

LASER INTERFEROMETER GRAVITATIONAL WAVE OBSERVATORY
- LIGO -
CALIFORNIA INSTITUTE OF TECHNOLOGY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Document Type LIGO-T010024-00-Z	15-22 January 2001
MPI Mock Data Challenge	
Warren Anderson, Maria Barnes, Kent Blackburn, Patrick Brady, Duncan Brown, Jolien Creighton, Teviet Creighton, Phil Ehrens, Albert Lazzarini, Isaac Salzman and Alan Wiseman	

Distribution of this draft:
MPI working group

California Institute of Technology
LIGO Project - MS 51-33
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/>

Contents

1	Test Definition	4
1.1	Overview	4
1.2	Hardware and Software Components to be tested	4
1.3	Software components to be tested	4
1.4	Required input data	4
1.5	Output data to be generated	6
1.6	Required Hardware	6
1.7	Required Software	6
1.8	Required Personnel	6
2	Test Results	7
3	Criteria for evaluation	7
3.1	Software	7
3.2	Hardware	7
4	Execution Plan	7
4.1	Environment	7
4.2	Build test	8
4.3	Documentation test	8
4.4	API level tests	8
4.5	LALWrapperInterface tests	8
4.6	Inspiral search code tests	8
4.7	Excess-power search code tests	9
4.8	Pipeline tests	9
4.9	Long-term operation	9
5	Conclusions and recommendations	9
5.1	Overview	9
5.2	Failures and open bugs	10
5.3	Recommendations arising from the MDC	10
A	Workplans	11
B	Test checklists	13
B.1	00prep	13
B.1.1	MDCPREP	14
B.1.2	LDASBUILD	15
B.1.3	LALBUILD	16
B.1.4	LALWRAPPERBUILD	17
B.2	01doc	18
B.2.1	MPIAPIDOC	19
B.2.2	WRAPPERAPIDOC	20
B.2.3	LALDOC	21
B.2.4	LALWRAPPERDOC	22
B.3	02wrapper	23
B.3.1	WRAPPER01	24

B.4	03mpi	41
	B.4.1 MPIAPI01	42
B.5	04lalwrapper	45
	B.5.1 LALWRAPPER01	46
B.6	05mpiWrapper	55
	B.6.1 MPIWRAPPER01	56
B.7	06inspiral	58
	B.7.1 Command Line Arguments	58
	B.7.2 Test Check Program	58
	B.7.3 INSPIRAL01	60
	B.7.4 INSPIRAL02	63
	B.7.5 INSPIRAL03	70
	B.7.6 INSPIRAL04	73
	B.7.7 INSPIRAL05	75
	B.7.8 INSPIRAL06	79
	B.7.9 INSPIRAL07	81
B.8	07power	84
	B.8.1 Command Line Arguments	84
	B.8.2 POWER01	85
	B.8.3 POWER02	92
	B.8.4 POWER03	95
B.9	08ldasInspiral	98
B.10	09ldasPower	99
B.11	10longTermInspiral	100
	B.11.1 LTINSPIRAL01	101
	B.11.2 LTINSPIRAL02	104
B.12	11longTermPower	107
	B.12.1 LTPOWER01	108
	B.12.2 LTPOWER02	111
B.13	12simultaneous	113
B.14	13longTerm	114
	B.14.1 MPILONGTERM01	115

1 Test Definition

1.1 Overview

The purpose of this MDC is to verify the functionality of the MPI and Wrapper APIs, focusing on

- Correctness of implementation;
- Integration of programming elements (C++, Tcl, LAL and LALwrapper);
- Integration of system elements (MPI, Wrapper, Assistant Manager, and Manager APIs);
- Ability to handle parallel, concurrent analyses;
- Through-put and capacity.

1.2 Hardware and Software Components to be tested

The LDAS prototype Beowulf at Caltech will be used for this MDC. The MDC will exercise the hardware, but it is not intended to provide benchmark information suitable for future planning. Any timing or throughput information should be used for guidance only.

1.3 Software components to be tested

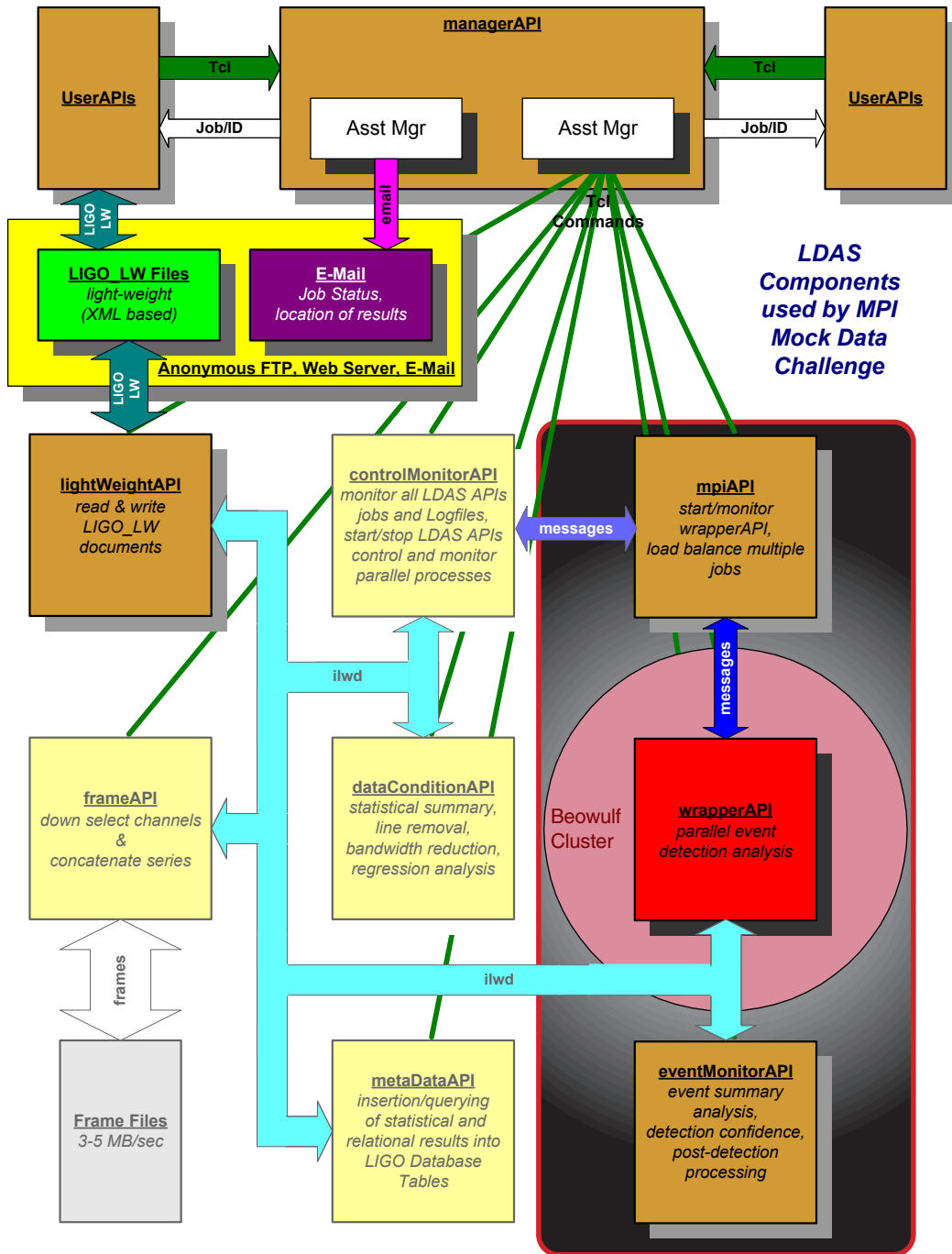
- Correctness of implementation: A subset of the functionality described in the MPI API baseline (LIGO T990086) and the Wrapper API baseline (LIGO T990097);
- Integration of programming elements: the functionality described above and the Tcl management of the MPI API;
- Integration of system elements: the ability to drive the MPI API from user commands submitted to the ManagerAPI;
- Ability to handle parallel, concurrent analyses: the ability of the MPI API to successfully dispatch and manage multiple instantiations of the wrapper API. This will be evaluated using an inspiral and burst search code;
- Thought-put and capacity: ability of the API's to function in unison without overloading the system.

1.4 Required input data

The MDC will require a variety of input data to verify the various software components. Simulated Level 3 data (i.e., whitened GW strain data together with whitening filter coefficients and calibration data, including noise power spectral density) will be used for the long term tests. In all cases, the data will be expected in ILWD format suitable for ingestion by the wrapperAPI. The data will be read in from files which will be placed under CVS with all documentation, user commands and results. The details of the noise character and the simulated signals will be chosen for the purpose of verifying and diagnosing the MPI and wrapper API functionality.

The following data is provided in the ldasmc CVS repository discussed above

Figure 1: LDAS components involved in the advanced analysis processes based on MPI and executing on the LDAS distributed computing parallel cluster of nodes. The functionality to be tested in this MDC is indicated inside the red box, i.e. wrapperAPI and mpiAPI. A stub for the event monitor API will be used during the MDC. The integration of these system elements, the ability to handle parallel concurrent analyses, and the throughput and capacity of the system will be tested. Other components of LDAS which will be exercised during the MDC are indicated; those components not used during the MDC, but which will form part of the full system are indicated by lighter colors than the others.



- API Tests:
`bad_boolean.ilwd`, `bad_both_interval.ilwd`, `bad_domain.ilwd`,
`bad_freq_interval.ilwd`, `bad_time_interval.ilwd`, `data.tar.gz`,
`empty.ilwd`, `input.ilwd`, `no_data.ilwd`, `no_domain.ilwd`,
`no_sequence.ilwd`, `no_startfreq.ilwd`, `no_starttime.ilwd`,
`no_stepsize.ilwd`, `no_stopfreq.ilwd`, `no_stoptime.ilwd`
- Search Tests:
`c_1.40_1.40_11.00.ilwd`, `c_1.85_1.51_13.79.ilwd`, `cn_1.40_1.40_20.00.ilwd`,
`cn_1.51_1.10_17.43.ilwd`, `cn_1.51_1.10_17.43_bank.ilwd`,
`cn_1.85_1.51_13.79.ilwd`, `cn_3.79_3.33_4.61.ilwd`, `cn_3.79_3.33_4.61_bank.ilwd`,
`cn_4.16_0.93_5.79.ilwd`, `cn_4.16_0.93_5.79_bank.ilwd`,
`n_1048576_1024Hz.ilwd`, `n_128_1024Hz.ilwd`, `n_128_8192Hz.ilwd`,
`n_64_1024Hz.ilwd`, `n_64_2048Hz.ilwd`, `n_64_7369Hz.ilwd`, `n_64_8192Hz.ilwd`,
`trivial_huge.ilwd`, `trivial_long.ilwd`, `trivial_long_bank.ilwd`,
`trivial_medium.ilwd`, `trivial_short.ilwd`, `n1026slr1024.ilwd`,
`power_trivial.ilwd`
- Long-term operations test:
`cn_chunk3.ilwd.gz`, `chirp_power.ilwd.gz`, `cn_chunk3.ilwd.gz`, `im-`
`pulse_chunk.ilwd.gz`, `impulse_power.ilwd.gz`, `masslist.ilwd.gz`,
`n_chunk2.ilwd.gz`, `noise_power.ilwd.gz`

1.5 Output data to be generated

Event data to populate the LDAS data base tables will be communicated to the Event Monitor API through a data socket from the wrapperAPI. This output data will then be written to a file. Other diagnostic information will be extracted from the LDAS logs; `stderr` and `stdout` will be captured and logged for later analysis where necessary.

1.6 Required Hardware

LDAS prototype Beowulf at Caltech. The system used during the MDC will be a 16 node beowulf system with a single master node which can communicate with the other machines outside the Beowulf's private network.

1.7 Required Software

Required third party software packages include: FFTW 2.1.3, LAM 6.3.2 (with LDAS patches), FrameL v4.22. The LDAS, LAL and LALwrapper software systems for both Sun Solaris and Intel Linux environments.

1.8 Required Personnel

The execution of the MDC will require the presence of the MPI working group, the LDAS development team and support personnel to provide system administration when software or hardware components are updated.

2 Test Results

Each test that forms a part of this MDC is fully described in a test checklist (see appendix B). Each step in any particular test is described on the appropriate checklist. The successful or unsuccessful completion of each step is recorded on the checklist together with any responses. If a given test is unsuccessful, a recommendation to proceed with subsequent tests without resolving the failure (and providing a fix) may be made if the failure is not critical. All such failures will be reported in the problem tracking system should they go unaddressed. The checklists will be fully completed and form part of the official record of this MDC.

3 Criteria for evaluation

3.1 Software

- Correctness of results: output data generated by search codes should be correctly parsed by the wrapperAPI. Differences between the output received by the Event Monitor API and the expected results are considered a failure.
- Error handling: All errors should be correctly handled by each software layer. From low to high level errors, the hierarchy for detecting and handling is mpiAPI, wrapperAPI, LALwrapperInterface, and LAL search codes.
- Throughput: 24h continuous operation at full throughput with no evidence of memory leaks.
- Capacity: be able to provide job control sufficient to handle two concurrent searches of different character.

3.2 Hardware

The performance of the mater node on the prototype Beowulf will be monitored to insure that it is not overloaded with communication and management tasks for multiple searches.

4 Execution Plan

Prior to beginning this MDC assemble all the required components (cf. ??). The versions of LDAS, LAL and LALwrapper being tested are tagged in the repository with PRE02MDCYYMMDD, where YY is the current year, MM the two-digit current month and DD the current day. These versions of LDAS, LAL and LALwrapper is referred to here as the test-version. The tag should be interpreted as indicating start of the second MDC on MM/DD/YY.

All supplementary materials including this document, test checklists, the associated scripts containing LDAS user commands, input data, result data and output data where appropriate will be organised as described in be placed in the ldasmdc CVS repository at `gravity.phys.uwm.edu:/usr/local/cvs/ldasmdc`.

4.1 Environment

This MDC will be executed on the LDAS Beowulf clusters at Caltech. It assumes an ldas user account, which is referred to here and henceforth as `ldas_mdc`. On the LDAS system file server the following directory hierarchy under the LDAS anonymous FTP area (`/ldas_outgoing/jobs`) will be created. It will hold the contents of the ldasmdc CVS repository currently stored at `gravity.phys.uwm.edu:/usr/local/cvs/ldasmdc`.

This will create the shared directory `/ldas_outgoing/jobs/ldasmdc` which will contain the input data, the test scripts built out of LDAS user commands, the documentation explaining the usage for each of the tests executed during this MDC. See worksheet MDCPREP in Sec. B.1

4.2 Build test

The wrapper and mpi API and their tests should build successfully within LDAS with no errors or warnings: Worksheets LDASBUILD in Sec. B.1. LAL, LALwrapper and their tests should build successfully with no errors or warnings: Worksheets LALBUILD and LALWRAPPERBUILD in Sec. B.1.

4.3 Documentation test

Verify that each components documentation is complete: Worksheets MPIAPIDOC, WRAPPERAPIDOC, LALDOC and LALWRAPPERDOC in Sec. B.2.

4.4 API level tests

- Verify the correctness of implementation and error handling in the wrapperAPI: Worksheet WRAPPER01 in Sec. B.3.
- Verify the correctness of implementation and error handling in the mpiAPI: Worksheet MPIAPI01 in Sec. B.4.
- Verify correct interaction between the mpiAPI and the wrapperAPI: Worksheet MPIWRAPPER01 in Sec. B.6.

4.5 LALWrapperInterface tests

Verify the correctness of implementation and error handling in the LALWrapperInterface: Worksheet LALWRAPPER01 in Sec. B.5.

4.6 Inspiral search code tests

- Verify that LDAS can run a binary inspiral search using the inspiral shared object and trivial input data. Successively longer trivial input files should be filtered by the shared object against one template: Worksheet INSPIRAL01 in Sec. B.7.
- Verify that the inspiral shared object correctly rejects improperly formed filter parameters, the failure is correctly logged by LDAS and the job cleanly shutdown: Worksheet INSPIRAL02 in Sec. B.7.
- Verify that LDAS can run a binary inspiral search using the inspiral shared object and trivial input data with multiple templates: Worksheet INSPIRAL03 in Sec. B.7.
- Verify that LDAS can run a binary inspiral search using the inspiral shared object and find known inspiral signals at the correct effective distance using a single template: Worksheet INSPIRAL04 in Sec. B.7.
- Verify that LDAS can run a binary inspiral search using the inspiral shared object and white gaussian noise input data: Worksheet INSPIRAL05 in Sec. B.7.

- Verify that LDAS can run a binary inspiral search using the inspiral shared object and find known inspiral signals in gaussian noise at the correct effective distance using a single template: Worksheet INSPIRAL06 in Sec. B.7.
- Verify that LDAS can run a binary inspiral search using the inspiral shared object and find known inspiral signals in gaussian noise at the correct effective distance using a bank of templates: Worksheet INSPIRAL07 in Sec. B.7.

4.7 Excess-power search code tests

- Verify that the power shared object correctly rejects improperly formed filter parameters, the failure is correctly logged by LDAS and the job cleanly shutdown: Worksheet POWER01 in Sec. B.8.
- Verify that LDAS can run excess power search using the power shared object and trivial input data with various filter parameters: Worksheet POWER02 in Sec. B.8.
- Verify that LDAS can run excess power search using the power shared object and trivial input data with various filter parameters: Worksheet POWER03 in Sec. B.8.

4.8 Pipeline tests

- Verify that the inspiral DSO under LDAS can filter multi-segment chunks of data of various lengths, using Gaussian noise as input: Worksheet LTINSPIRAL01 in Sec. B.11.
- Verify that the inspiral DSO under LDAS can filter multi-segment chunks containing various inspiral chirps embedded in Gaussian noise: Worksheet LTINSPIRAL02 in Sec. B.11.
- Verify that libpower.so can analyze multi-segment chunks of data of various lengths, using Gaussian noise as input: Worksheet LTPOWER01 in Sec. B.12.
- Verify that libpower.so can analyze multi-segment chunks of data on different numbers of nodes, using Gaussian noise plus impulses as input: Worksheet LTPOWER02 in Sec. B.12.

4.9 Long-term operation

The long-term operations test will consist of the two wrapped clients identified in section 4.8 running “unattended” for 24h: Worksheet MPILONGTERM01 in Sec. B.14.

5 Conclusions and recommendations

5.1 Overview

The MPI MDC took place during the week 15-19 January 2001. It was a success. The implementation of the wrapperAPI, the mpiAPI and two shared objects were verified. In broad terms, the conclusions of the MDC can be summarised as follows:

1. The LDAS team and LAL developers can work together to develop and implement gravitational-wave search codes to run in the LDAS parallel analysis environment.
2. The wrapperAPI provides both the requisite functionality and flexibility needed to make LDAS a powerful data analysis engine.

Moreover, the long-term run described in test checklist LONGTERM01 demonstrated that concurrent analyses (of different types) can be executed using LDAS. Several failure modes (described below) were identified during this run which continued from 03:45 on 19 January 2001 until the morning of 22 January 2001. Nevertheless, this test demonstrated that the system should be able to run real-time analyses as soon as the LDAS pipeline is complete.

5.2 Failures and open bugs

Each test executed during the MDC was fully described in a test checklist in appendix B; failures were resolved as time permitted. At the end of the week, several failures remained outstanding. They are summarized in Table 5.2 in which their current status is also outlined.

Table 1: A list of failures encountered during the MDC. A **Conditional Pass** was assigned to a test if a bug was encountered that could not be fixed due to time constraints and would not interfere with the continued execution of the MDC.

Test	Result	Synopsis of problem	Status
LALDOC	Fail	Documentation for <code>findchirp</code> and <code>burstsearch</code> is not up to date or included in <code>lsd.pdf</code>	Open (lal#6,#7)
LALWRAPPERDOC	Fail	Documentation for <code>load</code> , <code>inspiral</code> and <code>power</code> shared objects is not available	Open (lal#2,#3)
WRAPPER01	Conditional Pass	Test 5(a) wrapperAPI segfaults if input ILWD file contains an empty container.	Closed (ldas621)
LALWRAPPER01	Conditional Pass	Tests 8 and 10 failed; these tests are considered to be “too sick” so a conditional pass was assessed to be reconsidered later.	Open
INSPIRAL02	Conditional Pass	Shared object raises SIGSEGV when the template list is incorrectly specified in the command line arguments.	Open
INSPIRAL07	Conditional Pass	Test 01 reports $\chi^2 = 0$ for the event. All other values are correct.	Open
LONGTERM01	Conditional Pass	Four modes of failure identified. See Sec. B.14.	Closed (ldas647)

5.3 Recommendations arising from the MDC

Several issues remain to be addressed after the MDC. First, we list some recommendations arising directly from the experience with LDAS:

1. Search user accounts should be hard mounted on all nodes. Latency in auto-mounting seemed to cause intermittent problems when starting jobs using the mpiAPI.

