



Global Diagnostics System

NSF Review, February 26, 2001

Daniel Sigg, LIGO Hanford Observatory

John Zweizig, LIGO Caltech



Diagnostics Tasks

□ Detector Characterization

- Calibration
- Detector response, inter-system dependencies & cross-couplings
- Machine artifacts

□ Maintain Performance

- System identification (Feedback control)
- Continuous operation monitoring

□ Detection Confidence

- Understand the physical environment
- Understand the auxiliary degrees-of-freedom

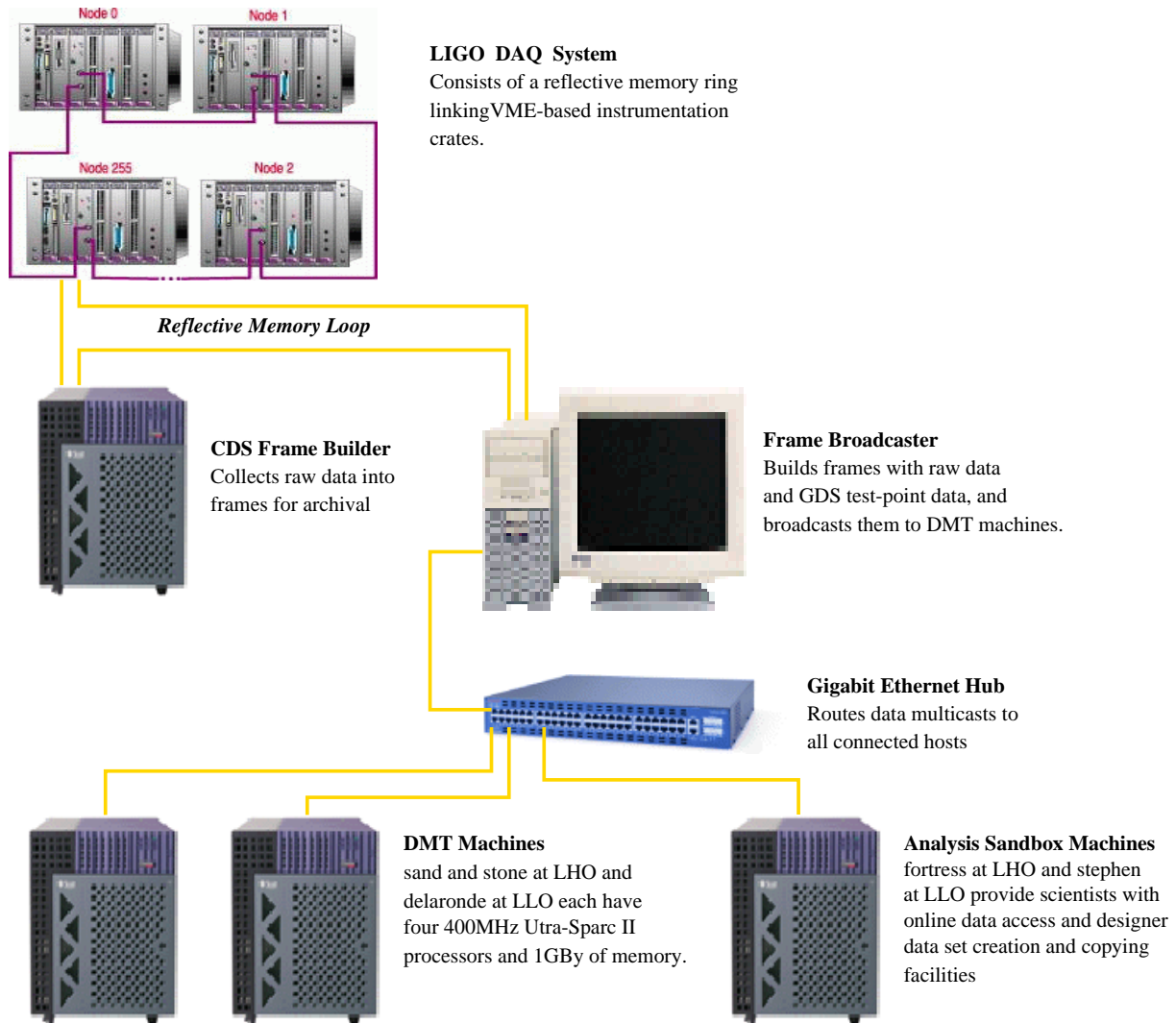
⇒ GW signal \therefore 1% of data rate (3MB/s/ifo)



