Ben Johnson and the data manipulation package known as mDv written by Justin Garofoli, which allow users to request data snippets by start time and duration, load those channels into Matlab memory, and then carry out manipulations like time-frequency mapping, with scripts. Those scripts can be modified at will by the user, making it easy to carry out quite sophisticated operations, thanks to the power of the Matlab toolboxes, including system identification and adaptive filtering.

Such tools have tended not to be robust, however, against changes in Matlab versions, and the authors have had little time to support them, given other, higher-priority duties. At the same time, there is a substantial user base in the LSC data analysis community for Matlab tools, with a software librarian devoted to Matlab applications (MatApps). There seems to be an opportunity to bring together common interests of commissioners and data analysts to strengthen and unify the collaboration's Matlab infrastructure. This is an area where LSC institutions could well prove critical in supplementing LIGO Lab staff.

We also note that the GEO Collaboration's Matlab-based data viewer has proven powerful and flexible (and already includes the location-independent data access recommended above for LIGO GUI tools). Martin Hewitson has adapted this data viewer to the LIGO NDS, using Ben Johnson's client. The resulting program (called ligoDV) can read and display LIGO data from the observatory caches and is under active development by Josh Smith.

It is possible that some Matlab tools could eventually develop well enough to supplant the data viewer and many DTT functions (excluding excitations). We think that there should be no obstacles put in the way of such development. At the same time we recognize that making a robust, versatile tool like the DTT from scratch is no small task; it is a long road from a GUI written to do one task well to a general-purpose utility with the DTT's power. Users will vote their feet; if the old tools wither away because new tools evolve to become superior, that's a good thing.

One specific area where LIGO/LSC-developed tools have proven unsatisfactorily slow is real-time display of spectrograms. The more satisfactory linux-based commercial program baudline suggests one avenue of development if in-house or Matlab-based solutions can't be made to serve our needs.

5. **Implementation of rapid inspiral & burst searches** – In the Advanced LIGO era, we will want aggressively low latency for detecting astrophysical signals. We believe the online DMT system offers a natural platform already for rapid burst searches and should be seriously considered for that purpose. At the same time, cluster-based solutions will be necessary for the computationally demanding inspiral searches (and perhaps for some burst searches). We recommend that the LSC explore using the online DMT system for the lowest latency searches and avoid reliance upon only a cluster-based approach. Pushing the most rapid searches onto the DMT would allow the latency requirements on cluster-based solutions to be relaxed, simplifying their development for Advanced LIGO. In addition, we recommend that well defined cluster computing resources be dedicated to rapid astrophysical searches.
   Responsibility: LSC
   Time scale: Advanced LIGO

   Background: In the Advanced LIGO era we expect to see gravitational waves frequently enough that it's plausible LIGO will start reporting high-SNR candidates automatically via the internet to the astronomical community. The Robotic Optical Transient Survey Experiment (ROTSE) now provides a response time of seven seconds to GRB alerts received from NASA satellites, which themselves have a latency of 20 seconds. If we wish in 2015 to match the state of the art in GRB alerts in 2008, we will need to issue alerts within half a minute of receiving a gravitational burst.

Speed of analysis will be of the essence. The online DMT system offers such speed now and could serve as a platform for burst searches in that era. The latency of cluster-based analyses will probably be much reduced from the current minutes (occasionally hours / days when malfunctions occur), but it will still be hard for cluster-based analyses to compete with the DMT on speed and robustness. We think it makes sense to ask the burst group to look seriously at implementing one or more pipelines in the online DMT system. We note that the existing BurstMon DMT monitor already supports most of the infrastructure of an incoherent burst search and that the offline KleineWelle program is also written for the DMT environment. We also note that a C++ DMT program can be used as a "wrapper" to call existing LAL/LALApps C software, making it unnecessary to rewrite some current pipelines from scratch.

The online DMT's present low latency in examining single-interferometer data is not sufficient alone, however. In order for us to provide a pointing accuracy usable by ROTSE or similar telescope networks, data will need to be brought together rapidly from multiple interferometers, as addressed in recommendation 6.

While we think that flagship burst searches could be run in an online DMT system like that of S5, the inspiral searches need much greater computing resources. A cluster-based solution will be necessary for those searches. A successor to the present ONASYS software infrastructure will be developed under DASWG to improve robustness and latency. We believe that both improvements would be aided greatly by the dedication of computing resources to the rapid searches, to avoid unpredictable interference from lower-priority offline analyses. We also emphasize the importance of ensuring that results from those cluster-based searches be piped back to the control rooms quickly.