2. Job dependent logging of `stderr` and `stdout` would be extremely useful for debuggin purposes. This should be handled by the `mpiAPI` if at all possible. An alternative method would be adding another command line argument to the `wrapperAPI` to open a file and send this information to that file. The filename would be supplied by the `mpiAPI` via this command line argument.
3. At the time of the MDC, job monitoring was not as easy as one would like. The control and monitor API should be made a high priority for development. As of writing, Kent Blackburn indicated that this was now in hand and a useful implementation is available in LDAS 0.0.14 which was released on Friday 23 February 2001.

Along with these general recommendations for the LDAS system, the following recommendations arose for the software components that were tested during the MDC.

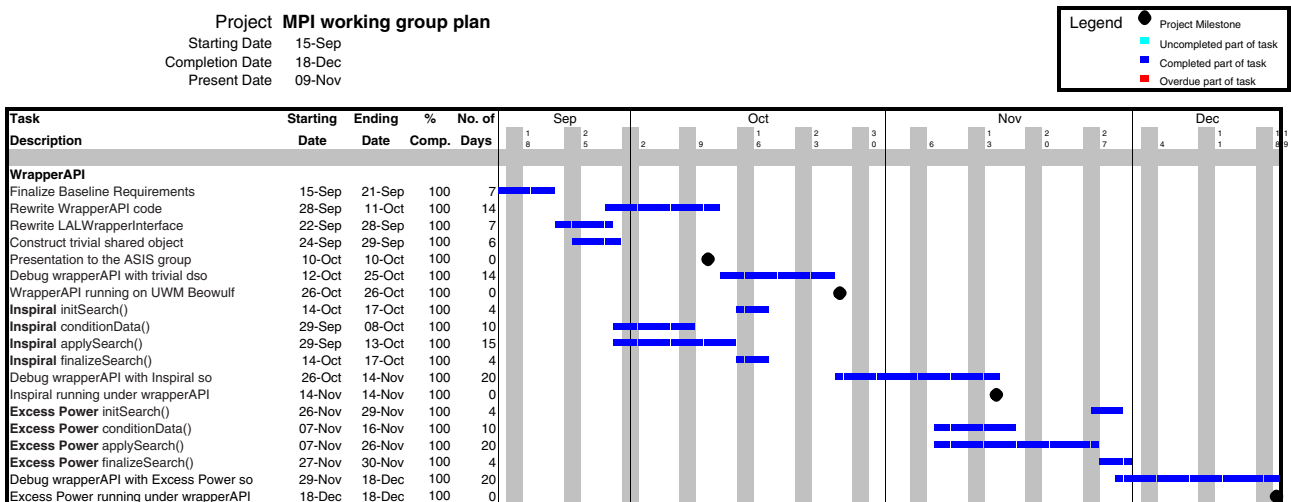
Error messages: The length of error messages returned by the `wrapperAPI` should be changed to 16 kb. (This is implemented and available in LDAS 0.0.14)

Documentation: The LAL and LALwrapper documentation should be brought up to date with the software that is available within these packages. The *LAL-LDAS Interface Coding Specification* (LIGO-T010003-00-E) document should be completed.

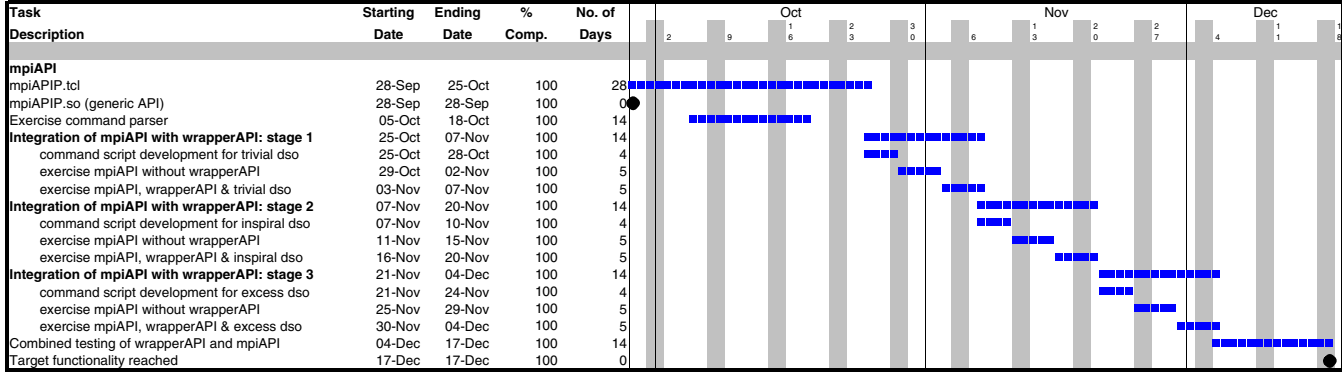
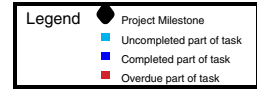
Development HOWTO: A development HOWTO document should be written to include instructions for building the `wrapperAPI` executable in standalone mode, and instructions for adding the `HelloWorld.so` to the LALwrapper package.

Load balancing: A shared object should be developed which supports the full functionality of load-balancing as described in the `wrapperAPI` baseline specifications. The load-balancing functionality should be tested in the `wrapperAPI` and `mpiAPI`.

A Workplans



Project **MPI working group plan**
 Starting Date 28-Sep
 Completion Date 17-Dec
 Present Date 09-Jan



B Test checklists

The following pages contain detailed procedures to be followed for testing the functionality of the wrapper-API and mpiAPI.

B.1 00prep

In preparation for the MDC, an official release version of LDAS shall be built and installed on the LDAS development, test, Hanford, and Livingston systems without errors. In addition, an official release version of each of lal and lalwrapper shall be built, checked, installed and mirrored onto the same LDAS systems

B.1.1 MDCPREP

Test Case: MDCPREP
Purpose: Prepare the environment for the MPI API MDC.
Tester: [Kent Blackburn and Duncan Brown](#)
Test machine: [ldas-dev, beowulf](#)
Date (mm/dd/yy): [01/15/01](#) **Time:** [14:00](#)

ENVIRONMENT AND PREREQUISITES

Third party software needed for the MDC by both LAL and LDAS shall be pre-installed and available to all LDAS computers and developer workstations through the LDAS /ldcg filesystem. This include the availability of the input data to be used by MDC tests in a subdirectory accessible to all LDAS computers and available through anonymous FTP and the web to the LIGO and LSC workstations. The required third party software consists of FFTW version 2.1.3, LAM version 6.3.2 (with LDAS specified patches), and FrameL i version 4.22. The root directory for the location of input data shall be /ldas_outgoing/jobs which shall have the subdirectories listed in the PROCEDURE section below.

PROCEDURE

1. On the LDAS system file server create the following directory hierarchy under the LDAS anonymous FTP area (/ldas_outgoing/jobs), which will hold the contents of the ldasmdc CVS repository currently stored at [gravity.phys.uwm.edu:/usr/local/cvs/ldasmdc](#). This will create the shared directory /ldas_outgoing/jobs/ldasmdc which will contain the input data, the test scripts built out of LDAS user commands, the documentation explaining the usage for each of the tests executed during this MDC, and the results of running these test scripts:

```
ldasmdc repository checked out
```

Pass

2. Verify correct installation of required software packages

- (a) FFTW 2.1.3

Pass

- (b) LAM 6.3.2 (with LDAS patches)

Pass

- (c) FrameL v4.22

Pass**SUMMARY**

Known faults encountered – list bug IDs: None

New faults submitted – list bug IDs: None

TEST RESULT**Pass**

B.1.2 LDASBUILD**Test Case:** LDASBUILD**Purpose:** Verify that the LDAS software system properly built for both Sun Solaris and Intel Linux environments; that the LDAS software system has been properly installed on the LDAS development fileserver; that the LDAS software system has been properly pushed (rsync) onto all LDAS servers and each node of the Beowulf Clusters. Also verify that all LDAS APIs (managerAPI, mpiAPI, wrapperAPI, control-MonitorAPI, prototype-eventMonitorAPI, etc.) are running and properly configured on all LDAS systems.**Tester:** Kent Blackburn and Duncan Brown**Test machine:** ldas-sw, ldas-dev, beowulf, ldas-test**Date (mm/dd/yy):** 01/15/01 **Time:** 14:00**ENVIRONMENT AND PREREQUISITES**

This test is performed from the user accounts install and ldas on the LDAS development, test, Hanford and Livingston system.

PROCEDURE

1. Inspect the log-files from the release build of the LDAS S/W system and insure that LDAS, including all APIs, documentation pages, and automated make checks completed without errors. builds without errors. Correct the LDAS source if necessary until a successful build results.

Pass

2. Inspect the LDAS API web based log-files for each LDAS system to verify that each API is running and properly reporting status to the system by following the webpage links at <http://www.ldas-sw.ligo.caltech.edu> to the API Status page and the individual API log-files for each LDAS system.

Pass

3. Mark the current, successfully built version of the LDAS S/W using the CVS tag PRE02MDCYYMMDD, where YY is the two digit year, MM the two digit month and DD the two digit day of the beginning of this MDC.

Pass**SUMMARY****Known faults encountered – list bug IDs:** None**New faults submitted – list bug IDs:** None**TEST RESULT****Pass**

B.1.3 LALBUILD

Test Case: LALBUILD
Purpose: Verify LAL build, check and installation
Tester: [Kent Blackburn and Duncan Brown](#)
Test machine: m71, beowulf
Date (mm/dd/yy): 01/15/01 **Time:** 14:00

ENVIRONMENT AND PREREQUISITES

This test is executed from the LDAS account install on the LDAS development server and verified to have been pushed to all LDAS system servers.

PROCEDURE

1. Follow the installation instructions in LAL-0.6. Verify that the build completes without errors. Verify that the LDAS development file server has properly presented (exported or mirrored) the LAL installation to all computers mounting /ldcg, including the LDAS system servers and LDAS development workstations: `make`, `make check`, `make dvi`. Make corrections as necessary. `make install`
Pass
2. Mark the current, successfully built version of LAL using the CVS tag `PRE02MDCYYMMDD`, where YY is the two digit year, MM the two digit month and DD the two digit day of the beginning of this MDC.
Pass

SUMMARY

Known faults encountered – list bug IDs: None

New faults submitted – list bug IDs: None

TEST RESULT

Pass

B.1.4 LALWRAPPERBUILD

Test Case: LALWRAPPERBUILD
Purpose: Verify LALwrapper build, check and installation
Tester: Kent Blackburn and Duncan Brown
Test machine: m71, beowulf
Date (mm/dd/yy): 01/15/01 **Time:** 14:00

ENVIRONMENT AND PREREQUISITES

This test is executed from the LDAS account install on the LDAS development server and verified to have been pushed to all LDAS system servers.

PROCEDURE

1. Get lalwrapper from CVS archive at `gravity.phys.uwm.edu:/usr/local/cvs/lalwrapper`. In the lalwrapper directory, type `./00boot`. Follow the installation instructions in LALwrapper-1.0. Verify that the build completes without errors. Verify that the LDAS development file server has properly presented (exported or mirrored) the LALwrapper installation to all computers mounting /ldcg, including the LDAS system servers and LDAS development workstations: `make`, `make check`. Correct as necessary. `make install`
2. Mark the current, successfully built version of LAL using the CVS tag `PRE02MDCYYMMDD`, where YY is the two digit year, MM the two digit month and DD the two digit day of the beginning of this MDC.

Pass**Pass****SUMMARY**

Known faults encountered – list bug IDs: None

New faults submitted – list bug IDs: The following was need to correctly configure the package:
`./configure --prefix=/ldcg --enable-frame --enable-mpi --with-extra-cppflags="-I/ldcg/include" --with-extra-ldflags="-L/ldcg/lib -Wl, -rpath=/ldcg/lib"`

TEST RESULT**Pass**

B.2 01doc

The following tests are designed to verify the existence and correctness of documentation for mpiAPI, wrapperAPI, LAL and lalwrapper.

B.2.1 MPIAPIDOC

Test Case: MPIAPIDOC
Purpose: Verify the existence of the mpiAPI web-based documentation.
Tester: [Philip Ehrens](#)
Test machine: marfik.ligo.caltech.edu
Date (mm/dd/yy): 01/15/01 **Time:** 17:30

ENVIRONMENT AND PREREQUISITES

This test is executed from any LIGO user account with internet access and a web browser. It depends on the completion of MDCPREP and LDASBUILD.

PROCEDURE

1. Using a web-browser verify that the MPI API documentation is built and complete for LDAS release version 0.0.13. The MPI API documentation must be access from the top level LDAS home page on the web server: www.ldas-dev.ligo.caltech.edu

Pass

SUMMARY

Known faults encountered – list bug IDs: None

New faults submitted – list bug IDs: None

TEST RESULT

Pass

B.2.2 WRAPPERAPIDOC

Test Case: WRAPPERAPIDOC
Purpose: Verify the existence of the wrapperAPI web-based documentation.
Tester: [Maria Barnes](#)
Test machine: ldas-dev.ligo.caltech.edu
Date (mm/dd/yy): [01/15/01](#) **Time:** 17:30

ENVIRONMENT AND PREREQUISITES

This test is executed from any LIGO user account with internet access and a web browser. It depends on the completion of MDCPREP and LDASBUILD.

PROCEDURE

1. Using a web-browser verify that the Wrapper API documentation is built and complete for LDAS release version 0.0.13. The Wrapper API documentation must be access from the top level LDAS home page on the web server: www.ldas-dev.ligo.caltech.edu

Conditional Pass

SUMMARY

Known faults encountered – list bug IDs: Bugs #550 and #601: problem with percepts correctly building documentation

New faults submitted – list bug IDs: None

TEST RESULT

Conditional Pass

B.2.3 LALDOC

Test Case: LALDOC
Purpose: Verify the existence of the LAL documentation.
Tester: Teviet Creighton
Test machine: ldas-pcdev1.ligo.caltech.edu
Date (mm/dd/yy): 01/15/01 **Time:** 18:20

ENVIRONMENT AND PREREQUISITES

This test is executed from any LIGO user account from a LDAS workstation which mounts the /ldcg filesystem. It depends on the completion of MDCPREP and LALBUILD.

PROCEDURE

1. Confirm the existence of the LAL documentation file /ldcg/doc/lal-0.6/lal.pdf

Pass

2. Using a portable document format (PDF) viewer verify that the LAL documentation is built and complete for package findchirp

Fail

Notes: Error codes should conform to the LAL standard. Documentation incomplete, should be completed.

3. Using a portable document format (PDF) viewer verify that the LAL documentation is built and complete for package burstsearch

Fail

Notes: Documentation is not present, should be documented.

SUMMARY

Known faults encountered – list bug IDs: Documentation for findchirp and burstsearch is not up to date or included in lsd.pdf

New faults submitted – list bug IDs:

TEST RESULT**Fail**

B.2.4 LALWRAPPERDOC

Test Case: LALWRAPPERDOC
Purpose: Verify the existence of the LALwrapper documentation.
Tester: [Teviet Creighton](#)
Test machine: [ldas-pcdev1.ligo.caltech.edu](#)
Date (mm/dd/yy): [01/15/01](#) **Time:** 18:20

ENVIRONMENT AND PREREQUISITES

This test is executed from any LIGO user account from a LDAS workstation which mounts the /ldcg filesystem. It depends on the completion of MDCPREP and LALWRAPPERBUILD.

PROCEDURE

1. Confirm the existence of the LALwrapper documentation file /ldcg/doc/lalwrapper-1.0/lalwrapper.pdf
Pass
2. Using a portable document format (PDF) viewer verify that the LAL documentation is built and complete for shared object `trivial`
Pass
3. Using a portable document format (PDF) viewer verify that the LAL documentation is built and complete for shared object `sick`
Pass
4. Using a portable document format (PDF) viewer verify that the LAL documentation is built and complete for shared object `load`
Fail
Notes: **Documentation incomplete, should be completed**
5. Using a portable document format (PDF) viewer verify that the LAL documentation is built and complete for shared object `inspiral`
Fail
Notes: **Documentation incomplete, should be completed**
6. Using a portable document format (PDF) viewer verify that the LAL documentation is built and complete for shared object `power`
Fail
Notes: **Documentation not present, should be documented**

SUMMARY

Known faults encountered – list bug IDs: Documentation for load, inspiral and power shared objects is not available

New faults submitted – list bug IDs: None

TEST RESULT**Fail**

B.3 02wrapper

Verify the wrapperAPI in stand alone mode. Basic tests of wrapper command execution, error trapping and handling, correctness of data translation, and loading of shared objects.

B.3.1 WRAPPER01

Test Case: WRAPPER01

Purpose: Verify the correctness of implementation and error handling in the wrapperAPI

Tester: Maria Barnes

Test machine: beowulf@ldas-test.ligo.caltech.edu

Date (mm/dd/yy): 01/16/01 **Time:** 12:30

ENVIRONMENT AND PREREQUISITES

This test is executed from the user account mdc on the Beowulf hardware. It depends on the successful completion of MDCPREP and LDASBUILD.

The input data should be stored in `.../ldasmdc/mpi/test/02wrapper/input/`

PROCEDURE

Run `.../ldasmdc/mpi/test/02wrapper/command/mpirun_wrap` script to execute all error schemas. In order to run a particular schema, pass the root name as an argument to the script, for example:
`mpirun_wrap error2-a.`

`.../ldasmdc/mpi/test/02wrapper/command/mpirun_wrap`

```
#!/bin/sh
# $Id: mpirun_wrap,v 1.3 2001/01/19 23:11:51 mbarnes Exp $
# shell commands should appear in line continued form here: \
PATH=/ldas/ldas-0.0/bin:/ldcg/bin \
LD_LIBRARY_PATH=/ldas/ldas-0.0/lib:/ldas/ldas-0.0/lib/genericAPI:/ldcg/lib \
export PATH LD_LIBRARY_PATH \
# \
exec tcsh "$0" ${1+"$@"}

# darn. the jobid used by the wrapper is an int, not a string
set RUNCODE MPIMDC

set i 0

set retval [ list ]

## PUT YOUR COMMAND FILENAMES HERE
set cmds {
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error1.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-a.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-bl.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-b2.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-c.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-d.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-e.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-f.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-g.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-h.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-i.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-j.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-k.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-l.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-m.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-n.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-o.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-p.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-q.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-r.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-s.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error2-t.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error3-a.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error3-b.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error4-a.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error4-b.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error4-c.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error5-a.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error5-b.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error5-c.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error5-d.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error5-e.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error5-f.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error5-g.schema
}
```



```

/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error5-h.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error5-i.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error5-j.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error5-k.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error5-l.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error5-m.schema
/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/command/error5-n.schema
}

foreach cmd [ split $cmds "\n" ] {
  set tmp [ list ]
  set cmd [ string trim $cmd ]
  if { ! [ string length $cmd ] } { continue }
  if { [ regexp -- {^#} $cmd ] } { continue }

  ;## if command line arguments are given, use them
  ;## to filter the scripts.
  if { [ llength $argv ] } {
    set root [ file rootname [ file tail $cmd ] ]
    if { [ lsearch -exact $argv $root ] < 0 } {
      continue
    }
  }

  set fname $cmd
  if { [ catch {
    set fid [ open $cmd r ]
    set cmd [ read $fid ]
    catch { close $fid }
  } err ] } {
    puts stderr "$fname: $errorInfo"
    catch { close $fid }
    continue
  }

  foreach line [ split $cmd "\n" ] {
    set line [ string trim $line ]
    if { ! [ string length $line ] } { continue }
    if { [ regexp -- {^#} $line ] } { continue }
    append tmp "$line\n"
  }

  if { ! [ string length $tmp ] } { continue }
  set cmd $tmp

  incr i
  regsub -all -- {[\s\n]+} $cmd { } cmd
  set cmd [ subst -nocommands -nobackslashes $cmd ]
  puts stderr $cmd
  exec lamclean -v
  if { [ catch {
    set retval [ eval exec mpirun -v $cmd ]
  } err ] } {
    puts stderr "$fname: $err"
  }

  if { [ string length $retval ] } {
    puts stderr "$fname: $retval"
    set retval [ list ]
  }
}

```

1. Illegal number of nodes in COMM_WORLD. - Less than two nodes were specified to run parallel job on.

```

/.../ldasmdc/mpi/test/02wrapper/command/error1.schema

```

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.1 error handler:
# "Illegal number of nodes in COMM_WORLD."
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange        log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates

```

```
#
-np 1 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwrapp
per/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/input.ilwd
```

Pass

2. Error specifications for parsing command line arguments

- (a) Bad node value. - Node value exceeds the total amount of nodes in the global communicator COMM_WORLD.