Basic Approach

- ❑ Consists of two sub-systems (tools)
 - ❑ Diagnostic test tool (DTT)
 - Optional direct excitation, injecting calibration signals
 - Standard analysis tools (psd, coherence, transfer function)
 - Data received directly from DAQ system
 - Graphical Interface
 - ❑ Data Monitoring tool (DMT)
 - Passive monitoring
 - Online access to all IFO+PEM data
 - Supports complex algorithms
 - Interactive use (via root) or background running

Online Data Access



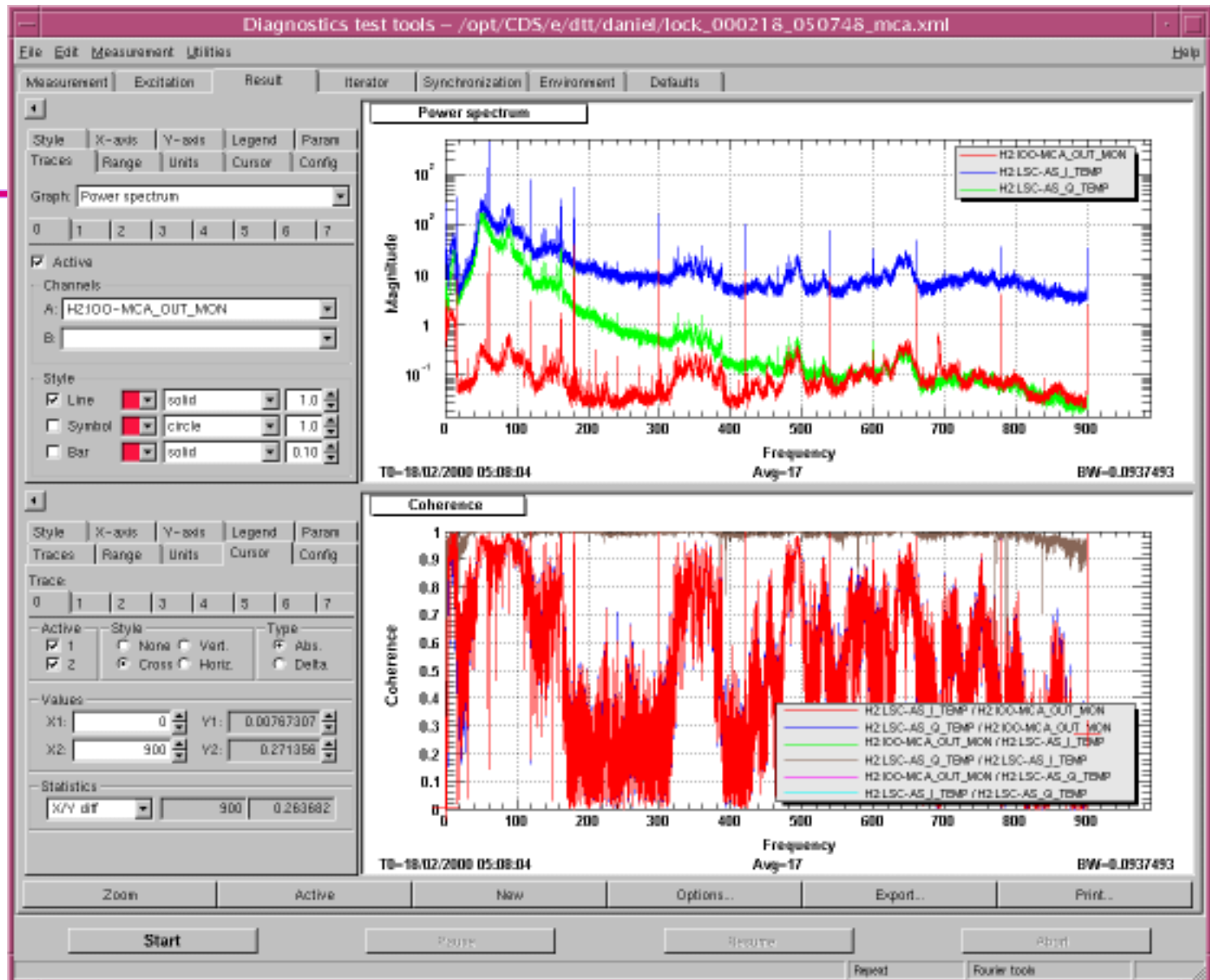
- GDS frames built by dedicated frame broadcaster.
- Data broadcast over G-bit Ethernet to DMT hosts.



