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

Document Type	**LIGO-T040175-00-Z**	2005/12/19

Specification for Storing Calibration Data for LIGO

Duncan Brown, Alexander Dietz, Gabriela Gonzàlez, Mike Landry, Brian O'Reilly, Patrick Sutton and John Zweizig

Distribution of this draft:

LIGO Scientific Collaboration

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 NW17-161
Cambridge, MA 01239
Phone (617) 253-4824
Fax (617) 253-7014
E-mail: info@ligo.mit.edu

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

Processed with LaTeX on 2005/12/19

Contents

1 Introduction

The document is intended to define specifications for the calibration data products for the LIGO interferometers exchanged within the LIGO Scientific Collaboration. Calibration products exchanged within the LSC include reference frequency series, containing the detector calibration at certain reference times, and time series describing how the calibration of the detector varies during the course of a science run. This information can be exchanged in the ASCII, XML or IGWD Frame format files described in this document.

2 Definitions

The response function $R(f)$ is the frequency domain transfer function from a particular channel $\tilde{v}(f)$ (e.g. `Xn:LSC-AS_Q` or `Xn:LSC-DARM_ERR`) to the gravitational strain $\tilde{h}(f)$, that is,

$$\tilde{h}(f) = R(f)\tilde{v}(f). \tag{1}$$

The response function at a given time t is defined in terms of the quantities

- $\alpha(t)$ is the measured cavity (optical) gain factor,
- $\gamma(t)$ is the measured open loop gain factor,
- $G_0(f)$ is the reference open-loop gain,
- $C_0(f)$ is the reference sensing (cavity) function.

The response runction needed to calibrate the channel `AS_Q` is

$$R_{\texttt{AS\_Q}}(f;t) = \frac{1+\gamma(t)G_0(f)}{\alpha(t)C_{Q0}(f)}, \tag{2}$$

and the response function needed to calibrate the channel `DARM_ERR` is

$$R_{\texttt{DARM\_ERR}}(f;t) = \frac{1+\gamma(t)G_0(f)}{\gamma(t)C_{D0}(f)}. \tag{3}$$

The reference functions $G_0(f)$ and $C_0(f)$ are determined during calibration runs performed before or during a science run and are considered fixed. The reference response function $R_0(f)$ may be calculated from the reference sensing function and open loop gain by

$$R_0(f) = \frac{1+G_0(f)}{C_0(f)}. \tag{4}$$

To compute the response function at any given time, the time dependent α and γ factors must be computed. $\alpha(t)$ and $\gamma(t)$ can be determined by monitoring the `AS_Q` channel, the excitation channel in which the line is injected, and various IFO feedback channels. Formulae for computing α and γ are given in, for example, LIGO-T040060-01 (available from the ”Documents” subsection of the calibration homepage). The data required as input to these calculations are important for online monitoring, and the resulting $\alpha(t)$ and $\gamma(t)$ are necessary for offline calibration. The following is necessary for the production of calibration line amplitudes and strain data using the LALApps code:

- Calibration line frequency
- Real and imaginary parts of the open loop gain and digital servo filter at the calibration line frequency
- The filter coefficients for the inverse sensing, actuation and possibly the digital servo
- The name of the excitation channel in which the calibration line is injected.

The following data is necessary for the computation of α and γ using the DMT SenseMon program:

- Calibration line data:
  - the name of the excitation channel
  - the frequency of the line
  - the amplitude at which it was injected into the excitation channel when the reference calibrations were made
  - the amplitude at which it was measured in the readout channel when the reference calibrations were made
- DARM gain data:
  - the number of DARM gain channels that must be monitored
  - the name of each DARM gain channel
  - the value of each DARM gain channel when the reference calibrations were measured

3 Calibration program maintenance

All code used to produce calibration data and exchange it between different formats must reside in a CVS repository. The calibration CVS archive should contain

1. Matlab interferometer models used to produce the original calibration data.
2. Matlab code used to digitize the analog filters in the interferometer models for use in calibrated strain data.
3. Condor submit files or pipeline initialization files used to generate Condor DAGs which produce calibration factors or strain data using LALApps code.
4. Reference calibration data measurements (e.g. raw ASCII time series measurements).
5. Any calibration products released by the calibration team (e.g. ASCII time series, frequency series and frame data).

