

---

# LIGO Data Analysis System (LDAS)

## Status

A. Lazzarini

S. Anderson (LIGO/SRL)

J. K. Blackburn

W. Majid

T. A. Prince(LIGO/SRL)

S. Tilav

B. Allen(U.Wisc/Milwaukee)

B. Mours(VIRGO)

J. Salmon (Caltech/CACR)

B. Sears (Caltech/SRL)

R. Williams(Caltech/CACR)

PAC Meeting

LIGO Livingston Observatory (LLO)

23 - 24 April 1998



# Data Analysis System for LIGO I Status

---

- Design Requirements Review completed 12/97
  - ›› Provide on-line analysis at the observatories; data distribution from on-line cache -- diagnostics.
  - ›› Process and reduce the raw LIGO datasets at the off-line center to prepare the data for archival storage and retrieval.
  - ›› Provide computational and storage resources for off-line analysis using the archived data
  - ›› Provide a flexible design which can be reconfigured to reflect new analysis or computational requirements as they evolve.
  - ›› Provide access to LIGO data from all LIGO Laboratory sites and also from member institutions of the LIGO Scientific Collaboration for the LIGO I search.

# Data Analysis System for LIGO I Status

- MOU drafted [LIGO-M980029]
  - ›› Under review by LIGO (Barish) and CACR (Messina)
  - ›› CACR would be an institutional member of the LSC
  - ›› Identifies specific areas of mutually beneficial collaborative research:
    - Data archival and retrieval
      - > 20TB systems
      - > 20MB/s throughput
    - Linux/PC (Dec Alpha?) based compute cluster development
      - Low cost supercomputers
      - ~ 20 GFLOPS
      - ~ 32 - 64 nodes
      - Fast ethernet dedicated interconnection
      - MPI-based parallel computation
    - Data server/data distribution over the net
    - Use of CACR supercomputer resources for advanced LIGO data analysis going beyond LIGO computing resources
      - Resources will be made available to LIGO Laboratory (all sites) and members of LSC involved in LIGO I search
    - Mutual commitment to developing future research programs/proposals to expand resource base as data analysis needs grow

# Data Analysis System for LIGO I

## Status: Testing & Prototyping

---

### 1. Networking & Storage Systems

- A list of networks and storage facilities being investigated for use by LIGO:
  - ›› **HIPPI**: High Performance Parallel Interface.
    - 800Mb/s "super computer" network at Caltech (CACR).
    - HIPPI-FP Framing Protocol: Low level HIPPI protocol used when superior performance to vendor supplied TCP/IP is required (often an order of magnitude).
  - ›› **ATM**: Asynchronous Transfer Mode.
    - 155Mb/s fibre based network being adopted by LIGO.
  - ›› **100BaseT**: Twisted pair Ethernet.
    - 100Mb/s Ethernet wire used to network "pile-of-PCs".
  - ›› **HPSS**: High Performance Storage Solution.
    - Hierarchical Storage Management (HSM) system being investigated as the primary archive for LIGO data.

# Data Analysis System for LIGO I

## Status: Testing & Prototyping

### 2. Point-to-Point Bandwidth Tests

#### >> ATM ↔ ATM

- Sun ↔ Sun: 12MB/s
- Sun ↔ IBM: 6-14MB/s (TCP vs UDP)
- IBM ↔ IBM: 10-15MB/s (TCP vs UDP)
- 

#### >> 100BaseT ↔ 100BaseT

- Pile-of-PCs ↔ IBM: 4.5-6MB/s

#### >> HIPPI ↔ HIPPI

- IBM ↔ IBM: 8.5MB/s (TCP)
- Paragon ↔ Paragon: 0.6-2MB/s (TCP)
- Paragon ↔ Paragon: 65MB/s (interface nodes HIPP-FP)
- Paragon ↔ Paragon: 30MB/s (generic nodes HIPP-FP)
- Paragon parallel filesystem ↔ IBM: 30MB/s

#### >> ATM ↔ HIPPI (using Gigarouter)

- Sun ↔ IBM: 1MB/s
- 

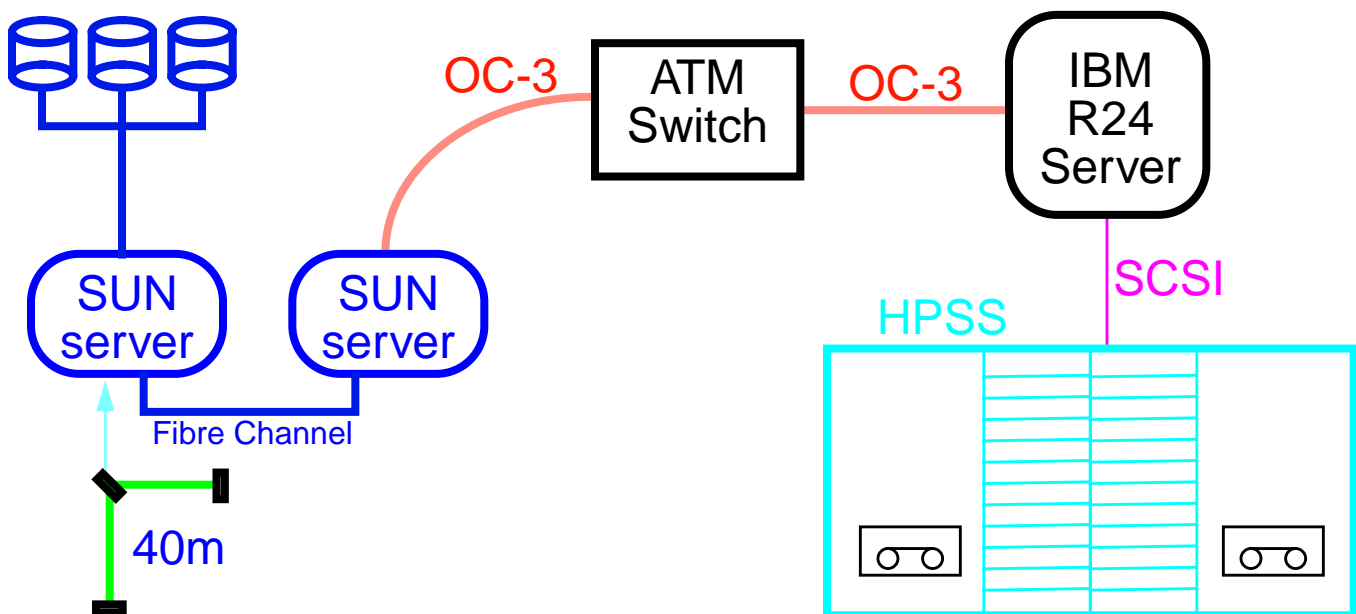
- Note: single connection bandwidths.
  - Some rates suffer from relatively high software latencies and may be significantly improved by running multiple parallel connections.

# Data Analysis System for LIGO I

## Status: Testing & Prototyping

### 3. 40m Data Transmission Tests

- Data rate of 2.4MB/s has been obtained in archiving 40m frames into HPSS.
  - ›› 600kB/s for sequential transfer (presently relatively small) 705 kB frame files
  - ›› Data transfer occurs in bursts of 2.4MB/s as verified with larger files
  - ›› Could (presumably) be obtained for small files with multiple concurrent ftp sessions
  - ›› Current performance is limited by the sequential aspect of the HPSS ftpd.
    - Will be overcome by changing from ftp to ftpd once additional ATM switching hardware is installed at CACR.

