Science Analysis Software Overview

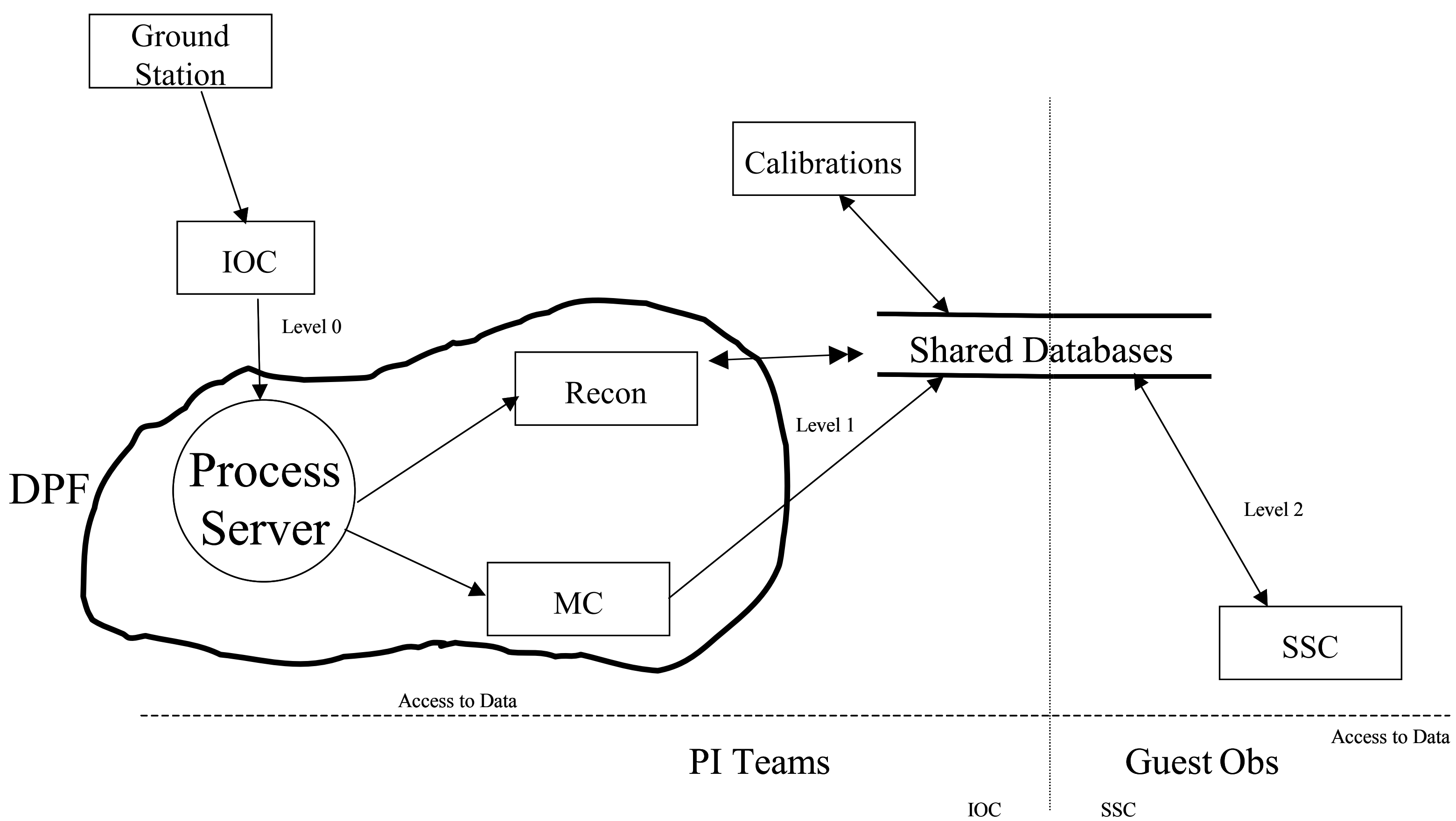
- Requirements
- Conceptual Design
- Deliverables
- Organization