Calibration data will be released from the calibration home page via links to the relevant data products in the calibration CVS archive. The following codes should also be maintained in the appropriate CVS archives (lscsoft or DMT):

1. LAL/LALApps C code used to compute calibration line amplitudes or calibration calibration update time series.
2. LALApps pipeline generation scripts used to produce Condor DAGs which generate calibration factors or strain data.
3. LAL/LALApps C code used to generate calibrated strain data.
4. LAL/LALApps C or DMT C++ code used to generate calibration frame data.

5. LAL/LALApps C or DMT C++ code used to generate calibration XML data.

Matlab scripts should be checked into the calibration section of the matapps archive, stand alone or LAL dependent C code should be checked into the calibration module of the LALApps CVS archive and DMT calibration code shout be checked into the DMT CVS. This will allow calibration data products to contain a CVS tag which will allow the pedigree of the calibration data to be traced.

4 File naming convention

In order to simplify metadata publishing and allow replication using the Lightweight Data Replicator (LDR), calibration data files should use the naming convention decsribed in this section. For frame files, this convention is consistent with the frame file naming convention described in LIGO-T010150-00. Calibration files should have a filename of the type

`basename.ext`

where the three character extension `ext` after the period determines the file type as

- `txt` flat ASCII format,
- `xml` XML format, or
- `gwf` frame data format.

The `basename` of the file is divided into the hypen ("–") separated fields

`SITE-TAG-GPSSTART-DURATION`

where

- `SITE` is one or more observatory letter codes corresponding to the source of the data within the file. If more than one letter is given, obervatory letters should be listed alphabetically. For example if the file contains data from only the L1 detector, then `SITE` should be `L`. If the file contains data from either the H1 or H2 detectors, or both, then `SITE` will be `H`. If the file contains data from both the H1 and L1 detectors, then `SITE` will be `HL`
- `TAG` is a string containing only the ASCII letters (a–z, A–Z), digits (1–9) and underscores ("_") which describes the contents of the file. The exact format of this tag string is described below.
- `GPSSTART` is the integer GPS time of the file. For time series data, this is the GPS time of the first time stamp in the file. For reference time series data, this is the time at which the reference data was taken.
- `DURATION` is the duration of the file in integer seconds. For time series data, this is the duration of the data in the file. That is the file is assumed to cover the half open interval [`GPSSTART`,`GPSSTART`+`DURATION`). For reference frequency series data, this is the equivalent length of the data in the time domain.

For files containing time series data, the TAG field in the file name is divided into the underscore ("_") separated fields

IFO_CAL_FAC_RUN_VERSION_TIME

where

- IFO is an alphanumerically sorted list of inteferometers for which this calibration file contains data, for example H1, H1H2L1, etc.
- CAL_FAC is a fixed string that identifies the file as containing calibration factor update time series data.
- RUN is indicating the run this file is valid for, e.g. S4 or S5.
- VERSION is a calibration version string in the format V1, V2, . . ., V27, ... The version type may be U if the file contains unity calibration coefficients for the entire diration of the file, that is $\alpha = \gamma = 1$.
- TIME is the value of the time spacing that is used, for example 001 for a second trend or 060 for a minute trend.

For XML and frame files containing frequency series data, the TAG field in the filename is divided into the underscore ("_") separated fields

IFO_CAL_REF_CHANNEL_RUN_VERSION

where

- IFO is an alphanumerically sorted list of inteferometers for which this calibration file contains data, for example H1, H1H2L1, etc.
- CAL_REF is a fixed string that identifies the file as containing reference calibration frequency series data.
- CHANNEL is the channel name that has been used for the reference calibration frequency series, e.g. DARM_ERR or AS_Q.
- RUN is indicating the run where this file belongs, e.g. S4 or S5.
- VERSION is a calibration version string in the format V1, V2, . . ., V27, ...

For ASCII files containing frequency series data, where only one frequency series can be stored per file, the TAG field in the filename is divided into the underscore ("_") separated fields

IFO_CAL_REF_CHANNEL_RUN_VERSION

- IFO is an alphanumerically sorted list of inteferometers for which this calibration file contains data, for example H1, H1H2L1, etc.
- CAL_REF is a fixed string that identifies the file as containing reference calibration frequency series data.

- `CHANNEL` is the name of the frequency series calibration data in the file, for example `CAV_GAIN_AS_Q`, `CAV_GAIN_DARM_ERR`, `RESPONSE_AS_Q`, `RESPONSE_DARM_ERR`, or `OLOOP_GAIN`.
- `RUN` is indicating the run this file is valid for, e.g. `S4` or `S5`.
- `VERSION` is a calibration version string in the format `V1`, `V2`, . . ., `V27`, ...