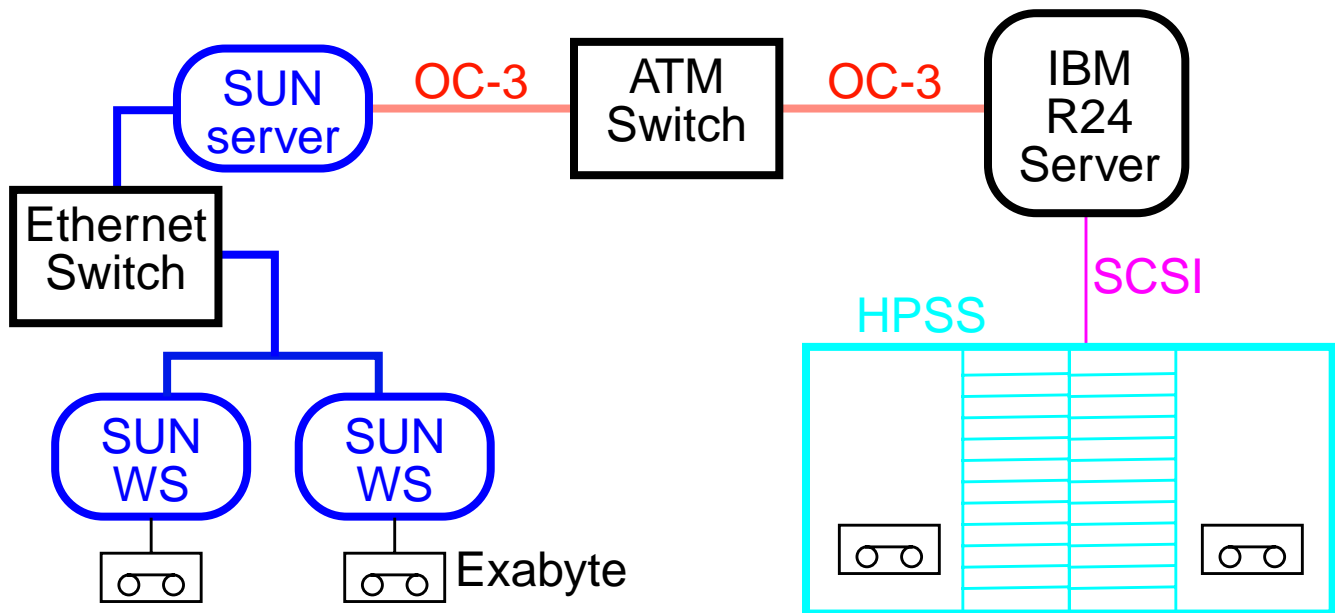


# Data Analysis System for LIGO I

## Status: Testing & Prototyping

### 4. Radio Pulsar Tests

- ›› As a test in robustness and as an interim data set until the 40m network connectivity was finalized in April 1998, 1 TB of radio pulsar data has been archived into HPSS.
  - During the transfer of 1,016,304,861,184 Bytes as 3041 separate files several bugs in HPSS, ftp, and the HIPPI driver have been isolated and have been or are planned to be fixed by IBM.



# Data Analysis System for LIGO I

## Status: Testing & Prototyping

---

### 5. Fourier Transform Tests

- ›› Small (1K complex): single node
  - PentiumPro 200MHz (FFTW): 105MFLOP/s
  - Ultra II 300MHz (FFTW): 275MFLOP/s
  - PA8000 180MHz (VECLIB): 255MFLOP/s
  - RS6000 132 (ESSL): 344MFLOP/s
- ›› **Medium (1M complex):** single node
  - PentiumPro 200MHz (FFTW): 40MFLOP/s
  - Ultra II 300MHz (FFTW): 87MFLOP/s
  - PA8000 180MHz (VECLIB): 193MFLOP/s
  - RS6000 132MHz (ESSL): 196MFLOP/s
- ›› Large (1G complex): multiple node
- ›› 512 node Delta: 1GFLOP/s (hypercube code)
- ›› 16 node Exemplar: 620MFLOP/s (1 hypercube)
  - HP's parallel FFT library does not work well on multiple hypernodes (16 processors each) by design.
  - Note, the MPI version of FFTW is under testing as a possibility for a portable large in-core FFT library. However, the current version has uncovered a bug in the Intel implementation of MPI.
- ›› Extra Large (1TB): multiple node + disk
  - Work in progress on porting existing code to CACR Beowulf (rewrite as a portable MPI library?)



# Data Analysis System for LIGO I

## Status: Testing & Prototyping

---

### 6. Data distribution server

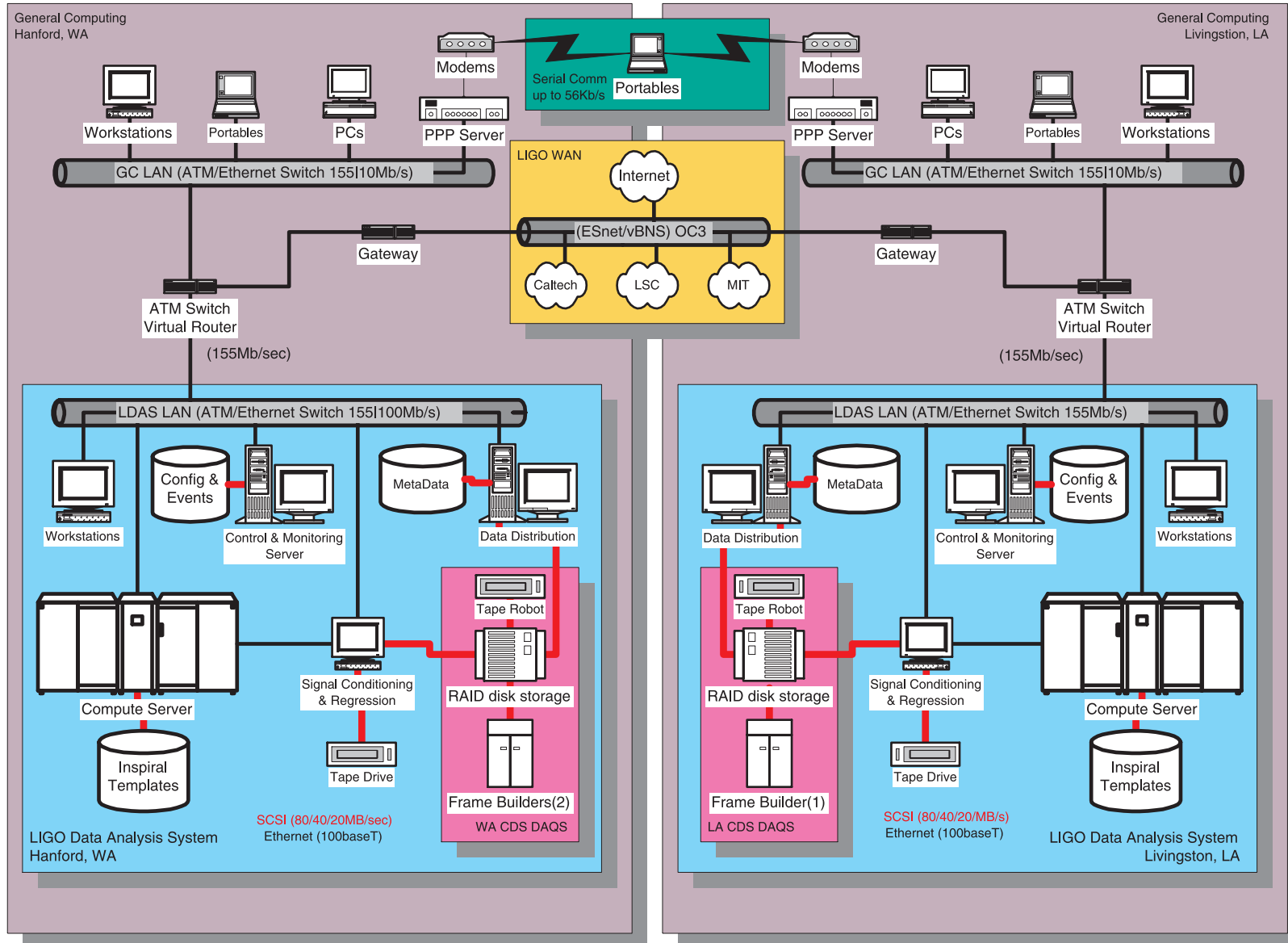
- 50GB RAID system (seamlessly extensible to 250 GB)
  - ›› Will be back end of 40m DAQS prototype
  - ›› Serves as interface to LDAS - per architecture plan
  - ›› Hardware at 40m lab; awaiting availability of servers & personnel to implement
  - ›› Two 300 MHz UNIX servers ordered to stage data of RAID system via SCSI interfaces
    - Replaces present pathway from 40m DAQS to CACR/HPSS
- Example web client interface built by CACR staff
  - ›› Limited distribution of data at present
  - ›› Available for downloading/viewing/listening to 40m data
  - ›› Starting to develop a metadata catalog model for summary descriptors which will be used for LIGO frame data