One could try to implement a high-SNR inspiral search in the online DMT system, using coarse template spacing and deemphasizing lower frequencies, in order to achieve very low latency. But we understand from the Inspiral Group that with expected Advanced LIGO sensitivity curves, much of a candidate's SNR will be accumulated more than a minute before the merger event, anyway. Such a scenario would not only relax requirements on analysis latency, it would offer the exciting prospect of *anticipating* a merger event. One could imagine alerting the control room to an impending chirp and thereby ensure no controls settings are changed (as we do now for GRB alerts, in order to collect clean background data samples). Looking farther ahead, one could even envision responding to such an alert by tuning interferometer spectral sensitivity to maximize final SNR, based on preliminary estimates of signal parameters. Hence we expect that when Advanced LIGO nears its design sensitivity at low frequencies, a cluster-based solution will suffice for inspiral searches.

6. **Merging of h(t) in realtime at the Observatories** – To facilitate rapid coincidence analysis, h(t) from each observatory (including from Virgo and GEO) should be brought together in each control room with very low latency O(10 seconds). This requirement will demand robust bandwidth and development of new infrastucture.
   Responsibility: Lab
   Time scale: Advanced LIGO

   Background: The most serious obstacle to carrying out rapid online DMT coincidence searches at the moment is that the online DMT systems at each observatory see only data from that observatory. At present, carrying out coincidence analysis, particularly coherent multi-IFO analysis, can be done easily only on the clusters and with substantial latency, because of the way in which archival h(t) is produced. Whether a coincidence search is carried out with online DMT or cluster infrastructure, a more rapid production and distribution of h(t) is needed.

   Fortunately, even now the bandwidth between the sites is high enough that streaming h(t) across the network in real-time is feasible, and in the Advanced LIGO era it should be trivial. What is lacking is the infrastructure to receive those data streams and merge them together in a synchronized way with a latency of seconds so that real-time analysis programs can make use of them. In principle, the task is straightforward, but this is another area where close coordination between the

CDS and Computing groups is appropriate.

We note that tools exist now in the DMT infrastructure to produce h(t), offering one means for low-latency production that could serve real-time analyses, but not necessarily produce archival h(t).

Similarly, piping additional diagnostic figures of merit from each observatory to the others (including Virgo and GEO) would be useful and should be encouraged.

7. **Improvement of online science segment definition** – Better online science segment defining infrastructure is needed to eliminate present "edge artifacts" at ends of lock stretches.
   Responsibility: Lab and LSC
   Time scale: Enhanced LIGO

   Background: Because of latencies in the EPICS channels used presently to keep track of interferometer state and therefore to define starts and stops of science mode segments, the segment lists generated online typically are inaccurate at the 1-2 second level on the end times of science mode, requiring later offline patching in the form of data quality flags. This is a seemingly minor but nonetheless quite annoying feature of the present infrastructure, which has grown somewhat organically. Some prior thought and modest preparation should eliminate this problem before S6 begins.

8. **Improved data quality availability** – Issuing astronomical alerts in near real-time will require confidence in the quality of the data. The latency of S5 data quality information (weeks to months for most DQ flags) will be completely inadequate for Advanced LIGO analysis. Three distinct improvements will be needed: 1) migration of offline DQ flagging to the online DMT system; 2) streamlining of the insertion and extraction of DQ flags using the segment database; and 3) rapid translation of operator / scimon concerns to the DQ database, including the ability to disable astronomical alerts when data quality is deemed unreliable.
   Responsibility: LSC
   Time scale: Enhanced LIGO

   Background: Although a handful of S5 DQ flags are produced by online DMT monitors, most flags are based on offline analyses by detector characterization investigators. A large number of these flags, however, can be defined via automated algorithms (although setting appropriate thresholds may require human judgement). Their creation should be migrated to the online DMT system on the Enhanced LIGO timescale via the upgrade of existing monitors or the creation of new ones.

Making use of the data quality flags in real-time analysis requires that the database accept insertions rapidly and respond to queries rapidly. The present segment database, as currently configured, does not provide sufficient speed. It's possible that a reconfiguration of the existing database or a new set of clients can produce acceptable response time.

In addition to flags with definitions amenable to automation, there will likely always be a need for flags defined by operators and scimons based on judgement. The translation of such data quality information entered by operators and scimons in the elog into DQ flags is presently quite tedious and prone to error, as a different person typically does the translation. An automated but flexible program is needed, to allow automatic entry of information directly into the database. Because of the variety of persons entering that data (and the lateness of the hour when some entries will be made), some simple error correction capability will also be essential. Again, this need should be addressed before S6 and may be well suited to the efforts of an LSC institution.