5 File formats

Calibration data may be stored in flat ASCII format files organized by rows and columns, as defined in section 5.1, in XML files, as described in section 5.2 and LIGO-T990023, or in IGWD frame files, as described in section 5.3 and LIGO-T970130.

5.1 ASCII calibration data

Calibration data may be stored in ASCII text files organized by whitespace separated rows and columns. A single file may contain either time or frequency series data, but not both. If the file contains frequency domain data, only one frequency series may be stored in a single file. Each row is either a comment or a line containing time or frequency series calibration data. The first character of a comment row must be a percent symbol ("%", ASCII 0x25). Rows are to be terminated by the UNIX newline convention (lines ended by a new line character ASCII 0x0A). Columns are to be separated by one or more horizontal tab characters (ASCII 0x11) or one or more space characters (ASCII 0x40).

5.1.1 Mandatory comments

The first row of the ASCII file must contain a comment of the form

```
% deltaT = %f
```

for time series data and

```
% deltaF = %f
```

for frequency series data, where `%f` is a floating point number corresponding to the time sample interval Δt or frequency sample interval Δf of the time or frequency series. The second row of the file must contain a comment of the form

```
% $Name:  $
```

where `$Name:  $` is CVS tag of the code used to produce the calibration data. The third row of the file must contain the names of the columns in the file. Optional additional comment lines may be included after this row.

5.1.2 Calibration update parameters

The calibration update parameters α and γ describe how the reference calibration series change with time. They should be stored as time series where non-comment rows in the file contain the following columns:

```
GPSTime     alpha     gamma    other columns
```

where

- `GPSTime` is the time stamp of the start of the half open interval that this duration of validity of this row. The time stamp should be a GPS time in the format `gpsSeconds` or `gpsSeconds.gpsNanoSeconds`. The datum in each rows spans the half-open interval $[\texttt{GPSTime}, \texttt{GPSTime} + \texttt{deltaT})$
- `alpha` is the real part of the calibration update parameter α in the format `%18.9Lf`
- `gamma` is the real part of the calibration update parameters γ in the format `%18.9Lf`
- `other columns` are additional whitespace separated columns that may be included for each timestamp. These columns must be described in a comment at the top of the file and may include such items as SenseMon range, the imaginary parts of the parameters α, γ, or additional line amplitudes.

The difference between the timestamps t_1 and t_2 of sucessive rows must be greater than or equal to `deltaT`. If $t_2 - t_1 > \texttt{deltaT}$, then there is considered to be a gap in the calibration data (i.e. no data is available) in the half-open interval $[t_1 + \texttt{deltaT}, t_2)$.

5.1.3 Reference frequency series

The reference functions $G_0(f)$, $C_0(f)$ and $R_0(f)$ are needed to construct the response function and can be stored as ASCII frequency series. The frequency series $C_0(f)$ and $R_0(f)$ may be unique to a specific channel, e.g. `AS_Q` or `DARM_ERR` and stored in an ASCII file. Each frequency series must be stored in an individual file, with metadata about the channel stored in the file name as described in section 4. The reference functions should be stored as complex frequency series as modulus r and phase ϕ following the convention $R = r\exp(i\phi)$ where r is the modulus and $phi = \arg(R)$ is the principle value of the argument of R. Note the branch point is at $R = 0$ with branch cut along the negative real axis so that ϕ is in the range $-\pi < \phi <= \pi$ in radians. The reference functions should be stored as complex frequency series where the non comment rows in the file contain the following columns:

```
frequency        modulus          phase
```

where

- `frequency` is the sampled frequency in Hz of each row in the file.
- `modulus` is the modulus of the complex frequency series data at the frequency given by the first column of this row.

Name	Type	Object	Comments
`Channel`	string	param	Name of (readout) channel to be calibrated
`Version`	string	param	Version identification string
`Comment`	string	param	Comment string
`EXCChannel`	string	param	Excitation channel name
`CalLineFreq`	double	param	Calibration line frequency
`CalLineAmplASQ`	double	param	Nominal calibration line amplitude in readout channel.
`CalLineAmplEXC`	double	param	Nominal calibration line amplitude in excitation channel.
`StartTime`	time	param	Start of calibration validity.
`Duration`	double	param	Length of validity.
`DARMChannels`	-	table	Nominal gain settings and readback channels. Col1: Readback channel (string), Col2: Nominal value (double).
`OpenLoopGain`	double	table	Open loop gain transfer function, Col1: Frequency (Hz), Col2: Gain, Col3: Phase (in degrees).
`ResponseFunction`	double	table	IFO response transfer function, Col1: Frequency (Hz), Col2: Gain, Col3: Phase (in degrees).
`SensingFunction`	double	table	Sensing transfer function, Col1: Frequency (Hz), Col2: Gain, Col3: Phase (in degrees).
`Alpha`	double	table	Gain factors cvs time, Col1: GPS start time, Col2: α, Col3: $\alpha\dot{\beta}$.