### 7. Linux PC compute cluster

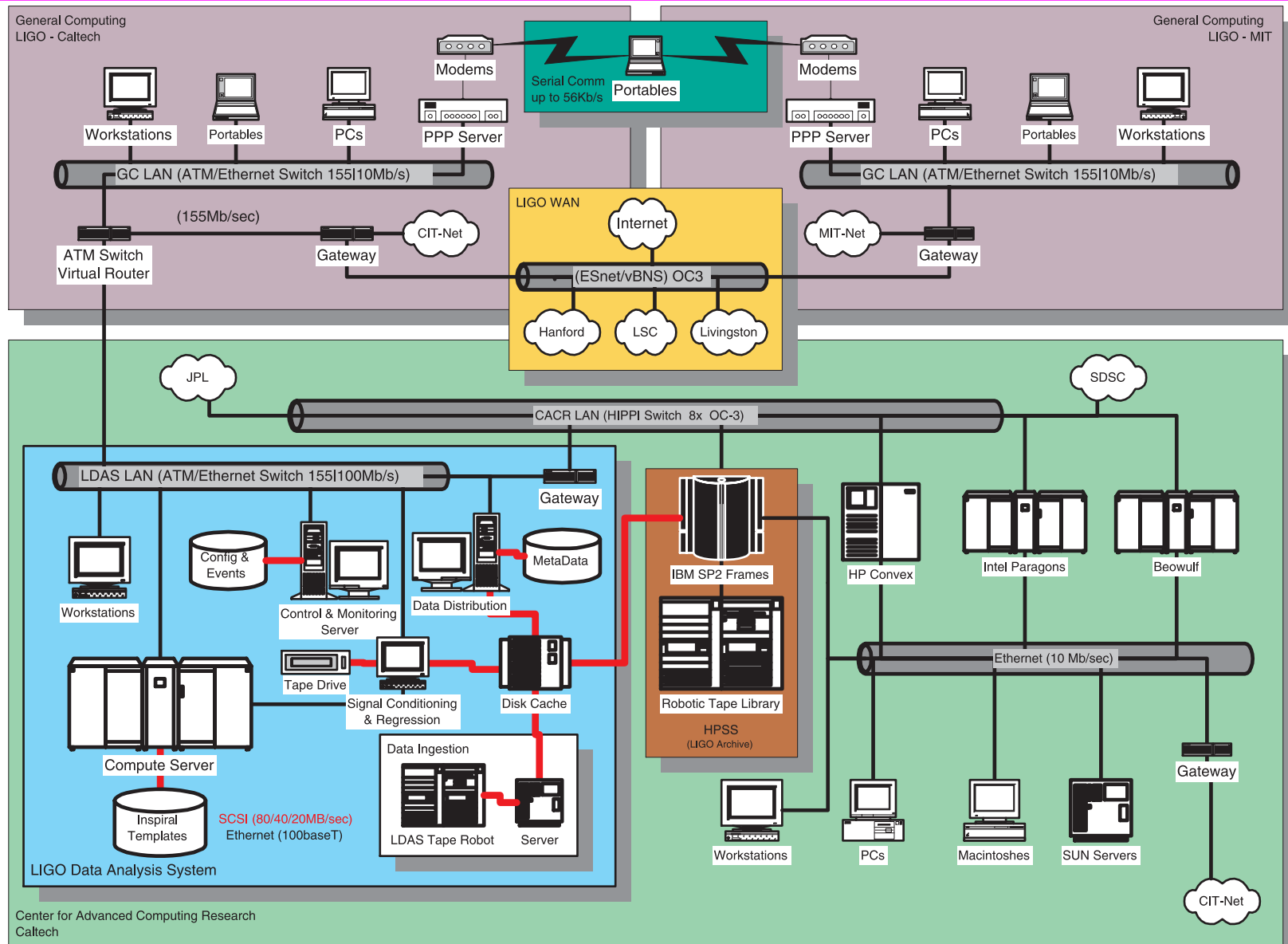
- ›› In-house development deferred in favor of data transmission and data distribution activities
  - Prototyping will likely first take place at U. Wisc. (Allen) as an LSC-based activity.

# LIGO Data Analysis System

## On-line architecture



# LIGO Data Analysis System Off-line Architecture



# Data Analysis System for LIGO I

## Status: Design

---

- **Software system design**

- Data analysis -- scope: LSC + LIGO Laboratory**

- ›› Data analysis flows - sizing of requirements:
    - ›› Data analysis software prototyping - GRASP code
    - ›› Data usage model

- Data management -- scope: LIGO Laboratory**

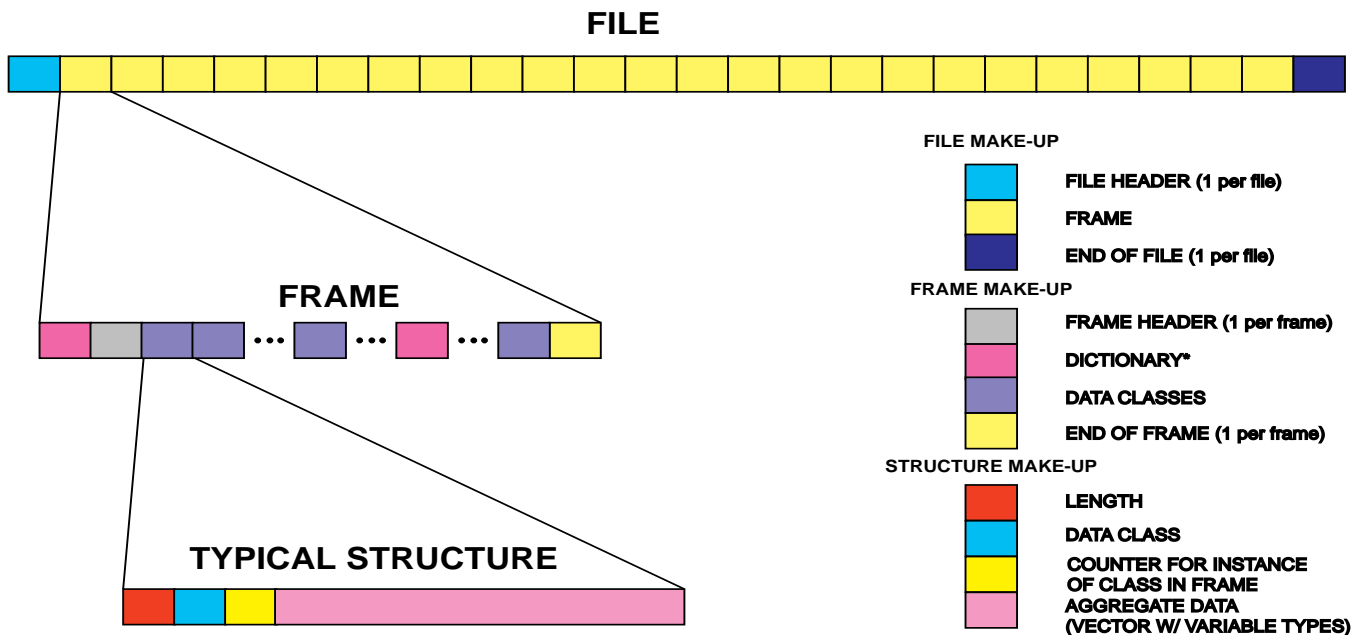
- ›› Design and definition of architecture & components:
    - ›› Data distribution & access
      - Storage systems & archives
      - Data transmission & downloading
    - ›› Metadata creation/archival/retrieval
    - ›› API design/development
      - Data ingestion (incorporation of new/recent data)
      - LDAS command language
      - Interprocess communications -- LDAS distributed data analysis manager
      - Disk cache management
      - Access to data libraries - Frame/LW/Metadata/Event
      - Filtering/MPI/Conditioning
      - Control/Reporting