Level 3 Requirements

- Instrument Simulation and Reconstruction
  - Detailed modeling of instrument and incident fluxes
  - Event level reconstruction to interpret instrument response and identify particle content
- Data Pipeline
  - Prompt processing of Level 0 data through to Level 1 event quantities
  - Providing near real time monitoring information to the IOC
  - Monitoring and updating instrument calibrations
  - Reprocessing of instrument data
  - Performing bulk production of Monte Carlo simulations
- Higher Level Analysis
  - Creating high level science products from Level 1 for the PI team
  - Providing access to event and photon data for higher level data analysis
- Interfacing with other sites
  - mirror PI team site(s)
  - SSC
- Supporting Engineering Model and Calibration tests
- Supporting the collaboration for the use of these tools

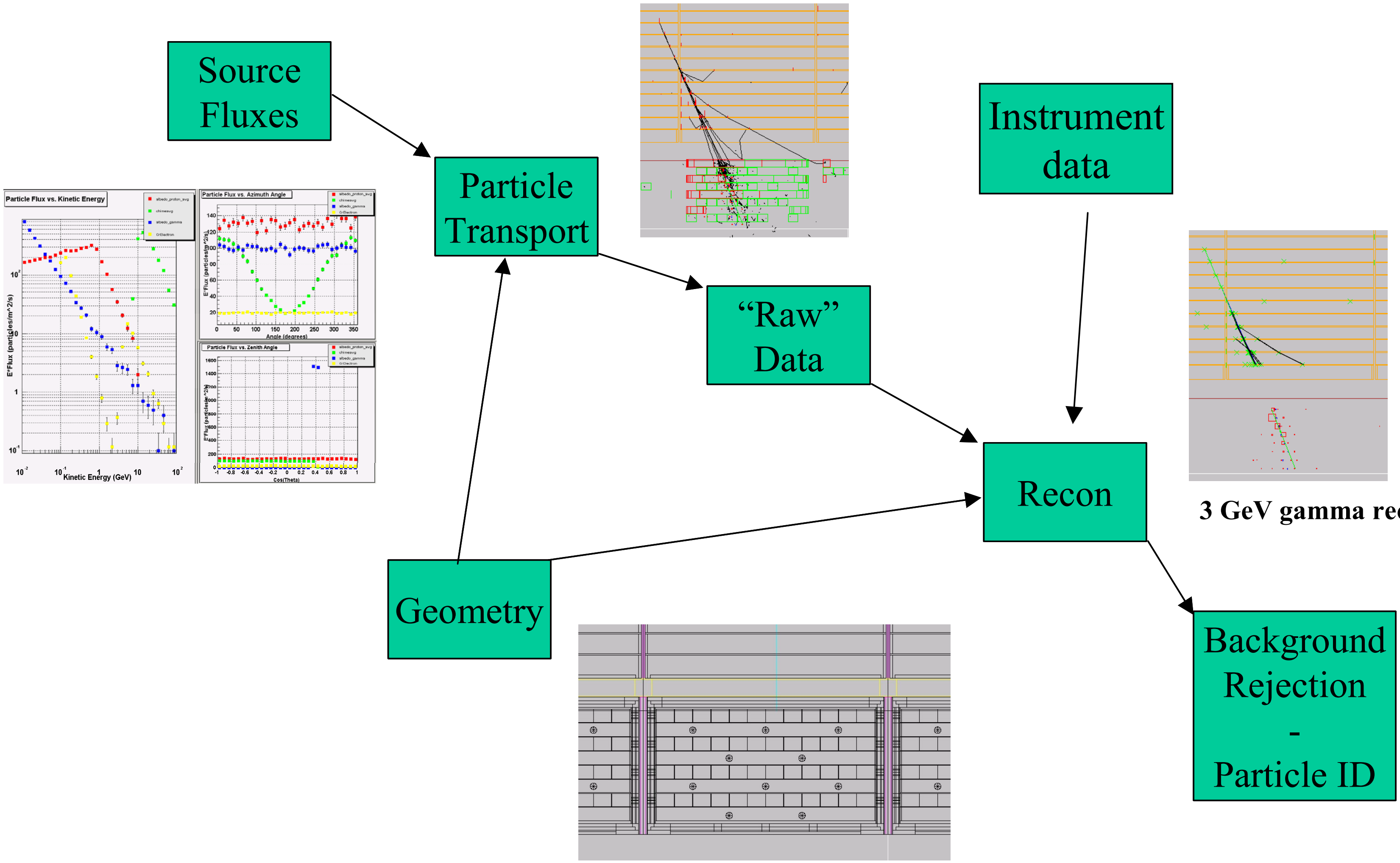
Data Flow

- Automated server to accept error-corrected data from IOC
- do Level 1 processing into Event database(s) shared w/SSC



