LIGO - LSC Meeting on Data Analysis

0900 - 1630 PST **26 February 1999**

Agenda

- LDAS Overview by LIGO 0900 - 1200
  - Requirements
  - Software architecture & design
  - Hardware implementation
  - Development timeline
  - Procurement strategies
- Lunch 1200
- Models for organizing the LSC data analysis effort 1300 - 1400
- Allocation of LIGO Laboratory resources for various searches 1400 - 1430
- CACR Involvement in LIGO 1430 - 1500
- Other LSC issues/activities 1500 - 1630
  - Astrophysical signatures
  - Detection confidence andvalidation
  - Detector characterization
  - Plans for the Data Analysis White Paper

LIGO Data Analysis System (LDAS)
Requirements *(from 12/97 review)*

- Assumptions/Dependencies
  - Detector delivers a fully functional DAQS. Data are written in frame format to a disc cache system available to the on-line LDAS. Iincludes the availability from DAQS of data-valid logic flags to identify saturated or aliased waveforms.
  - Detector implements interferometer diagnostics system. LDAS does not need to provide real-time (signal) feedback information to the LIGO interferometers. LDAS - Diagnostics System interface shall be primarily through the operator or scientist. Data or parameters derived by diagnostics routines will be done ~~through~~ using frame-based data.
  - LDAS, together with DAQS, will provide for an on-line (volatile) data storage system capable of accommodating a volume of data sufficient to provide overlap between shifts.
  - LDAS goal shall be to process datastream in real-time and on-line. This includes providing for the exchange of detection event lists between LIGO sites.
  - The off-line system does not directly interface to the on-line system. Connectivity is not precluded; not part of network topology at present.
  - Data reduction shall be accomplished as far upstream in the data acquistion process as possible in order to enable LIGO to archive reduced datasets for at least ~~5~~ 2 years. As a target, a minimum volume reduction of 10X is assumed. As a minimum, the GW channel, calibrated in strain, shall be archived permanently.

LIGO Data Analysis System (LDAS)
Requirements *(from 12/97 review)*

- Assumptions/Dependencies
  - Specifically not considered to be within the scope of the LDAS are:
    - Data analysis functions performed at centers other than the LIGO Laboratory Facilities.
    - The on-line diagnostics system used for stimulus-response characterization, transfer function determination, and calibration functions. However, it is expected that software developed for the LDAS will find utility within the diagnostics system (e.g., numerical algorithm libraries, database tools).
    - Simulations shall be provided separately from, but coordinated with, the LDAS. The interface shall be using frame-based representations of simulation outputs.

LIGO Data Analysis System (LDAS)
Requirements *(from 12/97 review)*

- Mission-critical services:

1. Provide on-line analysis at the observatories.

- Physical strain extraction possibly using relevant ancillary channels (e.g., PEM) to remove instrumental or environmental signatures.
- Processing of strain data through real-time detection algorithms for both performance monitoring and scientific purposes.
- A means to cross-correlate data (either time series or event lists) from multiple interferometers.
- A means to store data frames and analysis results (local to the Observatory LAN) to short term storage media. This functionality will be provided by the LIGO DAQS resources, with augmentation by LDAS.
- A means to access both “live” and short term archived data via the Observatory LAN and the LIGO WAN. Access shall be subject to available bandwidth and demand.
- Means to retrieve, concatenate and extract specific channels of recent data from the on-line storage system.
- Sufficient automation to run continuously and autonomously during periods of normal operation.
- *A means to display and visualize results of analyses over the Observatory LAN.*
  - *Extensive graphics available with Matlab*
  - *Primitive graphics with LDAS userAPIs built on Tcl/Tk*

LIGO Data Analysis System (LDAS)
Requirements *(from 12/97 review)*

- Mission-critical services:

2. Provide for extended off-line processing capabilities:

- A means to reduce the raw data to science data representing calibrated GW strain data and a reduced subset of ancillary data and a data quality descriptor (will also be available on-line).
- A means to archive, retrieve and distribute reduced datasets acquired over a period of time at least ~~5~~ 2 years in duration.
- A means for duplicating reduced datasets either for backup or for distribution.
- Sufficient computing margin to enable multiple analyses to be conducted in parallel.

3. Provide a means to access the data archive via the LIGO WAN by the LIGO Laboratory and LIGO Scientific Collaboration to support ~~database~~ data products (frames; dBs; LigoLW objects) manipulation at the off-line site by remote users.

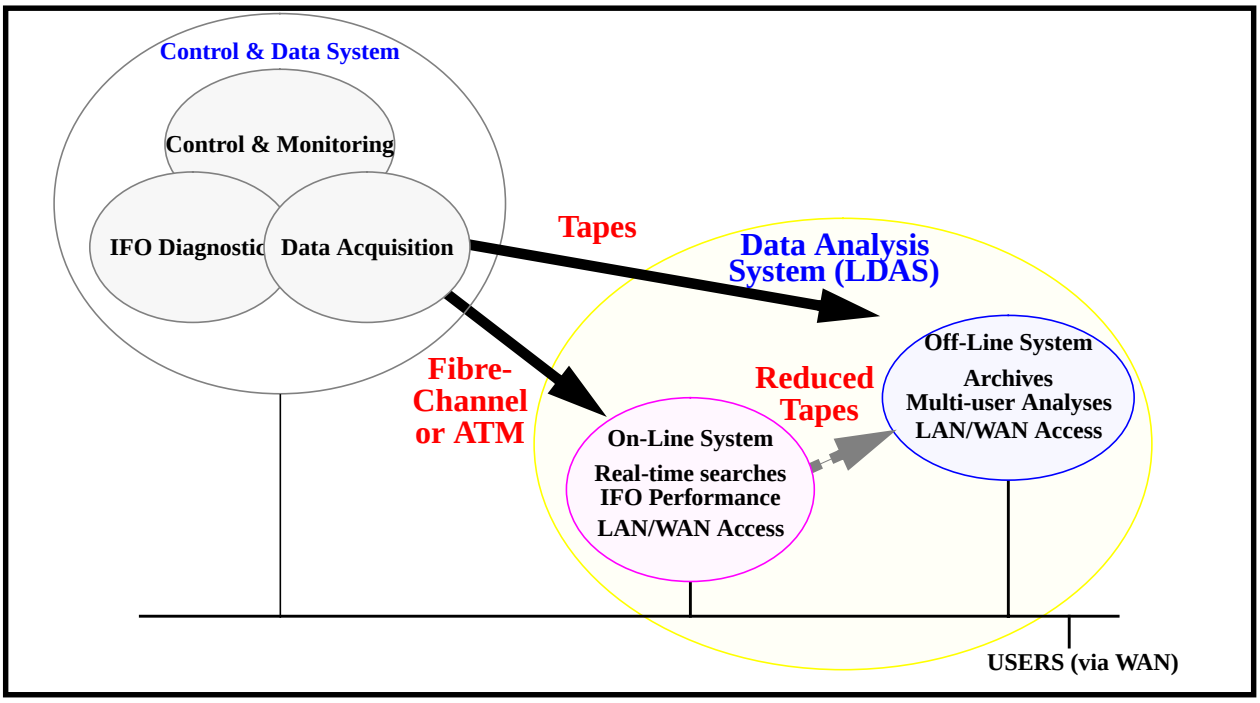
4. Provide a flexible design which can be reconfigured to reflect new analysis or computational requirements as they evolve.

LIGO Data Analysis System (LDAS)
Requirements *(from 12/97 review)*

5. Implementation goals:

- Flexibility => No (or very little) custom hardware with custom software interfaces
- Extensibility => Modular (not function specific) component design
- Portability => Upgradable hardware under same software or vice-versa => POSIX compliance, software standards, etc.
- Maintainability => Object oriented programming design ("reusable software components")
- Reliability => Distributed (redundant) components, independent of SW components
  - implied by previous design features

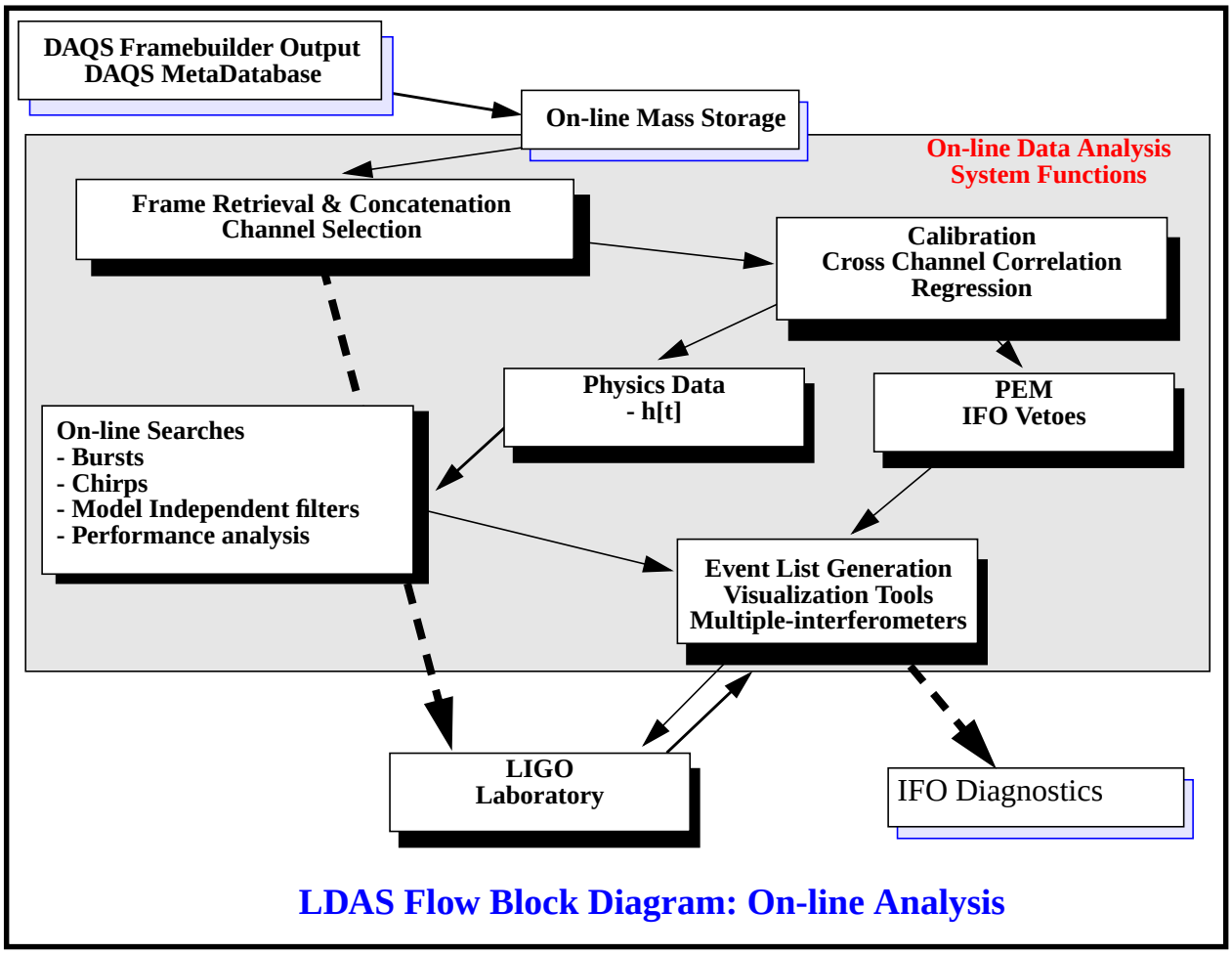
LIGO Data Analysis System (LDAS)
Functional Units