# LIGO Data Analysis System Software Design

## DATA FORMAT STANDARDS

### FRAMES:

- Developed by VIRGO (Benoit Mours)
- Standardization defined by LIGO and VIRGO (*LIGO-T970130-B*)
- Accepted by GEO600 and TAMA
- Primarily developed for data acquisition and data archival
- Frame data attributes grow out of C language structures
- Frame I/O library (in C) and documentation available from Web see (<http://lapph.in2p3.fr/virgo/FrameL>)
- MATLAB I/O interface to Frames also available
- C++ version of library and API under development at LIGO & VIRGO



\* Dictionary structure behavior is unique in that:

1. It precedes header for first frame of file;
2. Dictionary is built up incrementally as additional structures are incorporated into frame
3. It is valid for entire file (persistent)

# LIGO Data Analysis System Software Design

---

## *DATA FORMAT STANDARDS*

### **Light-Weight Format:**

- Candidates - SDF being developed by CACR (John Salmon)  
CDL/NetCDF developed by Unidata Program Center
- Primarily intended as an easy to use data format
- May also be used for communication of data between distributed processes

### **Other Data Formats:**

- Meta Data - data about data; candidates include database solutions, distributed, accessible through web browsers
- Event Data - data about events (analysis/filter results) includes triggers, diagnostic filters, GW filters
- MPI Data - data communicated between MPI processes (MPI standard)

# Data Analysis System for LIGO I Metadata (Partial List)

Basis of size estimate

SOURCE	Data	Data Types	#Parameters		#Bytes	#/Hr	Data Volume/Year [GB]
			#Bins	#Pixels #Samples/s			
LIGO - Interferometer	Machine state vector	Text: Parameters	1000		4	10	0.4
	Snapshots	Video	2500		1	10	0.3
		Audio	40000		1	0.002	3
		Spectra; fast scopes	1024		4	100	4
	Frame Data Catalog	Text/Strings	1024		1	10	0.1
	Operator Logs	Text/Strings	512		1	20	0.1
		Graphics	250000		1	20	50
		Spectra	2048		4	20	2
	Diagnosics - Calibrations	Spectra	2048		4	20	2
	Diagnosics - Triggers	Matrices/coefficients	100		4	1	0.004
		Text: Model parameters	1024		1	1	0.01
		Discrete logic	100		4	3600	14
LIGO - PEM	Facilities state vector	Text: Parameters	512		1	3600	18
	Seismometers	Spectra	1024		4	360	15
	Magnetometers	Spectra	2048		4	360	29
	Tiltmeters	Time Series	0.016666667		2	3600	0.001
	Microphones	Spectra	10000		2	360	72.0
	Diagnosics - Calibrations Diagnosics - Triggers	Matrices/coefficients	100		4	1	0.004
		Text: Model parameters	1024		1	1	0.01
		Discrete logic	100		4	3600	14

# Data Analysis System for LIGO I Metadata (Partial List)

## LDAS Metadata List - Preliminary

Basis of size estimate

SOURCE	Data	Data Types	#Parameters		#Bytes	#/Hr	Data Volume/Y ear [GB]
			#Bins	#Pixels #Samples/s			
Non-LIGO	Seismic	Text: Parameters	512		1	10	0.1
	Electromagnetic storms	Text/Strings	256		1	100	0.3
	Astrophysics - GRBs	Text/Strings/Parameters	256		1	0.04	0.00
	Astrophysics - neutrinos	Text/Strings	256		1	0.04	0.00
	Astrophysics - visible	Text/Strings	256		1	0.0416667	0.00
	Astrophysics - gravitational	Text/events	2048		1	10	0.2
LDAS Events	Event Lists	Text: Parameters	2048		1	10	0.2
		GW channel snap shots	1638400		2	2	66
		Images/Graphics	500000		1	10	50
LDAS Specific	Catalogs	Text					0.0
<b>Total Metadatabase == &gt;</b>							<b>341</b>



# LIGO Data Analysis System Software Design

---

## *SOFTWARE STANDARDS*

### Languages:

- ANSI C++ will be primary development language for compiled components to be maintained and supported by LIGO
- ANSI C will be used where wrappers are needed to bind C++ components with C, FORTRAN and TCL components
- TCL (Tool Control Language) will be used in the LDAS components responsible for controlling the LDAS software system
- TK will be used for the Graphical User Interface components
- Tclets (TCL/TK plug-ins) will be used for web browser connectivity into the LDAS software system.
- A TBD database will be used to integrate metadata into the LDAS software system

### Communications:

- TCL layer sockets will be used to communicate commands and messages between components (processes) of the LDAS
- C++ socket class library will be used to communicate data between components (processes) of the LDAS

### Libraries:

- C++ Class Libraries, numerical libraries, I/O libraries will be dynamically linked and built as shared libraries on supporting platforms for efficient use of hardware resources

# LIGO Data Analysis System Software Design

---

## *VERSION CONTROL*

### Software Development:

- Development cycle will include CVS (Concurrent Version System) as the software management system
- LIGO will be responsible for the LDAS CVS software repository
- A secure CVS shell will allow remote check-ins and check-outs from the LIGO LDAS repository

### Software Style:

- The LIGO systems integration team has written a recommended software style guide (*LIGO-T970211-00-E*)

### Releases:

- Releases of the LDAS software system will be in the public domain and available via FTP and WWW
- User Interfaces to the LDAS (shells and GUIs) will be available via FTP and WWW
- New releases will be announced over the Web and via email

# LIGO Data Analysis System Software Design

---

## *SOFTWARE ARCHITECTURE*

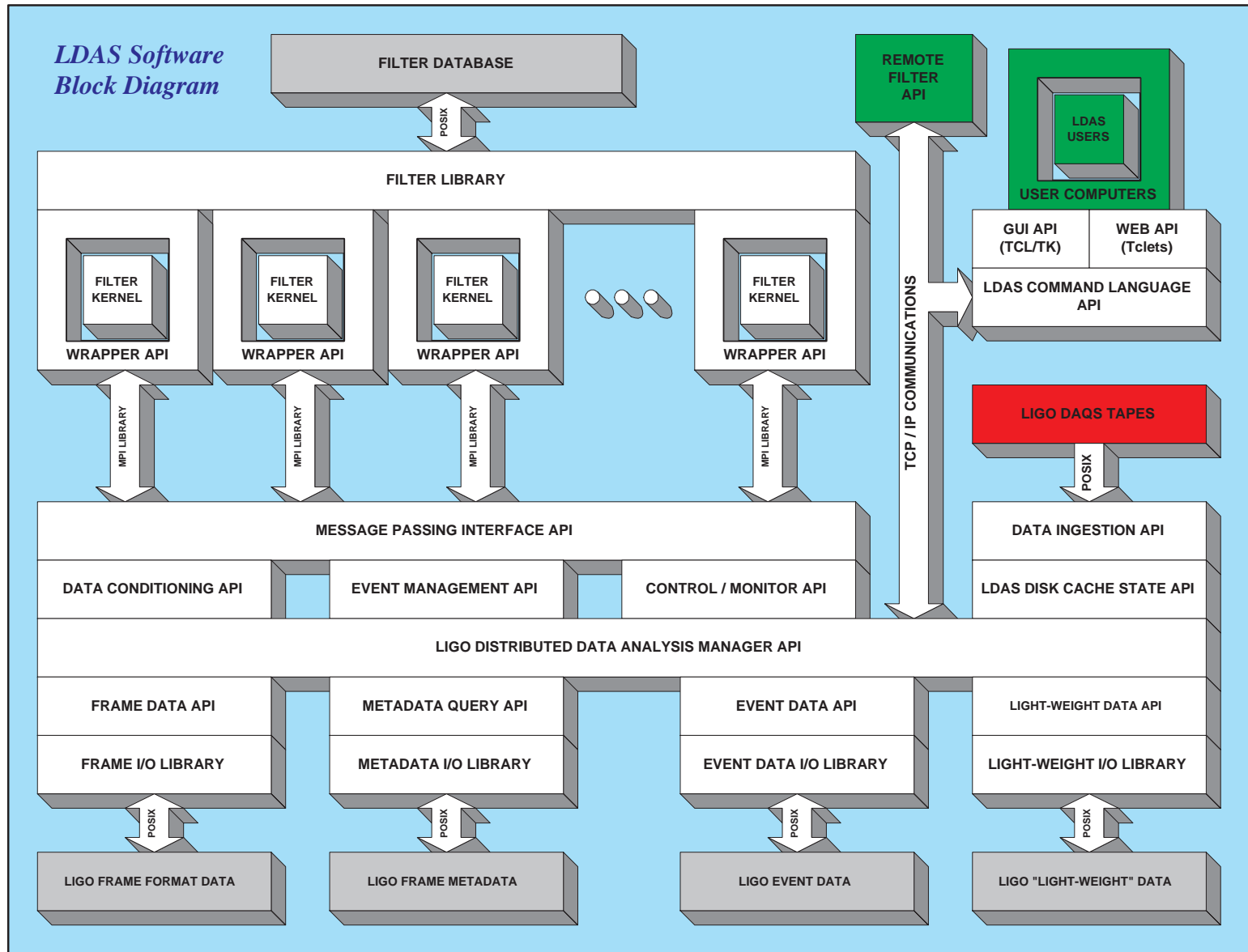
### Distributed Components:

