The LDAS Database Mock Data Challenge

Peter Shawhan
LIGO / Caltech

NSF Review
April 30–May 2, 2001

LIGO-G010188-00-E

Goals of the Database MDC

This MDC covered the core database software (IBM DB2), LDAS database interface (Metadata API), file formatting (Lightweight API), and job control (Manager API)

Goals:

- Validate the database table design
- Rigorously test the functionality of the database system
- Evaluate the performance of the database system
- Test the process of inserting triggers and other data generated by the Data Monitoring Tool (DMT) system into the database
- Foster the creation of user interface tools
- Exercise various database administration tasks
- Ensure that accurate documentation exists

Database MDC Logistics

Planning document was written in August 2000

MDC was expected to be complete by end of 2000

However, manpower limitations led to change in strategy: tasks were addressed gradually over a longer period of time

MDC finally completed in April 2001

Contributors included: Peter Shawhan, Philip Charlton, Phil Ehrens, Mary Lei, Maria Barnes, Ed Maros, Greg Mendell, John Zweizig, Roy Williams

Database Table Design

No major changes to schema since November 1999

Significant expansion and tuning of indexes

Many minor changes in response to user requests

Overall design is mature, but there are certain to be many more changes (new or modified columns; additional tables) as system is put into more active use

Database System Functionality

A set of test scripts were created for automated testing

Scripts use 97 files of test data, and submit 217 LDAS jobs

General emphasis on “exhaustive” testing, including error handling

Major categories:

- Communication with LDAS managerAPI
- Data insertion
- Database queries
- LIGO_LW formatting rules

Various problems were reported and resolved

Web-based documentation for LDAS user commands was improved

Database System Performance

Measured insertion rates are easily adequate

- Event candidates: ~ few hundred per second
- Spectra: 300-400 kb/sec

More quantitative tests will be done now that final RAID storage has been installed

Query speed depends on complexity of query

Appropriate indexes make a big difference

Data Insertion by Data Monitoring Tool

DMT “Trigger Manager” collects environmental triggers, etc., from several monitor processes, and sends them to LDAS for insertion

System tested during engineering runs

Demonstrated successful long-term operation, and the ability to handle a high instantaneous trigger rate

User Interfaces

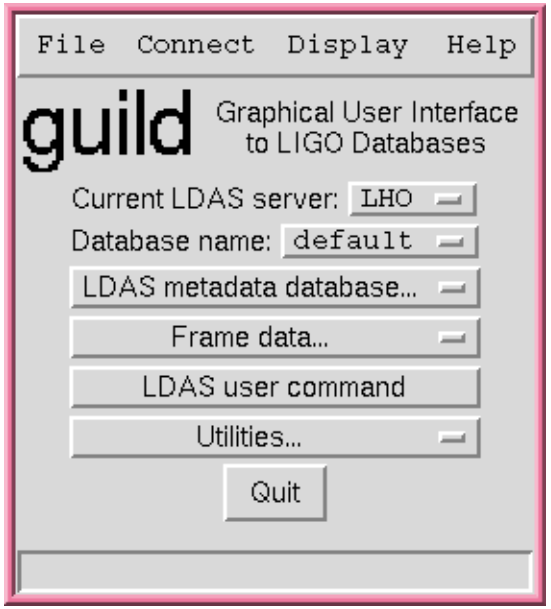
Development of user interfaces was a major goal of the MDC

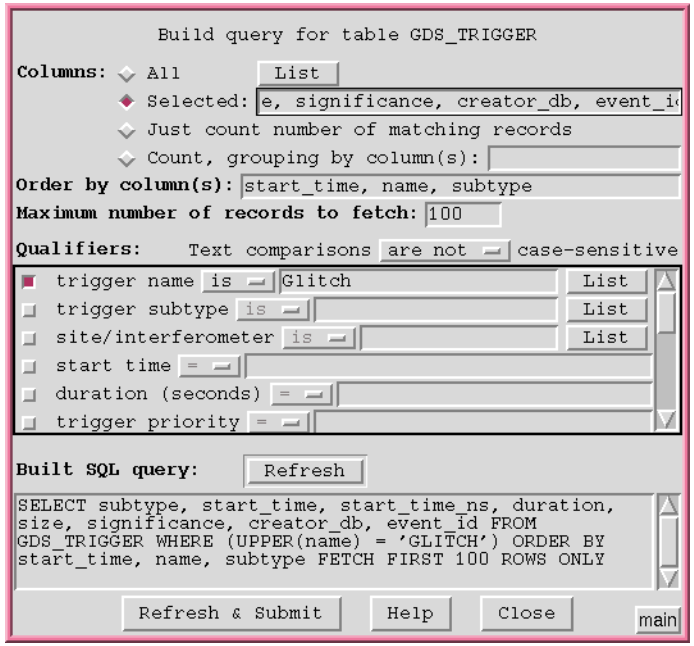
Now available:

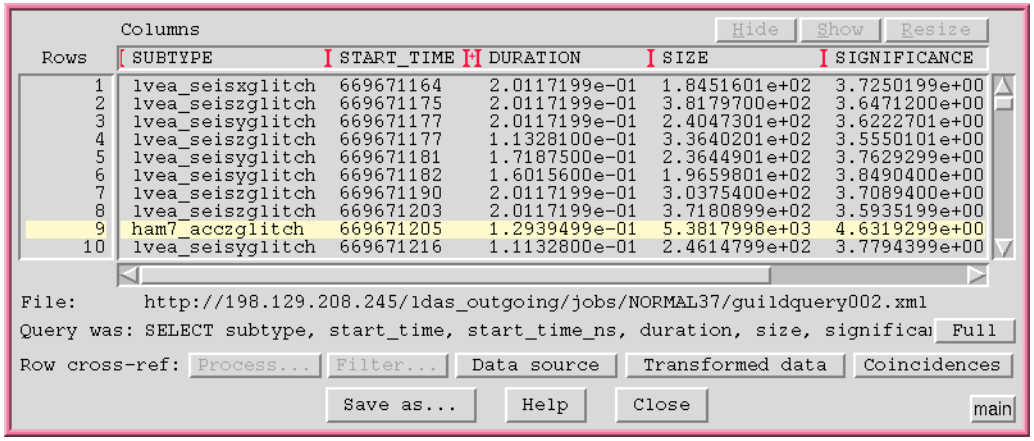
- Graphical user interface: guild
- Command-line utilities: putMeta, getMeta
- C library (“metaio”) to parse LIGO_LW files
- Command-line utilities for LIGO_LW files: lwtscan, lwtprint, lwtdiff
- MATLAB MEX-file to read LIGO_LW file contents into MATLAB arrays

These tools (and others) are being distributed as part of the “LIGOtools” software suite