LIGO Data Analysis System (LDAS)
Design

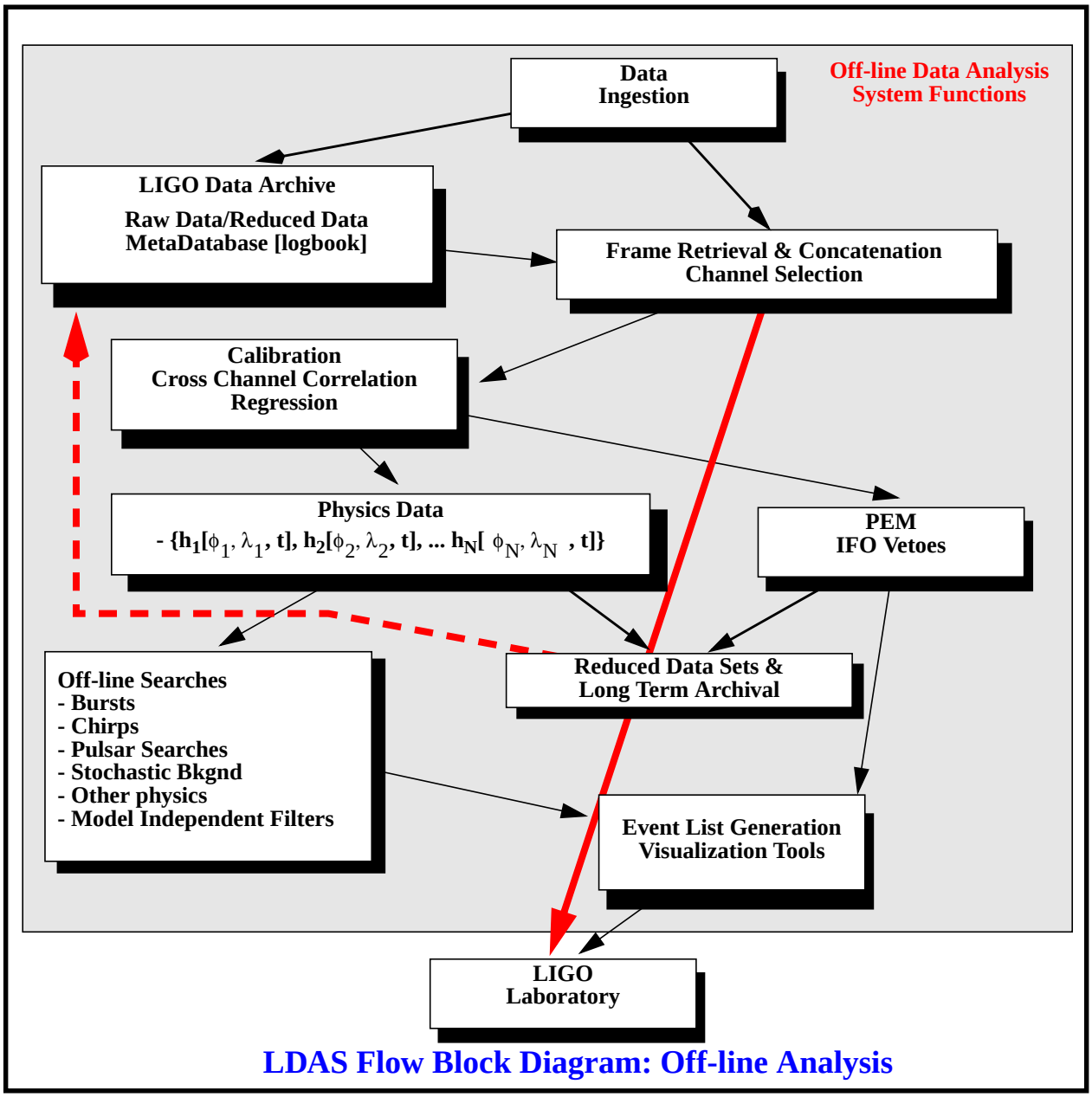
- Two LDAS components
  - On-line LDAS
    - Two systems, one for Hanford, and one for Livingston
    - Hanford system handles 2 interferometers
    - Provide computational power at the observatories to support diagnostics, detection, expansion/growth,...
  - Off-line LDAS
    - Collaborative arrangement with CACR
      - Dedicated LIGO hardware within CACR on scale of observatory systems
      - Database archive
      - Strategic use of other CACR facilities as available
    - Transparent access for off-line analysis of archived data
      - LIGO Laboratory
      - LIGO Scientific Collaboration
- Wide area network (WAN) to enable inter-site communications
  - University scientific and engineering support to Observatories
  - Access to archive database
  - Access to real-time data from observatories
  - Inter-observatory event sharing

LIGO Data Analysis System (LDAS)
On-line Functions



LDAS Flow Block Diagram: On-line Analysis

LIGO Data Analysis System (LDAS)
Off-Line Functions



LDAS Flow Block Diagram: Off-line Analysis

The LIGO Data Analysis System: LDAS

"A Users' Perspective"

KENT BLACKBURN
LIGO LABORATORY
CALIFORNIA INSTITUTE OF TECHNOLOGY
FEBRUARY 23RD, 1999
LIGO-G990011-00-E

The LDAS Team:

- Stuart Anderson
- Kent Blackburn
- Phil Ehrens
- Dave Farnham
- Xiao Hu
- Barbara Kratochwill
- Albert Lazzarini
- Walid Majid
- Ed Maros
- Tom Prince
- Bruce Sears
- Roy Williams

Overview of Talk:

- Primary Purpose, Goals & Demands
- Tour of Hardware Design
- Tour of Software Design
- Supporting Users
- Further Reading
- Closing Remarks

Primary Purpose:

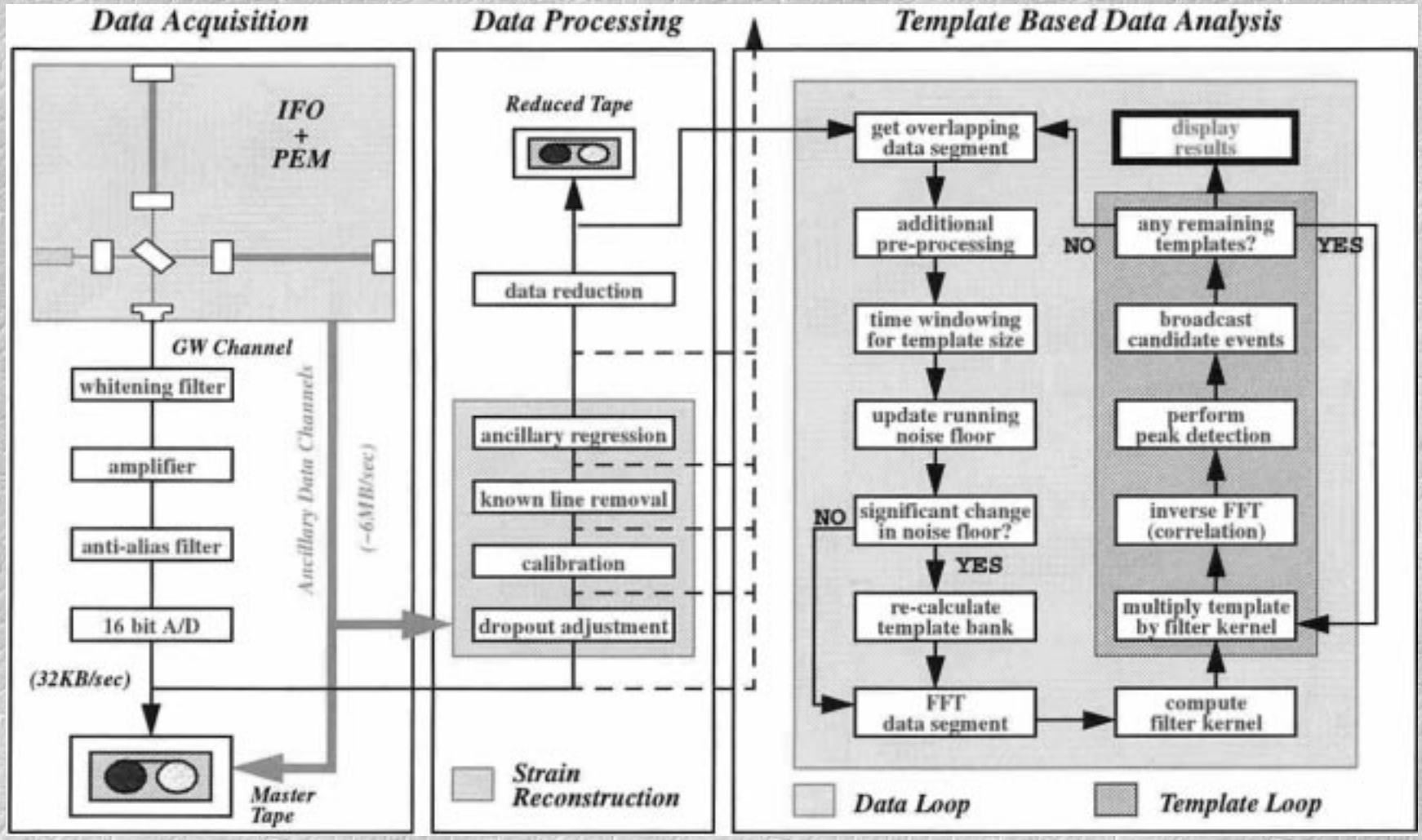
- First and foremost, the LIGO Data Analysis System is being implemented to detect and characterize gravitational waves from astrophysical sources.
- In addition, the LDAS will perform:
  1. raw frame data archival,
  2. database management functions for
     - raw frame data descriptors,
     - diagnostic trigger descriptors,
     - and astrophysical filter (template) event descriptors,
  3. data & metadata distribution services.

Principal Sources:

- Binary Inspiral of Compact Stellar Objects
  - neutron star - neutron star
  - black hole - neutron star
  - black hole - black hole

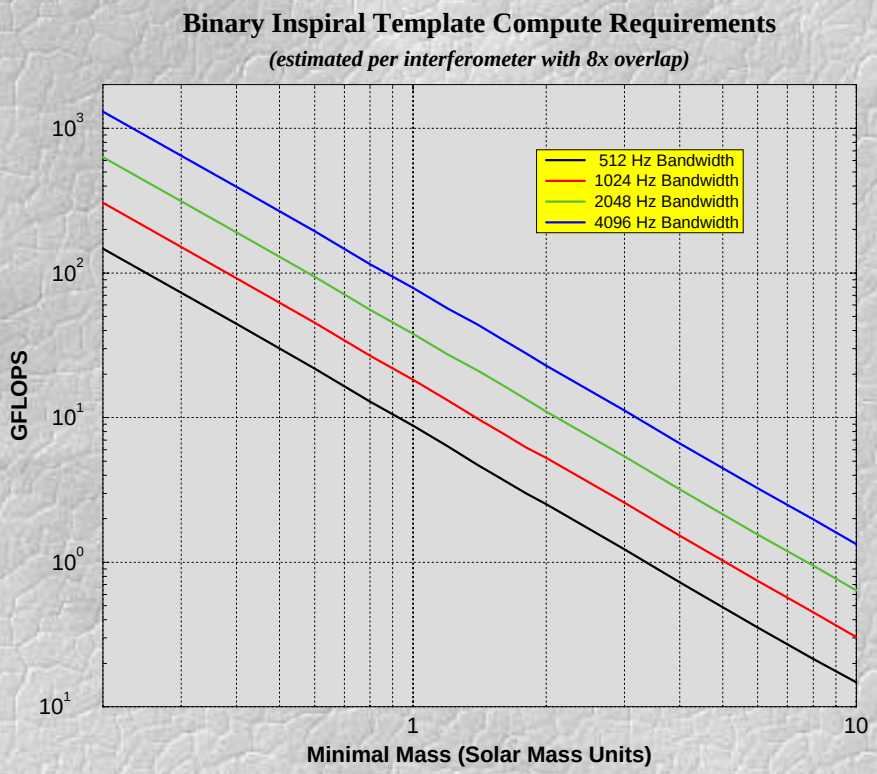
  } dominate LDAS requirements
- Burst Events
  - supernovae (requiring coincidence between sites)
- Periodic Sources
  - pulsars (all sky unlikely, but targeted searches easily carried out)
- Others
  - black hole ring-downs
  - black hole mergers
  - stochastic background
  - serendipitous discoveries!

Data Flow Model:

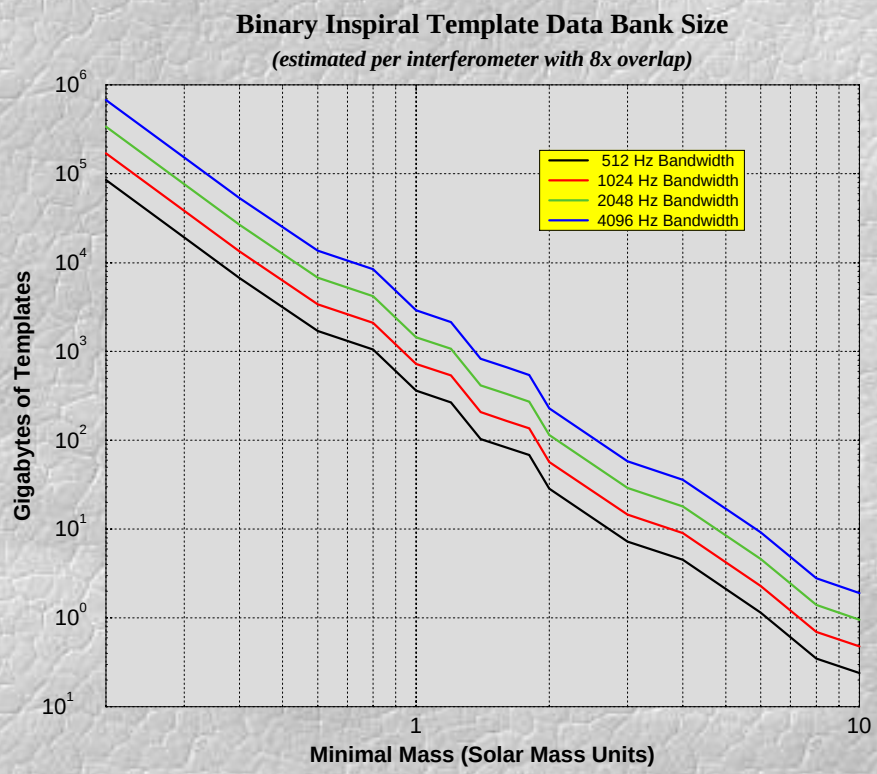


Optimal Filtering Demands:

Computation:



Templates:



Hardware Design Needs:

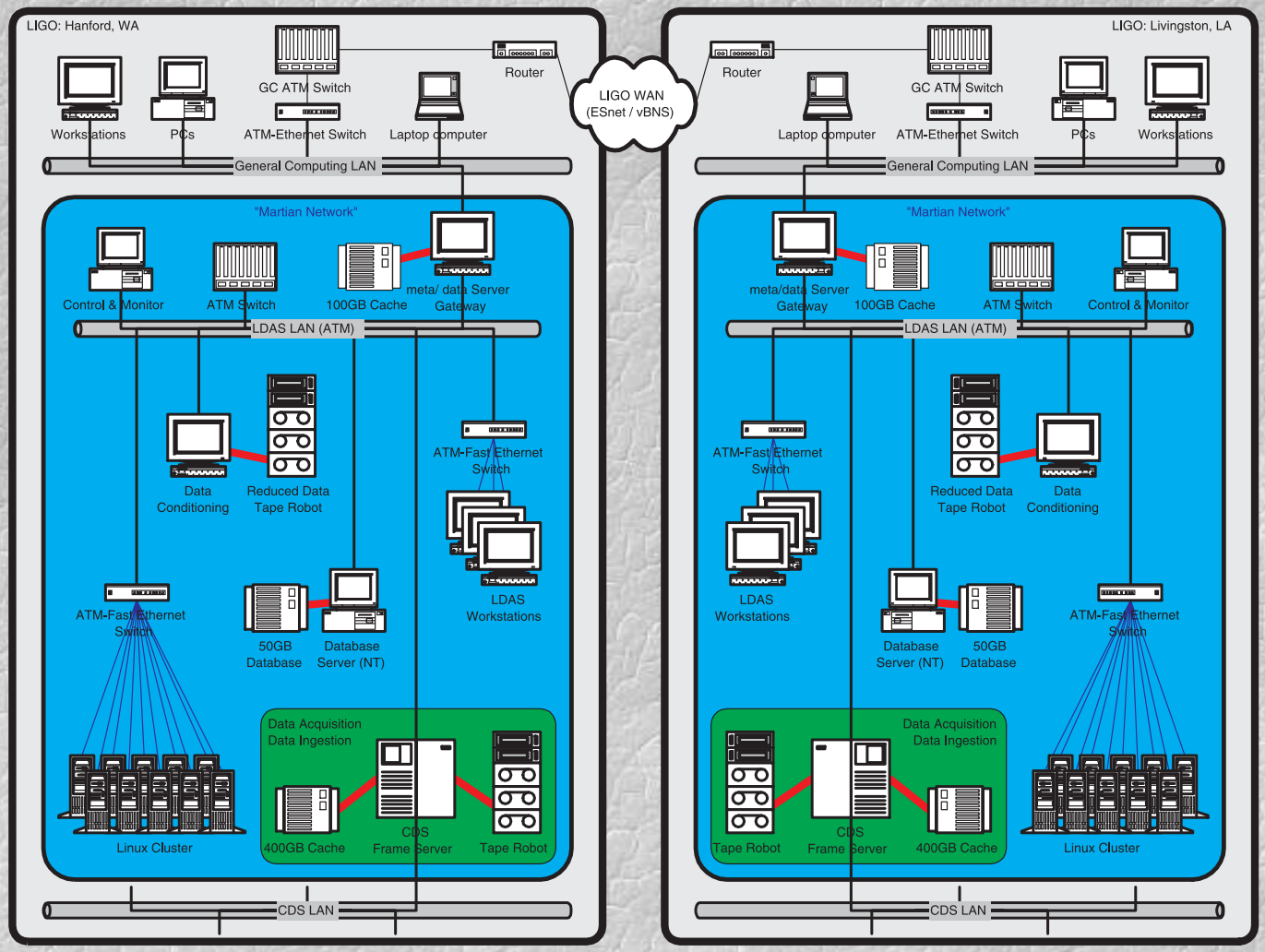
Requirements:

1. *10-100 GFLOPS of Compute Performance & 0.5-5 TB of uncompressed Templates*
2. *100-620 megabits per second point-to-point*
3. *500 GB of On-Line Disk Cache*
4. *50-500 TB Archived Data per Year*
5. *50-500 GB of metadata per Year*
6. *LAN & WAN Networks*

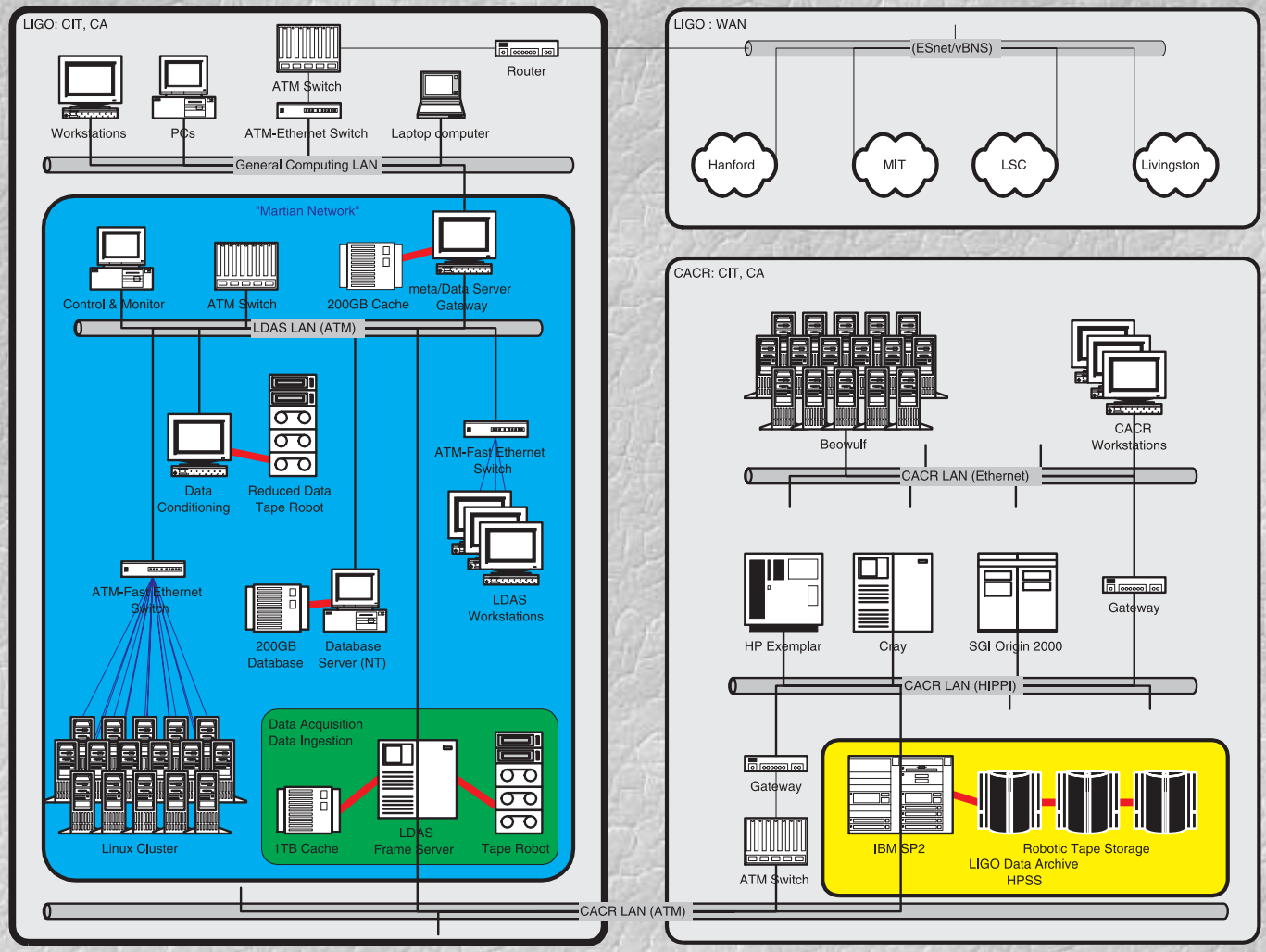
Technologies:

1. *Beowulf (PC-Linux Clusters)*
2. *Switched ATM & Fast-Ethernet for LANs*
3. *SCSI Hard Disk Storage*
4. *HPSS, Tapes (optical?)*
5. *Database Capable of handling this Volume*
6. *LIGO ATM & Fast-Ethernet LANs with ESNet and vBNS connectivity between the Observatories & Archive*

Hardware Architecture(online):



Hardware Architecture(offline):



Hardware Status:

- LDAS Hardware Design Relatively Complete
  1. Procurement being delayed/staged as much as possible
  2. PDR tentatively scheduled for March 11th
- WAN to Sites Established
  1. T1 links at both Hanford, WA & Livingston, LA
- First LAN Hardware (ATM Switch) Arrived at Hanford
- Compact Beowulf Cluster Received from ALTA Technologies for Prototyping
- meta/Data Server Hardware Identified in conjunction with CDS Frame-Builder
- Establishing Archival Storage Specifications in collaboration with CACR

Software Design Needs:

Requirements:

1. *Portability -*
   - POSIX for UNIX I/O
   - ANSI C/C++ for Compilers
   - TCL/TK for Steering
   - MPI for Parallel Computing
   - ODBC for Database Clients
2. *Extensibility -*
   - Modular/Reusable Code
   - OOP Design
3. *Maintainability -*
   - OO Languages Where Practical
   - CVS Source Code Management
4. *Flexibility -*
   - Class/Object Design
   - Modular Libraries
   - Distributed Processing

Components:

1. *Data Formats -*
   - IGWD Frames
   - Lightweight (Metadata, Events, Templates, Communications ...)
2. *Libraries -*
   - Supported Data Format I/O
   - Numerical (FFT's, filters, etc.)
   - POSIX & Socket interfaces
3. *LDAS API's -*
   - Supervisors to Data I/O
   - Control, Monitor, & Management
   - MPI Level Communications
   - Filtering and Analysis
4. *UI's -*
   - TCL/TK (Wish Shell) GUI
   - TCL Interpreter Command Line
   - TCLet Plug-ins to Web Browsers

Standardized Data Formats:

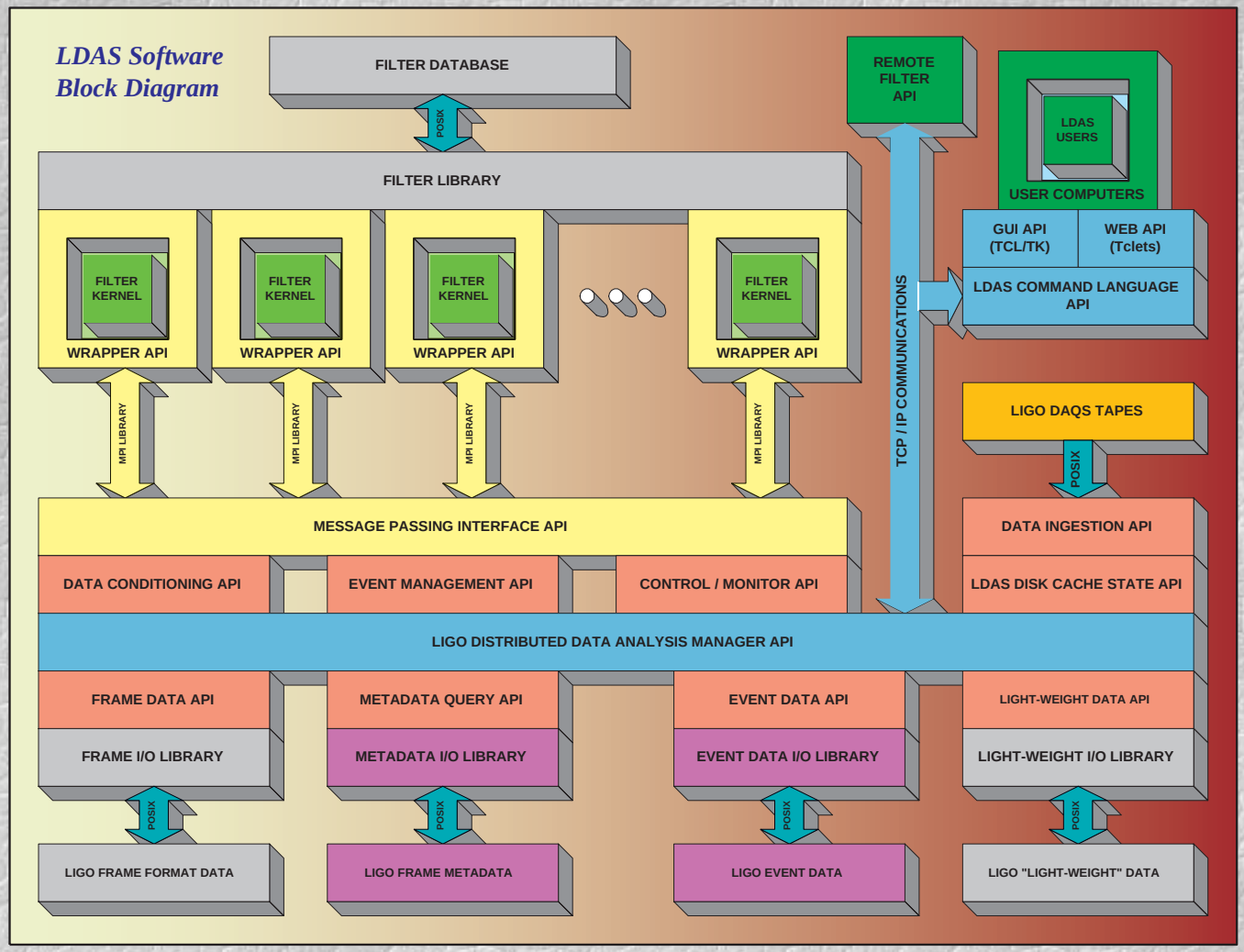
Frame:

1. Structured (C-like) Format
   - Samples of these structures:
     - ✓ ADC, Static, Detector, Trigger, ...
     - ✓ Simulated, History, Logs, ...
2. Jointly Developed with VIRGO
   - LIGO-T970130-B-E
   - VIRGO-SPE-LAP-5400-102
3. Original I/O Library in C
4. C++ Class I/O Library
5. Primary Uses:
   - Data Acquisition
   - Data Archival
   - *Subsystem for Diagnostics*

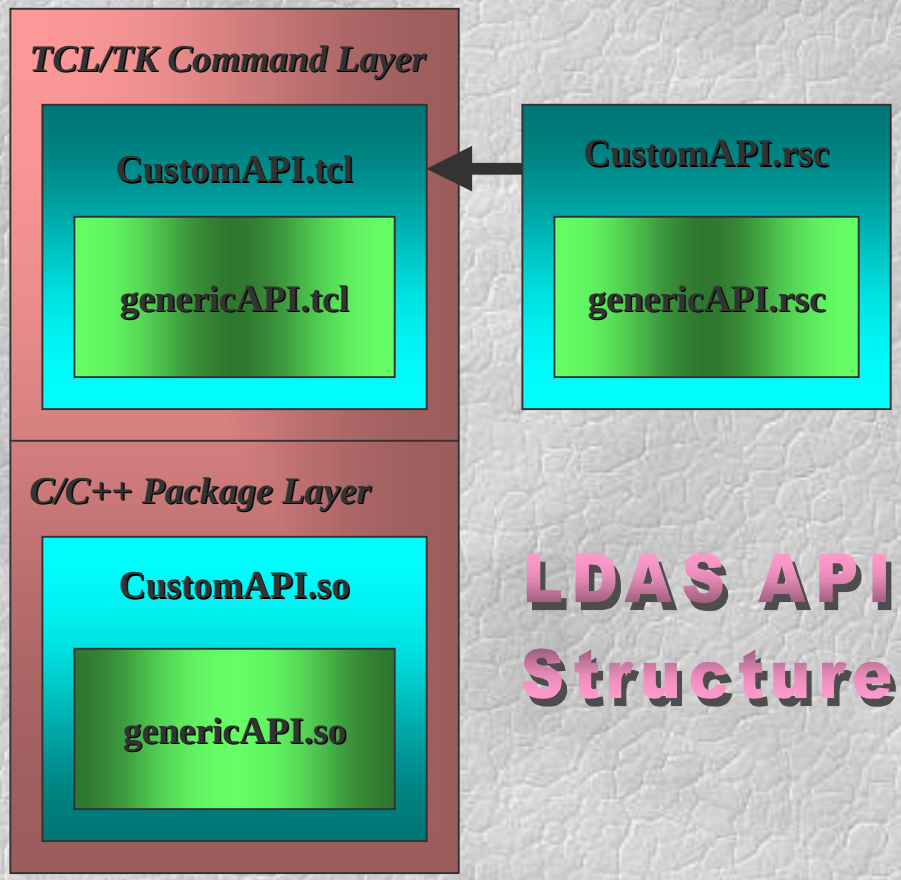
Lightweight:

1. Tagged (XML-based) Format
2. Easily Parsed *(and written)*
   - <int_s format="ascii">57, 7, 15, 12, 15, 31, 755</int_s>
3. LIGO Defined Objects
   - tables (n-tuplets),
   - arrays (matrix, vector),
   - vectors (time-series, power-spectra)
   - Some Revisions expected
4. Use Complement Frames!
   - LIGO Event data
   - LIGO Metadata
   - Spectra & Time-series data
   - Inter-process Communications

Software Architecture(block):



Layered LDAS API Design:



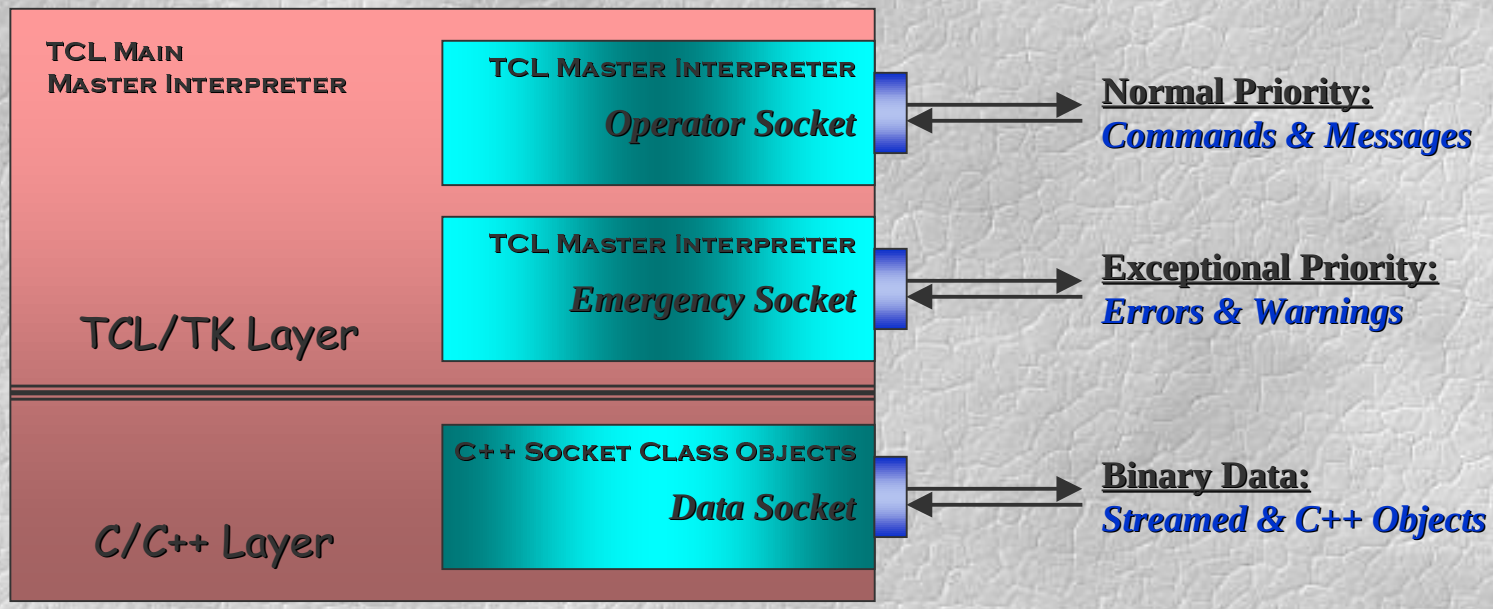
LDAS API Structure

LDAS API's:

1. Two Layers:
   - TCL/TK
   - C/C++ *(extends TCL Language)*
   - SWIG Unifies Layers
2. GenericAPI *(core)* Module:
   - Communications
     - ✓ *TCL <-> C++*
     - ✓ *API <-> API*
   - Common TCL proc's:
     - ✓ *Help*
     - ✓ *Logging*
     - ✓ *Command Socket Management*
     - ✓ *Resource Management*
   - Common C/C++ methods:
     - ✓ *Data Socket Management*
     - ✓ *Internal Data Management*
     - ✓ *Class Save & Restore*
3. *Custom (specialization) Module*

LDAS API Communications:

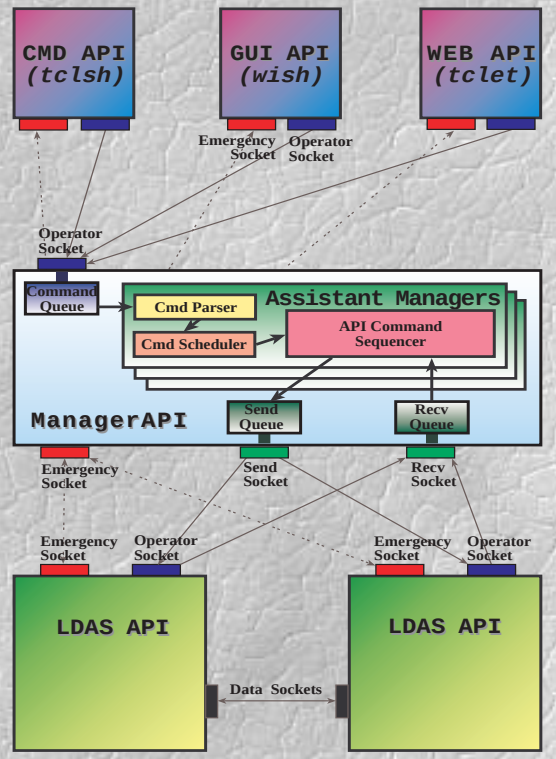
- **3 Types of Socket Communications in API's:**
  1. Operator Sockets - Normal Inter-process Commands & Messages
  2. Emergency Sockets - Error & Warning Commands & Messages
  3. Data Sockets - Binary Data in either Raw Streams or C++ Objects



Distributed API Supervisor:

Manager API:

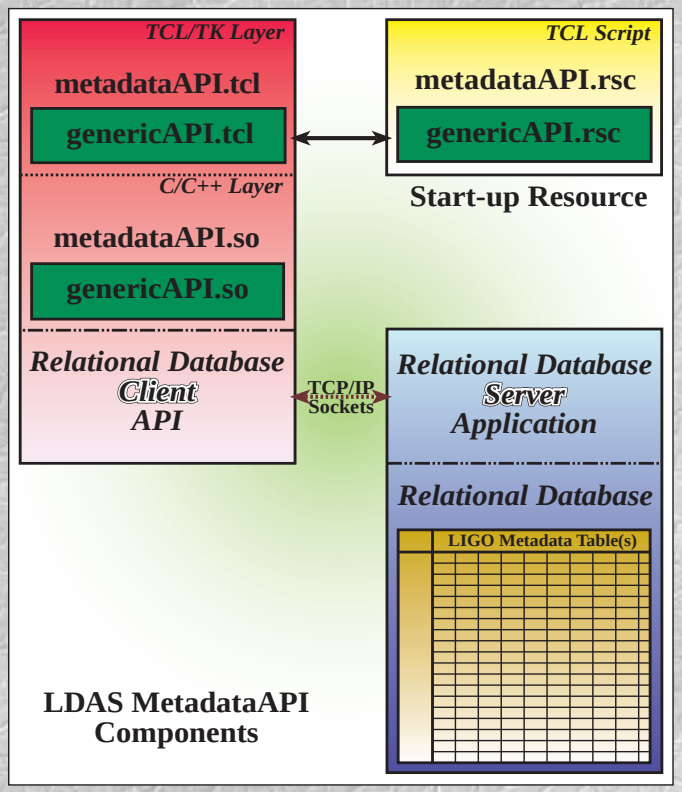
1. Two Supervisory Layers:
   - Single "top-level" Manager
   - Multiple (3-10) Assistant Managers
2. Manager Layer:
   - Communications
     - ✓ *Manages sockets & queues*
     - ✓ *Interface for User API's*
     - ✓ *Point of contact for All API's*
   - Exception Management
   - Log File Management
   - System Operator Interface
3. Assistant Manager Layer:
   - Command Execution:
     - ✓ *Parsing*
     - ✓ *Schedule Building*
     - ✓ *Command Sequencing*
   - Reports to Manager Layer