- Software components will be integrated into a distributed system using MPI and sockets for inter-process communications
- A distributed software manager will act as the master, serving users and coordinating the connections between all system components
- User interfaces based on TCL/TK will act as clients, making requests for services from the LDAS system through the distributed software manager

### API Components:

- The manager will see a standard interface(API) to all the software components

# LIGO Data Analysis System Software Design



# LDAS SOFTWARE DESIGN

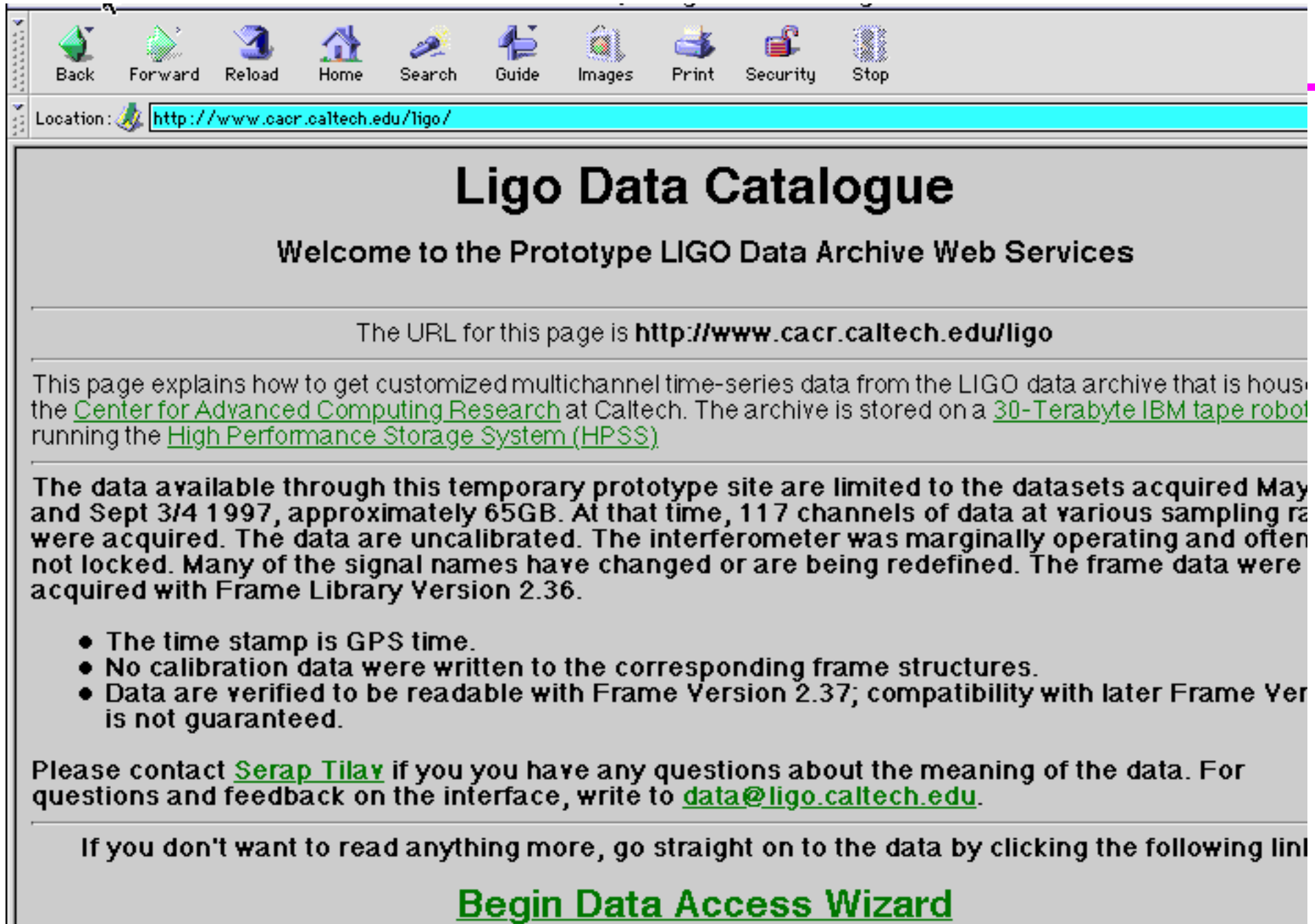
## Software Prototyping: TCL/TK Layer

**Demo:** TCL Calculator Engine (*server*) running on a Sun Workstation computes for TCL/TK clients running in *netscape* and a *wish shell* on a PC...also a keyboard interface to the server's socket using the *telnet session!*

The screenshot displays a desktop environment with three main windows:

- Netscape:** A web browser window showing a page titled "LIGO Data Analysis System Tclet Demo" by Dr. James Kent Blackburn at the California Institute of Technology. The page contains text explaining a client-server example for math operations and a calculator interface with inputs 5 and 7, and a result of 35.
- frontend:** A window showing a calculator interface with inputs 5 and 7, and a result of 0.714285714286.
- Telnet - polaris:** A terminal window showing a telnet session. The user 'kent@polaris' connects to 'polaris.ligo.caltech.edu' and performs calculations: 'add 5 9 %v' resulting in '14', and 'subtract 24 19 {difference is %v}' resulting in 'difference is 5'.

## S/W Prototyping: Data Server Model



The screenshot shows a web browser window with a toolbar containing icons for Back, Forward, Reload, Home, Search, Guide, Images, Print, Security, and Stop. The address bar shows the URL <http://www.cacr.caltech.edu/ligo/>. The main content area features the title "Ligo Data Catalogue" and a subtitle "Welcome to the Prototype LIGO Data Archive Web Services". Below this, it states the URL for the page and provides a detailed description of the data archive, including its location at Caltech and its storage on a 30-Terabyte IBM tape robot. A list of bullet points describes the data's characteristics, and a contact information section is provided for further inquiries. A large green link "Begin Data Access Wizard" is prominently displayed at the bottom of the page content.

Location: <http://www.cacr.caltech.edu/ligo/>

# Ligo Data Catalogue

## Welcome to the Prototype LIGO Data Archive Web Services

The URL for this page is <http://www.cacr.caltech.edu/ligo>

This page explains how to get customized multichannel time-series data from the LIGO data archive that is housed at the [Center for Advanced Computing Research](#) at Caltech. The archive is stored on a [30-Terabyte IBM tape robot](#) running the [High Performance Storage System \(HPSS\)](#).

The data available through this temporary prototype site are limited to the datasets acquired May and Sept 3/4 1997, approximately 65GB. At that time, 117 channels of data at various sampling rates were acquired. The data are uncalibrated. The interferometer was marginally operating and often not locked. Many of the signal names have changed or are being redefined. The frame data were acquired with Frame Library Version 2.36.

