



How To Get Started With DATABASE I/O



Gregory Mendell
LIGO Hanford Observatory



DATABASE Basics

- We are using IBM DB2 relational databases.
- All I/O is currently done via LDAS.
- DMT Monitors and DSOs have built-in ways to submit data to LDAS databases. Otherwise:
 - To output data one runs `getMetaData`.
 - To input data one runs `putMetaData`.
- Need an LDAS password:

<https://www.ldas-wa.ligo.caltech.edu/passwordRequest/ldaspassword.cgi>



LIGOTOOLS and LDAS Jobs

- LIGOTOOLS provides front ends for doing I/O and analysis with the databases.
- Go here to install ligotools:

<http://www.ldas-sw.ligo.caltech.edu/ligotools/>

- Install tclshexe, ldasjobs, dataflow, root, eventool, and ilwdread packages.

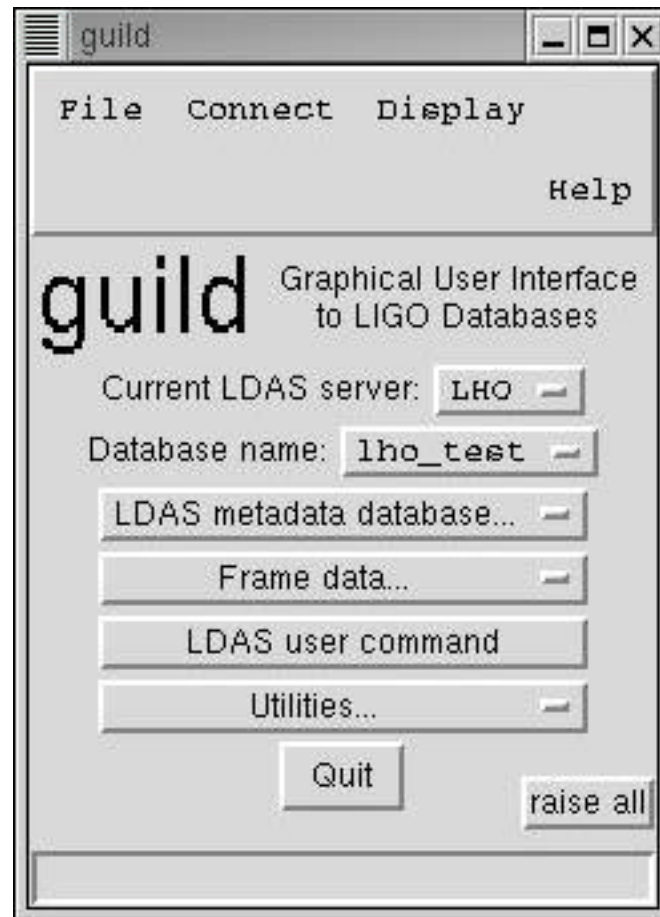


LIGOTOOLS Packages

- Start with GUILD!
- The dataflow package provides getMeta and putMeta command line tools for database I/O.
- You will be dealing with data in xml files.
- I have not used the eventool or ilwdread but they look useful.
- You can write your own tcl scripts that use the Idasjobs package.



GUILD





GUILD Build Query

Build query for table **SNGL_DPERIODIC**

Columns: All Selected:

Just count number of matching records

Count, grouping by column(s):

Order by column(s):

Maximum number of records to fetch: (LDAS max: 10000)

Data epoch:

Qualifiers: Text comparisons case-sensitive

<input type="checkbox"/>	LDAS job ID	<input type="button" value="is"/> <input type="text"/>	<input type="button" value="List"/>
<input type="checkbox"/>	LDAS job user tag	<input type="button" value="is"/> <input type="text"/>	<input type="button" value="List"/>
<input type="checkbox"/>	interferometer	<input type="button" value="is"/> <input type="text"/>	<input type="button" value="List"/>
<input type="checkbox"/>	target name	<input type="button" value="is"/> <input type="text"/>	<input type="button" value="List"/>
<input type="checkbox"/>	right ascension (degrees)	<input type="button" value="="/> <input type="text"/>	
<input type="checkbox"/>	declination (degrees)	<input type="button" value="="/> <input type="text"/>	
<input type="checkbox"/>	start time	<input type="button" value="="/> <input type="text"/>	
<input type="checkbox"/>	end time	<input type="button" value="="/> <input type="text"/>	
<input type="checkbox"/>	duration (seconds)	<input type="button" value="="/> <input type="text"/>	

Built SQL query:

```
SELECT * FROM SNGL_DPERIODIC ORDER BY start_time,
start_time_ns, target_name, end_time FETCH FIRST 100 ROWS ONLY
```



GUILD Query Result

Entries in table SNGL_DPERIODIC (LHO, ldas_tst)

Column selected: TARGET_NAME

Rows	IFO	START_TIME	END_TIME	TARGET	SKY_RA	SKY_D	FREQU	STAT_NAME	STAT	STAT_PROB
1	H1	729976096	729984288	J1939	5.1471+	3.766+	1.283+	Norm JKS P+	9.63+	9.5741624e+
2	H1	729976096	729984288	J1939	5.1471+	3.766+	1.283+	Norm JKS P+	9.63+	9.5741624e+
3	H1	729976096	729984288	J1939	5.1471+	3.766+	1.283+	Norm JKS P+	9.63+	9.5741624e+