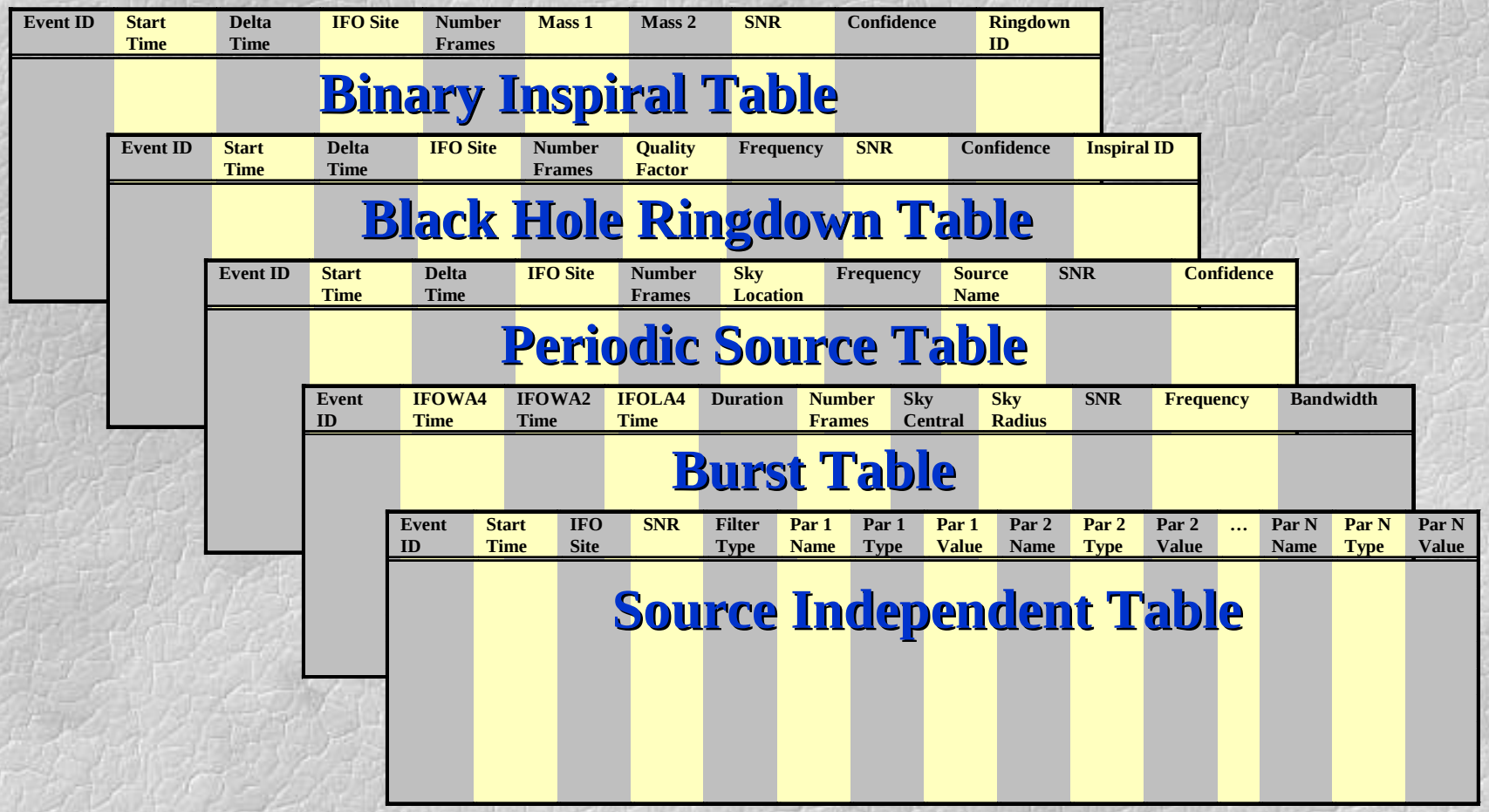
Distributed Database API:

Metadata API:

1. Relational Database:
   - Using ODBC Standard for Calls
   - Currently Developing with DB2
   - Server can be Unix or NT Based
2. Table Contents:
   - Frame Characterization:
     - ✓ *Descriptors*
     - ✓ *Simple Statistical Summary*
   - Event Characterization:
     - ✓ *Optimal Filter Results*
     - ✓ *Astrophysical parameters*
   - GDS Trigger Results
   - CDS State Vectors
   - LDAS Logs
3. Metadata Query Sevices
   - LDAS Processing Queries
   - Data-Mining Queries

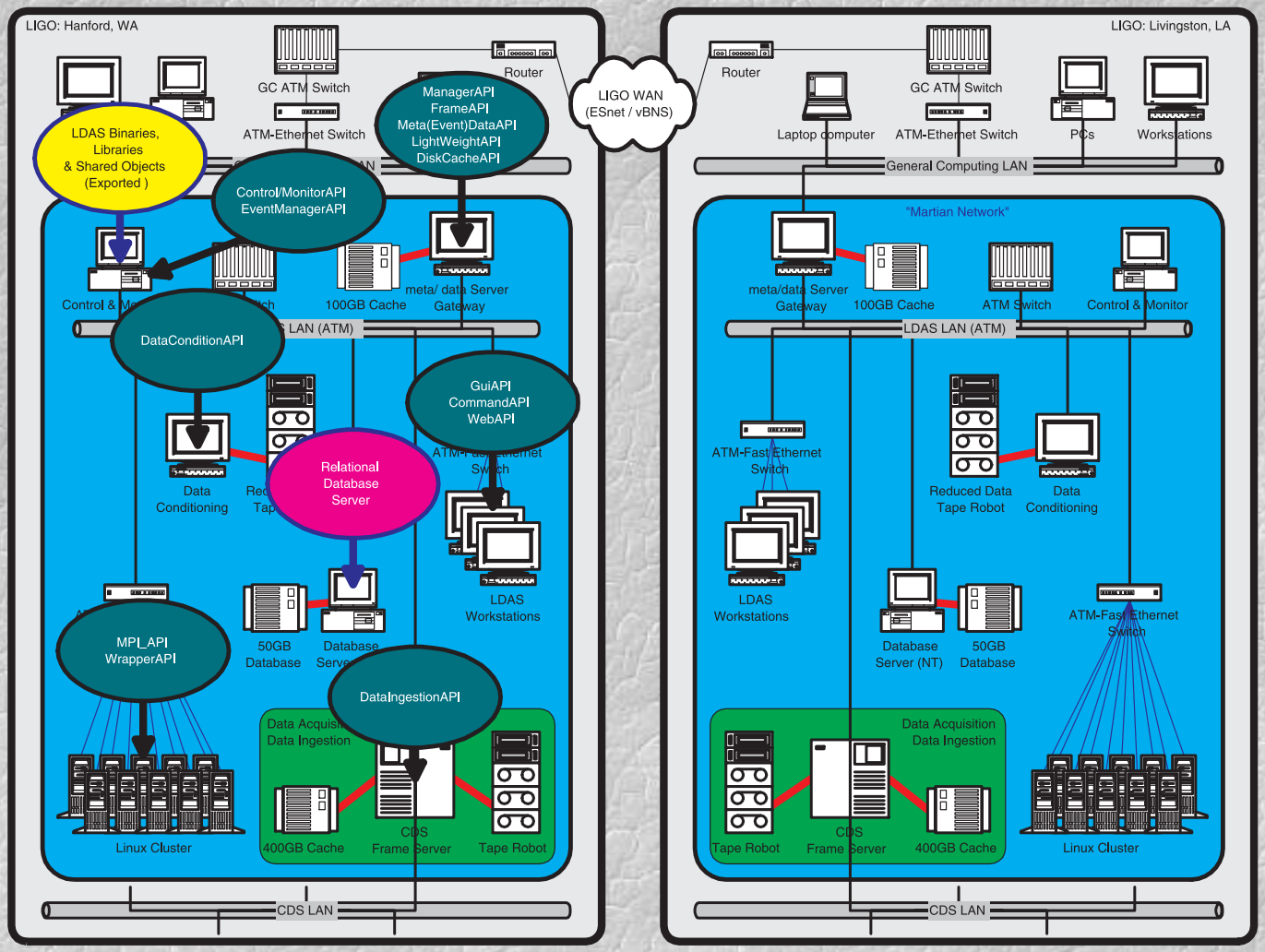


Database Event Tables:

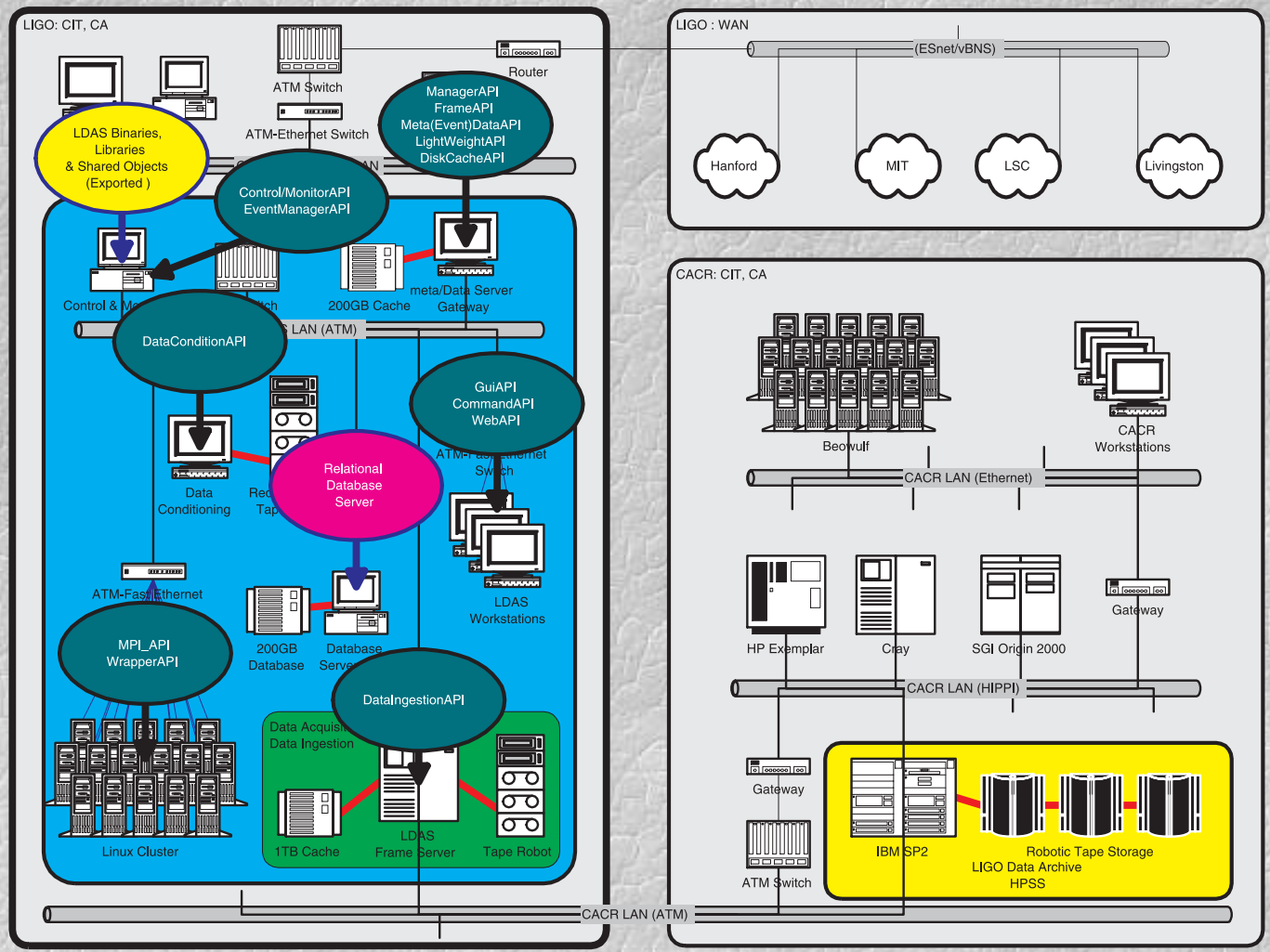


see URL: http://www.ligo.caltech.edu/~xhu/index.html for Event, Frame, GDS, Log, CDS Table Descriptions

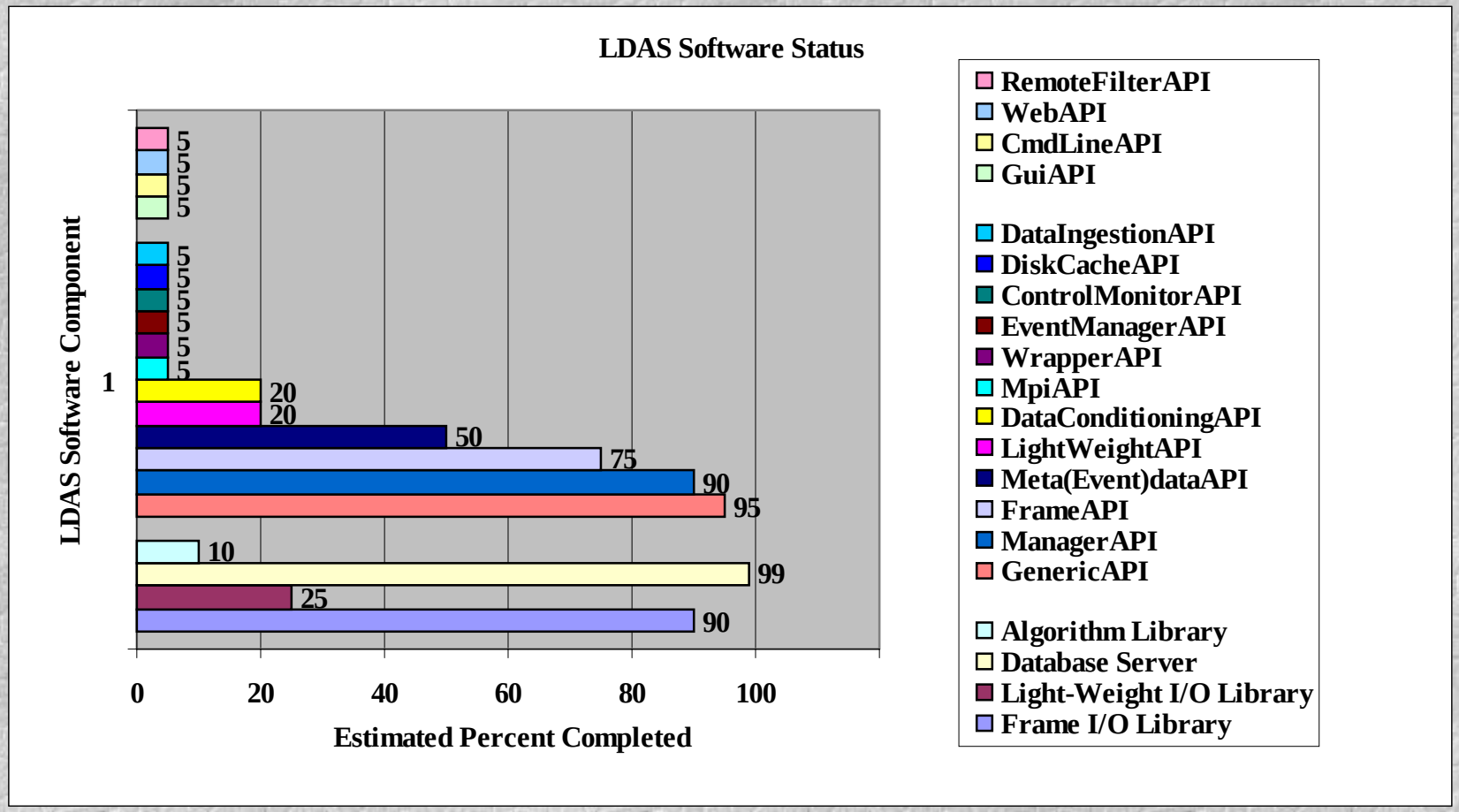
Software Mapping(online):