9. **Improved archiving of figures of merit** – Although most DMT monitors running in background 24/7 performed very well during S5, there were sporadic problems in archiving their figures of merit. The present infrastructure for producing permanent summaries needs to be overhauled.
Responsibility: Lab
Time scale: Enhanced LIGO

Background: The DMT monitors running in background during S5 performed for the most part very well during the run, with data dropouts and machine reboots quite rare. A number of monitors were upgraded during the run, and new monitors were introduced. More problematic were the display and archiving of figures of merit. The display problems are discussed above w.r.t. the DMT viewer, but there were also annoying and repeated glitches in the archiving of daily summary plots. Those glitches arose from the cobbled-together system used and should be eliminated with a robust, dedicated system for S6. That system should include not only graphs, but also data files to permit accurate localization of features seen in the graphs. One possibility, already used by GEO, is archiving of Matlab figure files via cron scripts.

10. **Improved support for DMT monitor development** – Existing barriers to the creation and maintenance of DMT monitors need to be lowered.
    Responsibility: Primary = Lab, secondary = LSC
    Time scale: Enhanced LIGO

    <u>Background:</u> Writing and maintaining DMT programs is more difficult than it needs to be. Problems include: 1) non-uniformity of DMT installations on different observatory computers; 2) difficulty in installing DMT software at LSC institutions (discussed above), and 3) the automake structure of the GDS software tree. Even experienced DMT authors find that upgrading and testing a monitor after a new DMT release can be a bit painful. Streamlining the DMT development process would make it more attractive to new scientists to develop DMT programs and would encourage existing DMT authors to make useful upgrades.

11. **Increased programming support for all of the above** – A person or persons should be identified to serve the role of software librarian(s) for control room diagnostic tools. Ideally, this librarian role would be filled by one or more senior physicists available to spend substantial effort on ensuring uniformity of tool distributions, on incorporation of new tools or tool enhancements, and on resolution of platform incompatibilities. The support of one or more full-time programmers, ideally programmers with physics backgrounds, will be essential to make such a program realistic.

    More specifically, we recommend that substantial programmer resources be provided by the Lab to support John Zweizig in establishing and maintaining full portability of the C and C++ control room tools that include the data viewer, DTT, DMT viewer, and the DMT monitor infrastructure itself.

    We also recommend that the LSC (and Lab) increase physicist and programmer support for development of Matlab tools for commissioning and for use in the control room during science runs. There are already "home-brew" Matlab tools for data access and exploration, along with a variety of Matlab tools for offline data analysis. We hope these tools will provide a foundation to make a robust suite of distributed tools useful in the Observatory control rooms, R&D facilities, and at LSC institutions.

We believe that dedicated professional programming support will be needed to establish platform portability of existing tools, especially for Windows.
Responsibility: Lab and LSC
Time scale: Pre-Enhanced LIGO **- urgent**

Background: The creation or augmentation of the tools described above will require a major effort from many persons in the Lab and in certain LSC institutions. But in addition, there remain the issues of maintaining a uniform environment at Lab and LSC institutions and supporting several different platforms. We think it would be highly desirable to find persons (probably two) to take responsibility for enforcing that portability and uniformity, persons who would test new code, oversee releases, and prevent changes that cause standard tools to break on any supported platform. (Ideally, one person would oversee all control room tool releases, but it's probably more realistic to split the job between two persons, one to oversee C/C++ tools, the other to oversee Matlab tools.) An experienced senior physicist would be preferable to a professional programmer. Nonetheless, the additional support of professional programmers is likely to be helpful, especially if some tools are ported to Windows platforms.

4 Final Comments

While some of the above recommendations could be implemented by the CDS group in collaboration with certain LSC institutions, there are other recommendations concerning data transfer / access, and portability of tools, where we believe close cooperation will be be needed between the Lab's CDS and Computing groups. The practical necessity to establish divisions in responsibilities between these groups has led over the years to somewhat arbitrary and sometimes counterproductive divisions in software functionality. These divisions have at times impeded detector characterization and data analysis more generally. We urge that Lab management help find ways to make these natural divisions of responsibility transparent to the LSC user community.

Finally, while it is desirable to involve LSC institutions in developing and maintaining software infrastructure, it is also important to ensure stable long-term support. Code development by graduate students or postdocs can cause headaches when those persons leave the collaboration or move to new positions within the collaboration. The choice of which jobs to "farm out" to the LSC should take into account the real-world constraints of university grant support and personnel turnover. That said, there are very encouraging examples where LSC institutions have taken on and successfully fulfilled major responsibilities in software and computer infrastructure. In any case, ample documentation of deployed programs, interfaces, and protocols should be a strict requirement on contributing groups.