Setup



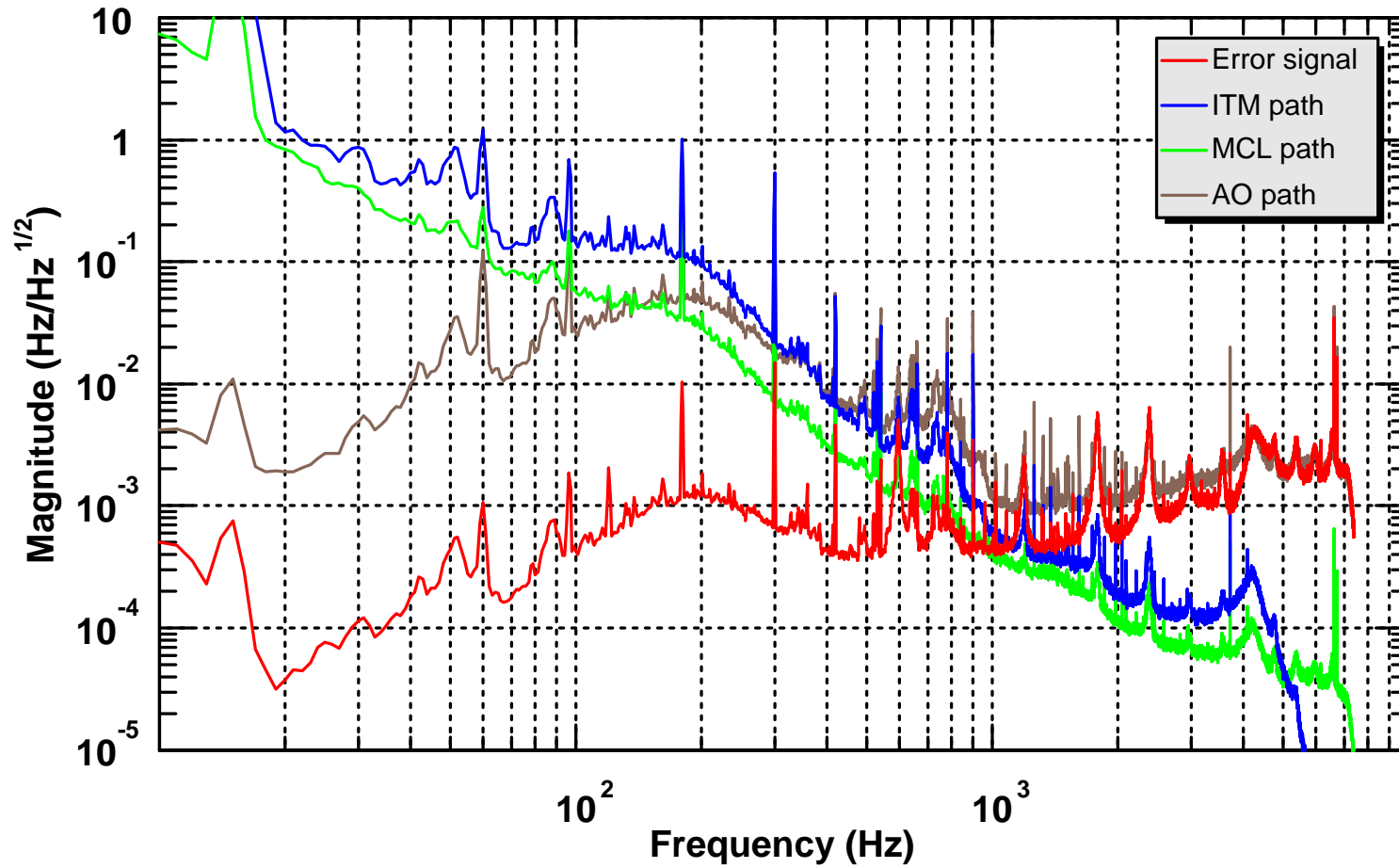
Plot





Frequency Noise Measurement

Power spectrum estimate: Frequency Noise



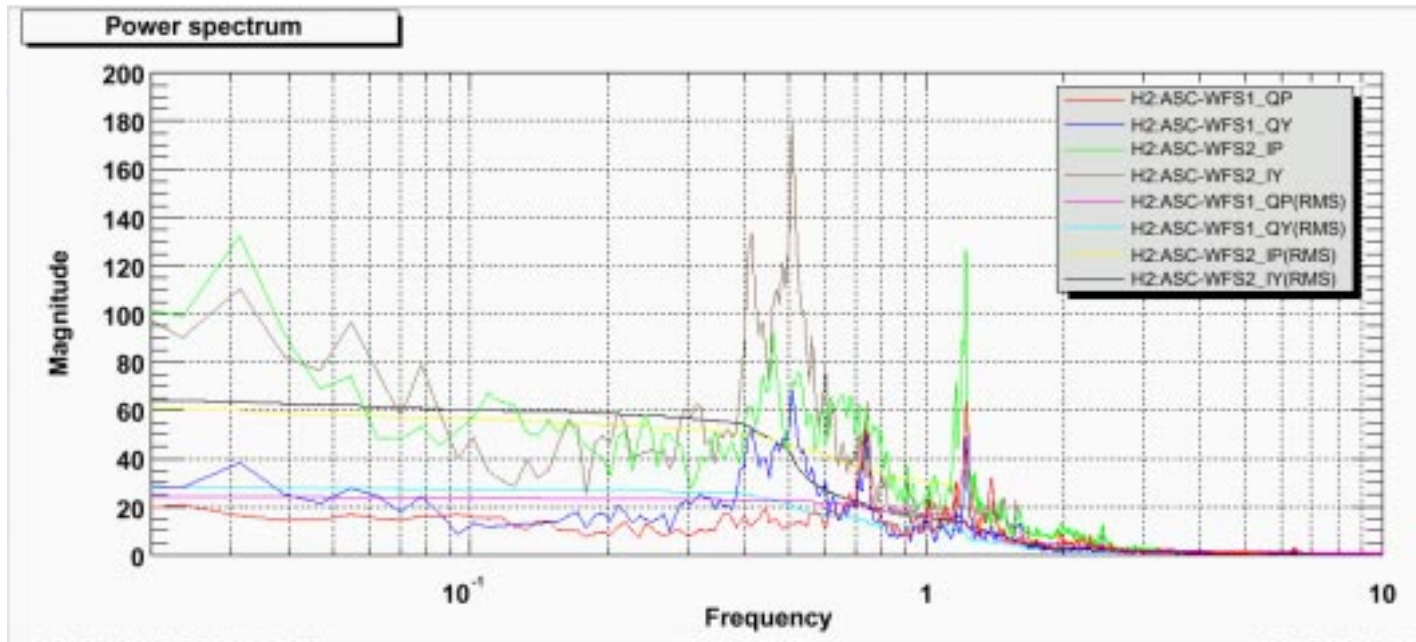
T0=14/04/2000 03:30:00

Avg=100

BW=1.49999

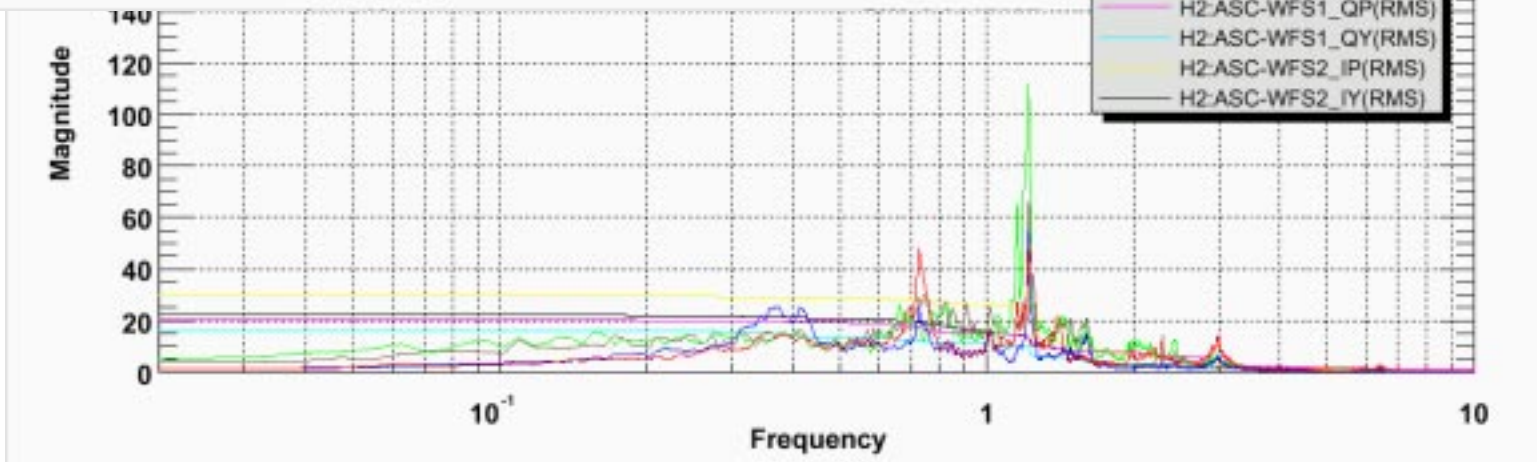


Alignment Fluctuations



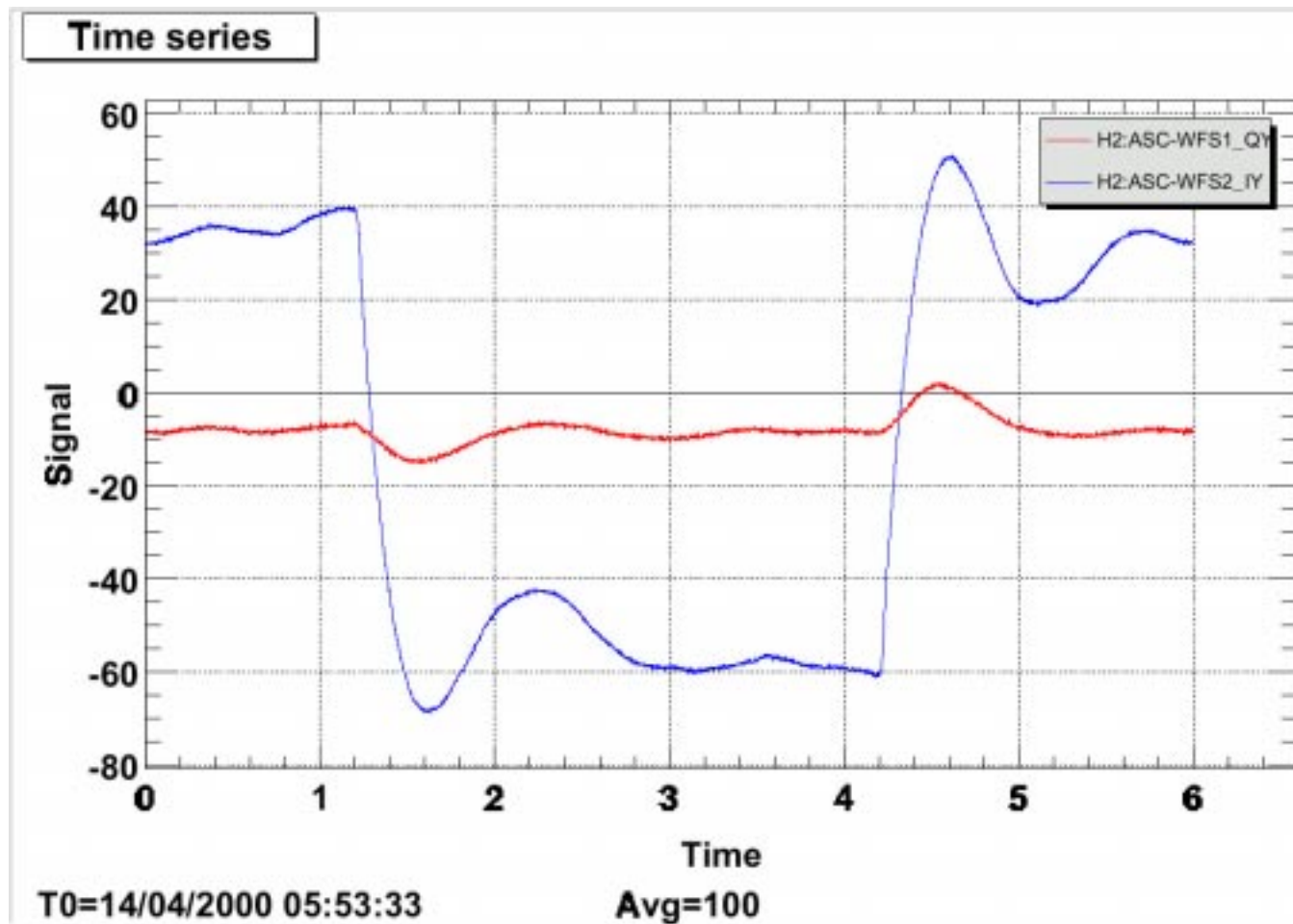
before

after





Auto-Alignment: Step Response

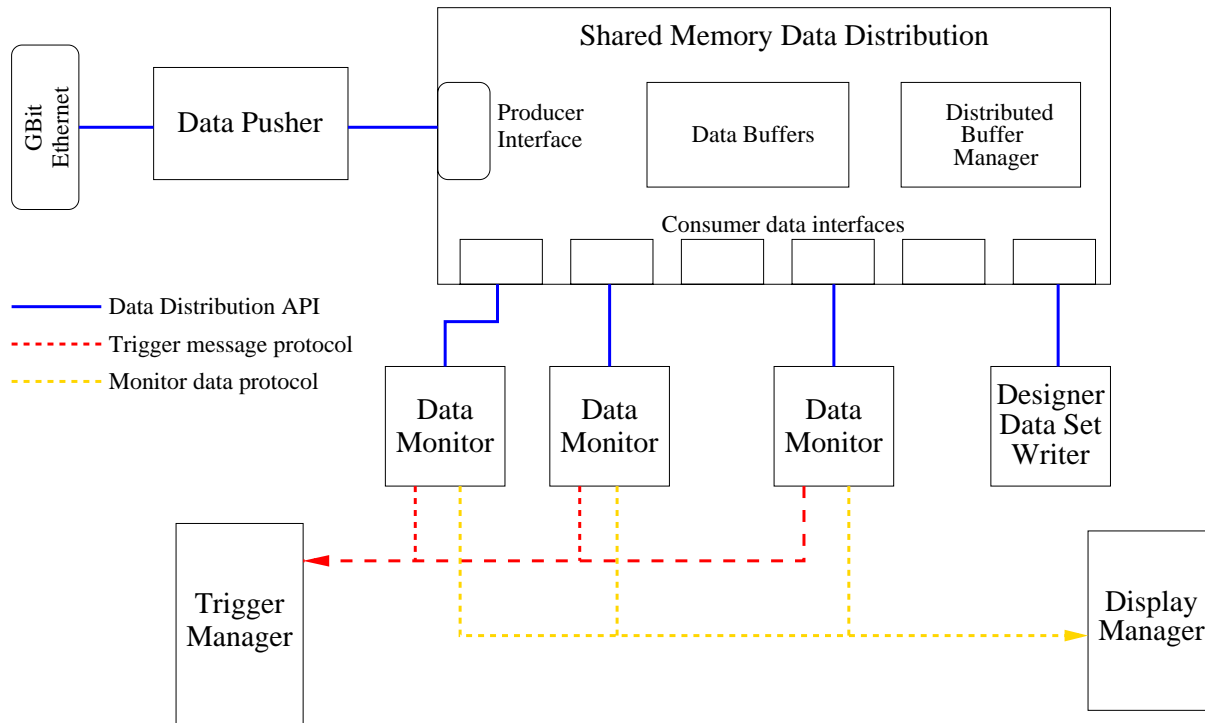




Data Monitor Tool

- ❑ Detect and tag known signals and disturbances