Software Mapping(offline):

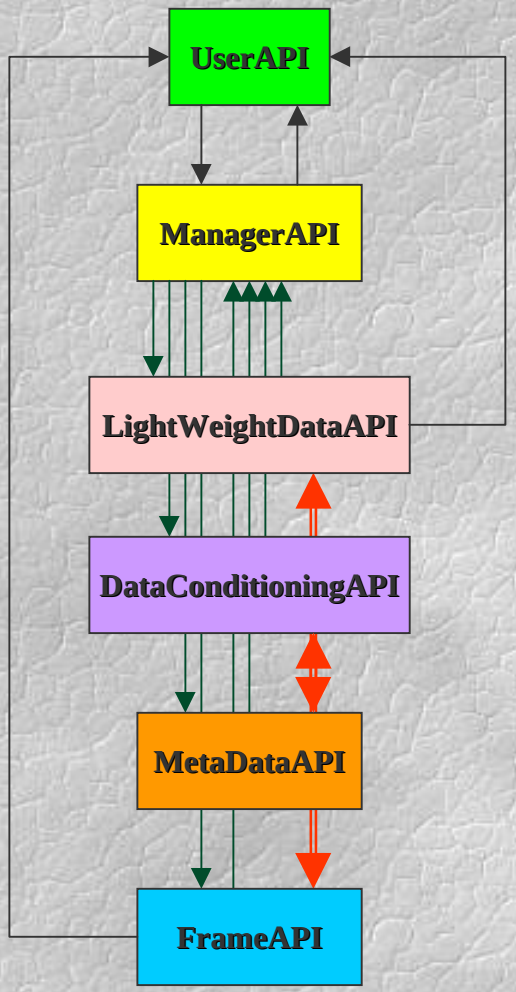


Software Status:



First Software Deliverable:

- **α-release Spring 99?**
  1. LDAS Components to be included:
     - GenericAPI
     - FrameAPI
     - Meta(Event)dataAPI
     - LightWeightDataAPI
     - DataConditioningAPI
     - ManagerAPI
     - Simple UserAPI
  2. Some "Quick Look" Capabilities
  3. Support of Site Installation Activities
  4. Provide User Base for Testing and Debugging System



LDAS User Classifications:

- Perusers - Browsers of Data Products
  1. Connection: Modem or Ethernet
  2. Tools of Choice: GUI & Web Browser
- Users - meta/Data Flow Manipulators *+ Perusers*
  1. Connection: Modem, Ethernet, LDAS LAN or WAN
  2. Tools of Choice: GUI or Command Line Interface
- Developers - Algorithm/Infra-structure Coders
  1. Connection: Ethernet, LDAS LAN or WAN
  2. Tools of Choice: Raw Data or Sockets Directly into LDAS Data Flow
- Processes - Distributed Processes of LDAS
  1. Connections: LDAS LAN or WAN
  2. Tools of Choice: LDAS Command and Data Sockets Protocols

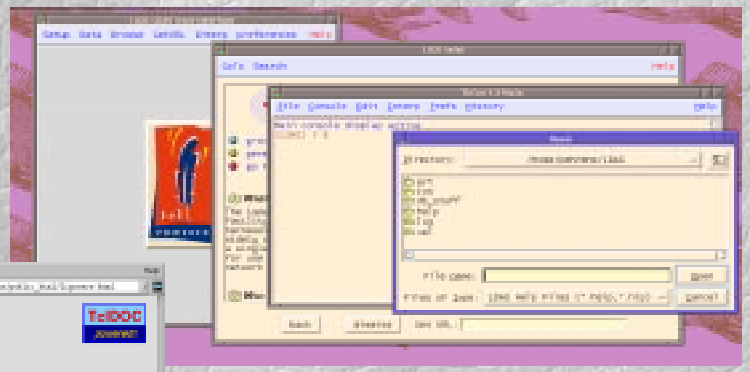
LDAS User Interfaces:

- **User Interface model based on TCL/TK**
  1. Doesn't preclude other software languages:
     - Any environment that can send LDAS Commands, properly formatted, to the ManagerAPI's Command Socket and receive LDAS Light-Weight Data Format can be made to work with LDAS!
  2. Portable between Unix / Windows / Mac Operating Systems
- **3 Types of User Interfaces Based on TCL/TK to be provided:**
  1. Command Line Interfaces - TCL shell scripts which communicate with the LDAS ManagerAPI sockets
  2. Graphical User Interfaces - TK Wish shell widgets which communicate with the LDAS ManagerAPI's sockets
  3. Web Browser Interfaces - TCLet plug-ins that display widgets in web browsers & communicate with the LDAS ManagerAPI's sockets

"Smart" User Interfaces:

Initial Limited UI

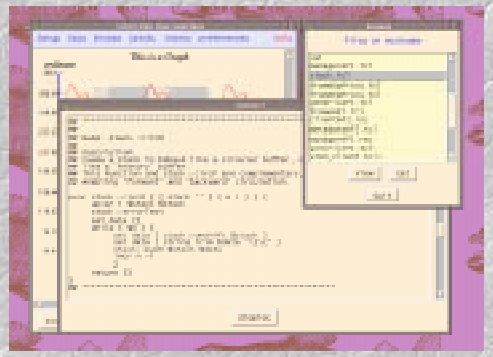
1. Client/Server Socket Links
2. Server sends TCL/TK Code

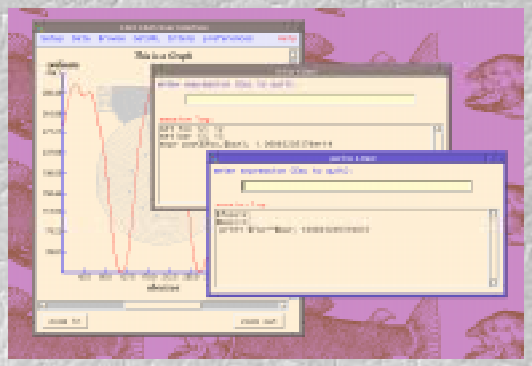


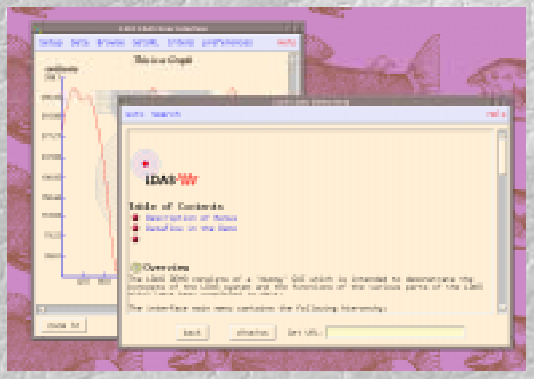
Socket Based Communications

Final Custom UI

1. Appended Functionality to UI
2. Provideds centralized UI Code Management







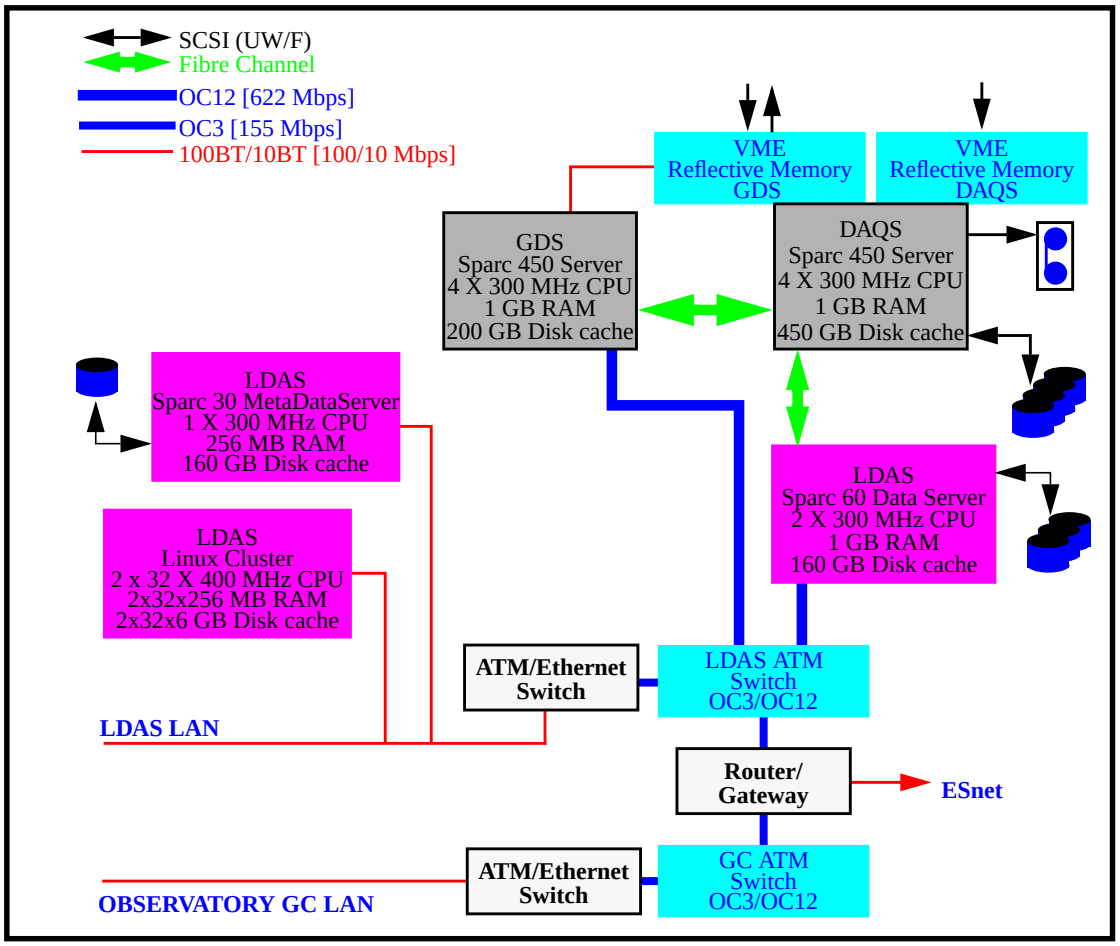
LDAS Documentation:

- On the Web: LIGO Data Working Group Bulletin Board
  1. http://www.ligo.caltech.edu/~prince/LDCG_lsc/LDCG.html
     - LDAS Technical Review Documents
     - LDAS Software Guidelines
     - LDAS Software Requirements
     - LDAS Software Specifications
     - LDAS Code Formatted as HTML
     - LDAS Presentations
     - Other Useful Documentation
  1. LDAS Source Code Presentation
     - C++ to HTML using DOC++
     - TCL/TK to HTML using TCLDoc
     - CVS Server to LDAS Software Repository (password protected)

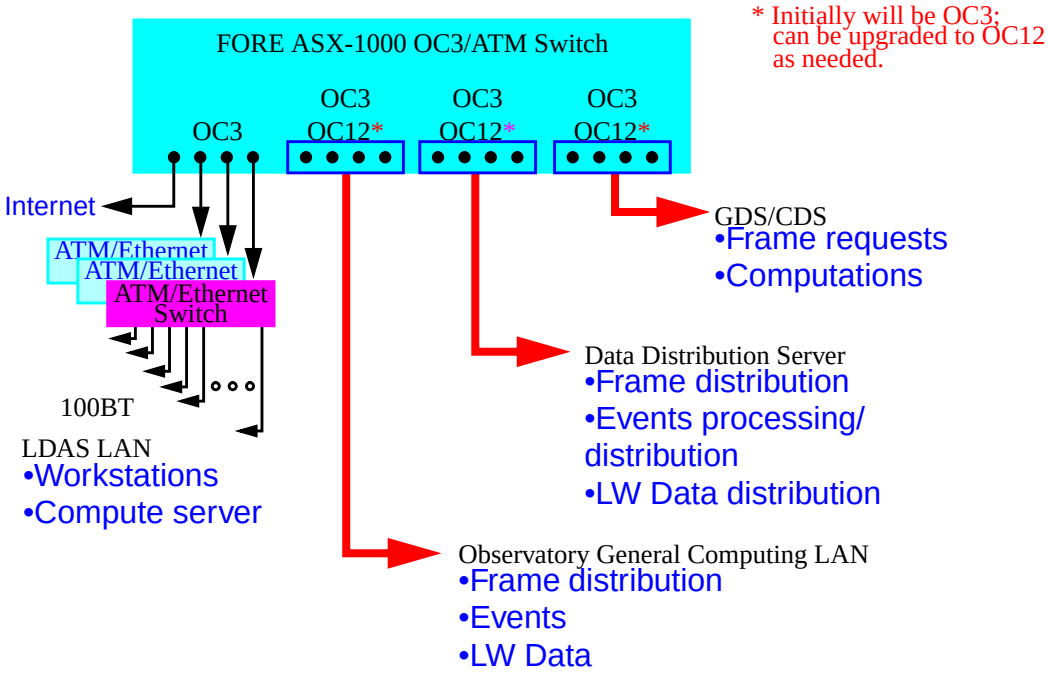
Closing Remarks:

- Primary Purpose - Detection Of Gravitational Waves Using LIGO Data
- Design Is Portable, Extensible, Maintainable & Flexible
- Supports Users At Multiple Levels
  - e.g., Perusers, Users, Developers, Processes
- Powerful "Smart" User Interfaces Based On TLC/TK Have Been Prototyped
- Significant Amounts Of Coding Still Ahead!