```
/. . . /ldasmdc/mpi/test/02wrapper/command/error2-a.schema
```

```
#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2-a error handler:
# "Bad node value."
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1,3) -dynlib=/ldcg/lib/lalwrapp
per/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/input.ilwd
```

Pass

- (b) Bad node range. - Invalid range is specified (node value exceeds the total amount of nodes in the global communicator or range limits are invalid).

```
/. . . /ldasmdc/mpi/test/02wrapper/command/error2-b1.schema
```

```
#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2-b1 error handler:
# "Bad node range."
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-3) -dynlib=/ldcg/lib/lalwrapp
per/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/input.ilwd
```

```
/. . . /ldasmdc/mpi/test/02wrapper/command/error2-b2.schema
```

```
#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2-b2 error handler:
# "Bad node range."
```

```

#
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(2-1) -dynlib=/ldcg/lib/lalwr
pper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/input.ilwd

```

Pass

- (c) Multiple occurrences of rank r in node list. - Rank r is specified multiple number of times.
 /.../ldasmdc/mpi/test/02wrapper/command/error2-c.schema

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2c error handler:
# "Multiple occurrences of rank R in nodelist"
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1,1) -dynlib=/ldcg/lib/lalwr
pper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/input.ilwd

```

Pass

- (d) Bad -nodelist format. - Invalid format specified for -nodelist command line argument.
 /.../ldasmdc/mpi/test/02wrapper/command/error2-d.schema

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2d error handler:
# "Bad -nodelist format".
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1&2) -dynlib=/ldcg/lib/lalwr
pper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/input.ilwd

```

Pass

- (e) Rank 0 cannot be included in nodelist, it is reserved for wrapperAPI master. - Rank 0 is reserved for the wrapperAPI master.

/.../ldasmdc/mpi/test/02wrapper/command/error2-e.schema

```
#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2e error handler:
# "Rank 0 cannot be included in nodelist"
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(0,1-2) -dynlib=/ldcg/lib/lalwra
pper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0
.00097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTi
meRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoi
ng/jobs/ldasmdc/mpi/test/02wrapper/input/input.ilwd
```

Pass

- (f) Bad (hostname,port) format. - Invalid format specified for host/port command line argument.

/.../ldasmdc/mpi/test/02wrapper/command/error2-f.schema

```
#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2f error handler:
# "Bad (host,port) format"
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwra
pper/libinspiral.so -dataAPI=(data1234) -resultAPI=(data\ server,10032) -filterparams=(0,4096,1,8,0
.00097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTi
meRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoi
ng/jobs/ldasmdc/mpi/test/02wrapper/input/input.ilwd
```

Pass

- (g) Bad -dynlib format. - Invalid format specified for -dynlib command line argument.

/.../ldasmdc/mpi/test/02wrapper/command/error2-g.schema

```
#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2g error handler:
# "Bad -dybllib format."
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
```

```
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwr
apper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.
0097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTi
meRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoin
g/jobs/ldasmdc/mpi/test/02wrapper/input/input.ilwd
```

Pass

- (h) Bad -filterparams format. - Invalid format specified for -filterparams argument.
/.../ldasmdc/mpi/test/02wrapper/command/error2-h.schema

```
#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2h error handler:
# "Bad -filterparams format"
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow;           Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwra
pper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/input.ilwd
```

Pass

- (i) Bad -realTimeRatio argument. - Invalid value specified for -realTimeRatio command line argu-
ment.
/.../ldasmdc/mpi/test/02wrapper/command/error2-i.schema

```
#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2i error handler:
# "Bad -realTimeRatio format"
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow;           Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwra
pper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=x -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing/j
obs/ldasmdc/mpi/test/02wrapper/input/input.ilwd
```

Pass

- (j) Bad -doLoadBalance argument.- Invalid value specified for -loadBalance command line argu-
ment. One of $T|TRUE|F|FALSE$.
/.../ldasmdc/mpi/test/02wrapper/command/error2-j.schema

```
#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2j error handler:
# "Bad -doLoadBalance argument."
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
```

```

# argv[5] = deltaT           Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwra
pper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0,0
0097656,0,0,3,0,200,0,0.001,0,0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=A -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing/job
s/ldasmdc/mpi/test/02wrapper/input/input.ilwd

```

Pass

- (k) Bad -dataDistributor argument.- Invalid value specified for -dataDistributor command line argument. One of *W|WRAPPER||S|SEARCHMASTER*.

/.../ldasmdc/mpi/test/02wrapper/command/error2-k.schema

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2k error handler:
# "Bad -dataDistributor argument."
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwra
pper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0,0
0097656,0,0,3,0,200,0,0.001,0,0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=Who -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoi
ng/jobs/ldasmdc/mpi/test/02wrapper/input/input.ilwd

```

Pass

- (l) Bad -communicateOutput argument.- Invalid value specified for -communicateOutput command line argument. One of *A|ALWAYS||O|ONCE*.

/.../ldasmdc/mpi/test/02wrapper/command/error2-l.schema

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2l error handler:
# "Bad -communicateOutput argument."
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwra
pper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0,0
0097656,0,0,3,0,200,0,0.001,0,0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/input.ilwd -communicateOutput=ONLY

```

Pass

- (m) Unknown command line argument. - Unknown command line argument specified.
- /.../ldasmdc/mpi/test/02wrapper/command/error2-m.schema

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2m error handler:
# "Unknown command line argument."
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwrapp
er/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/input.ilwd -abc=123

```

Pass

- (n) -nodelist must be specified. - -nodelist command line argument must be specified.
 /.../ldasmdc/mpi/test/02wrapper/command/error2-n.schema

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2n error handler:
# "-nodelist must be specified."
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -dynlib=/ldcg/lib/lalwrapp
er/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.00097656,0,0,3.0,
200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTimeRatio=0.9 -doLo
adBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing/jobs/ldasmdc/mpi
/test/02wrapper/input/input.ilwd

```

Pass

- (o) -dynlib must be specified. - -dynlib command line argument must be specified.
 /.../ldasmdc/mpi/test/02wrapper/command/error2-o.schema

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2o error handler:
# "-dynlib must be specified."
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dataAPI=(data,1234) -re
sultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.00097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0
\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTimeRatio=0.9 -doLoadBalance=FALSE -dataDistrib
utor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/input/input
.ilwd

```

Pass

- (p) -jobID must be specified. -jobId command line argument must be specified.
 /.../ldasmdc/mpi/test/02wrapper/command/error2-p.schema

```
#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2p error handler:
# "-jobID must be specified."
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -dynlib=/ldcg/lib/lalwrapper/libinspiral
.so -nodelist=(1-2) -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -uniqueID=9.0 -inputFile=/ldas_outgoing/jobs/ldas
mdc/mpi/test/02wrapper/input/input.ilwd
```

Pass

- (q) -uniqueID must be specified. -uniqueId command line argument must be specified.
 /.../ldasmdc/mpi/test/02wrapper/command/error2-q.schema

```
#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2q error handler:
# "-uniqueID must be specified."
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -dynlib=/ldcg/lib/lalwrapper/libinspiral
.so -nodelist=(1-2) -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -jobID=$i -dataDistributor=W -inputFile=/ldas_outgoing/jobs/ldasmdc/
mpi/test/02wrapper/input/input.ilwd
```

Pass

- (r) -mpiAPI must be specified. -mpiAPI command line argument must be specified.
 /.../ldasmdc/mpi/test/02wrapper/command/error2-r.schema

```
#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2r error handler:
# "-mpiAPI must be specified."
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
```



```
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -dynlib=/ldcg/lib/lalwrapper/libinspiral.so -nodelist=(1-2) -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.00097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTimeRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/input/input.ilwd
```

Pass

- (s) -dataAPI must be specified. - -dataAPI command line argument must be specified.
/.../ldasmdc/mpi/test/02wrapper/command/error2-s.schema

```
#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2s error handler:
# "-dataAPI must be specified."
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -dynlib=/ldcg/lib/lalwrapper/libinspiral.so -nodelist=(1-2) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.00097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTimeRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/input/input.ilwd
```

Pass

- (t) -resultAPI must be specified. - -resultAPI command line argument must be specified.
/.../ldasmdc/mpi/test/02wrapper/command/error2-t.schema

```
#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.2t error handler:
# "-resultAPI must be specified."
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -dynlib=/ldcg/lib/lalwrapper/libinspiral.so -nodelist=(1-2) -dataAPI=(data,1234) -filterparams=(0,4096,1,8,0.00097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTimeRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/input/input.ilwd
```

Pass

3. Error specifications for loading dso and calling functions from dso:

- (a) Error loading dynamic library: error_message. - Error loading dynamic shared object library specified by dynlib command line argument followed by the error_message.
/.../ldasmdc/mpi/test/02wrapper/command/error3-a.schema

```
#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.3a error handler:
# "Error loading dynamic library: msg"
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
```

```

# argv[2] = numPoints      Number of data points in a segment
# argv[3] = numSegments   Number of overlapping data segments
# argv[4] = numChisqBins  Number of frequency bands for chisq veto
# argv[5] = deltaT        Sampling interval
# argv[6] = overlap       Overlap between segments (# of points)
# argv[7] = invSpecTrunc  Duration of inverse spectrum in time domain
# argv[8] = fLow          Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange     log_2( dynamic range scaling )
# argv[12] = numTplts     number of templates
# argv[13] = (m1,m2;...)  list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwra
pper/libinspiralghost.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,
8,0.00097656,0,0,3.0,200.0,0.001,0,0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -re
alTimeRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_out
going/jobs/ldasmdc/mpi/test/02wrapper/input/input.ilwd

```

Pass

- (b) (foo function): msg_str. - Node failed with msg_str error when calling foo function.
/.../ldasmdc/mpi/test/02wrapper/command/error3-b.schema

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.3b error handler:
# "( initSearch ): msg"
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch  Number of coarse templates to exchange
# argv[2] = numPoints      Number of data points in a segment
# argv[3] = numSegments   Number of overlapping data segments
# argv[4] = numChisqBins  Number of frequency bands for chisq veto
# argv[5] = deltaT        Sampling interval
# argv[6] = overlap       Overlap between segments (# of points)
# argv[7] = invSpecTrunc  Duration of inverse spectrum in time domain
# argv[8] = fLow          Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange     log_2( dynamic range scaling )
# argv[12] = numTplts     number of templates
# argv[13] = (m1,m2;...)  list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwra
pper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0,0
0097656,0,0,3.0,200.0,0.001,0,0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/input.ilwd

```

Pass

4. Error specifications for communication and data sockets:

- (a) MsgSocket error: error_message. - Error occurred when using mpiAPI socket for communica-
tion. error_message contains the problem description.
/.../ldasmdc/mpi/test/02wrapper/command/error4-a.schema

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.4a error handler:
# "MsgSocket error: error"
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch  Number of coarse templates to exchange
# argv[2] = numPoints      Number of data points in a segment
# argv[3] = numSegments   Number of overlapping data segments
# argv[4] = numChisqBins  Number of frequency bands for chisq veto
# argv[5] = deltaT        Sampling interval
# argv[6] = overlap       Overlap between segments (# of points)
# argv[7] = invSpecTrunc  Duration of inverse spectrum in time domain
# argv[8] = fLow          Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange     log_2( dynamic range scaling )
# argv[12] = numTplts     number of templates
# argv[13] = (m1,m2;...)  list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,12345) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwra
pper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0,0
0097656,0,0,3.0,200.0,0.001,0,0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/input.ilwd

```

Pass

- (b) connect_failure - The socket was unable to connect to the designated address.
 /.../ldasmdc/mpi/test/02wrapper/command/error4-c.schema

```
#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.4c error handler:
# "connection_failure"
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwra
pper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,12345) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/input.ilwd
```

Pass

- (c) invalid_host - The host to which the address refers can not be found.
 /.../ldasmdc/mpi/test/02wrapper/command/error4-b.schema

```
#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.4b error handler:
# "invalid_host"
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwra
pper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(data,10032) -filterparams=(0,4096,1,8,0.0009765
6,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTimeRatio
=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing/jobs/
ldasmdc/mpi/test/02wrapper/input/input.ilwd
```

Pass

5. Error specifications for generating MPI data type representing inPut data structure:

- (a) Malformed ILWD input. - Input ILWD data is malformed: either element is NULL, or it is not an ILWD container or container is empty.

/.../ldasmdc/mpi/test/02wrapper/command/error5-a.schema

```
#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.5a error handler:
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
```

```

# argv[9] = rhosqThreshold      thresholds for SNR
# argv[10] = chisqThreshold     thresholds for chisqr
# argv[11] = dynRange          log_2( dynamic range scaling )
# argv[12] = numTplts          number of templates
# argv[13] = (m1,m2;...)       list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwrapp
pper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3,0,200,0,0,001,0,0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/empty.ilwd

```

Fail

Notes: **wrapperAPI segfaults if input ILWD file contains an empty container**

- (b) sequence is missing. - Sequence data is missing for inPut data structure.
 /.../ldasmdc/mpi/test/02wrapper/command/error5-b.schema

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.5b error handler:
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch      Number of coarse templates to exchange
# argv[2] = numPoints          Number of data points in a segment
# argv[3] = numSegments        Number of overlapping data segments
# argv[4] = numChisqBins       Number of frequency bands for chisq veto
# argv[5] = deltaT             Sampling interval
# argv[6] = overlap            Overlap between segments (# of points)
# argv[7] = invSpecTrunc       Duration of inverse spectrum in time domain
# argv[8] = fLow               Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold     thresholds for SNR
# argv[10] = chisqThreshold    thresholds for chisqr
# argv[11] = dynRange          log_2( dynamic range scaling )
# argv[12] = numTplts          number of templates
# argv[13] = (m1,m2;...)       list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwrapp
pper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3,0,200,0,0,001,0,0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/no_sequence.ilwd

```

Pass

- (c) data is missing - ILWD format element representing the data is missing.
 /.../ldasmdc/mpi/test/02wrapper/command/error5-c.schema

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.5c error handler:
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch      Number of coarse templates to exchange
# argv[2] = numPoints          Number of data points in a segment
# argv[3] = numSegments        Number of overlapping data segments
# argv[4] = numChisqBins       Number of frequency bands for chisq veto
# argv[5] = deltaT             Sampling interval
# argv[6] = overlap            Overlap between segments (# of points)
# argv[7] = invSpecTrunc       Duration of inverse spectrum in time domain
# argv[8] = fLow               Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold     thresholds for SNR
# argv[10] = chisqThreshold    thresholds for chisqr
# argv[11] = dynRange          log_2( dynamic range scaling )
# argv[12] = numTplts          number of templates
# argv[13] = (m1,m2;...)       list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwrapp
pper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3,0,200,0,0,001,0,0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/no_data.ilwd

```

Pass

- (d) domain is missing - ILWD format element representing the domain is missing.
 /.../ldasmdc/mpi/test/02wrapper/command/error5-d.schema

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.5d error handler:
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch      Number of coarse templates to exchange

```

```

# argv[2] = numPoints      Number of data points in a segment
# argv[3] = numSegments   Number of overlapping data segments
# argv[4] = numChisqBins  Number of frequency bands for chisq veto
# argv[5] = deltaT        Sampling interval
# argv[6] = overlap       Overlap between segments (# of points)
# argv[7] = invSpecTrunc  Duration of inverse spectrum in time domain
# argv[8] = fLow          Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange     log_2( dynamic range scaling )
# argv[12] = numTplts     number of templates
# argv[13] = (m1,m2;...)  list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwrapp
per/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/no_domain.ilwd

```

Pass

- (e) Undefined domain. - Undefined domain was specified. One of *TIME|FREQ|BOTH*.
 /.../ldasmdc/mpi/test/02wrapper/command/error5-e.schema

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.5e error handler:
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch  Number of coarse templates to exchange
# argv[2] = numPoints      Number of data points in a segment
# argv[3] = numSegments   Number of overlapping data segments
# argv[4] = numChisqBins  Number of frequency bands for chisq veto
# argv[5] = deltaT        Sampling interval
# argv[6] = overlap       Overlap between segments (# of points)
# argv[7] = invSpecTrunc  Duration of inverse spectrum in time domain
# argv[8] = fLow          Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange     log_2( dynamic range scaling )
# argv[12] = numTplts     number of templates
# argv[13] = (m1,m2;...)  list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwrapp
per/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/bad_domain.ilwd

```

Pass

- (f) Invalid boolean data type. - ILwd element representing boolean data must be of char_u type.
 /.../ldasmdc/mpi/test/02wrapper/command/error5-f.schema

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.5f error handler:
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch  Number of coarse templates to exchange
# argv[2] = numPoints      Number of data points in a segment
# argv[3] = numSegments   Number of overlapping data segments
# argv[4] = numChisqBins  Number of frequency bands for chisq veto
# argv[5] = deltaT        Sampling interval
# argv[6] = overlap       Overlap between segments (# of points)
# argv[7] = invSpecTrunc  Duration of inverse spectrum in time domain
# argv[8] = fLow          Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange     log_2( dynamic range scaling )
# argv[12] = numTplts     number of templates
# argv[13] = (m1,m2;...)  list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwrapp
per/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/bad_boolean.ilwd

```

Pass

- (g) start_time is missing. - Could not find elements with "start_time" name field.
 /.../ldasmdc/mpi/test/02wrapper/command/error5-g.schema

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.5g error handler:

```

```

#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwra
pper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/no_starttime.ilwd

```

Pass

- (h) stop_time is missing. - Could not find elements with stop_time name field.
 /.../ldasmdc/mpi/test/02wrapper/command/error5-h.schema

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.5h error handler:
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwra
pper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/no_stoptime.ilwd

```

Pass

- (i) step_size is missing. - Could not find element with step_size name field.
 /.../ldasmdc/mpi/test/02wrapper/command/errori-a.schema

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.5i error handler:
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwra
pper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0
0097656,0,0,3.0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/no_stepsize.ilwd

```

Pass

- (j) Malformed input format for gpsTimeInterval. - Invalid input format was specified for gp-
 sTimeInterval.
 /.../ldasmdc/mpi/test/02wrapper/command/error5-j.schema

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.5g error handler:
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwrapp
per/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0,0
0097656,0,0,3,0,200,0,0.001,0,0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/bad_time_interval.ilwd

```

Pass

- (k) start_freq is missing. - Could not find element with start_freq name field.
 /.../ldasmdc/mpi/test/02wrapper/command/error5-k.schema

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.5k error handler:
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwrapp
per/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0,0
0097656,0,0,3,0,200,0,0.001,0,0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/no_startfreq.ilwd

```

Pass

- (l) stop_freq is missing. - Could not find element with stop_freq name field.
 /.../ldasmdc/mpi/test/02wrapper/command/error5-l.schema

```

#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.5l error handler:
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)    list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwrapp
per/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0,0
0097656,0,0,3,0,200,0,0.001,0,0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTim
eRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing
/jobs/ldasmdc/mpi/test/02wrapper/input/no_stopfreq.ilwd

```

Pass

- (m) Malformed input format for frequencyInterval. - Invalid input format was specified for frequencyInterval.

/.../ldasmdc/mpi/test/02wrapper/command/error5-m.schema

```
#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.5m error handler:
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)   list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwrapper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0097656,0,0,3,0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTimeRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/input/bad_freq_interval.ilwd
```

Pass

- (n) Malformed input format for timeFreqInterval. - Invalid input format was specified for timeFreqInterval.