- The time stamp is GPS time.
- No calibration data were written to the corresponding frame structures.
- Data are verified to be readable with Frame Version 2.37; compatibility with later Frame Version is not guaranteed.

Please contact [Serap Tilay](#) if you have any questions about the meaning of the data. For questions and feedback on the interface, write to [data@ligo.caltech.edu](mailto:data@ligo.caltech.edu).

If you don't want to read anything more, go straight on to the data by clicking the following link

[Begin Data Access Wizard](#)

# S/W Prototyping: Data Server Model

**Begin Data Access Wizard**

The archive consists of a catalog hosted by the [Access](#) database. The Wizard below is driven by the [WebBase](#) middleware. The archive consists of a collection of *Datasets*. Each of these represents a run of an interferometer over a compact time period, with the machine parameters remaining constant. The first page of the wizard asks you select one of these from a list.

Next you are asked to select the output format for the data you are requesting. For more information, see the web page on [LIGO Data Formats](#). Choices are **Frame**, **SDF**, **netCDF** (not yet implemented), and **audio**. For the latter you will need to set up your browser to accept audio files such as [this one](#).

If you choose Frame format, there is no choice about which channels you want -- you get them all. Otherwise, you get a page listing all the available channels corresponding to the Dataset that you chose. The SDF and netCDF formats contain several channels which are *all the same frequency*, and the audio output contains only one channel. Select the channel(s) that you want and click the "Next" button to continue.

Next you get a list of data segments to choose from, each a few minutes long. Select one of these, then choose which part of that segment that you would like, by typing in the offset, in seconds, from the beginning of the segment, and number of seconds you would like sent to you. Click the "Next" button to continue.

The final page shows the structure of the query that you have generated. When you click on the "Get Data" button, the server is activated and data will begin to flow. Since the data is on a tape robot, it may take a couple of minutes to start. The data will come with a MIME type that depends on the format you chose:

Frame	application/x-igwd-frame
SDF	application/x-SDF
netCDF	application/x-netcdf
audio	audio/basic

# LIGO Data Analysis System Software Design

## LIBRARIES LINKED ON LDAS HARDWARE

	libmpi.a	libframe.a	libmeta.a	libevent.a	libsdf.a	libfilter.a	libposix.a
Computer Server	X				X	X	X
Control & Monitor Server				X	X		X
Data Distribution Server		X	X		X		X
User Interface					X		
Data Ingestion Server		X			X		X
Data Conditioning Server		X			X		X

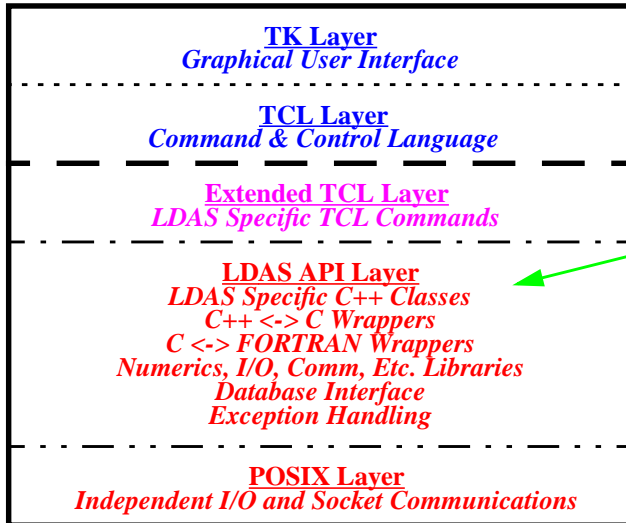
## LIBRARIES LINKED TO LDAS SOFTWARE

	libmpi.a	libframe.a	libmeta.a	libevent.a	libsdf.a	libfilter.a	libposix.a
Filter Wrapper Class	X				X	X	X
MPI API	X						X
Data Conditioning API					X		X
Event Management API					X		X
Control & Monitor API					X		X
Frame Data API		X			X		X
Meta-Data API			X		X		X
Event Data API				X	X		X
SDF API					X		X
Filter Kernel	Inherits				Inherits	Inherits	Inherits
GUI User Interfaces					TCL		
Remote Filter API					X	X	X
LDAS Command API					TCL		
Data Ingestion API		X			X		X
LDAS Disk Cache API					X		X
LDAS Manager API					X		X



# LIGO Data Analysis System Software Design

## APIs “GENERIC” LAYERED COMPOSITION

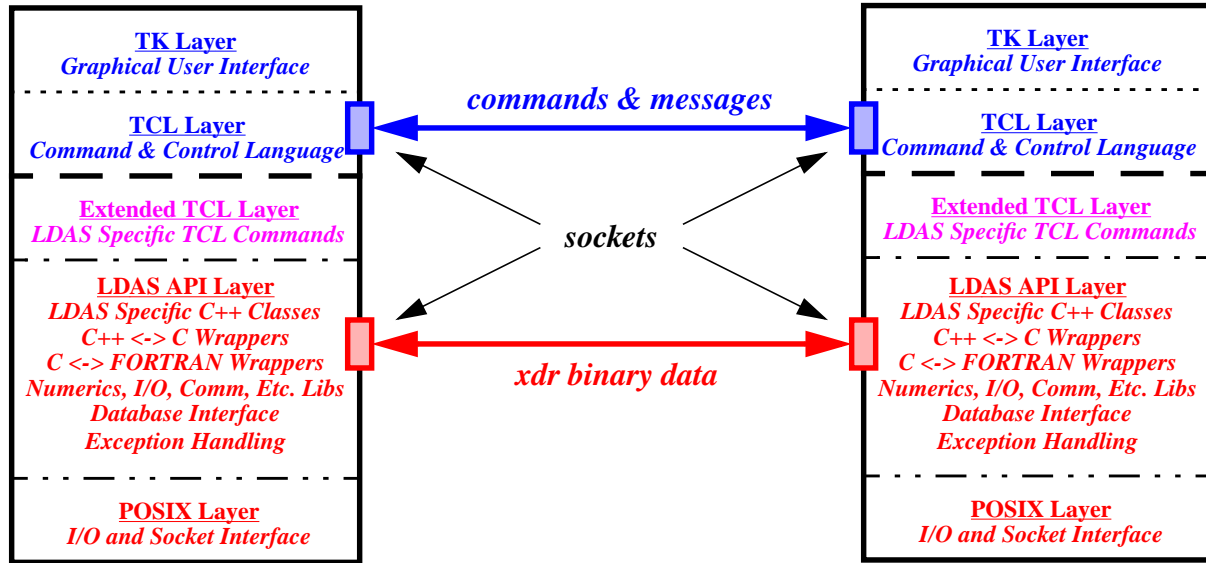


*LSC Designed Algorithms overload virtual functions found in LDAS C++ Classes*

APIs	TK (GUI)	TCL (Control)	Extended TCL(LDAS)	LDAS Applications
Message Passing Interface	OPTIONAL	YES	YES	YES
Data Conditioning	YES	YES	YES	YES
Event Manager	YES	YES	YES	YES
Control & Monitor	YES	YES	YES	YES
Frame Data	OPTIONAL	YES	YES	YES
Meta-Data	OPTIONAL	YES	YES	YES
Event Data	OPTIONAL	YES	YES	YES
Simple Data Format	OPTIONAL	YES	YES	YES
LDAS Command Language	GUI Front Ends	YES	OPTIONAL	YES
Remote Filter API	OPTIONAL	YES	YES	NO
Data Ingestion	YES	YES	YES	YES
LDAS Disk Cache	YES	YES	YES	YES
LDAS Manager	YES	YES	NO	NO