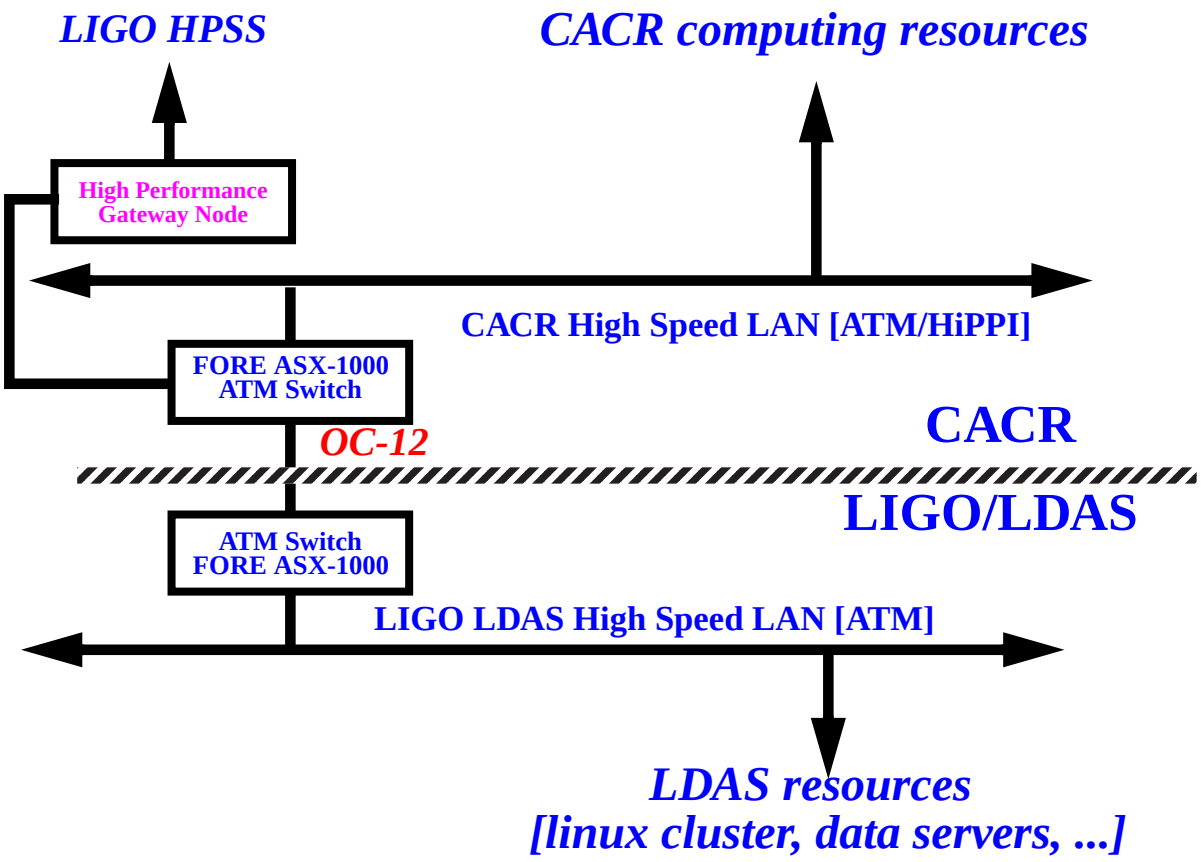
LDAS Interfaces to Detector Systems



LDAS Interfaces to Detector Systems (cont)



LDAS Interfaces to CACR Systems



LDAS Interfaces to CACR Systems - HPSS

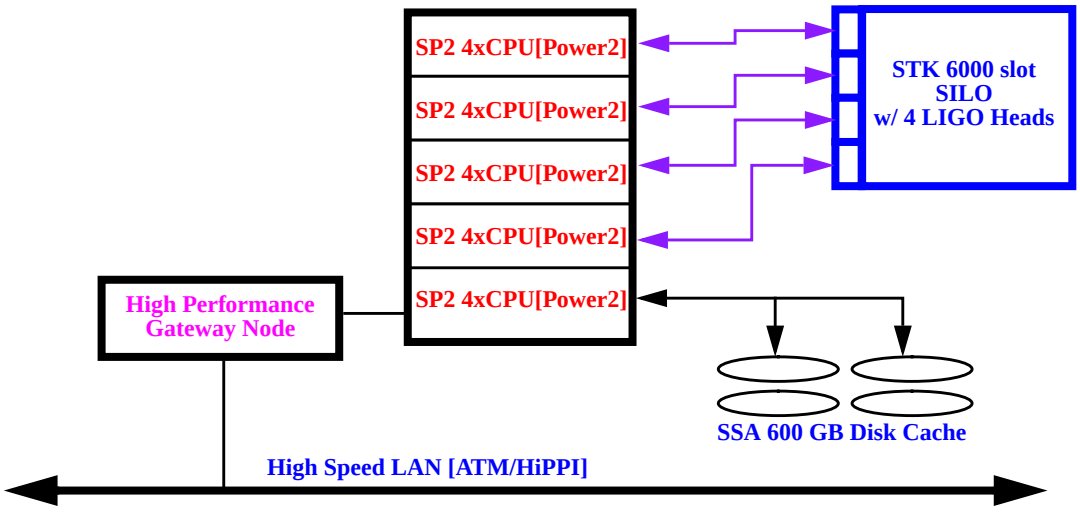


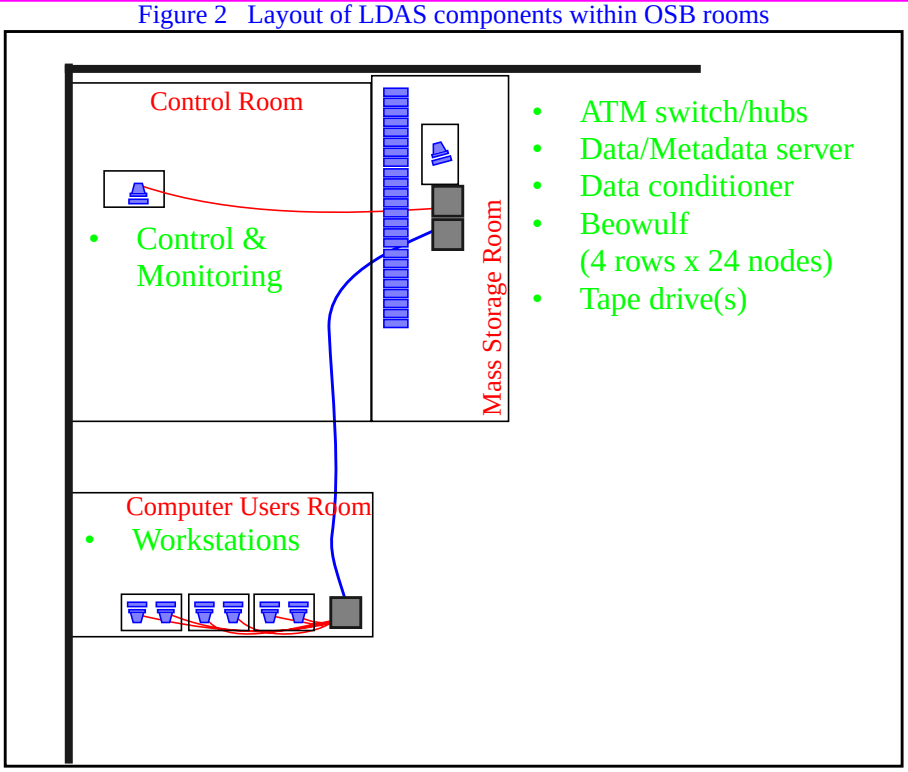
Figure 1 HPSS Configuration for LIGO

LDAS User Interface

- Details for the user APIs which provide the interfaces for users into LDAS are still in the definition process
  - When the requirements and specifications are available, they will be reviewed separately.
  - » "LIGO Command Language" (or LCL) based on Tcl scripts
  - » Understood by the managerAPI.
  - » LCL will consist of a number of high-level commands that imply a sequence of lower-level operations
  - » Commands can themselves be concatenated into more complex sequences.
  - » Follows the paradigm of a standard UNIX shell.
- Examples of such commands will include:
  - "re-sample channel X at rate R";
  - "heterodyne channel X with mixer frequency f0"
  - "regress channels {Y,Z,...} from channel X, pass resultant to socket S or process P";
  - "calibrate channel X (using the calibration data valid at the time channel X was acquired and which are available in the framed data)"
  - ; create a subframe for channels {X,Y,Z,...} from time interval {T1,T2} and re-sampled at rate R";
  - etc.

LDAS Layout at Observatories

Figure 2 Layout of LDAS components within OSB rooms



LIGO Data Archive
HPSS (High Performance StorageSystem)

- IBM System
  - Used by SDSC, FLNL, others
  - IBM OS/SW/CPUs; 3rd party robotics
  - scalable, starting from ~ few TB to 300+TB
- LIGO HPSS specification:
  - IBM SP2 rack with:
    - backplanse switch
    - five 4-way PCI nodes in rack
  - ~500+ GB SSA RAID disk cache to dump tapes
  - 1 high performance gateway node (HPGN)
  - 4 Redwood tape drives (=> 4 users)
  - few X 100GB disk stroage for non-HPSS data (metadata on spinning media)
  - buy tapes as needed
- Use CACR tape robotics/silo for cassette storage
  - focus LIGO funds on “smart” hardware: tapes drives; disk drives; CPUs; switches;...