/.../ldasmdc/mpi/test/02wrapper/command/error5-n.schema

```
#
# lam boot schema for inspiral shared object
# this schema is to verify wrapper01.5n error handler:
#
# libinspiral.so should be invoked with the following arguments
#
# argv[1] = numCoarseExch   Number of coarse templates to exchange
# argv[2] = numPoints       Number of data points in a segment
# argv[3] = numSegments     Number of overlapping data segments
# argv[4] = numChisqBins    Number of frequency bands for chisq veto
# argv[5] = deltaT          Sampling interval
# argv[6] = overlap         Overlap between segments (# of points)
# argv[7] = invSpecTrunc    Duration of inverse spectrum in time domain
# argv[8] = fLow            Low frequency cut-off in inverse spectrum
# argv[9] = rhosqThreshold  thresholds for SNR
# argv[10] = chisqThreshold thresholds for chisqr
# argv[11] = dynRange       log_2( dynamic range scaling )
# argv[12] = numTplts       number of templates
# argv[13] = (m1,m2;...)   list of templates
#
-np 3 /ldas/ldas-0.0/bin/wrapperAPI -mpiAPI=(beowulf,11000) -nodelist=(1-2) -dynlib=/ldcg/lib/lalwrapper/libinspiral.so -dataAPI=(data,1234) -resultAPI=(dataserver,10032) -filterparams=(0,4096,1,8,0.0097656,0,0,3,0,200.0,0.001,0.0,6,3,(1.0,1.0\;1.4,1.4\;2.0,2.0\;2.2,2.2\;2.4,2.4\;5.5,5.5)) -realTimeRatio=0.9 -doLoadBalance=FALSE -dataDistributor=W -jobID=$i -uniqueID=9.0 -inputFile=/ldas_outgoing/jobs/ldasmdc/mpi/test/02wrapper/input/bad_both_interval.ilwd
```

Pass

SUMMARY

Known faults encountered – list bug IDs: None

New faults submitted – list bug IDs: #621: problem handling NULL ILWD input data

TEST RESULT

Conditional Pass

B.4 03mpi

Verify the mpiAPI in stand alone mode. Basic tests of mpiAPI command execution, error trapping and handling, and logging.

B.4.1 MPIAPI01

Test Case: MPIAPI01

Purpose: Verify the correctness of implementation and error handling in the mpiAPI

Tester: Isaac Salzman

Test machine: tarazed.ligo.caltech.edu

Date (mm/dd/yy): 01/16/01 **Time:** 15:00

ENVIRONMENT AND PREREQUISITES

This test is executed from the user account `ldas_mdc` on the Beowulf hardware. It depends on the successful completion of MDCPREP and LDASBUILD.

PROCEDURE

Run the LDAS user command scripts listed in the tests below. Inspect the log files and verify that errors were logged. No output verification is required in this test as no output should be created by any of the jobs. The component parts of this test are executed as free-standing Tcl scripts. These scripts will establish socket communication with a local or remote LDAS system and issue a “user command” on that system. Any message returned by the system at the time the user command is issued is displayed on STDERR in the local terminal. All user command transactions are logged in the file `LDASmpi.log.html`, which is visible from any web browser.

1. Error specifications for parsing command line arguments

- (a) No arguments. - `mpiFile2File` user command called with no arguments. The manager log file should reflect the submission of an invalid request. No e-mail will be generated, the user is informed of the failure in the terminal from which the command is run.

```

/.../ldasmdc/mpi/test/03mpi/command/MPI0101.tclsh

#!/ldcg/bin/tclsh
#
# $Id: MPI0101.tclsh,v 1.2 2001/01/16 00:29:36 pehrens Exp $

set cmd "ldasJob
{
  -name ldas_mdc
  -password beowulf
  -email mpigroup@gravity.phys.uwm.edu
}
{
  mpiFile2File
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

- (b) Required option missing. - A required option was not provided. The mpiAPI log file will contain an entry referring to the missing option(s), and e-mail will be sent to the user describing the error.

```

/.../ldasmdc/mpi/test/03mpi/command/MPI0102.tclsh

#!/ldcg/bin/tclsh
#
# $Id: MPI0102.tclsh,v 1.2 2001/01/16 00:29:19 pehrens Exp $

set cmd "ldasJob
{
  -name ldas_mdc
  -password beowulf

```

```

    -email mpigroup@gravity.phys.uwm.edu
  }
  {
    mpiFile2File
    -returnprotocol http://mpi0102
    -dynlib          libtrivial.so
    -filterparams    {}
    -inputfile       trivial.ilwd
    -subject          {MPI0102}
  }"
set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

- (c) Malformed option. - Provided argument is not formatted as defined in the specification for the API. The mpiAPI log file will contain an entry referring to the malformed option(s), and e-mail will be sent to the user describing the error.

```

/.../ldasmdc/mpi/test/03mpi/command/MPI0103.tclsh

```

```

#!/ldcg/bin/tclsh
#
# $Id: MPI0103.tclsh,v 1.2 2001/01/16 00:29:02 pehrens Exp $

set cmd "ldasJob
{
  -name ldas_mdc
  -password beowulf
  -email mpigroup@gravity.phys.uwm.edu
}
{
  mpiFile2File
  -returnprotocol http://mpi0103
  -dynlib          libtrivial.so
  -filterparams    0,1,2,X
  -inputfile       trivial.ilwd
  -subject          {MPI0103}
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

- (d) Foreign option. - Provided argument is not an argument as defined in the specification for the user command. The managerAPI log file will contain an entry referring to the invalid option(s). No e-mail will be generated, the user is informed of the failure in the terminal from which the command is run.

```

/.../ldasmdc/mpi/test/03mpi/command/MPI0104.tclsh

```

```

#!/ldcg/bin/tclsh
#
# $Id: MPI0104.tclsh,v 1.2 2001/01/16 00:28:45 pehrens Exp $

set cmd "ldasJob
{
  -name ldas_mdc
  -password beowulf
  -email mpigroup@gravity.phys.uwm.edu
}
{
  mpiFile2File
  -returnprotocol http://mpi0104
  -dynlib          libtrivial.so
  -filterparams    0,1,2,3
  -inputfile       trivial.ilwd
  -issyvoo         1.0
  -subject          {MPI0104}
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

SUMMARY

Known faults encountered – list bug IDs: None

New faults submitted – list bug IDs: None

TEST RESULT

Pass

B.5 04lalwrapper

Verify the LALWrapperInterface in stand alone mode. Basic tests of error trapping and handling, correctness of data manipulation and trivial DSO.

B.5.1 LALWRAPPER01

Test Case: LALWRAPPER01

Purpose: Verify the correctness of implementation and error handling in the LALWrapperInterface

Tester: Isaac Salzman

Test machine: tarazed.ligo.caltech.edu

Date (mm/dd/yy): 01/17/01 **Time:** 19:00

ENVIRONMENT AND PREREQUISITES

This test is executed from the user account `ldas_mdc` on the Beowulf hardware. It depends on the successful completion of MDCPREP and LALWRAPPERBUILD.

PROCEDURE

1. illness 00: no illness

```
/. . . /ldasmdc/mpi/test/04lalwrapper/command/LALWRAPPER0101.tclsh
```

```
#!/ldcg/bin/tclsh
#
# $Id: LALWRAPPER0101.tclsh,v 1.5 2001/01/17 05:32:56 jolien Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libsick.so
  -filterparams (00)
  -inputfile /ldas_outgoing/jobs/temp/1026x1_1024.ilwd
  -subject LALWRAPPER0101
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Output to log files should contain no error messages.

Pass

2. illness 01: failed LAL routine

```
/. . . /ldasmdc/mpi/test/04lalwrapper/command/LALWRAPPER0102.tclsh
```

```
#!/ldcg/bin/tclsh
#
# $Id: LALWRAPPER0102.tclsh,v 1.5 2001/01/17 05:32:56 jolien Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libsick.so
  -filterparams (01)
  -inputfile /ldas_outgoing/jobs/temp/1026x1_1024.ilwd
  -subject LALWRAPPER0102
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Output to log files should contain the error message:

```
Level 1: $Id: 04test.tex,v 1.6 2001/02/12 23:31:56 jolien Exp $
Status code 16: Fail.
function LALInitSearch, file Sick.c, line 104
```

from initSearch.

Pass

3. illness 02: recursively failed LAL routine

```
./.../ldasmdc/mpi/test/04lalwrapper/command/LALWRAPPER0103.tclsh
```

```
#!/ldcg/bin/tclsh
#
# $Id: LALWRAPPER0103.tclsh,v 1.5 2001/01/17 05:32:56 jolien Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libsick.so
  -filterparams (02)
  -inputfile /ldas_outgoing/jobs/temp/1026x1_1024.ilwd
  -subject LALWRAPPER0103
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Output to log files should contain the error message:

```
Level 1: $Id: 04test.tex,v 1.6 2001/02/12 23:31:56 jolien Exp $
Status code -1: Recursive error
function LALInitSearch, file Sick.c, line 112

Level 2: $Id: 04test.tex,v 1.6 2001/02/12 23:31:56 jolien Exp $
Status code -1: Recursive error
function LALRecursiveFailFunction, file Sick.c, line 321

Level 3: $Id: 04test.tex,v 1.6 2001/02/12 23:31:56 jolien Exp $
Status code 16: Fail.
function LALRecursiveFailFunction, file Sick.c, line 326
```

from initSearch.

Pass

Notes: **Error message was truncated**

4. illness 03: status description too long

```
./.../ldasmdc/mpi/test/04lalwrapper/command/LALWRAPPER0104.tclsh
```

```
#!/ldcg/bin/tclsh
#
# $Id: LALWRAPPER0104.tclsh,v 1.5 2001/01/17 05:32:56 jolien Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libsick.so
  -filterparams (03)
  -inputfile /ldas_outgoing/jobs/temp/1026x1_1024.ilwd
  -subject LALWRAPPER0104
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Output to log files should contain the error message:

```
stringifyLALStatus: error message too long, file: LALWrapperInterface.c, line: 690, $Id: 04test.tex,v 1.6 2001/02/12 23:31:56 jolien Exp $
```

from `initSearch`.

Pass

5. illness 04: too many levels of failed LAL routines

```
/.../ldasmdc/mpi/test/04lalwrapper/command/LALWRAPPER0105.tclsh
```

```
#!/ldcg/bin/tclsh
#
# $Id: LALWRAPPER0105.tclsh,v 1.5 2001/01/17 05:32:56 jolien Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libsick.so
  -filterparams (04)
  -inputfile /ldas_outgoing/jobs/temp/1026x1_1024.ilwd
  -subject LALWRAPPER0105
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Output to log files should contain the error message:

```
stringifyLALStatus: too many levels in status structure, file: LALWrapperInterface.c, line: 648, $Id: 04test.tex,v 1.6 2001/02/12 23:31:56 jolien Exp $
```

from `initSearch`.

Pass

6. illness 05: detected memory leak

```
/.../ldasmdc/mpi/test/04lalwrapper/command/LALWRAPPER0106.tclsh
```

```
#!/ldcg/bin/tclsh
#
# $Id: LALWRAPPER0106.tclsh,v 1.5 2001/01/17 05:32:56 jolien Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libsick.so
  -filterparams (05)
  -inputfile /ldas_outgoing/jobs/temp/1026x1_1024.ilwd
  -subject LALWRAPPER0106
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Output to log files should contain the error message:

```
signalHandler: Caught SIGSEGV, file: LALWrapperInterface.c, line: 550, $Id: 04test.tex,v 1.6 2001/02/12 23:31:56 jolien Exp $
```

from `finalizeSearch`.

Pass

7. illness 06: freeOutput works on valid output

```
/.../ldasmdc/mpi/test/04lalwrapper/command/LALWRAPPER0107.tclsh
```



```
#!/ldcg/bin/tclsh
#
# $Id: LALWRAPPER0107.tclsh,v 1.5 2001/01/17 05:32:56 jolien Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libsick.so
  -filterparams (06)
  -inputfile /ldas_outgoing/jobs/temp/1026x1_1024.ilwd
  -subject LALWRAPPER0107
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Output to log files should contain no error messages.

Pass

8. illness 07: illegal output pointer

```
./.../ldasmdc/mpi/test/04lalwrapper/command/LALWRAPPER0108.tclsh
```

```
#!/ldcg/bin/tclsh
#
# $Id: LALWRAPPER0108.tclsh,v 1.5 2001/01/17 05:32:56 jolien Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libsick.so
  -filterparams (07)
  -inputfile /ldas_outgoing/jobs/temp/1026x1_1024.ilwd
  -subject LALWRAPPER0108
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Output to log files should contain the error message:

```
signalHandler: Caught SIGSEGV, file: LALWrapperInterface.c, line: 550, $Id: 04test.tex,v 1.6 2001/02/12 23:31:56 jolien Exp $
```

from freeOutput.

Fail

Notes: **Error came from** applySearch.

This is actually a bug in Sick.c, which has now been fixed. Now happyAPI runs and “correctly” causes a SIGSEGV to be raised in freeOutput, which is caught. However, I expect that wrapperAPI will generate a SIGSEGV even before freeOutput is called, since it will presumably try to do something with the output (unlike happyAPI) — so I still expect that this test will fail with wrapperAPI (this has not been checked).

It is perhaps unrealistic to expect wrapperAPI to deal with ill-formed output.

9. illness 08: output has a dodgy link to the middle of a linked-list

```
./.../ldasmdc/mpi/test/04lalwrapper/command/LALWRAPPER0109.tclsh
```

```
#!/ldcg/bin/tclsh
#
# $Id: LALWRAPPER0109.tclsh,v 1.5 2001/01/17 05:32:56 jolien Exp $

set cmd "ldasJob
```

```
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libsick.so
  -filterparams (08)
  -inputfile /ldas_outgoing/jobs/temp/1026x1_1024.ilwd
  -subject LALWRAPPER0109
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Output to log files should contain no error messages.

Pass

10. illness 09: tangled output

```
/.../ldasmdc/mpi/test/04lalwrapper/command/LALWRAPPER0110.tclsh
```

```
#!/ldcg/bin/tclsh
#
# $Id: LALWRAPPER0110.tclsh,v 1.5 2001/01/17 05:32:56 jolien Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libsick.so
  -filterparams (09)
  -inputfile /ldas_outgoing/jobs/temp/1026x1_1024.ilwd
  -subject LALWRAPPER0110
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Output to log files should contain the error message:

```
signalHandler: Caught SIGSEGV, file: LALWrapperInterface.c, line: 550, $Id: 04test.tex,v 1.6 2001/02/12 23:31:56 jolien Exp $
```

from freeOutput.

Fail

Notes: **No error message, job completed successfully.**

Actually, the job may have died due to a signal raised in wrapperAPI while trying to parse the tangled output, before freeOutput is run. This could be checked, but the test is probably too ill to worry about.

It is perhaps unrealistic to expect either wrapperAPI or LALWrapperInterface to deal with ill-formed output linked lists.

11. illness 10: SIGSEGV in initSearch

```
/.../ldasmdc/mpi/test/04lalwrapper/command/LALWRAPPER0111.tclsh
```

```
#!/ldcg/bin/tclsh
#
# $Id: LALWRAPPER0111.tclsh,v 1.5 2001/01/17 05:32:56 jolien Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libsick.so
  -filterparams (10)
  -inputfile /ldas_outgoing/jobs/temp/1026x1_1024.ilwd
  -subject LALWRAPPER0111
}"
```

```

}"
set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Output to log files should contain the error message:

```
signalHandler: Caught SIGSEGV, file: LALWrapperInterface.c, line: 550, $Id: 04test.tex,v 1.6 2001/02/12 23:31:56 jolien Exp $
```

from initSearch.

Pass

12. illness 11: SIGABRT in initSearch

```
./.../ldasmdc/mpi/test/04lalwrapper/command/LALWRAPPER0112.tclsh
```

```

#!/ldcg/bin/tclsh
#
# $Id: LALWRAPPER0112.tclsh,v 1.5 2001/01/17 05:32:56 jolien Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libsick.so
  -filterparams (11)
  -inputfile /ldas_outgoing/jobs/temp/1026x1_1024.ilwd
  -subject LALWRAPPER0112
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Output to log files should contain the error message:

```
signalHandler: Caught SIGABRT, file: LALWrapperInterface.c, line: 547, $Id: 04test.tex,v 1.6 2001/02/12 23:31:56 jolien Exp $
```

from initSearch.

Pass

13. illness 12: SIGSEGV in conditionData

```
./.../ldasmdc/mpi/test/04lalwrapper/command/LALWRAPPER0113.tclsh
```

```

#!/ldcg/bin/tclsh
#
# $Id: LALWRAPPER0113.tclsh,v 1.5 2001/01/17 05:32:56 jolien Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libsick.so
  -filterparams (12)
  -inputfile /ldas_outgoing/jobs/temp/1026x1_1024.ilwd
  -subject LALWRAPPER0113
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Output to log files should contain the error message:

```
signalHandler: Caught SIGSEGV, file: LALWrapperInterface.c, line: 550, $Id: 04test.tex,v 1.6 2001/02/12 23:31:56 jolien Exp $
```

from conditionData.

Pass

14. illness 13: SIGABRT in conditionData

`../../ldasmdc/mpi/test/04lalwrapper/command/LALWRAPPER0114.tclsh`

```
#!/ldcg/bin/tclsh
#
# $Id: LALWRAPPER0114.tclsh,v 1.5 2001/01/17 05:32:56 jolien Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libsick.so
  -filterparams (13)
  -inputfile /ldas_outgoing/jobs/temp/1026x1_1024.ilwd
  -subject LALWRAPPER0114
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Output to log files should contain the error message:

```
signalHandler: Caught SIGABRT, file: LALWrapperInterface.c, line: 547, $Id: 04test.tex,v 1.6 2001/02/12 23:31:56 jolien Exp $
```

from conditionData.

Pass

15. illness 14: SIGSEGV in applySearch

`../../ldasmdc/mpi/test/04lalwrapper/command/LALWRAPPER0115.tclsh`

```
#!/ldcg/bin/tclsh
#
# $Id: LALWRAPPER0115.tclsh,v 1.5 2001/01/17 05:32:56 jolien Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libsick.so
  -filterparams (14)
  -inputfile /ldas_outgoing/jobs/temp/1026x1_1024.ilwd
  -subject LALWRAPPER0115
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Output to log files should contain the error message:

```
signalHandler: Caught SIGSEGV, file: LALWrapperInterface.c, line: 550, $Id: 04test.tex,v 1.6 2001/02/12 23:31:56 jolien Exp $
```

from applySearch.

Pass

16. illness 15: SIGABRT in applySearch

`../../ldasmdc/mpi/test/04lalwrapper/command/LALWRAPPER0116.tclsh`

```
#!/lscg/bin/tclsh
#
# $Id: LALWRAPPER0116.tclsh,v 1.5 2001/01/17 05:32:56 jolien Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /lscg/lib/lalwrapper/libsick.so
  -filterparams (15)
  -inputfile /ldas_outgoing/jobs/temp/1026x1_1024.ilwd
  -subject LALWRAPPER0116
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Output to log files should contain the error message:

```
signalHandler: Caught SIGABRT, file: LALWrapperInterface.c, line: 547, $Id: 04test.tex,v 1.6 2001/02/12 23:31:56 jolien Exp $
```

from applySearch.

Pass

17. illness 16: SIGSEGV in finalizeSearch

```
../../ldasmdc/mpi/test/04lalwrapper/command/LALWRAPPER0117.tclsh
```

```
#!/lscg/bin/tclsh
#
# $Id: LALWRAPPER0117.tclsh,v 1.5 2001/01/17 05:32:56 jolien Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /lscg/lib/lalwrapper/libsick.so
  -filterparams (16)
  -inputfile /ldas_outgoing/jobs/temp/1026x1_1024.ilwd
  -subject LALWRAPPER0117
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Output to log files should contain the error message:

```
signalHandler: Caught SIGSEGV, file: LALWrapperInterface.c, line: 550, $Id: 04test.tex,v 1.6 2001/02/12 23:31:56 jolien Exp $
```

from finalizeSearch.

Pass

18. illness 17: SIGABRT in finalizeSearch

```
../../ldasmdc/mpi/test/04lalwrapper/command/LALWRAPPER0118.tclsh
```

```
#!/lscg/bin/tclsh
#
# $Id: LALWRAPPER0118.tclsh,v 1.5 2001/01/17 05:32:56 jolien Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /lscg/lib/lalwrapper/libsick.so
  -filterparams (17)
  -inputfile /ldas_outgoing/jobs/temp/1026x1_1024.ilwd
  -subject LALWRAPPER0118
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Output to log files should contain the error message:

```
signalHandler: Caught SIGABRT, file: LALWrapperInterface.c, line: 547, $Id: 04test.tex,v 1.6 2001/02/12 23:31:56 jolien Exp $
```

from finalizeSearch.