Table 1: XML format calibration records.

- `phase` is the phase in radians (wrapped) of the complex frequency series data at the frequency given by the first column of this row.

The frequency series may begin at eithr DC or at a higher frequency, but all rows should be separated by the same frequency interval `deltaF`.

5.2 XML calibration data

Calibration data may be stored in an LIGO Light-Weight XML format. The calibration record stored is as described in LIGO-T990111-B-D. The XML calibration record follows the LIGO Light-Weight specification and contains the fields described in table 1.

5.3 Frame calibration data

Calibration frequency series or time series data may be stored in IGWD Frame format (LIGO-T970130-F). For a given calibration version, all frequency domain calibration data should be contained in a single frame file and all time series data should be contained in a single frame file. Time and frequency domain data may not be mixed in a single frame file. Multiple series may be stored in a single Frame file by placing each series in a frame within the file. The frame file should contain a history record containing the version of the frame library and the CVS tag of the program used to create the frame, which may be the same or different to the tag of the program which created the calibration data.

5.3.1 Calibration update parameters

Calibration time series should be stored as FrProc processed data frames. The frame type should be 1 and the frame sub-type should be 3. The frame comment should contain the CVS tag of the program that *created the calibration data*, which may not be neccarily be the CVS tag of the program that created the frame file. The data must be stored as an FrVect within the FrProc structure with the dx set to the sample interval `deltaT` of the time series data. The names of the calibration channels in the frames must be:

- `Xn:CAL-CAV_FAC` The cavity gain factor $\alpha(t)$, where `Xn` is the name of the interferometer for which this data is valid, e.g. `L1:CAL-CAV_FAC`.
- `Xn:CAL-OLOOP_FAC` The open loop gain factor $\gamma(t)$, where `Xn` is the name of the interferometer for which this data is valid, e.g. `L1:CAL-OLOOP_FAC`.

The channels in the frame containing $\alpha(t)$ and $\gamma(t)$ either be of type `real_4`, since it is expected that $\alpha(t)$ and $\gamma(t)$ are real numbers.

If it is desired that the imaginary part of $\alpha(t)$ and $\gamma(t)$ be stored in the frames (for example, during online analysis), then these should be stored as addition `real_4` channels with the names

- `Xn:CAL-CAV_FAC_IMAG` The imaginary part of the cavity gain factor $\alpha(t)$, where `Xn` is the name of the interferometer for which this data is valid, e.g. `L1:CAL-CAV_FAC_IMAG`.
- `Xn:CAL-OLOOP_FAC_IMAG` The imaginary part of the open loop gain factor $\gamma(t)$, where `Xn` is the name of the interferometer for which this data is valid, e.g. `L1:CAL-OLOOP_FAC_IMAG`.

If the imaginary channels are present in the frame, then the normal channels should contain the real part of $\alpha(t)$ and $\gamma(t)$. Released versions of calibration frames will only contain real values of the update factors stored as `real_4` time series.

A calibration factor frame may contain calibration factor time series with gaps (missing data), by storing the data in sucessive frames with the correct GPS start times and values of dx.

5.3.2 Reference frequency series

Reference calibration frequency series should be stored as `complex_8` FrProc processed data frames. The frame type should be 2 and the frame sub-type should be 6. The frame comment should contain the CVS tag of the program that *created the calibration data*, which may not be neccarily be the CVS tag of the program that created the frame file. The data must be stored as an FrVect within the FrProc structure with the dx set to the sample interval `deltaF` of the frequency series data. The names of the calibration channels in the frames must be:

- `Xn:CAL-OLOOP_GAIN` The reference open loop gain function $G_0(f)$, where `Xn` is the name of the interferometer for which this data is valid, e.g. `L1:CAL-OLOOP_GAIN`.
- `Xn:CAL-CAV_GAIN_CHANNEL` The reference cavity gain function $C_0(f)$ *for a specific channel* `CHANNEL`, where `Xn` is the name of the interferometer for which this data is valid, e.g. `L1:CAL-CAV_GAIN_AS_Q` or `L1:CAL-CAV_GAIN_DARM_ERR`.