 - Find and record transients
 - Correlate external effects to operational parameters
- ❑ Measure and summarize the running state
 - Noise spectra, average power, other operational parameters
 - Rate and magnitude of known transient signals
- ❑ Notify operators of faults or abnormal conditions
 - Increases in all or part of noise spectrum
 - Other device specific problems
- ❑ Support interactive testing and diagnosis

Online Process Model



- Current data distributed via shared memory in each host.
- Any number of monitors access current data independently
- Trigger Manager routes triggers to operators/meta-Database.
- Monitors serve data for Status Reports & Displays.

Monitor Software Structure

- Skeleton (Base class)
 - Provides all house-keeping functions for monitor program
 - Fills time series with specified amount and type of data
 - Calls monitor processing function when requested data is available
- Monitor implemented as C++ class
- C++ Class Library
 - Input data (frames)
 - Output data (triggers, trends, html, frames)
 - Data containers (Time series, Spectra, Wavelet)
 - Signal processing (FIR & IIR filters, decimation, Line removal)
- Interactive operation with ROOT
 - All library classes available from ROOT
 - Macros for plotting LIGO data

Monitor Output Data Types

- Triggers (*meta*-Database records)
 - Document time & magnitude of transients / state changes.
 - Guide data analysis.
- Trend Frames
 - Contain averages, RMS & extrema vs. time.
 - Record long-term trends of channels or derived signals.
- Designer frames
- Status reports
 - Text or HTML.
 - Easy to implement; Useful but ugly.
- Monitor Data Displays
 - Data objects (time series, Spectra etc.) served by monitor
 - Monitor Display Manager displays selected objects.



Conclusions

Combination of

- High performance data acquisition system
- 24 hour disk cache
- New software and analysis tools

has enabled

- Fast learning curve
- Emphasis on analysis rather than data gathering
- Greatly enhanced remote diagnostics