Pass

SUMMARY

Known faults encountered – list bug IDs: None

New faults submitted – list bug IDs: None

TEST RESULT

Conditional Pass

B.6 05mpiWrapper

Verify correct interaction between the mpiAPI and the wrapperAPI: with and without trivial dso, error trapping, logging, data communication.

B.6.1 MPIWRAPPER01

Test Case: MPIWRAPPER01

Purpose: Verify correct interaction between the mpiAPI and the wrapperAPI

Tester: Isaac Salzman

Test machine: tarazed.ligo.caltech.edu

Date (mm/dd/yy): 01/16/01 **Time:** 16:00

ENVIRONMENT AND PREREQUISITES

This test is executed from the user account ldas_mdc on the Beowulf hardware. It depends on the successful completion of WRAPPER01 and MPI01.

The component parts of this test are executed as free-standing Tcl scripts. These scripts will establish socket communication with a local or remote LDAS system and issue a “user command” on that system. Any message returned by the system at the time the user command is issued is displayed on STDERR in the local terminal. Messages written to stdout and stderr by the .so, the wrapper API, and the mpi API are recorded in the file mpi.log. All user command transactions are logged in the file LDASmpi.log.html, which is visible from any web browser. The actual URL’s for the files described above are site dependent.

PROCEDURE

Run the LDAS user command scripts listed in the tests below. Inspect the log files and verify that errors were logged in the LDASmpi.log.html file. When output is produced, it’s existence should be verified by examining the results directory.

1. Error trapping and logging of wrapperAPI errors by the mpiAPI

- (a) Logging of wrapperAPI error - Verify that errors propagate from the wrapperAPI and then to the mpiAPI, and are logged in the mpiAPI log file. “Error loading dynamic library” is returned by the wrapperAPI to the mpiAPI where it is logged in both the LDASmpi.log.html and the LDASwrapper.log.html files.

```

/.../ldasmdc/mpi/test/05mpiWrapper/command/MPIWRAPPER0101.tclsh
#!/dcg/bin/tclsh
#
# $Id: MPIWRAPPER0101.tclsh,v 1.1 2001/01/16 22:07:05 pehrens Exp $

set cmd "ldasJob
{
  -name ldas_mdc
  -password beowulf
  -email mpigroup@gravity.phys.uwm.edu
}
{
  mpiFile2File
  -returnprotocol http://mpiwrapper0101
  -dynlib libtrivial.so
  -filterparams 0,1,2,3
  -inputfile trivial.ilwd
  -subject {MPIWRAPPER0101}
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

- (b) Logging of .so error - Verify that errors propagate correctly from the .so to the wrapperAPI and then to the mpiAPI, and are logged in the mpiAPI log file. “Number of templates in coarse

exchange is zero or negative” is returned by the .so to the wrapperAPI which relays it to the mpiAPI where it is logged in both the LDASmpi.log.html and the LDASwrapper.log.html files.

/.../ldasmdc/mpi/test/05mpiWrapper/command/MPIWRAPPER0102.tclsh

```
#!/ldcg/bin/tclsh
#
# $Id: MPIWRAPPER0102.tclsh,v 1.1 2001/01/16 22:07:19 pehrens Exp $

set cmd "ldasJob
{
  -name ldas_mdc
  -password beowulf
  -email mpigroup@gravity.phys.uwm.edu
}
{
  mpiFile2File
  -returnprotocol http://mpiwrapper0102
  -dynlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,2,1,1,0.00097656,0,0,1.0,1.0,1.0,1.0,1.0,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_short.ilwd
}
-subject {MPIWRAPPER0102}
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {\n\s|+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

2. Logging of normal mpiAPI/wrapperAPI communications

- (a) Logging of messages passed between wrapperAPI to mpiAPI - Verify that all communications between the wrapper and mpi API's are logged in the file LDASwrapper.log.html. /.../ldasmdc/mpi/test/05mpiWrapper/command/MPIWRAPPER0201.tclsh

```
#!/ldcg/bin/tclsh
#
# $Id: MPIWRAPPER0201.tclsh,v 1.1 2001/01/16 22:06:48 pehrens Exp $

set cmd "ldasJob
{
  -name ldas_mdc
  -password beowulf
  -email mpigroup@gravity.phys.uwm.edu
}
{
  mpiFile2File
  -returnprotocol http://mpiwrapper0201
  -dynlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,1024,1,1,0.00097656,0,0,1.0,1.0,1.0,1.0,1,1,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_medium.ilwd
}
-subject {MPIWRAPPER0201}
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {\n\s|+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

SUMMARY

Known faults encountered – list bug IDs: None

New faults submitted – list bug IDs: None

TEST RESULT

Pass

B.7 06inspiral

The purpose of these tests are to verify functionality of the binary inspiral shared object running under LDAS.

B.7.1 Command Line Arguments

The command line arguments for the inspiral shared object are as follows:

```
/*
 *
 * libinspiral.so should be invoked with the following arguments to wrapperAPI
 *
 * argv[0] = "--filterparams"
 * argv[1] = createRhosqVec      Create a vector of the last filter output
 * argv[2] = numPoints          Number of data points in a segment
 * argv[3] = numSegments        Number of overlapping data segments
 * argv[4] = numChisqBins        Number of frequency bands for chisq veto
 * argv[5] = deltaT              Sampling interval
 * argv[6] = overlap            Overlap between segments (# of points)
 * argv[7] = invSpecTrunc        Duration of inverse spectrum in time domain
 * argv[8] = fLow                Low frequency cut-off in inverse spectrum
 * argv[9] = rhosqThreshold      thresholds for SNR
 * argv[10] = chisqThreshold     thresholds for chisqr
 * argv[11] = dynRange           dynamic range scaling for condition data
 * argv[12] = numTplts           number of templates
 * argv[13] = numCoarseExch      Number of coarse templates to exchange
 * argv[14] = (m1,m2/...)        list of templates
 */
```

B.7.2 Test Check Program

This suite of tests uses a small c program `getStats` to check the output. written by the shared object. When run without arguments, this program reads an ILWD from standard input. It ignores the first 12 lines of input; the 13th line must begin with whitespace (any amount) followed by the tag:

```
<real_4 dims='#' ... >
```

where # is the number of data in the vector. The program reads in the vector, and prints out its maximum and minimum values, mean, and standard deviation (unbiased sample estimate). An example run of this program is shown.

```
[duncan@m81 result]\dir cat output_2.ilwd | ./getStats
Reading 1048576 data... finished.
Minimum   = 0.000000
Maximum   = 0.188205
Mean      = 0.005851
Std. Dev. = 0.023078
```

The source code and a makefile can be found in

```
./.../ldasmdc/mpi/test/06inspiral/input/results/getStats.c
```

```
/******
Program: getStats.c
Usage: getStats [ -h | --help ]

When run without arguments, this program reads an ILWD from standard input.
It ignores the first 12 lines of input; the 13th line must begin with
whitespace (any amount) followed by the tag:

<real_4 dims='#' ... >

where # is the number of data in the vector. The program reads in the
vector, and prints out its maximum and minimum values, mean, and standard
deviation (unbiased sample estimate).

$Id: getStats.c,v 1.2 2001/01/17 02:18:34 duncan Exp $
*****/

#include <math.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
```

```

int
main( int argc, char **argv )
{
    int i = 0;          /* an index */
    int c = 0;          /* current input character as an int */
    int length;        /* number of data */
    float *data = NULL; /* data array */
    float max, min, mean, stddev; /* statistics */

    /* Print usage or help message. */
    if ( argc > 1 ) {
        fprintf( stderr, "Usage: %s [ -h | --help ]\n", *argv );
        if ( ( !strcmp( argv[1], "-h" ) ) ||
              ( !strcmp( argv[1], "--help" ) ) ) )
            puts( "\n"
                  "When run without arguments, this program reads an ILWD"
                  " from standard input.\n"
                  "It ignores the first 12 lines of input; the 13th line"
                  " must begin with\n"
                  "whitespace (any amount) followed by the tag:\n\n"
                  "\t<real_4 dims='#' ... >\n\n"
                  "where # is the number of data in the vector. The"
                  " program reads in the\n"
                  "vector, and prints out its maximum and minimum values,"
                  " mean, and standard\n"
                  "deviation (unbiased sample estimate).");
        return 1;
    }

    /* Drop first 12 lines. */
    i = 12;
    while ( i-- ) {
        while ( c != (int)( '\n' ) )
            c = getchar();
        c = getchar();
    }

    /* Get number of datapoints, and skip to end of tag. */
    scanf( "%*lreal_4 dims='%i'", &length );
    while ( c != (int)( '>' ) )
        c = getchar();
    printf( "Reading %d data.. ", length );
    fflush( stdout );

    /* Allocate and read data. */
    if ( !( data = (float *)malloc( length*sizeof(float) ) ) )
        return 1;
    for ( i = 0; i < length; i++ )
        scanf( "%f", data + i );
    printf( "finished.\n" );

    /* Compute max, min, and mean. */
    max = min = mean = data[0];
    for ( i = 1; i < length; i++ ) {
        float datum = data[i];
        mean += datum;
        if ( min > datum )
            min = datum;
        if ( max < datum )
            max = datum;
    }
    mean /= (float)( length );
    printf( "Minumum   = %f\n"
           "Maximum     = %f\n"
           "Mean        = %f\n", min, max, mean );

    /* Compute standard deviation (unbiased sample estimator). */
    stddev = 0.0;
    for ( i = 0; i < length; i++ ) {
        float diff = data[i] - mean;
        stddev += diff*diff;
    }
    stddev = sqrt( stddev / ( (float)( length - 1 ) ) );
    printf( "Std. Dev. = %f\n", stddev );

    return 0;
}

```

B.7.3 INSPIRAL01**Test Case:** INSPIRAL01**Purpose:** Verify that LDAS can run a binary inspiral search using the inspiral shared object and trivial input data. Successively longer trivial input files should be filtered by the shared object against one template.**Tester:** [Duncan Brown](#)**Test machine:** [ldas-test](#)**Date (mm/dd/yy):** [01/19/2001](#) **Time:** 14:15**ENVIRONMENT AND PREREQUISITES**

This test is executed through the user `ldas_mdc` on LDAS. It depends on the successful completion all test checklists in subsections B.1–B.6.

The following input data is required for this test:

1. `trivial_short.ilwd` $n = 2$
2. `trivial_medium.ilwd` $n = 2^{10}$
3. `trivial_long.ilwd` $n = 2^{20}$
4. `trivial_huge.ilwd` $n = 2^{23}$

The trivial input files should contain one data segment of n points. This consists of three channels of data:

1. `ifodmro` of length n , sample rate 1024Hz, constant value of zero
2. `spectrum` of length $\frac{n}{2} + 1$, constant value of unity
3. `response` of length $\frac{n}{2} + 1$, constant value of unity

The input data should be stored in `.../ldasmdc/mpi/test/06inspiral/input/`

PROCEDURE

Run the LDAS user command scripts listed in the tests below. Inspect the job log files and verify that job execution completed without errors. For each LDAS job id `nnn` there should be two output files:

1. `NORMALnnn_0.txt` should contain the process information in `ilwd` format written by the wrapper.
2. `NORMALnnn_0.txt` should contain the output of the filter produced by the inspiral shared object.

Ensure that these output files have been created. Run the file containing the filter output through the `get-Stats` program and ensure that the minimum, maximum, mean and standard deviation are all 0.0

1. $n = 2$

```
.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0101.tclsh
```

```
#!/lscg/bin/tclsh
#
# $Id: INSPIRAL0101.tclsh,v 1.11 2001/02/05 07:09:19 duncan Exp $
set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
```

```

-dynlib /ldcg/lib/lalwrapper/libinspiral.so
-filterparams (1,2,1,1,0.00097656,0,0,1.0,1.0,1.0,1.0,1,1,(1.0,1.0))
-inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_short.ilwd
-subject INSPIRAL0101
}"
set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass2. $n = 2^{10}$

```

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0102.tclsh

```

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0102.tclsh,v 1.10 2001/02/05 07:09:19 duncan Exp $
set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,1024,1,1,0.00097656,0,0,1.0,1.0,1.0,1.0,1,1,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_medium.ilwd
  -subject INSPIRAL0102
}"
set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass3. $n = 2^{20}$

```

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0103.tclsh

```

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0103.tclsh,v 1.10 2001/02/05 07:09:19 duncan Exp $
set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,1048576,1,1,0.00097656,0,0,1.0,1.0,1.0,1.0,1,1,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_long.ilwd
  -subject INSPIRAL0103
}"
set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass4. $n = 2^{23}$

```

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0104.tclsh

```

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0104.tclsh,v 1.12 2001/02/05 07:09:20 duncan Exp $
set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libinspiral.so

```

```
-filterparams (1,8388608,1,1,0.00097656,0,0,1.0,1.0,1.0,1.0,1,1,(1.0,1.0))  
-inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_huge.ilwd  
-subject INSPiRAL0104  
}"  
  
set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]  
regsub -all -- {[\\n\\s]+} $cmd { } cmd  
puts $sid $cmd  
flush $sid  
puts [ read $sid ]  
close $sid
```

Pass

SUMMARY

Known faults encountered – list bug IDs: None

New faults submitted – list bug IDs: None

TEST RESULT

Pass

B.7.4 INSPIRAL02**Test Case:** INSPIRAL02**Purpose:** Verify that the inspiral shared object correctly rejects improperly formed filter parameters, the failure is correctly logged by LDAS and the job cleanly shutdown.**Tester:** Duncan Brown**Test machine:** ldas-dev**Date (mm/dd/yy):** 01/19/2001 **Time:** 14:15**ENVIRONMENT AND PREREQUISITES**

This test is executed through the user `ldas_mdc` on LDAS. It depends on the successful completion all test checklists in subsections B.1–B.6 and POWER01.

The following input data is required for this test:

1. `trivial_short.ilwd` $n = 2$

The trivial input files should contain one data segment of n points. This consists of three channels of data:

1. `ifodmro` of length n , sample rate 1024Hz, constant value of zero
2. `spectrum` of length $\frac{n}{2} + 1$, constant value of unity
3. `response` of length $\frac{n}{2} + 1$, constant value of unity

The input data should be stored in `.../ldasmdc/mpi/test/06inspiral/input/`

PROCEDURE

Run the LDAS user command scripts listed in the tests below. Inspect the job log files and verify that inspiral shared object rejected the incorrect filter parameter. Ensure that job exited cleanly.

1. Wrong number of arguments

- (a) Too few parameters

```

.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0201a.tclsh

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0201a.tclsh,v 1.6 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  {
    mpiFile2File
    -dynlib /ldcg/lib/lalwrapper/libinspiral.so
    -filterparams (0,2,1,1)
    -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_short.ilwd
    -subject INSPIRAL0201a
  }
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- [{"\n\s"}] $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

- (b) Too many parameters

```

.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0201b.tclsh

```

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0201b.tclsh,v 1.6 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,2,1,1,0.00097656,0,0,1.0,1.0,1.0,1.0,1,1,(1.0,1.0),1,2,3,4,5)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_short.ilwd
  -subject INSPIRAL0201b
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

2. Invalid numPoints: Data segment length is zero or negative

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0202.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0202.tclsh,v 1.9 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,-2,1,1,0.00097656,0,0,1.0,1.0,1.0,1.0,1,1,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_short.ilwd
  -subject INSPIRAL0202
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

3. Invalid numSegments: Number of data segments is zero or negative

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0203.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0203.tclsh,v 1.9 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,2,-1,1,0.00097656,0,0,1.0,1.0,1.0,1.0,1,1,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_short.ilwd
  -subject INSPIRAL0203
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

4. Invalid numChisqBins: Number of chi squared bins is zero or negative

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0204.tclsh


```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0204.tclsh,v 1.9 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,2,1,-1,0.00097656,0,0,1.0,1.0,1.0,1.0,1,1,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_short.ilwd
  -subject INSPIRAL0204
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

5. Invalid deltaT: deltaT is zero or negative

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0205.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0205.tclsh,v 1.11 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,2,1,1,-1.23456789,0,0,1.0,1.0,1.0,1.0,1,1,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_short.ilwd
  -subject INSPIRAL0205
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

6. Invalid overlap: Segment overlap is negative

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0206.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0206.tclsh,v 1.10 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,2,1,1,0.00097656,-1,0,1.0,1.0,1.0,1.0,1,1,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_short.ilwd
  -subject INSPIRAL0206
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

7. Invalid invSpecTrunc: Duration of inverse spectrum in time domain is negative

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0207.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0207.tclsh,v 1.10 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,2,1,1,0.00097656,0,-1,1.0,1.0,1.0,1.0,1,1,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_short.ilwd
  -subject INSPIRAL0207
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

8. Invalid fLow: Inverse spectrum low frequency cutoff is negative

```

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0208.tclsh

```

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0208.tclsh,v 1.9 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,2,1,1,0.00097656,0,0,-1.0,1.0,1.0,1.0,1,1,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_short.ilwd
  -subject INSPIRAL0208
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

9. Invalid rhoSqThreshold: RhoSq threshold is negative or zero

```

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0209.tclsh

```

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0209.tclsh,v 1.9 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,2,1,1,0.00097656,0,0,1.0,-1.0,1.0,1.0,1,1,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_short.ilwd
  -subject INSPIRAL0209
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

10. Invalid chiSqThreshold: ChiSq threshold is negative or zero

```

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0210.tclsh

```

```
#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0210.tclsh,v 1.9 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,2,1,1,0.00097656,0,0,1.0,1.0,-1.0,1.0,1,1,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_short.ilwd
  -subject INSPIRAL0210
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

11. Invalid dynRange: Dynamic range is negative

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0211.tclsh

```
#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0211.tclsh,v 1.4 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,2,1,1,0.00097656,0,0,1.0,1.0,1.0,-1.0,1,1,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_short.ilwd
  -subject INSPIRAL0212
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

12. Incorrectly formed template parameters.

(a) Invalid numTmplts: Number of templates is incorrect

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0212a.tclsh

```
#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0212a.tclsh,v 1.4 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,2,1,1,0.00097656,0,0,1.0,1.0,1.0,1.0,1,0,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_short.ilwd
  -subject INSPIRAL0212a
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

(b) Invalid numCoarseExch: Number of templates in coarse exchange is zero or negative

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0212b.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0212b.tclsh,v 1.4 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,2,1,1,0.00097656,0,0,1.0,1.0,1.0,0,1,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_short.ilwd
  -subject INSPIRAL0212b
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

- (c) Incorrect number of templates in list: Error parsing template string

```

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0212c.tclsh

```

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0212c.tclsh,v 1.4 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,2,1,1,0.00097656,0,0,1.0,1.0,1.0,1.0,1,1,(1.0/1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_short.ilwd
  -subject INSPIRAL0212c
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

- (d) Missing mass parameter: Error parsing template string

```

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0212d.tclsh

```

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0212d.tclsh,v 1.4 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,2,1,1,0.00097656,0,0,1.0,1.0,1.0,1.0,1,1,(1.0,1.0/1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_long.ilwd
  -subject INSPIRAL0212d
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Fail

- (e) Missing mass parameter: Error parsing template string

```

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0212e.tclsh

```

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0212e.tclsh,v 1.4 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so

```

```

-filterparams (1,2,1,1,0.00097656,0,0,1.0,1.0,1.0,1.0,1,2,(1.0,1.0/,1.0))
-inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_short.ilwd
-subject INSPIRAL0212e
}"
set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

(f) Rubbish in template string: Error parsing template string

```
./.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0212f.tclsh
```

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0212f.tclsh,v 1.4 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,2,1,1,0.00097656,0,0,1.0,1.0,1.0,1.0,1,1,(1.0,1.0/crap))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_short.ilwd
  -subject INSPIRAL0212f
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

(g) No templates in parameters or input: No template bank has been constructed

```
./.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0212g.tclsh
```

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0212g.tclsh,v 1.5 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,2,1,1,0.00097656,0,0,1.0,1.0,1.0,1.0,1)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_short.ilwd
  -subject INSPIRAL0212g
  -np 4
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Fail**SUMMARY****Known faults encountered – list bug IDs:** None**New faults submitted – list bug IDs:** Tests 12d and 12g cause the shared object to raise a SIGSEGV. This should be investigated and fixed.**TEST RESULT****Conditional Pass**

B.7.5 INSPIRAL03**Test Case:** INSPIRAL03**Purpose:** Verify that LDAS can run a binary inspiral search using the inspiral shared object and trivial input data with multiple templates.**Tester:** Duncan Brown**Test machine:** ldas-dev**Date (mm/dd/yy):** 01/19/2001 **Time:** 14:15**ENVIRONMENT AND PREREQUISITES**

This test is executed through the user `ldas_mdc` on LDAS. It depends on the successful completion all tests in section 00prep and test INSPIRAL01.