File: http://198.129.208.245/ldas_outgoing/jobs/LDAS-WA_89/LDAS-WA890251/guildquery004.xml

Query was: SELECT * FROM SNGL_DPERIODIC ORDER BY start_time, start_time_ns, target_name, end_time FETCH

Row cross-ref: [Process...](#) [Filter...](#) [Extra info...](#)

[Save as...](#) [Help](#) [Close](#) [main](#)



getMetaData

```
set cmd "{
  getMetaData
    -returnprotocol http://out.xml
    -outputformat LIGO_LW
    -database $DATABASE
    -sqlquery $SQLQUERY
}"
```

```
set LJcmd "LJrun thisJob -manager $SITE $cmd"
eval $LJcmd
LJcopy $thisJob(outputs) $FILENAME
```




putMetaData

```
set cmd "{
  putMetaData
    -ingestdata %FILE($FILENAME)
    -database $DATABASE
}"
set LJcmd "LJrun thisJob -manager $SITE $cmd"
eval $LJcmd
```



SNGL_DPERIODIC TABLE

COLNAME, TYPE, LENGTH, CODEPAGE, DEFAULT, NULLS ALLOWED:

```
"CREATOR_DB", "INTEGER", 4, 0, "1", "N", , ,
"PROCESS_ID", "CHARACTER", 13, 0, , "N", , ,
"FILTER_ID", "CHARACTER", 13, 0, , "Y", , ,
"IFO", "CHARACTER", 2, 819, , "N", , ,
"SEARCH", "VARCHAR", 24, 819, , "N", , ,
"CHANNEL", "VARCHAR", 64, 819, , "Y", , ,
"START_TIME", "INTEGER", 4, 0, , "N", , ,
"START_TIME_NS", "INTEGER", 4, 0, , "N", , ,
"END_TIME", "INTEGER", 4, 0, , "N", , ,
"END_TIME_NS", "INTEGER", 4, 0, , "N", , ,
"DURATION", "REAL", 4, 0, , "N", , ,
"TARGET_NAME", "CHARACTER", 32, 819, , "Y", , ,
"SKY_RA", "DOUBLE", 8, 0, , "N", , ,
"SKY_DEC", "DOUBLE", 8, 0, , "N", , ,
"FREQUENCY", "DOUBLE", 8, 0, , "N", , ,
"AMPLITUDE", "REAL", 4, 0, , "N", , ,
"PHASE", "REAL", 4, 0, , "N", , ,
"SNR", "REAL", 4, 0, , "Y", , ,
"CONFIDENCE", "REAL", 4, 0, , "Y", , ,
"STAT_NAME", "VARCHAR", 32, 819, , "Y", , ,
"STAT_VALUE", "REAL", 4, 0, , "Y", , ,
"STAT_PROB", "REAL", 4, 0, , "Y", , ,
"EVENT_ID", "CHARACTER", 13, 0, , "N", , ,
```

LIGO-G030296-00-W



SNGL_DPERIODIC XML File

```
...
<LIGO_LW Name="ligo:ldas:file">
<Table Name="ldas:SNGL_DPERIODIC:table">
  <Column Name="CREATOR_DB" Type="int_4s"/>
  <Column Name="PROCESS_ID" Type="char"/>
  <Column Name="IFO" Type="lstring"/>
  <Column Name="SEARCH" Type="lstring"/>
  <Column Name="CHANNEL" Type="lstring"/>
  <Column Name="START_TIME" Type="int_4s"/>
  <Column Name="START_TIME_NS" Type="int_4s"/>
  <Column Name="END_TIME" Type="int_4s"/>
  <Column Name="END_TIME_NS" Type="int_4s"/>
  <Column Name="DURATION" Type="real_4"/>
  <Column Name="TARGET_NAME" Type="lstring"/>
  <Column Name="SKY_RA" Type="real_8"/>
  <Column Name="SKY_DEC" Type="real_8"/>
  <Column Name="FREQUENCY" Type="real_8"/>
  <Column Name="AMPLITUDE" Type="real_4"/>
  <Column Name="PHASE" Type="real_4"/>
  <Column Name="SNR" Type="real_4"/>
  <Column Name="CONFIDENCE" Type="real_4"/>
  <Column Name="STAT_NAME" Type="lstring"/>
  <Column Name="STAT_VALUE" Type="real_4"/>
  <Column Name="STAT_PROB" Type="real_4"/>
  <Column Name="EVENT_ID" Type="char"/>
  <Stream Name="ldas:SNGL_DPERIODIC:table" Type="Local" Delimiter=",">
    1, "\000\000\000\000\000\000\000\000\000\000\000\000\000\000\000", "H2", "pulsar", "AS_Q", 729298800, 0, 729299040, 0, 240, "Test
Pulsar", 5.1471621479999996, 3.7669606900000002e-01, 1283.8499999999999, 1.5857173, 0.0, .80112284, 3.6735412e-01, "Norm JKS F
Statistic", 2.5144992, 3.6735412e-01, "\000\000\000\000\000\000\000\000\000\000\000\000\000\000\000"
  </Stream>
</Table>
</LIGO_LW>
```

Need to submit an entry for the process table as well, with unique PROCESS_ID.

Need to generate an EVENT_ID.