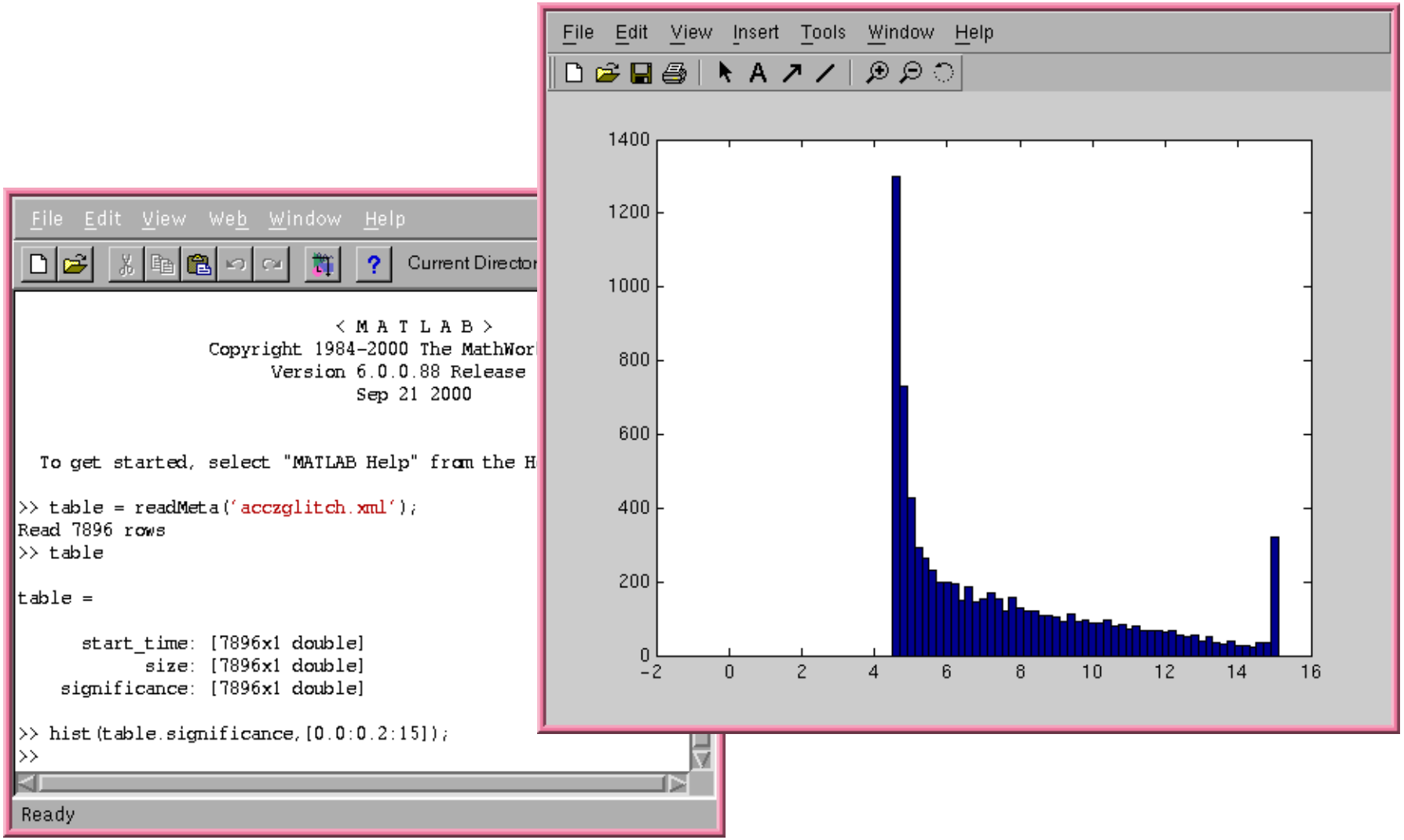
guild


File Connect Display Help
guild
Graphical User Interface to LIGO Databases
Current LDAS server: LHO
Database name: default
LDAS metadata database...
Frame data...
LDAS user command
Utilities...
Quit


Build query for table GDS_TRIGGER
Columns: All List
Selected: e, significance, creator_db, event_i
Just count number of matching records
Count, grouping by column(s):
Order by column(s): start_time, name, subtype
Maximum number of records to fetch: 100
Qualifiers: Text comparisons are not case-sensitive
trigger name is Glitch List
trigger subtype is List
site/interferometer is List
start time =
duration (seconds) =
trigger priority =
Built SQL query: Refresh
SELECT subtype, start_time, start_time_ns, duration, size, significance, creator_db, event_id FROM GDS_TRIGGER WHERE (UPPER(name) = 'GLITCH') ORDER BY start_time, name, subtype FETCH FIRST 100 ROWS ONLY
Refresh & Submit
Help
Close
main


Columns
Hide Show Resize
Rows SUBTYPE START_TIME DURATION SIZE SIGNIFICANCE
1 lvea_seisxglitch 669671164 2.0117199e-01 1.8451601e+02 3.7250199e+00
2 lvea_seiszglitch 669671175 2.0117199e-01 3.8179700e+02 3.6471200e+00
3 lvea_seisyglitch 669671177 2.0117199e-01 2.4047301e+02 3.6222701e+00
4 lvea_seiszglitch 669671177 1.1328100e-01 3.3640201e+02 3.5550101e+00
5 lvea_seisyglitch 669671181 1.7187500e-01 2.3644901e+02 3.7629299e+00
6 lvea_seisyglitch 669671182 1.6015600e-01 1.9659801e+02 3.8490400e+00
7 lvea_seiszglitch 669671190 2.0117199e-01 3.0375400e+02 3.7089400e+00
8 lvea_seiszglitch 669671203 2.0117199e-01 3.7180899e+02 3.5935199e+00
9 ham7_acczglitch 669671205 1.2939499e-01 5.3817998e+03 4.6319299e+00
10 lvea_seisyglitch 669671216 1.1132800e-01 2.4614799e+02 3.7794399e+00
File: http://198.129.208.245/ldas_outgoing/jobs/NORMAL37/guildquery002.xml
Query was: SELECT subtype, start_time, start_time_ns, duration, size, significa
Full
Row cross-ref: Process... Filter... Data source Transformed data Coincidences
Save as...
Help
Close
main

MATLAB Interface



Database Administration

Have exercised many tasks by now, such as:

- Creating, clearing, and deleting tables
- Creating indexes and triggers
- Adding a table to an existing database
- Adding a column to an existing table
- Deleting selected database entries (keyed by a unique ID assigned to each process which writes to the database)
- Backing up and restoring the entire database

Notes on procedures are being collected and will be assembled into a set of web pages for future reference

Summary

Database MDC took a long time, but was ultimately successful

Functionality was rigorously tested, and tests can be repeated at will

Performance seems fine

DMT - LDAS communication works well

Database table design is mature, but there will be more modifications as tables are put into active use

Several user interface tools are now available

Database system is ready for more widespread use!