The following input data is required for this test:

1. `trivial_long.ilwd` $n = 2^{20}$
2. `trivial_long_bank.ilwd` $n = 2^{20}$

The trivial input files should contain one data segment of n points. This consists of three channels of data:

1. `ifodmro` of length n , sample rate 1024Hz, constant value of zero
2. `spectrum` of length $\frac{n}{2} + 1$, constant value of unity
3. `response` of length $\frac{n}{2} + 1$, constant value of unity

The file `trivial_long_bank.ilwd` also contains a template bank in ILWD format. The input data should be stored in `./.../ldasmdc/mpi/test/06inspiral/input/`

PROCEDURE

Run the LDAS user command scripts listed in the tests below. Inspect the job log files and ensure that each command executed successfully without errors. No output is produced by any of the jobs.

1. `numTplts = 2, numCoarseExch = 1`

```
./.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0301.tclsh
```

```
#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0301.tclsh,v 1.12 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (0,1048576,1,1,0.00097656,0,0,1.0,1.0,1.0,1.0,1,2,(1.4,1.4/1.2,1,2))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/trivial_long.ilwd
  -np 3
  -subject INSPIRAL0301
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\n\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

B.7.6 INSPIRAL04

Test Case:	INSPIRAL04
Purpose:	Verify that LDAS can run a binary inspiral search using the inspiral shared object and find known inspiral signals at the correct effective distance using a single template.
Tester:	Isaac Salzman
Test machine:	tarazed.ligo.caltech.edu
Date (mm/dd/yy):	01/19/01
Time:	15:00

ENVIRONMENT AND PREREQUISITES

This test is executed through the user `ldas_mdc` on LDAS. It depends on the successful completion all test checklists in subsections B.1–B.6 and test INSPIRAL01.

The following input data is required for this test:

1. `c_1.40_1.40_11.00.ilwd` $1.400000M_{\odot}, 1.400000M_{\odot}$ at 11.000000 Mpc
2. `c_1.85_1.51_13.79.ilwd` $1.845162M_{\odot}, 1.508545M_{\odot}$ at 13.794533 Mpc

The input files should contain one data segment of $n = 2^{20}$ points. This consists of three channels of data:

1. `ifodmro` of length n , sample rate 1024Hz, whitened inspiral chirp of an m_1, m_2 solar mass binary system at an effective distance of d Mpc
2. `spectrum` of length $\frac{n}{2} + 1$, constant value of 2048.0
3. `response` of length $\frac{n}{2} + 1$, simulated LIGO I response function

The input data should be stored in `.../ldasmdc/mpi/test/06inspiral/input/`

PROCEDURE

Run the LDAS user command scripts listed in the tests below. Inspect the job log files and ensure that each command executed successfully without errors.

Inspect the result files produced and ensure that for each inspiral event the values of $\rho_{\max}^2, \chi_{\min}^2, d_{\text{eff}}$ and the GPS inspiral time correspond to those in the files

1. `c_1.40_1.40_11.00.out`
2. `c_1.85_1.51_13.79.out`

in the result directory `.../ldasmdc/mpi/test/06inspiral/result/`

1. $m_1 = 1.400000M_{\odot}, m_2 = 1.400000M_{\odot}, d = 11.000000$ Mpc
`.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0401.tclsh`

```
#!/lscg/bin/tclsh
#
# $Id: INSPIRAL0401.tclsh,v 1.10 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /lscg/lib/lalwrapper/libinspiral.so
  -filterparams (1,1048576,1,8,0.0009765625,0,0,40.0,64.0,3.0,69.0,1,1,(1.400000,1.400000))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/c_1.40_1.40_11.00.ilwd
  -subject INSPIRAL0401
```

```

}"
set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

2. $m_1 = 1.845162M_\odot$, $m_2 = 1.508545M_\odot$, $d = 13.794533$ Mpc

`../../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0402.tclsh`

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0402.tclsh,v 1.11 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,1048576,1,8,0.0009765625,0,0,40.0,64.0,3.0,69.0,1,1,(1.845162,1.508545))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/c_1.85_1.51_13.79.ilwd
  -subject INSPIRAL0402
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass**SUMMARY**

Known faults encountered – list bug IDs: The shared object reports GPS seconds incorrectly, but GPS nanoseconds correctly. The fix for this bug is known and should be implemented.

New faults submitted – list bug IDs: None

TEST RESULT**Pass**

B.7.7 INSPIRAL05

Test Case: INSPIRAL05

Purpose: Verify that LDAS can run a binary inspiral search using the inspiral shared object and white Gaussian noise input data.

Tester: [Isaac Salzman](#)

Test machine: tarazed.ligo.caltech.edu

Date (mm/dd/yy): 01/19/01 **Time:** 15:00

ENVIRONMENT AND PREREQUISITES

This test is executed through the user `ldas_mdc` on LDAS. It depends on the successful completion all test checklists in subsections B.1–B.6 and test INSPIRAL01.

The following input data is required for this test:

1. `n_64_1024Hz.ilwd` $\sigma^2 = 64$, sample rate 1024Hz
2. `n_128_1024Hz.ilwd` $\sigma^2 = 128$, sample rate 1024Hz
3. `n_64_2048Hz.ilwd` $\sigma^2 = 64$, sample rate 2048Hz
4. `n_64_8192Hz.ilwd` $\sigma^2 = 64$, sample rate 8192Hz
5. `n_128_8192Hz.ilwd` $\sigma^2 = 128$, sample rate 8192Hz
6. `n_64_7369Hz.ilwd` $\sigma^2 = 64$, sample rate 7369Hz
7. `n_1048576_1024Hz.ilwd` $\sigma^2 = 2^{20}$, sample rate 1024Hz

The input files should contain one data segment of $n = 2^{20}$ points. This consists of three channels of data:

1. `ifodmro` of length n , white Gaussian noise of given variance and sample rate.
2. `spectrum` of length $\frac{n}{2} + 1$, constant value of $2\sigma^2 \delta t$.
3. `response` of length $\frac{n}{2} + 1$, constant value of 1.

The input data should be stored in `/. . . /ldasmdc/mpi/test/06inspiral/input/`

PROCEDURE

Run the LDAS user command scripts listed in the tests below. Inspect the job log files and ensure that each command executed successfully without errors. For each LDAS job id `nnn` there should be two output files:

1. `NORMALnnn_0.txt` should contain the process information in `ilwd` format written by the wrapper.
2. `NORMALnnn_0.txt` should contain the output of the filter produced by the inspiral shared object.

Ensure that these output files have been created. Run the file containing the filter output through the `get-stats` and that the results match the values in the results files

1. `n_64_1024Hz.stats`
2. `n_128_1024Hz.stats`

3. n_64_2048Hz.stats
4. n_64_8192Hz.stats
5. n_128_8192Hz.stats
6. n_64_7369Hz.stats
7. n_1048576_1024Hz.stats

in the result directory `../../ldasmdc/mpi/test/06inspiral/result/`

1. $\sigma^2 = 64$, sample rate 1024Hz

`../../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0501.tclsh`

```
#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0501.tclsh,v 1.10 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,1048576,1,1,0.0009765625,0,0,40.0,100.0,1.0,1.0,1,1,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/n_64_1024Hz.ilwd
  -subject INSPIRAL0501
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

2. $\sigma^2 = 128$, sample rate 1024Hz

`../../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0502.tclsh`

```
#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0502.tclsh,v 1.10 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,1048576,1,1,0.00097656,0,0,40.0,100.0,1.0,1.0,1,1,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/n_128_1024Hz.ilwd
  -subject INSPIRAL0502
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

3. $\sigma^2 = 64$, sample rate 2048Hz

`../../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0503.tclsh`

```
#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0503.tclsh,v 1.10 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
```

```

mpiFile2File
-dynlib /ldcg/lib/lalwrapper/libinspiral.so
-filterparams (1,1048576,1,1,0.00048828,0,0,40.0,100.0,1.0,1.0,1,1,(1.0,1.0))
-inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/n_64_2048Hz.ilwd
-subject INSPIRAL0503
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\n\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass4. $\sigma^2 = 64$, sample rate 8192Hz

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0504.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0504.tclsh,v 1.10 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,1048576,1,1,0.00012207,0,0,40.0,100.0,1.0,1.0,1,1,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/n_64_8192Hz.ilwd
  -subject INSPIRAL0504
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\n\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass5. $\sigma^2 = 128$, sample rate 8192Hz

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0505.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0505.tclsh,v 1.10 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,1048576,1,1,0.00012207,0,0,40.0,100.0,1.0,1.0,1,1,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/n_128_8192Hz.ilwd
  -subject INSPIRAL0505
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\n\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass6. $\sigma^2 = 64$, sample rate 7369Hz

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0506.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0506.tclsh,v 1.10 2001/02/05 07:09:20 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File

```

```

-dynlib /ldcg/lib/lalwrapper/libinspiral.so
-filterparams (1,1048576,1,1,0.00013570,0,0,40.0,100.0,1.0,1.0,1,1,(1.0,1.0))
-inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/n_64_7369Hz.ilwd
-subject INSPIRAL0506
}"
set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\n\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass7. $\sigma^2 = 2^{20}$, sample rate 1024Hz

/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0507.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: INSPIRAL0507.tclsh,v 1.11 2001/02/05 07:09:20 duncan Exp $
set cmd "ldasJob
{
-name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,1048576,1,1,0.00097656,0,0,40.0,100.0,1.0,69.0,1,1,(1.0,1.0))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/n_1048576_1024Hz.ilwd
  -subject INSPIRAL0507
}"
set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\n\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass**SUMMARY****Known faults encountered – list bug IDs:** None**New faults submitted – list bug IDs:** None**TEST RESULT****Pass**

B.7.8 INSPIRAL06

Test Case:	INSPIRAL06
Purpose:	Verify that LDAS can run a binary inspiral search using the inspiral shared object and find known inspiral signals in Gaussian noise at the correct effective distance using a single template.
Tester:	Isaac Salzman
Test machine:	tarazed.ligo.caltech.edu
Date (mm/dd/yy):	01/19/01
Time:	15:00

ENVIRONMENT AND PREREQUISITES

This test is executed through the user `ldas_mdc` on LDAS. It depends on the successful completion all test checklists in subsections B.1–B.6 and test INSPIRAL01.

The following input data is required for this test:

1. `cn_1.40_1.40_20.00.ilwd` $1.400000M_{\odot}, 1.400000M_{\odot}$ at 11.000000 Mpc
2. `cn_1.85_1.51_13.79.ilwd` $1.845162M_{\odot}, 1.508545M_{\odot}$ at 13.794533 Mpc

The input files should contain one data segment of $n = 2^{20}$ points. This consists of three channels of data:

1. `ifodmro` of length n , sample rate 1024Hz, whitened inspiral chirp of an m_1, m_2 solar mass binary system at an effective distance of d Mpc in Gaussian noise.
2. `spectrum` of length $\frac{n}{2} + 1$, constant value of 2048.0
3. `response` of length $\frac{n}{2} + 1$, simulated LIGO I response function

The input data should be stored in `.../ldasmdc/mpi/test/06inspiral/input/`

PROCEDURE

Run the LDAS user command scripts listed in the tests below. Inspect the job log files and ensure that each command executed successfully without errors.

Inspect the result files produced and ensure that for each inspiral event the values of $\rho_{\max}^2, \chi_{\min}^2, d_{\text{eff}}$ and the GPS inspiral time correspond to those in the files

1. `cn_1.40_1.40_20.00.out`
2. `cn_1.85_1.51_13.79.out`

in the result directory `.../ldasmdc/mpi/test/06inspiral/result/`

1. $m_1 = 1.400000M_{\odot}, m_2 = 1.400000M_{\odot}, d = 11.000000$ Mpc
`.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0601.tclsh`

```
#!/ldcg/bin/tclsh
set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (0,1048576,1,8,9.765625e-04,0,0,40.0,64.0,30.0,69.0,1,1,(1.400000,1.400000))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/cn_1.40_1.40_20.00.ilwd
  -subject INSPIRAL0601
```

```
}"
set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {\n\s+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

2. $m_1 = 1.845162M_\odot, m_2 = 1.508545M_\odot, d = 13.794533 \text{ Mpc}$

```
/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0602.tclsh
```

```
#!/ldcg/bin/tclsh

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (0,1048576,1,8,9.765625e-04,0,0,40.0,64.0,30.0,69.0,1,1,(1.845162,1.508545))
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/cn_1.85_1.51_13.79.ilwd
  -subject INSPIRAL0602
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {\n\s+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass**SUMMARY**

Known faults encountered – list bug IDs: The shared object reports GPS seconds incorrectly, but GPS nanoseconds correctly. The fix for this bug is known and should be implemented.

New faults submitted – list bug IDs: None

TEST RESULT**Pass**

B.7.9 INSPIRAL07

Test Case:	INSPIRAL07
Purpose:	Verify that LDAS can run a binary inspiral search using the inspiral shared object and find known inspiral signals in Gaussian noise at the correct effective distance using a bank of templates.
Tester:	Duncan Brown
Test machine:	ldas-dev
Date (mm/dd/yy):	01/19/2001
Time:	14:15

ENVIRONMENT AND PREREQUISITES

This test is executed through the user `ldas_mdc` on LDAS. It depends on the successful completion all test checklists in subsections B.1–B.6 and test INSPIRAL01.

The following input data is required for this test:

1. `cn_4.16_0.93_5.79_bank.i155841M⊙,0.932539M⊙ at 5.785604 Mpc`
2. `cn_3.79_3.33_4.61_bank.i1787266M⊙,3.327534M⊙ at 4.610762 Mpc`
3. `cn_1.51_1.10_17.43_bank.i105872M⊙,1.096999M⊙ at 17.428312 Mpc`

The input files should contain one data segment of $n = 2^{20}$ points. This consists of four elements:

1. `ifodmro` of length n , sample rate 1024Hz, whitened inspiral chirp of an m_1, m_2 solar mass binary system at an effective distance of d Mpc in Gaussian noise.
2. `spectrum` of length $\frac{n}{2} + 1$, constant value of 2048.0
3. `response` of length $\frac{n}{2} + 1$, simulated LIGO I response function
4. `tmpltBank` a bank of 129 templates in ILWD format

The input data should be stored in `.../ldasmdc/mpi/test/06inspiral/input/`

PROCEDURE

Run the LDAS user command scripts listed in the tests below. Inspect the job log files and ensure that each command executed successfully without errors.

Inspect the result files produced and ensure that for each inspiral event the values of ρ_{\max}^2 , χ_{\min}^2 , d_{eff} and the GPS inspiral time correspond to those in the files

1. `cn_4.16_0.93_5.79.out`
2. `cn_3.79_3.33_4.61.out`
3. `cn_1.51_1.10_17.43.out`

in the result directory `.../ldasmdc/mpi/test/06inspiral/result/`

1. $m_1 = 4.155841M_{\odot}, m_2 = 0.932539M_{\odot}, d = 5.785604$ Mpc
`.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0701.tclsh`

```
#!/lscg/bin/tclsh

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /lscg/lib/lalwrapper/libinspiral.so
  -filterparams (0,1048576,1,1,9.765625e-04,0,0,40.0,3000.0,50.0,69.0,16)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/cn_4.16_0.93_5.79_bank.ilwd
  -subject INSPIRAL0701
  -np 8
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Fail

Notes: **Incorrectly reports $\chi^2 = 0$ for the event.**

- $m_1 = 3.787266M_\odot, m_2 = 3.327534M_\odot, d = 4.610762$ Mpc
/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0702.tclsh

```
#!/lscg/bin/tclsh

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /lscg/lib/lalwrapper/libinspiral.so
  -filterparams (0,1048576,1,8,9.765625e-04,0,0,40.0,3000.0,30.0,69.0,16)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/cn_3.79_3.33_4.61_bank.ilwd
  -subject INSPIRAL0702
  -np 4
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

- $m_1 = 1.505872M_\odot, m_2 = 1.096999M_\odot, d = 17.428312$ Mpc
/.../ldasmdc/mpi/test/06inspiral/command/INSPIRAL0703.tclsh

```
#!/lscg/bin/tclsh

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /lscg/lib/lalwrapper/libinspiral.so
  -filterparams (0,1048576,1,8,9.765625e-04,0,0,40.0,150.0,4.0,69.0,16)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/06inspiral/input/cn_1.51_1.10_17.43_bank.ilwd
  -np 4
  -subject INSPIRAL0703
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass**SUMMARY**

Known faults encountered – list bug IDs: None

New faults submitted – list bug IDs: Test 01 reports $\chi^2 = 0$ for the event. All other values are correct. This could be a bug in the shared object and should be investigated.

TEST RESULT

Conditional Pass

B.8 07power

These tests verify the functionality of the excess power shared object running under LDAS.

B.8.1 Command Line Arguments

The command line arguments for the inspiral shared object are as follows:

```

/*
 *
 * libpower.so should be invoked with the following arguments to wrapperAPI
 *
 * argv[0] = "--filterparams"
 * argv[1] = numPoints      Number of data points in a segment (2 * 2^p + 2)
 * argv[2] = numSegments   Number of overlapping segments
 * argv[3] = overlap       Number of points overlap between segments
 * argv[4] = overlapFactor Amount of overlap between neighboring TF tiles
 * argv[5] = minFreqBins   Smallest extent in freq of TF tiles to search
 * argv[6] = minTimeBins   Smallest extent in time of TF tiles to search
 * argv[7] = flow          Lowest frequency in Hz to be searched
 * argv[8] = deltaF        Frequency resolution of first TF plane
 * argv[9] = length        Length (N_F) of first TF plane (with N_T = 1)
 * argv[10] = numSigmaMin  threshold number of sigma
 * argv[11] = alphaDefault default alpha value for tiles with sigma < numSigmaMin
 * argv[12] = segDutyCycle Number of segments sent to slave for analysis
 * argv[13] = alphaThreshold Identify events with alpha less than this
 *
 */

```

B.8.2 POWER01

Test Case: POWER01

Purpose: Verify that the power shared object correctly rejects improperly formed filter parameters, the failure is correctly logged by LDAS and the job cleanly shutdown.

Tester: Isaac Salzman

Test machine: tarazed.ligo.caltech.edu

Date (mm/dd/yy): 01/18/01 **Time:** 15:00

ENVIRONMENT AND PREREQUISITES

This test is executed through the user `ldas_mdc` on LDAS. It depends on the successful completion all test checklists in subsections B.1–B.6.

The following input data is required for this test:

1. `n1026s1r1024.ilwd` $N = 1026$

This input file should contain one data segment of N points. This consists of three channels of data:

1. `ifodmro` of length N , sample rate 1024Hz, constant value of zero
2. `spectrum` of length $\frac{N}{2} + 1$, constant value of unity
3. `response` of length $\frac{N}{2} + 1$, constant value of unity

The input data should be stored in `.../ldasmdc/mpi/test/07power/input/`

PROCEDURE

Run the LDAS user command scripts listed in the tests below. Inspect the job log files and verify that power shared object rejected the incorrect filter parameter. Ensure that job exited cleanly.