# LIGO Data Analysis System Software Design

## APIs “TWO-LEVEL” SOCKET COMMUNICATIONS

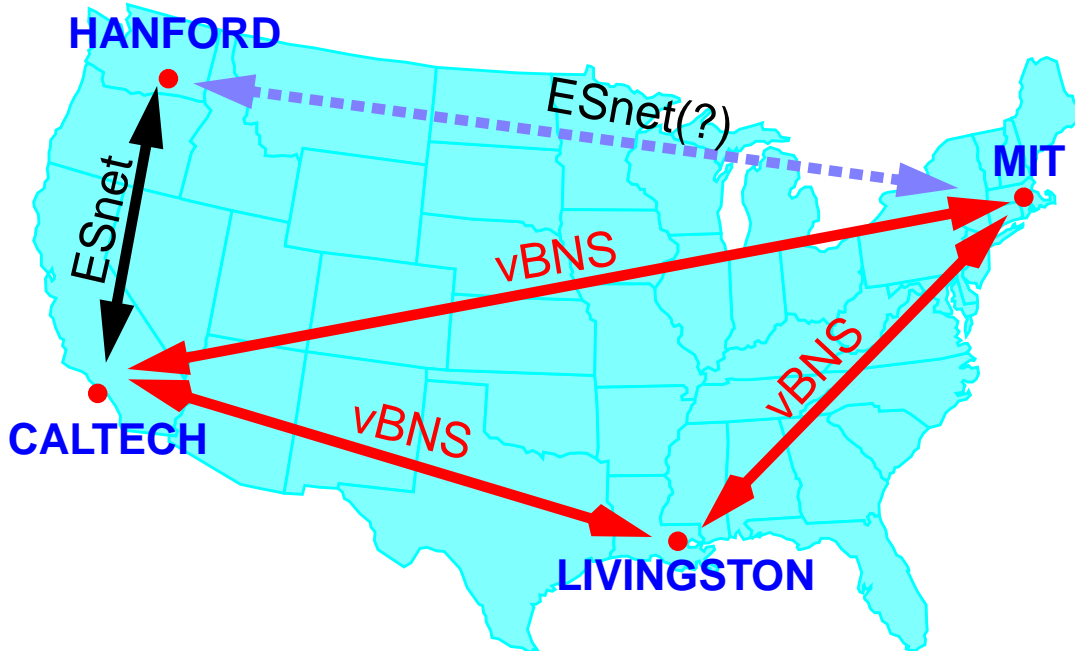


*Yellow boxes below indicate option to use SCSI*

<i>API</i>	FW	MPI	DC	EM	CM	FD	MD	ED	SDF	FK	COM	DI	DCS	MAN	RF
FW		mpi								inherit					
MPI	mpi	mpi	socket	socket	socket					inherit					
DC		socket			socket	socket	socket		socket					socket	socket
EM		socket		socket	socket			socket	socket					socket	socket
CM		socket	socket	socket		socket	socket	socket	socket					socket	socket
FD			socket		socket				socket				socket	socket	
MD			socket		socket		socket	socket	socket				socket	socket	
ED				socket	socket		socket	socket	socket					socket	
SDF			socket	socket	socket	socket	socket	socket					socket	socket	socket
FK	inherit	inherit													
COM														socket	
DI													socket		
DCS						socket	socket		socket			socket		socket	
MAN			socket	socket	socket	socket	socket	socket	socket		socket		socket		socket
RF			socket	socket	socket				socket					socket	

# LIGO Wide Area Network

## Plan



WAN/LAN Connectivity among LIGO Laboratory Sites

Site	Livingston, LA	Hanford, WA	MIT	Caltech
Caltech	vBNS(OC3)	ESnet (4 X T1) <-> vBNS(OC3)	vBNS(OC3)	OC3/ATM 100BT
MIT	vBNS(OC3)	MIT<->Caltech<->Hanford ESnet (4 X T1) <-> vBNS(OC3)	100BT OC3/ATM(?)	
Hanford, WA	ESnet (4 X T1) <-> vBNS(OC3)	OC3/ATM 100BT		
Livingston, LA	OC3/ATM 100BT			

# LIGO Wide Area Network

## Status

---

- LIGO proposed & drafted an MOU between NSF/DOE to provide access to ESnet at Hanford
  - ›› Final MOU complete: awaiting signatures at NSF, DOE
  - ›› Proceeding to implement initial (T1) capability; requested up to 4 x T1 BW (cost is an issue).
  - ›› SOW/PO with PNNL & Lockheed-Martin to procure switching & routing equipment almost complete
  - ›› Cross-over between ESnet and vBNS takes place at SDSC
  - ›› Routing: LHO-PNNL-SDSC-CACR-LIGO/Caltech
    - Awaiting IP address space release by ESnet
    - Testing has been done to establish quality of service between SDSC & PNNL: excellent.
    - Will conduct similar testing between CACR and SDSC - no surprises expected
  - ›› MIT may be added later as a separate addendum to MOU (reduce to several 2-body problems!)

# LIGO Wide Area Network

## Status

---

- Issue: ESnet link is not extensible beyond ~ 2 x T1 due to costs (we have requested extensibility to 4 x T1)
  - ›› WSU/Pullman (~ 100km NE) awarded an NSF grant to establish a vBNS hook-up
  - ›› UW/Seattle (~350 km W) has vBNS at present
  - ›› LIGO is exploring the possibility of a consortium to propose to NSF a direct vBNS hookup in Tri-Cities area with PNNL (EMSL: R. Bair) and WSU/Richland extension campus (J. Judy)
    - Follows model in place at Livingston
  - ›› Interest is “moderate”
    - EMSL needs better connectivity to Seattle, Caltech, SDSC
    - WSU/Richland has no present high-bandwidth connection to internet -- small budgets
  - ›› LIGO is an “outsider” and would like a local advocate who knows Washington state telecommunications infrastructure (State and commercial sectors)
  - ›› Not obvious 31 July 1998 deadline for proposal to NSF can be met