LIGO HPSS Capability

- Storage and access for ~ 100TB of LIGO data for indefinite period of time
- Support 4 independent users seeking data
- Data throughput to disk cache ~ 11 MB/s
- Use SP2 rack backplane as a high speed/low latency network to communicate between HPSS processes.
- One 4-way PCI node would support metadata services (dB2 is an IBM product)
- Four other nodes operate as parallel data movers, allowing for 4-way striped tapes (data striped across 4 tape cassettes for throughput).
- High performance gateway node (HPGN) would interface directly to SP2 backplane and allow high speed access to HPSS via ATM, HiPPI, gigabit or other high perf. network technologies

LIGO Development Timeline

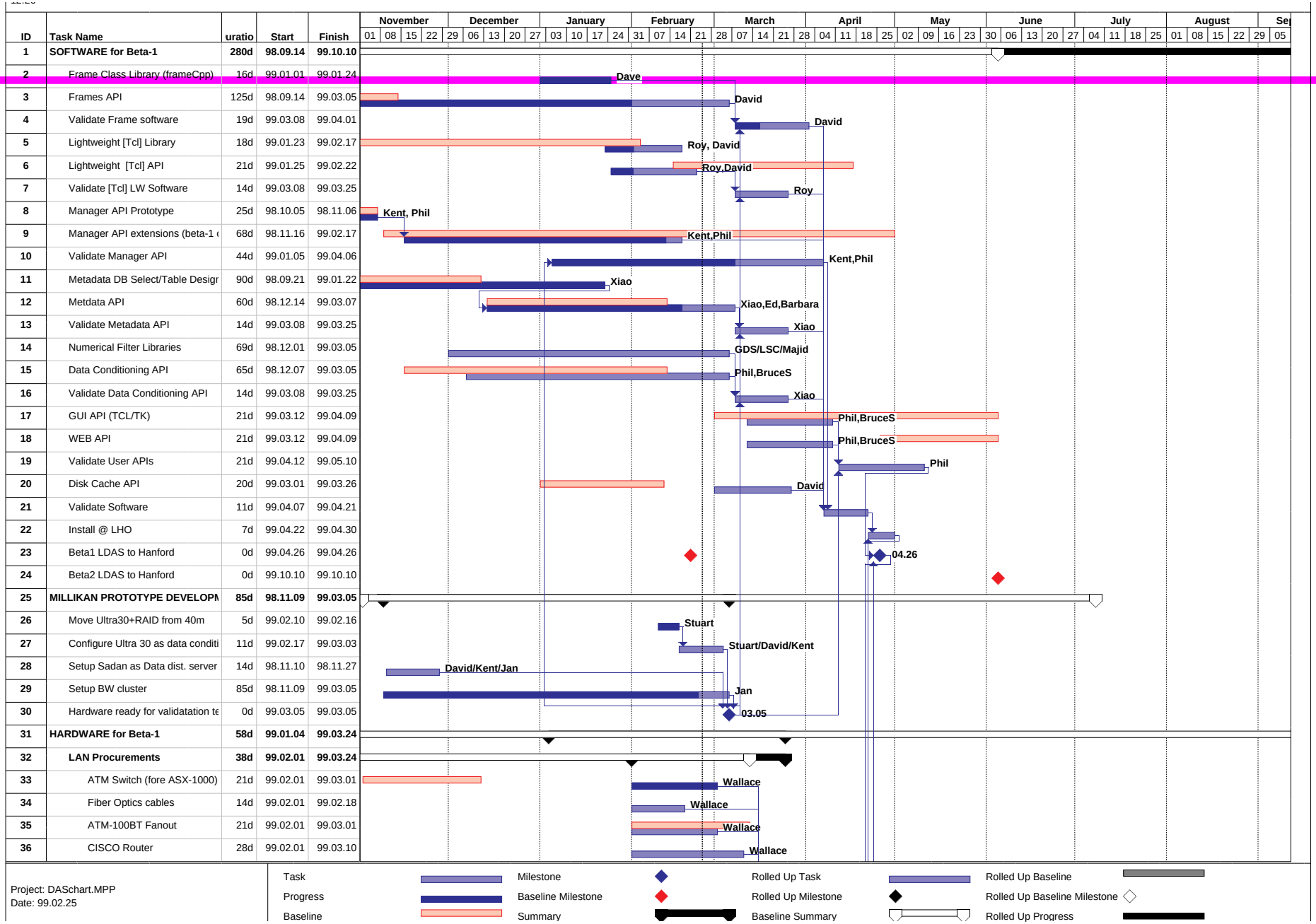
LIGO Installation Major Milestones

Milestone	Date	Comment
Vacuum Equipment Complete	1998.12.08	Both sites
BT Bakeout Complete	2000.02.21	Both sites
LHO 2km Start	1998.07.01	Begun
Power Recycled	1999.10.18	
Vertex Michelson complete		
LHO 2km IFO complete	2000.08.28	$h[f] < 10^{-20}$
LLO 4km Start	1999.06.01	
Power Recycled	2000.02.28	
Vertex Michelson complete		
LLO 4km IFO complete	2000.11.20	$h[f] < 10^{-20}$
LHO 4km Start	1998.07.01	Begun
Power Recycled	2000.03.20	
Vertex Michelson complete		
LHO 4km IFO complete	2000.10.16	$h[f] < 10^{-20}$
Design sensitivity	2001.11.05	$h[f] < 10^{-21}$
First science run	2002.01.01	3X operation

LDAS Development Strategy

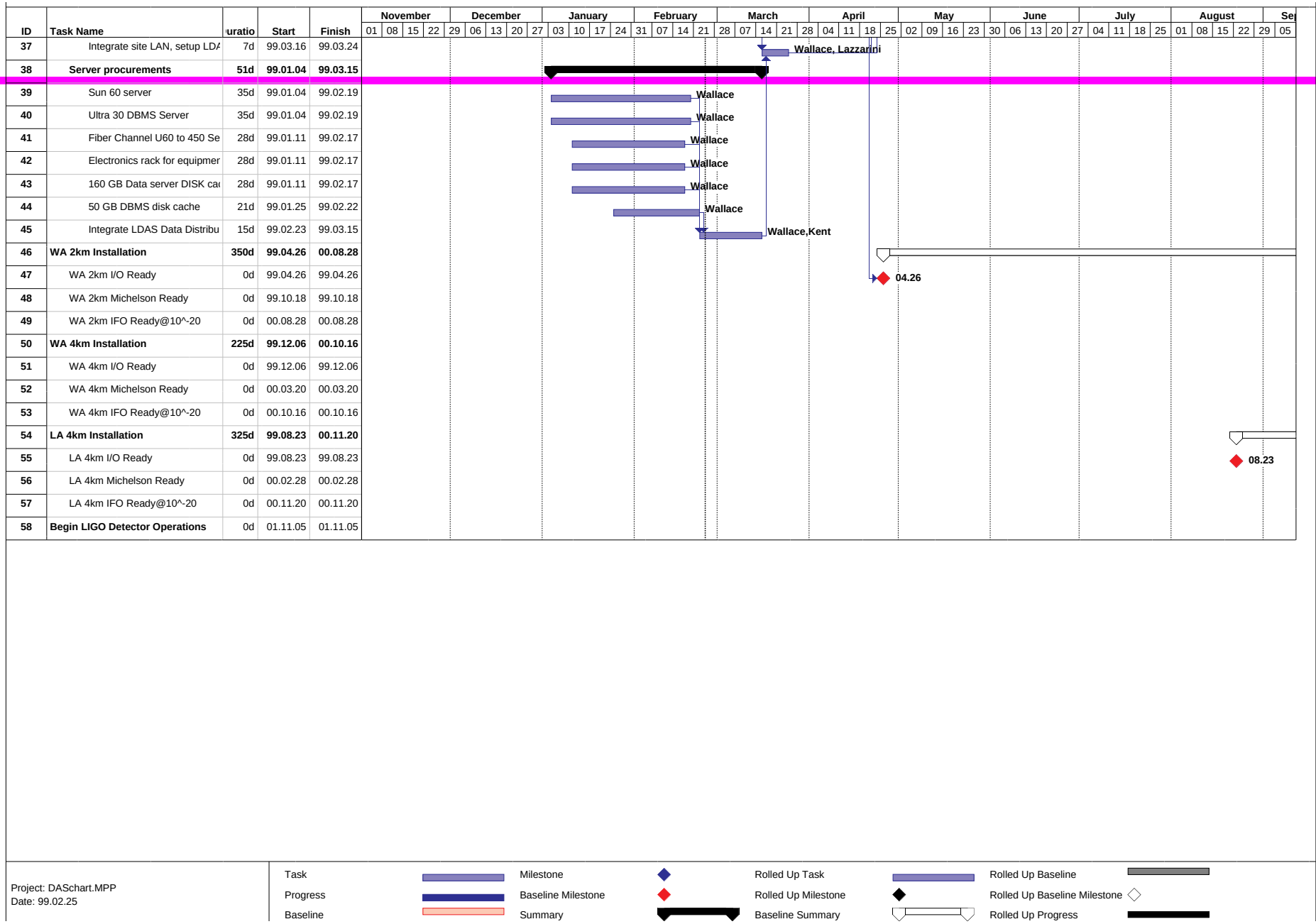
- Focus on initial installation for 2km IFO at Hanford
  - **Initial α release in late spring/summer 1999**
    - data distribution
    - data conditioning
    - metadata search/archival
  - **Support for initial commissioning of PSL + IO**
  - **Secondary β release in late fall 1999:**
    - 8 - 16 node beowulf for data analysis (diagnostics)
    - software update
- Replicate design, scaled as needed for other interferometers/site
- Defer off-line system availability until ~ 6 months before science run
    - large linux cluster
    - HPSS + data servers
    - data conditoning
    - data ingestion

LDAS Developement Timeline



ID	Task Name	uratio	Start	Finish
1	**SOFTWARE for Beta-1**	**280d**	**98.09.14**	**99.10.10**
2	Frame Class Library (frameCpp)	16d	99.01.01	99.01.24
3	Frames API	125d	98.09.14	99.03.05
4	Validate Frame software	19d	99.03.08	99.04.01
5	Lightweight [Tcl] Library	18d	99.01.23	99.02.17
6	Lightweight [Tcl] API	21d	99.01.25	99.02.22
7	Validate [Tcl] LW Software	14d	99.03.08	99.03.25
8	Manager API Prototype	25d	98.10.05	98.11.06
9	Manager API extensions (beta-1	68d	98.11.16	99.02.17
10	Validate Manager API	44d	99.01.05	99.04.06
11	Metadata DB Select/Table Desigr	90d	98.09.21	99.01.22
12	Metdata API	60d	98.12.14	99.03.07
13	Validate Metadata API	14d	99.03.08	99.03.25
14	Numerical Filter Libraries	69d	98.12.01	99.03.05
15	Data Conditioning API	65d	98.12.07	99.03.05
16	Validate Data Conditioning API	14d	99.03.08	99.03.25
17	GUI API (TCL/TK)	21d	99.03.12	99.04.09
18	WEB API	21d	99.03.12	99.04.09
19	Validate User APIs	21d	99.04.12	99.05.10
20	Disk Cache API	20d	99.03.01	99.03.26
21	Validate Software	11d	99.04.07	99.04.21
22	Install @ LHO	7d	99.04.22	99.04.30
23	Beta1 LDAS to Hanford	0d	99.04.26	99.04.26
24	Beta2 LDAS to Hanford	0d	99.10.10	99.10.10
25	**MILLIKAN PROTOTYPE DEVELOPM**	**85d**	**98.11.09**	**99.03.05**
26	Move Ultra30+RAID from 40m	5d	99.02.10	99.02.16
27	Configure Ultra 30 as data conditi	11d	99.02.17	99.03.03
28	Setup Sadan as Data dist. server	14d	98.11.10	98.11.27
29	Setup BW cluster	85d	98.11.09	99.03.05
30	Hardware ready for validatation te	0d	99.03.05	99.03.05
31	**HARDWARE for Beta-1**	**58d**	**99.01.04**	**99.03.24**
32	**LAN Procurements**	**38d**	**99.02.01**	**99.03.24**
33	ATM Switch (fore ASX-1000)	21d	99.02.01	99.03.01
34	Fiber Optics cables	14d	99.02.01	99.02.18
35	ATM-100BT Fanout	21d	99.02.01	99.03.01
36	CISCO Router	28d	99.02.01	99.03.10

LDAS Developement Timeline



ID	Task Name	uratio	Start	Finish
37	Integrate site LAN, setup LDA	7d	99.03.16	99.03.24
38	**Server procurements**	**51d**	**99.01.04**	**99.03.15**
39	Sun 60 server	35d	99.01.04	99.02.19
40	Ultra 30 DBMS Server	35d	99.01.04	99.02.19
41	Fiber Channel U60 to 450 Se	28d	99.01.11	99.02.17
42	Electronics rack for equipmer	28d	99.01.11	99.02.17
43	160 GB Data server DISK ca	28d	99.01.11	99.02.17
44	50 GB DBMS disk cache	21d	99.01.25	99.02.22
45	Integrate LDAS Data Distribu	15d	99.02.23	99.03.15
46	**WA 2km Installation**	**350d**	**99.04.26**	**00.08.28**
47	WA 2km I/O Ready	0d	99.04.26	99.04.26
48	WA 2km Michelson Ready	0d	99.10.18	99.10.18
49	WA 2km IFO Ready@10^-20	0d	00.08.28	00.08.28
50	**WA 4km Installation**	**225d**	**99.12.06**	**00.10.16**
51	WA 4km I/O Ready	0d	99.12.06	99.12.06
52	WA 4km Michelson Ready	0d	00.03.20	00.03.20
53	WA 4km IFO Ready@10^-20	0d	00.10.16	00.10.16
54	**LA 4km Installation**	**325d**	**99.08.23**	**00.11.20**
55	LA 4km I/O Ready	0d	99.08.23	99.08.23
56	LA 4km Michelson Ready	0d	00.02.28	00.02.28
57	LA 4km IFO Ready@10^-20	0d	00.11.20	00.11.20
58	**Begin LIGO Detector Operations**	0d	01.11.05	01.11.05

LSC ORGANIZATION FOR DATA ANALYSIS

- **PURPOSE**

  Generate the results and publications of the project

  Maintain the scientific integrity of results

  Minimize duplication but retain independent approaches

  Provide communication

  Foster collaborations

  Allocate resources and generate new resources

  Protect students

- **MECHANICS**

  Announcement of intent

  Solicitation for co-investigators

  Proposal for specific study

  Method

  People and tasks

  PhD Theses

  Estimate for computing resources needed

  Time to complete

  Outline of papers

- **ORGANIZATION**

  LSC Data Analysis Coordinator

  Receives announcements of intent

  Fosters collaborations

  Reviews proposals

  Allocates computational resources with advice of review board

  LSC Software Coordinator

  Approves software for general use

  Assigns groups for validation

  Data Analysis Review Board

  Data Analysis Coordinator,

  Software Coordinator,

  Laboratory director

  Shared facility directors

  Data committee chairs

  Spokesperson