1. Wrong number of arguments

- (a) Too few parameters

```

.../ldasmdc/mpi/test/07power/command/POWER0101a.tclsh

#!/ldcg/bin/tclsh
#
# $Id: POWER0101a.tclsh,v 1.3 2001/01/18 23:51:25 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1,1026,1,513,3,1,1,1,0,1,0,32,3,0,0.5,1,0.001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026s1r1024.ilwd
  -subject POWER0101a
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- [{"\n\s}+] $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

- (b) Too many parameters

```

.../ldasmdc/mpi/test/07power/command/POWER0101b.tclsh

```

```

#!/ldcg/bin/tclsh
#
# $Id: POWER0101b.tclsh,v 1.2 2001/01/18 23:55:25 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1,1026,1,513,3,1,1,1.0,1.0,32,3.0,0.5,1,0.001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026s1r1024.ilwd
  -subject POWER0101b
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

2. Invalid numPoints: Data segment length is zero or negative

/.../ldasmdc/mpi/test/07power/command/POWER0102.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: POWER0102.tclsh,v 1.3 2001/01/18 23:51:25 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (-1,1,513,3,1,1,1.0,1.0,32,3.0,0.5,1,0.001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026s1r1024.ilwd
  -subject POWER0102
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

3. Invalid numSegments: Number of data segments is zero or negative

/.../ldasmdc/mpi/test/07power/command/POWER0103.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: POWER0103.tclsh,v 1.3 2001/01/18 23:51:25 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,-1,513,3,1,1,1.0,1.0,32,3.0,0.5,1,0.001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026s1r1024.ilwd
  -subject POWER0103
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

4. Invalid overlap: overlap of data segments is negative

/.../ldasmdc/mpi/test/07power/command/POWER0104.tclsh

```
#!/ldcg/bin/tclsh
#
# $Id: POWER0104.tclsh,v 1.4 2001/01/19 01:03:50 isalzman Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,-1,3,1,1,1.0,1.0,32,3.0,0.5,1,0.001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026s1r1024.ilwd
  -subject POWER0104
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

5. Invalid overlapFactor: Overlap of TF tiles is negative

```
../../ldasmdc/mpi/test/07power/command/POWER0105.tclsh
```

```
#!/ldcg/bin/tclsh
#
# $Id: POWER0105.tclsh,v 1.3 2001/01/18 23:51:25 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,-1,1,1,1.0,1.0,32,3.0,0.5,1,0.001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026s1r1024.ilwd
  -subject POWER0105
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

6. Invalid minFreqBins: Smallest extent in time of TF tile is zero

```
../../ldasmdc/mpi/test/07power/command/POWER0106.tclsh
```

```
#!/ldcg/bin/tclsh
#
# $Id: POWER0106.tclsh,v 1.3 2001/01/18 23:51:26 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,1,-1,1.0,1.0,32,3.0,0.5,1,0.001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026s1r1024.ilwd
  -subject POWER0106
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

7. Invalid minTimeBins: Smallest extent in frequency of TF tile is zero

```
../../ldasmdc/mpi/test/07power/command/POWER0107.tclsh
```

```

#!/ldcg/bin/tclsh
#
# $Id: POWER0107.tclsh,v 1.3 2001/01/18 23:51:26 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,-1,1,1.0,1.0,32,3.0,0.5,1,0.001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026s1r1024.ilwd
  -subject POWER0107
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass8. Invalid flow: Lowest frequency to be searched is ≤ 0

/.../ldasmdc/mpi/test/07power/command/POWER0108.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: POWER0108.tclsh,v 1.3 2001/01/18 23:51:26 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,1,1,-1,1.0,32,3.0,0.5,1,0.001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026s1r1024.ilwd
  -subject POWER0108
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass9. Invalid deltaF: Frequency resolution of 1st time-freq plane is ≤ 0

/.../ldasmdc/mpi/test/07power/command/POWER0109.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: POWER0109.tclsh,v 1.3 2001/01/18 23:51:26 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,1,1,1.0,-1,32,3.0,0.5,1,0.001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026s1r1024.ilwd
  -subject POWER0109
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass10. Invalid length: Length (N_F) of first TF plane (with N_T=1) is ≤ 0

/.../ldasmdc/mpi/test/07power/command/POWER0110.tclsh


```
#!/ldcg/bin/tclsh
#
# $Id: POWER0110.tclsh,v 1.3 2001/01/18 23:51:26 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,1,1,1,0,1.0,-1,3.0,0.5,1,0.001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026s1r1024.ilwd
  -subject POWER0110
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass11. Invalid numSigmaMin: Threshold number of sigma is ≤ 1

```
/. . . /ldasmdc/mpi/test/07power/command/POWER0111.tclsh
```

```
#!/ldcg/bin/tclsh
#
# $Id: POWER0111.tclsh,v 1.3 2001/01/18 23:51:26 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,1,1,1,0,1.0,32,0.1,0.5,1,0.001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026s1r1024.ilwd
  -subject POWER0111
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

12. Invalid alphaDefault: Default alpha value is out of range

```
/. . . /ldasmdc/mpi/test/07power/command/POWER0112.tclsh
```

```
#!/ldcg/bin/tclsh
#
# $Id: POWER0112.tclsh,v 1.3 2001/01/18 23:51:26 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,1,1,1,0,1.0,32,3.0,1.2,1,0.001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026s1r1024.ilwd
  -subject POWER0112
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

13. Invalid segDutyCycle: Number of segments sent to slave is zero

```
/. . . /ldasmdc/mpi/test/07power/command/POWER0113a.tclsh
```

```

#!/ldcg/bin/tclsh
#
# $Id: POWER0113.tclsh,v 1.3 2001/01/18 23:51:26 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,1,1,1.0,1.0,32,3.0,0.5,0,0.001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026s1r1024.ilwd
  -subject POWER0113
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

14. Invalid alphaThreshold: The threshold value of alph is negative

/.../ldasmdc/mpi/test/07power/command/POWER0114.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: POWER0114.tclsh,v 1.3 2001/01/18 23:51:26 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,1,1,1.0,1.0,32,3.0,0.5,1,-0.5)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026s1r1024.ilwd
  -subject POWER0114
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

15. Got less data (in the ILWD container) than expected

/.../ldasmdc/mpi/test/07power/command/POWER0115.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: POWER0115.tclsh,v 1.3 2001/01/18 23:51:26 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1032,1,513,3,1,1,1.0,1.0,32,3.0,0.5,1,0.001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026s1r1024.ilwd
  -subject POWER0115
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

SUMMARY

Known faults encountered – list bug IDs: None

New faults submitted – list bug IDs: None

TEST RESULT

Pass

B.8.3 POWER02**Test Case:** POWER02**Purpose:** Verify that LDAS can run excess power search using the power shared object and trivial input data with various filter parameters.**Tester:** Alan Wiseman**Test machine:** ldas-test**Date (mm/dd/yy):** 01/19/01 **Time:** 13:00**ENVIRONMENT AND PREREQUISITES**

This test is executed through the user `ldas_mdc` on LDAS. It depends on the successful completion all test checklists in subsections B.1–B.6 and POWER01.

The following input data is required for this test:

1. `trivial_power.ilwd` $N = 1026$

The trivial input file contains one data segment of N points. This consists of three channels of data:

1. `ifodmro` of length N , sample rate 1024Hz, constant value of zero
2. `spectrum` of length $\frac{N}{2} + 1$, constant value of unity
3. `response` of length $\frac{N}{2} + 1$, constant value of unity

The input data should be stored in `.../ldasmdc/mpi/test/07power/input/`

PROCEDURE

Run the LDAS user command scripts listed in the tests below. Inspect the job log files and ensure that each command executed successfully without errors. No output is produced by any of the jobs.

1. `flow = 1.0, deltaF = 2.0, length = 16`

```
.../ldasmdc/mpi/test/07power/command/POWER0201.tclsh
```

```
#!/dcg/bin/tclsh
#
# $Id: POWER0201.tclsh,v 1.1 2001/01/19 20:54:19 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,1,1,1,0,2,0,16,3,0,0,5,1,0,001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/power_trivial.ilwd
  -subject POWER0201
}"

set sid [ socket ldas-test.ligo.caltech.edu 10001 ]
regsub -all -- {\n\s|+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

2. `flow = 1.0, deltaF = 2.0, length = 64`

```
.../ldasmdc/mpi/test/07power/command/POWER0202.tclsh
```

```

#!/ldcg/bin/tclsh
#
# $Id: POWER0202.tclsh,v 1.1 2001/01/19 20:54:19 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,1,1,1,0,2,0,64,3,0,0,5,1,0,001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/power_trivial.ilwd
  -subject POWER0202
}"

set sid [ socket ldas-test.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

3. flow = 1.0, deltaF = 4.0, length = 16

/.../ldasmdc/mpi/test/07power/command/POWER0203.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: POWER0203.tclsh,v 1.1 2001/01/19 20:54:19 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,1,1,1,0,4,0,16,3,0,0,5,1,0,001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/power_trivial.ilwd
  -subject POWER0203
}"

set sid [ socket ldas-test.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

4. flow = 1.0, deltaF = 8.0, length = 16

/.../ldasmdc/mpi/test/07power/command/POWER0204.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: POWER0204.tclsh,v 1.1 2001/01/19 20:54:20 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,1,1,1,0,8,0,16,3,0,0,5,1,0,001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/power_trivial.ilwd
  -subject POWER0204
}"

set sid [ socket ldas-test.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

5. flow = 10.0, deltaF = 1.0, length = 16

/.../ldasmdc/mpi/test/07power/command/POWER0205.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: POWER0205.tclsh,v 1.1 2001/01/19 20:54:20 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,1,1,10.0,1.0,16,3.0,0.5,1,0.001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/power_trivial.ilwd
  -subject POWER0205
}"

set sid [ socket ldas-test.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

6. flow = 100.0, deltaF = 1.0, length = 16

/.../ldasmdc/mpi/test/07power/command/POWER0206.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: POWER0206.tclsh,v 1.1 2001/01/19 20:54:20 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,1,1,100.0,1.0,16,3.0,0.5,1,0.001)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/power_trivial.ilwd
  -subject POWER0206
}"

set sid [ socket ldas-test.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

SUMMARY

Known faults encountered – list bug IDs: The shared object reports GPS seconds incorrectly, but GPS nanoseconds correctly. The fix for this bug is known and should be implemented.

New faults submitted – list bug IDs: None

TEST RESULT

Pass

B.8.4 POWER03

Test Case: POWER03

Purpose: Verify that LDAS can run excess power search using the power shared object and trivial input data with various filter parameters.

Tester: Alan Wiseman

Test machine: ldas-test

Date (mm/dd/yy): 01/19/01 **Time:** 14:30

ENVIRONMENT AND PREREQUISITES

This test is executed through the user `ldas_mdc` on LDAS. It depends on the successful completion all test checklists in subsections B.1–B.6 and POWER01.

The following input data is required for this test:

1. `n1026s1r1024.ilwd` $N = 1026$

The trivial input file contains one data segment of N points. This consists of three channels of data:

1. `ifodmro` of length N , sample rate 1024Hz, constant value of zero
2. `spectrum` of length $\frac{N}{2} + 1$, constant value of unity
3. `response` of length $\frac{N}{2} + 1$, constant value of unity

The input data should be stored in `./.../ldasmdc/mpi/test/07power/input/`

PROCEDURE

Run the LDAS user command scripts listed in the tests below. Inspect the job log files and ensure that each command executed successfully without errors.

1. `flow = 1.0, deltaF = 2.0, length = 16`
`./.../ldasmdc/mpi/test/07power/command/POWER0301.tclsh`

```
#!/ldcg/bin/tclsh
#
# $Id: POWER0301.tclsh,v 1.1 2001/01/19 21:01:07 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,1,1,1,0,2,0,16,3,0,0.5,1,0.1)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026s1r1024.ilwd
  -subject POWER0301
}"

set sid [ socket ldas-test.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Compare the output to that in the file `./.../test/07power/results/output0301.ilwd`; the `snr`, confidence may only agree to 6 significant figures. The job-id will be different.

Pass

2. flow = 1.0, deltaF = 2.0, length = 64

```
./.../ldasmdc/mpi/test/07power/command/POWER0302.tclsh
```

```
#!/ldcg/bin/tclsh
#
# $Id: POWER0302.tclsh,v 1.1 2001/01/19 21:01:07 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,1,1,1.0,2.0,64,3.0,0.5,1,0.1)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026slr1024.ilwd
  -subject POWER0302
}"

set sid [ socket ldas-test.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Compare the output to that in the file `./test/07power/results/output0302.ilwd`; the snr, confidence may only agree to 6 significant figures. The job-id will be different.

Pass

3. flow = 1.0, deltaF = 4.0, length = 16

```
./.../ldasmdc/mpi/test/07power/command/POWER0303.tclsh
```

```
#!/ldcg/bin/tclsh
#
# $Id: POWER0303.tclsh,v 1.1 2001/01/19 21:01:07 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,1,1,1.0,4.0,16,3.0,0.5,1,0.1)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026slr1024.ilwd
  -subject POWER0303
}"

set sid [ socket ldas-test.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Compare the output to that in the file `./test/07power/results/output0303.ilwd`; the snr, confidence may only agree to 6 significant figures. The job-id will be different.

Pass

4. flow = 1.0, deltaF = 8.0, length = 16

```
./.../ldasmdc/mpi/test/07power/command/POWER0304.tclsh
```

```
#!/ldcg/bin/tclsh
#
# $Id: POWER0304.tclsh,v 1.1 2001/01/19 21:01:07 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,1,1,1.0,8.0,16,3.0,0.5,1,0.1)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026slr1024.ilwd
  -subject POWER0304
}"

set sid [ socket ldas-test.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```


Compare the output to that in the file `../test/07power/results/output0304.ilwd`; the snr, confidence may only agree to 6 significant figures. The job-id will be different.

Pass

5. flow = 10.0, deltaF = 1.0, length = 16

`../ldasmdc/mpi/test/07power/command/POWER0305.tclsh`

```
#!/ldcg/bin/tclsh
#
# $Id: POWER0305.tclsh,v 1.1 2001/01/19 21:01:07 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,1,1,10.0,1.0,16,3.0,0.5,1,0.1)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026s1r1024.ilwd
  -subject POWER0305
}"

set sid [ socket ldas-test.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

No output should be produced by this test. The job-id will be different.

Pass

6. flow = 100.0, deltaF = 1.0, length = 16

`../ldasmdc/mpi/test/07power/command/POWER0306.tclsh`

```
#!/ldcg/bin/tclsh
#
# $Id: POWER0306.tclsh,v 1.1 2001/01/19 21:01:07 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libpower.so
  -filterparams (1026,1,513,3,1,1,100.0,1.0,16,3.0,0.5,1,0.1)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/07power/input/n1026s1r1024.ilwd
  -subject POWER0306
}"

set sid [ socket ldas-test.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Compare the output to that in the file `../test/07power/results/output0306.ilwd`; the snr, confidence may only agree to 6 significant figures. The job-id will be different.

Pass

SUMMARY

Known faults encountered – list bug IDs: The shared object reports GPS seconds incorrectly, but GPS nanoseconds correctly. The fix for this bug is known and should be implemented.

New faults submitted – list bug IDs: None

TEST RESULT

Pass

B.9 08ldasInspiral

This test was absorbed into 06inspiral described in subsection B.7.

B.10 09ldasPower

This test was absorbed into test 07power in subsection B.8.

B.11 10longTermInspiral

Verify that LDAS can run a binary inspiral search using the inspiral search on long, multi-segment chunks of data with multiple templates.

B.11.1 LTINSPIRAL01

Test Case:	LTINSPIRAL01
Purpose:	Verify that the inspiral DSO under LDAS can filter multi-segment chunks of data of various lengths, using Gaussian noise as input.
Tester:	Duncan Brown
Test machine:	ldas-dev
Date (mm/dd/yy):	02/05/2001
Time:	18:00

ENVIRONMENT AND PREREQUISITES

This test is executed through the user `ldas_mdc` on LDAS. It depends on the successful completion all tests in Sec. B.7.

The following input data are required for this test:

1. `n_chunk1.ilwd` $s = 2$ segments of length $n = 2^{20}$
2. `n_chunk2.ilwd` $s = 8$ segments of length $n = 2^{20}$
3. `n_chunk3.ilwd` $s = 2$ segments of length $n = 2^{10}$
4. `n_chunk4.ilwd` $s = 9$ segments of length $n = 2^{13}$

The input files contain multiple segments of white Gaussian noise of variance 64.0 sampled at 1024Hz, where each subsequent segment overlaps the previous one by half its length. This consists of three elements

1. `ifodmro` of length $n(s+1)/2$, white Gaussian noise
2. `spectrum` of length $\frac{n}{2} + 1$, constant value of 0.125
3. `response` of length $\frac{n}{2} + 1$, constant value of unity
4. `tmplBank` a bank of 129 templates in ILWD format

The input data should be stored in `.../ldasmdc/mpi/test/10longTermInspiral/input/`

For inspection of the data, one should use the `getStats` program discussed in section B.7.2. The source code and a makefile can be found in `.../ldasmdc/mpi/test/06inspiral/results/`

PROCEDURE

Run the LDAS user command scripts listed in the tests below. Inspect the job log files and ensure that each command executed sucesfully without errors. For each LDAS job ID n there should be six output files:

1. `NORMALn_0.txt` should contain the process information in ilwd format written by the wrapper.
2. `NORMALn_x.txt` should contain the output of the filter produced by slave x of the inspiral shared object.

Ensure that these output files have been created. Run the file containing the filter output through `getStats` and check that the results match the values in the results files

1. `n_chunk1.x.stats`

2. n_chunk2.x.stats
3. n_chunk3.x.stats
4. n_chunk4.x.stats

in the result directory `./.../ldasmdc/mpi/test/10longTermInspirial/result/`

1. 2 segments of length $n = 2^{20}$,
`./.../ldasmdc/mpi/test/10longTermInspirial/command/LTINSPIRAL0101.tclsh`

```
#!/lscg/bin/tclsh
#
# $Id: LTINSPIRAL0101.tclsh,v 1.7 2001/02/06 00:59:54 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /lscg/lib/lalwrapper/libinspiral.so
  -filterparams (1,1048576,2,1,0.00097656,524288,0,40.0,100.0,1.0,1.0,32)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/10longTermInspirial/input/n_chunk1.ilwd
  -subject LTINSPIRAL0101
  -np 6
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- [{"\n\s"}] $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

2. 8 segments of length $n = 2^{20}$,
`./.../ldasmdc/mpi/test/10longTermInspirial/command/LTINSPIRAL0102.tclsh`

```
#!/lscg/bin/tclsh
#
# $Id: LTINSPIRAL0102.tclsh,v 1.6 2001/02/06 00:59:54 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /lscg/lib/lalwrapper/libinspiral.so
  -filterparams (1,1048576,8,1,0.00097656,524288,0,40.0,100.0,1.0,1.0,32)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/10longTermInspirial/input/n_chunk2.ilwd
  -subject LTINSPIRAL0102
  -np 6
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- [{"\n\s"}] $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

3. 2 segments of length $n = 2^{10}$,
`./.../ldasmdc/mpi/test/10longTermInspirial/command/LTINSPIRAL0103.tclsh`

```
#!/lscg/bin/tclsh
#
# $Id: LTINSPIRAL0103.tclsh,v 1.6 2001/02/06 00:59:54 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
```

```

-dynlib /ldcg/lib/lalwrapper/libinspiral.so
-filterparams (1,1024,2,1,0.00097656,512,0,40.0,100.0,1.0,1.0,32)
-inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/10longTermInspiral/input/n_chunk3.ilwd
-subject LTINSPIRAL0103
-np 6
}"
set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

4. 9 segments of length $n = 2^{13}$,

/.../ldasmdc/mpi/test/10longTermInspiral/command/LTINSPIRAL0104.tclsh