# LIGO Wide Area Network

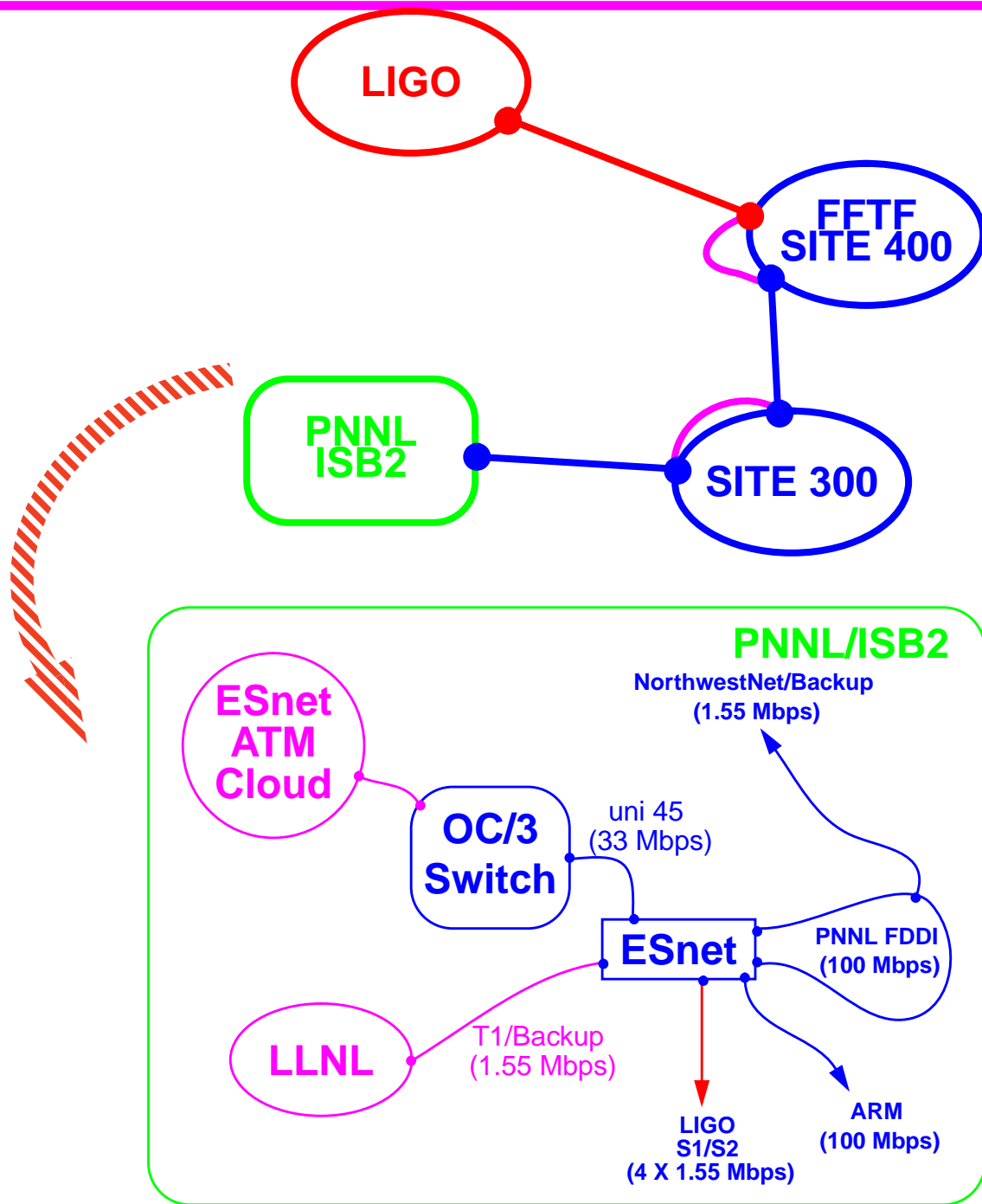
## Status

---

- T1 link to Livingston Observatory is in place
  - ›› LSU provides gateway service
  - ›› Recent proposal by LSU to NSF for vBNS connection includes LIGO access at Livingston
  - ›› Future FO link from observatory to campus via Bell South switch near Livingston [present link uses Cu].

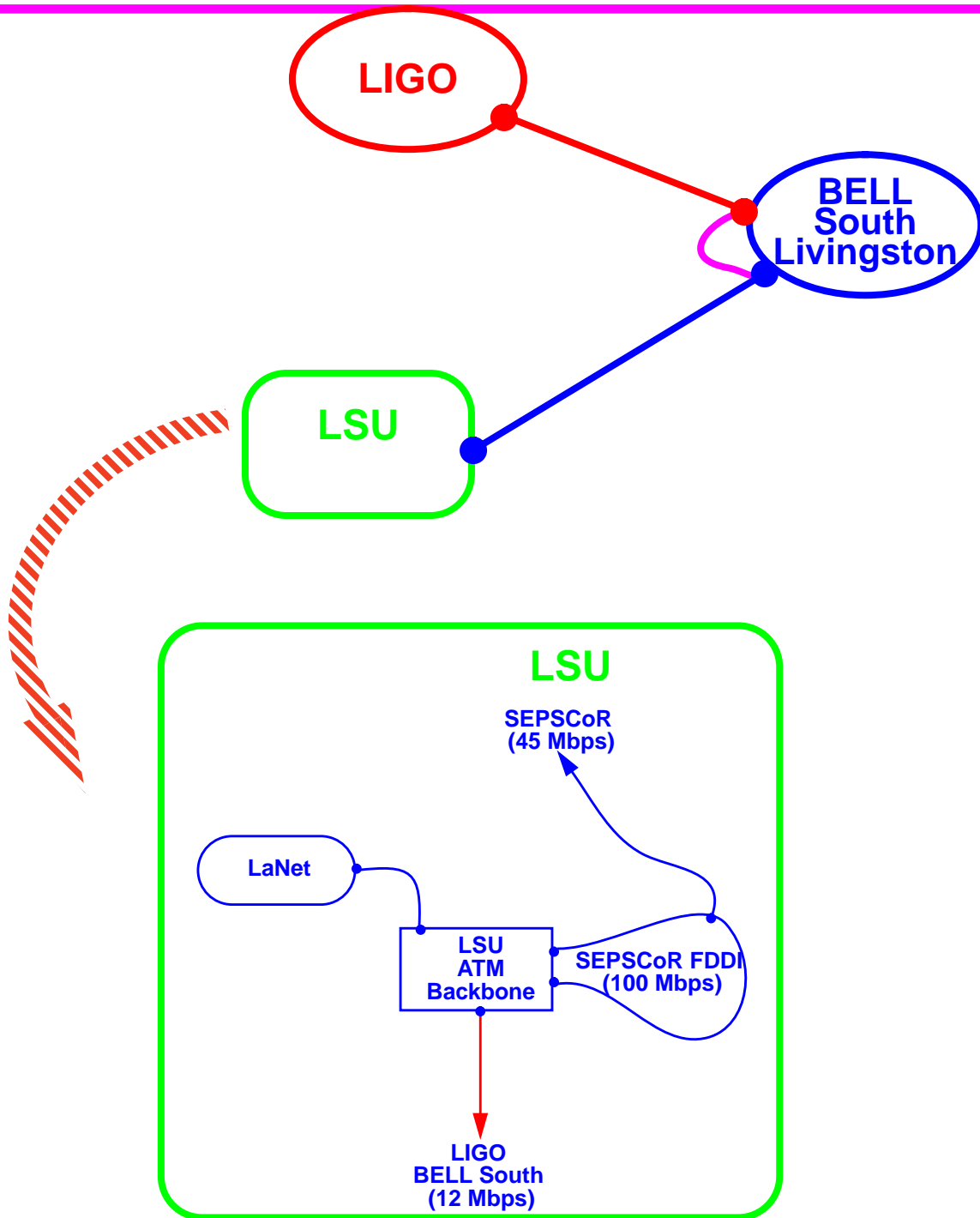
# LIGO Hanford WAN

## Link to ESnet



# LIGO Livingston WAN

## Link to LSU/vBNS





# LIGO Data Analysis System To-Do List

- Design & definition

- ›› Data -- channel lists/frame contents/types of frames/...
- ›› Metadata -- contents/environment
- ›› Algorithms -- hierarchical searches/periodic searches/f-t processing/wavelets/...
- ›› Events -- definition
- ›› LDAS architecture -- complete design/definition

- Development & prototyping

- ›› LDAS command language syntax
- ›› Scripting language implementation -- interprocess control & communication
- ›› Data distribution -- 40m implementation
- ›› Compute server -- BEOWULF cluster; integrate ~8 node cluster
- ›› Algorithms -- same as above
- ›› Benchmarks -- algorithms; data distribution;...
- ›› Visualization tools
  - Applets
  - Plug-ins
  - AP Interfaces to commercial/extant products
    - Matlab, IDL, Triana(GEO), ...
- ›› Data transmission -- quantify WAN performance/limitations

# LDAS Development Timeline

- Highest priority: staged implementation of on-line systems to support detector commissioning:

Detector Milestone:	Date	LDAS Need
>> Data Acquisition System, 2km:	9/98	Min. data dist.
>> PSL/Input Optics	2/99	“
>> Vertex Michelson, first light	7/99	Full data dist.
>> 2km operational	6/00	On-line system

- 4 km interferometers staggered in time by 3 & 6 mos.
- Staged installation at CACR of off-line system in period 6/99 - 12/01

— .

# Data Analysis System for LIGO I

## Summary of Recommendations from PAC Report of 11/97

---

- Data Model

- ›› Have started prototyping an access/filter/retrieval example session
- ›› Cooperating with CDS/Diagnostics group
  - Ensuring uniformity of software (where reasonable)
  - Data access & retrieval
  - Data type definition
- ›› Developing metadata list
- ›› Revised interface to CDS/Detector tape storage
  - CDS to provide on-line large RAID disk system
  - LDAS to provide tape writing capability to ensure seamless compatibility with tape archiving system at CACR
- ›› Frame format specification under configuration control
  - see URL [http://docuser.v.ligo.caltech.edu/~prince/data\\_dump.html](http://docuser.v.ligo.caltech.edu/~prince/data_dump.html); document # T970130.pdf
- ›› C++ class library definition/design in process
- ›› CACR MOU under review
- ›› Data archive system (e.g., HPSS) still under evaluation
  - May consider a “home-grown” *datawolf* system
- ›› Data compression will be implemented when needed
- ›› Have begun hiring necessary programming staff to carry forth implementation