- `Xn:CAL-RESPONSE_CHANNEL` The reference response function $C_0(f)$ in strain/count *for a specific channel* `CHANNEL`, where `Xn` is the name of the interferometer for which this data is valid, and `CHANNEL` is the name of the channel for which this response function is valid e.g. `L1:CAL-RESPONSE_AS_Q` or `L1:CAL-RESPONSE_DARM_ERR`.

6 Storage, dissemination and retrieval of calibration data

6.1 Offline use of calibration data

Calibration data should be available for download in all formats from the calibration home page at:

`http://blue.ligo-wa.caltech.edu/engrun/Calib_Home/`

The CVS tag should be available and the calibration products should be distributed by creating links to the files in the CVS. Frame format calibration data should be published into LDR at LHO and LLO and from there replicated to CIT and other tier 2 centers who request it.

6.2 Online use of calibration data

TBD.

A Calibration file manipulation utilities

A.1 LAL/LALApps utilities

A.2 DMT utilities

A utility to read and write calibration files in various formats is included in the DMT distribution. The program mkcalibfile has the following command syntax

```
mkcalibfile [-c <channel>] [-cav <sensing-text>] [-f <frame-in>]
            [-fmax <max-freq>] [-i <ifo>] [-o <out>]
            [-olg <open-loop-gain>] [-x <xml-in>]
```

All input files specified by the `-cav`, `-olg` or `-x` arguments are combined into a single calibration record. The calibration record is then written to the file specified by the `-o` argument. At present the output file is xml, but once the calibration frame is completely specified, the desired output format will be inferred from the output file extension.

B Example Calibration Files

B.1 Example ASCII calibration data

B.2 Frame dump of valid calibration data

B.3 Example XML calibration data

```
<?xml version="1.0"?>
<!DOCTYPE LIGO_LW SYSTEM
"http://www.cacr.caltech.edu/projects/ligo_lw.dtd">
<LIGO_LW>
  <LIGO_LW Name="Calibration[1]">

    <Param Name="Channel" Type="string">H2:LSC-AS_Q</Param>
    <Time Name="StartTime" Type="GPS">730123456</Time>
    <Param Name="Duration" Type="int">0</Param>
    <Param Name="Unit" Type="string">counts/strain</Param>
    <Param Name="Comment" Type="string">comment tag for version</Param>

    <Param Name="EXCChannel" Type="string">H2:LSC-ETMX_EXC</Param>
    <Param Name="CalLineFreq" Type="float">973.8</Param>
    <Param Name="CalLineAmplASQ" Type="float">0.00123</Param>
    <Param Name="CalLineAmplEXC" Type="float">0.08</Param>

    <Table Name="DARMChannels">
      <Column Name="ChannelName" Type="string"/>
     <Column Name="RefValue" Type="float"/>
      <Stream>
        "H2:LSC-ICMTRX_01" 0.011
        "H2:LSC-DARM_GAIN" -1.5
      </Stream>
    </Table>

    <Array Name="SensingFunction" Type="float">
      <Dim Name="Frequency">6</Dim>
      <Dim Name="F_Mod_Phase">3</Dim>
      <Stream>
        0 1e17 3.14
        500 1e17 1.5
        1000 1e17 0
        1500 1e16 -1.5
        2000 1e15 -3.14
        2500 1e14 -4.6
      </Stream>
    </Array>
    <Array Name="OpenLoopGain" Type="float">
      <Dim Name="Frequency">6</Dim>
      <Dim Name="F_Mod_Phase">3</Dim>
      <Stream>
        0 1000 3.14
        500 100 1.5
        1000 10 0
        1500 1 -1.5
        2000 0.1 -3.14
        2500 0.01 -4.6
      </Stream>
    </Array>
    <Array Name="ResponseFunction" Type="float">
      <Dim Name="Frequency">6</Dim>
      <Dim Name="F_Mod_Phase">3</Dim>
      <Stream>
        0 1e-15 3.14
        500 1e-16 1.5
        1000 1e-17 0
        1500 1e-17 -1.5
        2000 1e-16 -3.14
        2500 1e-15 -4.6
      </Stream>
    </Array>

    <Array Name="Alpha" Type="double">
      <Dim Name="GPS">2</Dim>
      <Dim Name="GPS_Alpha_AlphaBeta">3</Dim>
      <Stream>
        720000000 1.01 1.00
        720000060 1.10 0.99
        720000120 1.11 1.01
        720000180 1.23 0.97
      </Stream>
    </Array>

  </LIGO_LW>
</LIGO_LW>
```