```

#!/ldcg/bin/tclsh
#
# $Id: LTINSPIRAL0104.tclsh,v 1.6 2001/02/06 00:59:54 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (1,8192,9,1,0.00097656,4096,0,40.0,100.0,1.0,1.0,32)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/10longTermInspiral/input/n_chunk4.ilwd
  -subject LTINSPIRAL0104
  -np 6
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid

```

Pass

SUMMARY

Known faults encountered – list bug IDs: None

New faults submitted – list bug IDs: None

TEST RESULT

Pass

B.11.2 LTINSPIRAL02

Test Case:	LTINSPIRAL02
Purpose:	Verify that the inspiral DSO under LDAS can filter multi-segment chunks containing various inspiral chirps embedded in Gaussian noise.
Tester:	Duncan Brown
Test machine:	ldas-dev
Date (mm/dd/yy):	02/05/2001
Time:	18:00

ENVIRONMENT AND PREREQUISITES

This test is executed through the user `ldas_mdc` on LDAS. It depends on the successful completion all tests in Sec. B.7.

The following input data are required for this test:

1. `cn_chunk1.ilwd` 1 chirp embedded in noise
2. `cn_chunk2.ilwd` 2 chirps embedded in noise
3. `cn_chunk3.ilwd` 5 chirps embedded in noise

The input files contain 8 segments of data each of length $n = 2^{20}$, with subsequent segments overlapping the previous ones by half their length. The data consist of noise with variance 1048576 sampled at 1024Hz, with one or more chirp signals superimposed. The files contain four elements:

1. `ifodmro` of length $n(s+1)/2$, white Gaussian noise with whitened binary inspiral chirps; binaries have various masses (m_1, m_2) and are at various distances d .
2. `spectrum` of length $\frac{n}{2} + 1$, constant value of 2048.0
3. `response` of length $\frac{n}{2} + 1$, simulated LIGO 1 response function
4. `tmplBank` a bank of 129 templates in ILWD format

The input data should be stored in `.../ldasmdc/mpi/test/10longTermInspiral/input/`

PROCEDURE

Run the LDAS user command scripts listed in the tests below. Inspect the job log files and ensure that each command executed successfully without errors.

Inspect the result files produced and ensure that, for each chirp event, the values of ρ_{\max}^2 , χ_{\min}^2 , d_{eff} , and the GPS inspiral time agree with those in the files

1. `cn_chunk1.out1`
2. `cn_chunk2.out1`
3. `cn_chunk2.out2`
4. `cn_chunk3.out1`
5. `cn_chunk3.out2`

6. cn_chunk3.out3
7. cn_chunk3.out4
8. cn_chunk3.out5

in the result directory `/. . . /ldasmdc/mpi/test/10longTermInspirational/result/`

Note that each sample output file gives the results of detecting a single chirp event; thus, chunks with multiple chirps will have multiple sample output files. Also, the GPS inspiral times in each output file are offset by an amount recorded at the top of the file (this is an artifact of how the chunks were sliced into manageable segments when the sample output was being generated).

1. $m_1 = 1.113126M_\odot, m_2 = 0.877036M_\odot, d = 16.109306$ Mpc

`/. . . /ldasmdc/mpi/test/10longTermInspirational/command/LTINSPIRAL0201.tclsh`

```
#!/ldcg/bin/tclsh
#
# $Id: LTINSPIRAL0201.tclsh,v 1.6 2001/02/06 00:59:54 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (0,1048576,8,8,0.00097656,524288,0,40.0,150.0,2.0,69.0,32)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/10longTermInspirational/input/cn_chunk1.ilwd
  -subject LTINSPIRAL0201
  -np 6
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[ \n \s ]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

2. $m_1 = 4.379699M_\odot, m_2 = 1.183518M_\odot, d = 15.757040$ Mpc
 $m_1 = 1.790110M_\odot, m_2 = 1.506015M_\odot, d = 17.012230$ Mpc

`/. . . /ldasmdc/mpi/test/10longTermInspirational/command/LTINSPIRAL0202.tclsh`

```
#!/ldcg/bin/tclsh
#
# $Id: LTINSPIRAL0202.tclsh,v 1.6 2001/02/06 00:59:54 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (0,1048576,8,8,0.00097656,524288,0,40.0,400.0,4.0,69.0,32)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/10longTermInspirational/input/cn_chunk2.ilwd
  -subject LTINSPIRAL0202
  -np 6
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[ \n \s ]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

3. $m_1 = 4.096742M_\odot, m_2 = 1.201902M_\odot, d = 16.378759$ Mpc
 $m_1 = 4.544707M_\odot, m_2 = 4.346353M_\odot, d = 8.449714$ Mpc
 $m_1 = 1.074584M_\odot, m_2 = 1.054919M_\odot, d = 17.656496$ Mpc

$$m_1 = 3.965107M_{\odot}, m_2 = 3.753730M_{\odot}, d = 18.464674 \text{ Mpc}$$

$$m_1 = 4.996519M_{\odot}, m_2 = 1.468517M_{\odot}, d = 13.015765 \text{ Mpc}$$

(a) This script should find the chirps in result files 1 and 3—5.

`/.../ldasmdc/mpi/test/10longTermInspirals/command/LTINSPIRAL0203a.tclsh`

```
#!/ldcg/bin/tclsh
#
# $Id: LTINSPIRAL0203a.tclsh,v 1.3 2001/02/06 00:59:54 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (0,1048576,8,8,0.00097656,524288,0,40.0,150.0,7.0,69.0,32)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/10longTermInspirals/input/cn_chunk3.ilwd
  -subject LTINSPIRAL0203a
  -np 6
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

(b) This script should find the chirps in result file 2.

`/.../ldasmdc/mpi/test/10longTermInspirals/command/LTINSPIRAL0203b.tclsh`

```
#!/ldcg/bin/tclsh
#
# $Id: LTINSPIRAL0203b.tclsh,v 1.3 2001/02/06 00:59:54 duncan Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so
  -filterparams (0,1048576,8,8,0.00097656,524288,0,40.0,7000.0,35.0,69.0,32)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/10longTermInspirals/input/cn_chunk3.ilwd
  -subject LTINSPIRAL0203b
  -np 6
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Pass

SUMMARY

Known faults encountered – list bug IDs: GPS Seconds were incorrect for event due to known bug in ConditionData

New faults submitted – list bug IDs: None

TEST RESULT

Pass

B.12 11longTermPower

Verify that LDAS can run an excess power search using the libpower.so on long, multi-segment chunks of data.

B.12.1 LTPOWER01

Test Case: LTPOWER01

Purpose: Verify that libpower.so can analyze multi-segment chunks of data of various lengths, using Gaussian noise as input.

Tester: Alan Wiseman

Test machine: ldas-dev

Date (mm/dd/yy): 01/19/01 **Time:** 15:58

ENVIRONMENT AND PREREQUISITES

This test is executed through the user `ldas_mdc` on LDAS. It depends on the successful completion all test checklists in subsections B.1–B.6 and in Sec. B.8.

The following input data are required for this test:

1. `noise_power.ilwd` $N = 360459$

The input file contain 10 segments of white Gaussian noise of variance 64.0 sampled at 1024Hz, where each subsequent segment overlaps the previous one by half its length. This consists of three channels of data:

1. `ifodmro` of length $n(s + 1)/2$, white Gaussian noise
2. `spectrum` of length $\frac{n}{2} + 1$, constant value of 0.125
3. `response` of length $\frac{n}{2} + 1$, constant value of unity

The input data should be stored in `/. . . /ldasmdc/mpi/test/13longTerm/input/`

PROCEDURE

Run the LDAS user command scripts listed in the tests below. Inspect the job log files and ensure that each command executed sucesfully without errors. Inspect the output files produced and compare with the results where indicated.

1. 2 segments of length $N = 65538$,

`/. . . /ldasmdc/mpi/test/11longTermPower/command/LTPOWER0101.tclsh`

```
#!/dcg/bin/tclsh
#
# $Id: LTPOWER0101.tclsh,v 1.1 2001/01/19 23:00:41 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynlib /beowulf/ldas/lib/lalwrapper/libpower.so
  -filterparams (65538,2,32769,3,1,1,1.0,0.1,16384,3.0,0.5,1,1e-7)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/13longTerm/input/noise_power.ilwd
  -subject LTPOWER0101
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[ \n\ } $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Takes approximately 90 seconds to complete. There should be one output file:

- (a) `NORMALn_0.txt` should contain the process information in `ilwd` format written by the wrapper.

Pass2. 4 segments of length $N = 65538$,

```
./.../ldasmdc/mpi/test/11longTermPower/command/LTPOWER0102.tclsh
```

```
#!/ldcg/bin/tclsh
#
# $Id: LTPOWER0102.tclsh,v 1.1 2001/01/19 23:00:41 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /beowulf/ldas/lib/lalwrapper/libpower.so
  -filterparams (65538,4,32769,3,1,1,1,0,0.1,16384,3,0,0.5,1,1e-7)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/13longTerm/input/noise_power.ilwd
  -subject LTPOWER0101
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Takes approximately 180 seconds to complete. There should be one output file:

- (a) `NORMALn_0.txt` should contain the process information in `ilwd` format written by the wrapper.

Pass3. 10 segments of length $N = 65538$,

```
./.../ldasmdc/mpi/test/11longTermPower/command/LTPOWER0103.tclsh
```

```
#!/ldcg/bin/tclsh
#
# $Id: LTPOWER0103.tclsh,v 1.1 2001/01/19 23:00:41 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -dynamlib /beowulf/ldas/lib/lalwrapper/libpower.so
  -filterparams (65538,10,32769,3,1,1,1,0,0.1,16384,3,0,0.5,1,1e-7)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/13longTerm/input/noise_power.ilwd
  -subject LTPOWER0101
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[\\n\\s]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

Takes approximately 450 seconds to complete. There should be three output file:

- (a) `NORMALn_0.txt` should contain the process information in `ilwd` format written by the wrapper.
- (b) `NORMALn_1.txt` should contain a list of possible events. Compare the output to that in the file `.../test/11longTermPower/results/output0103_1.ilwd`; the `snr`, `confidence` may only agree to 6 significant figures. The job-id will be different.
- (c) `NORMALn_2.txt` should contain a list of possible events. Compare the output to that in the file `.../test/11longTermPower/results/output0103_2.ilwd`; the `snr`, `confidence` may only agree to 6 significant figures. The job-id will be different.

Pass

SUMMARY

Known faults encountered – list bug IDs: None

New faults submitted – list bug IDs: None

TEST RESULT

Pass

B.12.2 LTPOWER02

Test Case: LTPOWER02

Purpose: Verify that libpower.so can analyze multi-segment chunks of data on different numbers of nodes, using Gaussian noise plus impulses as input.

Tester: Patrick Brady

Test machine: ldas-dev

Date (mm/dd/yy): 01/18/01 **Time:** 21:30

ENVIRONMENT AND PREREQUISITES

This test is executed through the user `ldas_mdc` on LDAS. This test is executed through the user `ldas_mdc` on LDAS. It depends on the successful completion all test checklists in subsections B.1–B.6 and in Sec. B.8.

The following input data are required for this test:

1. `impulse_power.ilwd` $N = 360459$

The input file contain 10 segments of white Gaussian noise of variance 64.0 with 5 impulses injected and sampled at 1024Hz, where each subsequent segment overlaps the previous one by half its length. The data consists of three channels:

1. `ifodmro` of length $n(s + 1)/2$, white Gaussian noise
2. `spectrum` of length $\frac{n}{2} + 1$, constant value of 0.125
3. `response` of length $\frac{n}{2} + 1$, constant value of unity

The input data should be stored in `.../ldasmdc/mpi/test/13longTerm/input/`

PROCEDURE

Run the LDAS user command scripts listed in the tests below. Inspect the job log files and ensure that each command executed sucesfully without errors. Inspect the output files produced and compare with the results where indicated.

1. `np=4`, 10 segments of length $N = 65538$,
`.../ldasmdc/mpi/test/11longTermPower/command/LTPOWER0201.tclsh`

```
#!/ldcg/bin/tclsh
#
# $Id: LTPOWER0201.tclsh,v 1.1 2001/01/30 21:28:41 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -np 4
  -dynamlib /beowulf/ldas/lib/lalwrapper/libpower.so
  -filterparams (65538,10,32769,3,1,1,1.0,0.1,16384,3.0,0.5,1,1e-7)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/13longTerm/input/impulse_power.ilwd
  -subject LONGTERM004P
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {[ \n \s ]+} $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

There should be seven output file:

- (a) `NORMALn_0.txt` should contain the process information in `ilwd` format written by the wrapper.
- (b) `NORMALn_1.txt` – `NORMALn_6.txt` should contain lists of possible events. Compare the contents of these files with `output02_1.ilwd` – `output02_6.ilwd` stored in the directory `.../test/11longTermPower/results/`; the `snr` and `confidence` may only agree to 6 significant figures. The job-id will be different.

Pass

2. `np=5`, 10 segments of length $N = 65538$,

`.../ldasmdc/mpi/test/11longTermPower/command/LTPOWER0102.tclsh`

```
#!/lscg/bin/tclsh
#
# $Id: LTPOWER0202.tclsh,v 1.1 2001/01/30 21:28:41 patrick Exp $

set cmd "ldasJob
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }
{
  mpiFile2File
  -np 5
  -dynamlib /beowulf/ldas/lib/lalwrapper/libpower.so
  -filterparams (65538,10,32769,3,1,1,1.0,0.1,16384,3.0,0.5,1,1e-7)
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/13longTerm/input/impulse_power.ilwd
  -subject LONGTERM004P
}"

set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]
regsub -all -- {\n\s}+ $cmd { } cmd
puts $sid $cmd
flush $sid
puts [ read $sid ]
close $sid
```

There should be seven output file: There should be seven output file:

- (a) `NORMALn_0.txt` should contain the process information in `ilwd` format written by the wrapper.
- (b) `NORMALn_1.txt` – `NORMALn_6.txt` should contain lists of possible events. Compare the contents of these files with `output02_1.ilwd` – `output02_6.ilwd` stored in the directory `.../test/11longTermPower/results/`; the `snr` and `confidence` may only agree to 6 significant figures. The job-id will be different.

Pass

SUMMARY

Known faults encountered – list bug IDs: None

New faults submitted – list bug IDs: None

TEST RESULT

Pass

B.13 12simultaneous

This test was absorbed into test 13longTerm in subsection B.14.

B.14 13longTerm

Simultaneously run inspiral and power dso unattended for 24 hours.

B.14.1 MPILONGTERM01

Test Case: MPILONGTERM01

Purpose: Simultaneously run inspiral and power dso unattended for 24 hours.

Tester: MPI working group

Test machine: ldas-dev.ligo.caltech.edu

Date (mm/dd/yy): 01/19/01 **Time:** 03:45

ENVIRONMENT AND PREREQUISITES

This test is executed on the Beowulf hardware from ldas@ldas-dev. It depends on the successful completion of everything before, except the documentation.

PROCEDURE

The scripts /ldas_outgoing/jobs/mpimdc/mpi/test/13longTerm/command/*.tclsh will be executed sequentially by the script /ldas_outgoing/mpi_mdc_pacing_script. Each job should successfully execute and produce output. The tclsh scripts were generated by the program genscript.c in the command directory:

```
#include <stdio.h>
#include <sys/types.h>
#include <sys/stat.h>

const char *power =
"!/ldcg/bin/tclsh\n"
#\n\
# $Id: 13test.tex,v 1.5 2001/03/01 23:01:27 patrick Exp $ \n\
\n\
set cmd \"ldasJob \n\
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu } \n\
{ \n\
  mpiFile2File \n\
  -np $d \n\
  -dynamlib /ldcg/lib/lalwrapper/libpower.so \n\
  -filterparams (65538,10,32769,3,1,1,1.0,0.1,16384,3.0,0.5,1,1e-7) \n\
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/13longTerm/input/%s.ilwd\n\
  -subject LONGTERM%03dP \n\
}\n" \n\
\n\
set sid [ socket ldas-dev.ligo.caltech.edu 10001 ] \n\
regsub -all -- {[\\n\\s]+} $cmd { } cmd \n\
puts $sid $cmd \n\
flush $sid \n\
puts [ read $sid ] \n\
close $sid \n";

const char *inspiral[2] = {
"!/ldcg/bin/tclsh\n"
set cmd \"ldasJob\n\
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }\n\
{ \n\
  mpiFile2File\n\
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so\n\
  -filterparams (0,1048576,8,8,9.765625e-04,524288,0,40,150,5,69,25)\n\
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/13longTerm/input/%s.ilwd\n\
  -subject LONGTERM%03dI1\n\
  -np 10\n\
}\n" \n\
\n\
set sid [ socket ldas-dev.ligo.caltech.edu 10001 ]\n\
regsub -all -- {[\\n\\s]+} $cmd { } cmd\n\
puts $sid $cmd\n\
flush $sid\n\
puts [ read $sid ]\n\
close $sid\n",
"!/ldcg/bin/tclsh\n"
\n\
set cmd \"ldasJob\n\
{ -name ldas_mdc -password beowulf -email mpigroup@gravity.phys.uwm.edu }\n\
{ \n\
  mpiFile2File\n\
  -dynamlib /ldcg/lib/lalwrapper/libinspiral.so\n\
  -filterparams (0,1048576,8,8,9.765625e-04,524288,0,40,5000,50,69,30)\n\
  -inputfile /ldas_outgoing/jobs/ldasmdc/mpi/test/13longTerm/input/%s.ilwd\n\
  -subject LONGTERM%03dI2\n\
}
```

```

        -np 10\n\
    }\n\
\n\
set sid { socket ldas-dev.ligo.caltech.edu 10001 }\n\
regsub -all -- {[\n\s]+} $cmd { } cmd\n\
puts $sid $cmd\n\
flush $sid\n\
puts [ read $sid ]\n\
close $sid\n"
};

char *powerinp[3] = { "noise_power", "chirp_power", "impulse_power" };
char *inspiralinp[3] = { "n_chunk2", "cn_chunk3", "impulse_chunk" };

int main( void )
{
    char fname[64];
    int scriptno = 0;
    int powerno = 0;
    int inspiralno = 0;
    FILE *fp;

    while ( scriptno < 911 )
    {
        if ( scriptno++ % 5 )
        {
            /* power */
            snprintf( fname, sizeof( fname ), "LONGTERM%03d.tclsh", scriptno );
            fp = fopen( fname, "w" );
            fprintf( fp, power, powerno % 2 ? 4 : 5, powerinp[powerno % 3],
                scriptno );
            fclose( fp );
            fprintf( stdout, "%03d power:\tnodes=%d, inputfile=%s\n", scriptno,
                powerno % 2 ? 4 : 5, powerinp[powerno % 3] );
            ++powerno;
        }
        else
        {
            /* inspiral */
            snprintf( fname, sizeof( fname ), "LONGTERM%03d.tclsh", scriptno );
            fp = fopen( fname, "w" );
            fprintf( fp, inspiralinp[inspiralno % 2], inspiralinp[inspiralno % 3],
                scriptno );
            fclose( fp );
            fprintf( stdout, "%03d inspiral:\tcmd=%d, inputfile=%s\n", scriptno,
                inspiralno % 2, inspiralinp[inspiralno % 3] );
            ++inspiralno;
        }
        chmod( fname, S_IRUSR | S_IWUSR | S_IXUSR | S_IRGRP | S_IXGRP | S_IROTH |
            S_IXOTH );
    }

    return 0;
}

```

SUMMARY

Known faults encountered – list bug IDs: None

New faults submitted – list bug IDs: #647: There were four error conditions which arose during the long-term test which made it necessary to kill jobs by hand to return node or user resources to the system:

job 1220 - 01/20/01 2:30 a.m. The wrapperAPI did not reply to the mpiAPI, and when the mpiAPI attempted to kill the wrappers ssh permission failed between ldas@beowulf and search01@beowulf. 15 hours after the failure I tried to kill this job manually, and it triggered completion of the job after a 15 hour idle time!

job 1231 - 01/20/01 3:00 a.m. undetermined string format error. This one is a complete mystery to me, I was able to run this script individually without a hitch. Once this job was hung, the system was idle, waiting for nodes until I killed this job at 5:30 p.m. As soon as I killed the job everything started running again, beginning with the resurrected job 1220.

job 1318 - 01/20/01 9:50 p.m. ssh permission denied between ldas@beowulf and search03@beowulf when the mpiAPI attempted to kill this job due to inactivity. The ssh call which failed was the second of four such calls within the procedure.

job 1477 - 01/21/01 6:40 a.m. ssh permission denied between ldas@beowulf and search02@beowulf when the mpiAPI attempted to kill this job due to inactivity. The ssh call which failed was the second of four such calls within the procedure.

After writing the above report, Phil Ehrens investigated the result of running multiple ssh1 calls in succession and discovered that the LAST ssh1 call in a group made with the -f option will fail to return properly. This does not result in a "Permission denied" error, but does bear on an unusual phenomenon noticed where an ssh call made from inside of a script sometimes seems to return a null string. When the -n option is used instead of -f there are no problems. It is uncertain whether this raises issues for potentially long running commands executed via ssh. Openssh and ssh2 exhibit this same behaviour. The recommendation in the man page for ssh that -f be used when running from inside of a script is undeservedly unqualified. If the command run from ssh is expected to return in a reasonable amount of time, and the return value is to be captured, -f should *not* be used.

TEST RESULT

Conditional Pass