Instrument Simulations and Reconstruction

3 GeV gamma interaction

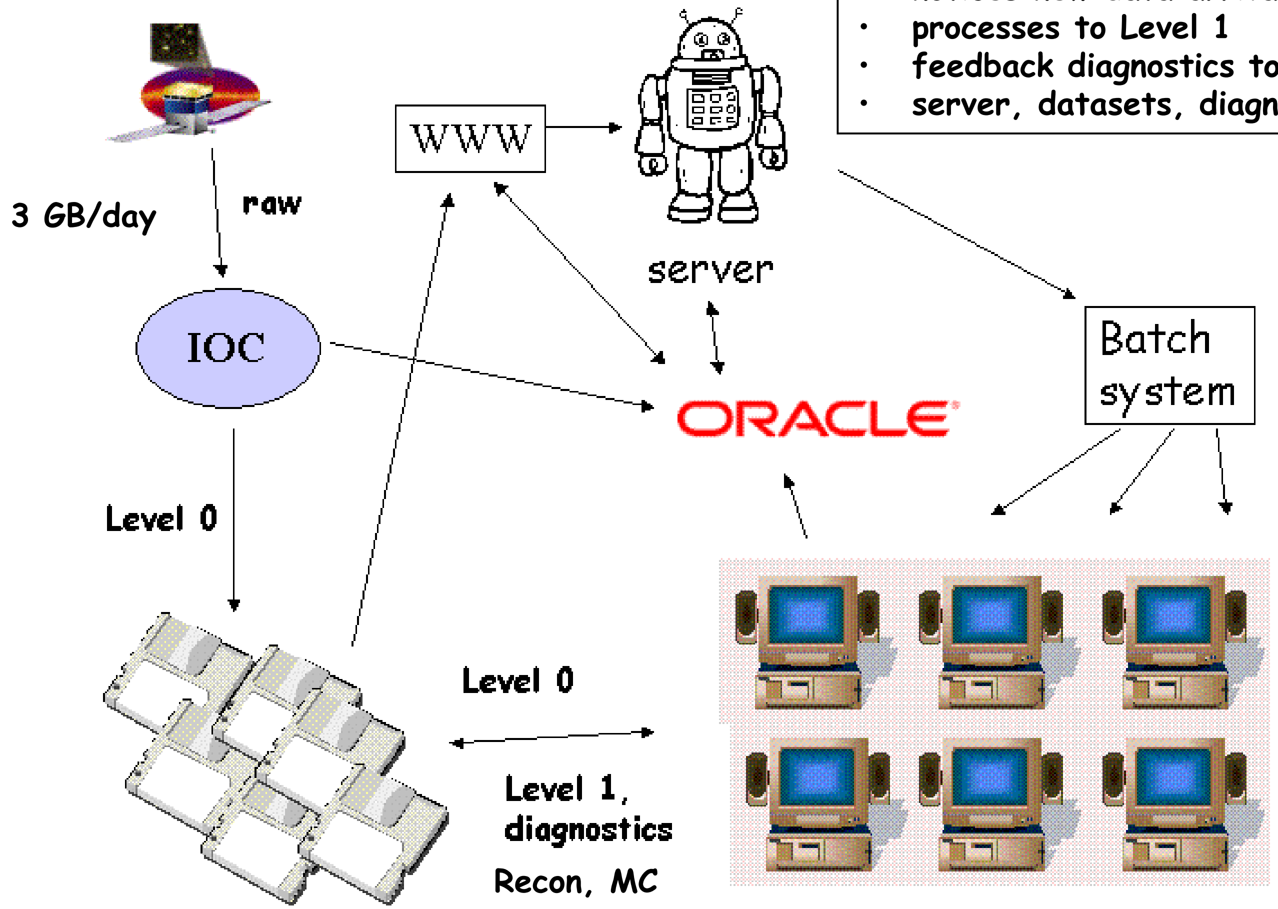


CAL Detail

Data Processing Facility

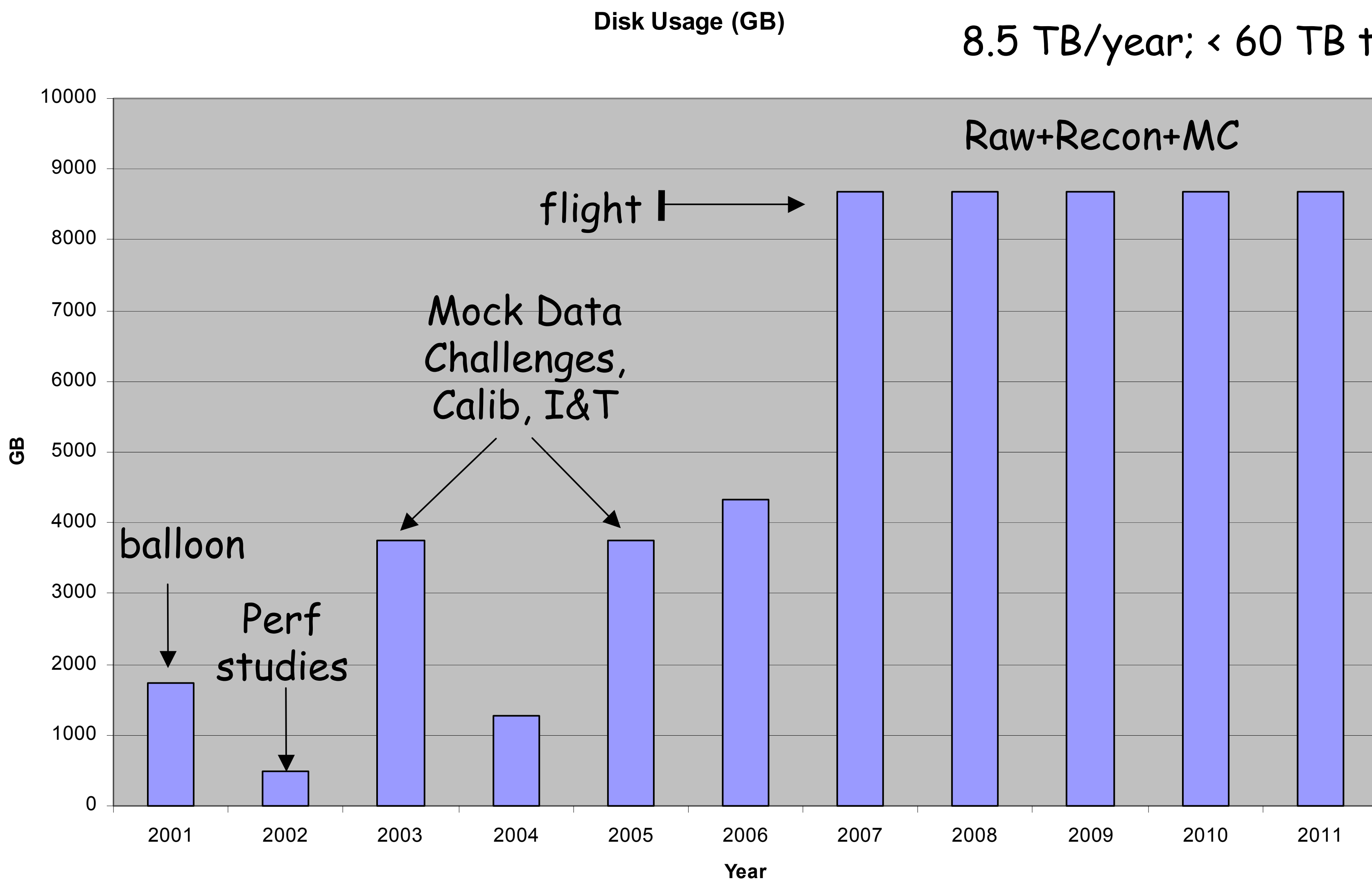
Fully automated server

- notices new data arrival
- processes to Level 1
- feedback diagnostics to IOC
- server, datasets, diagnostics tracked in DB



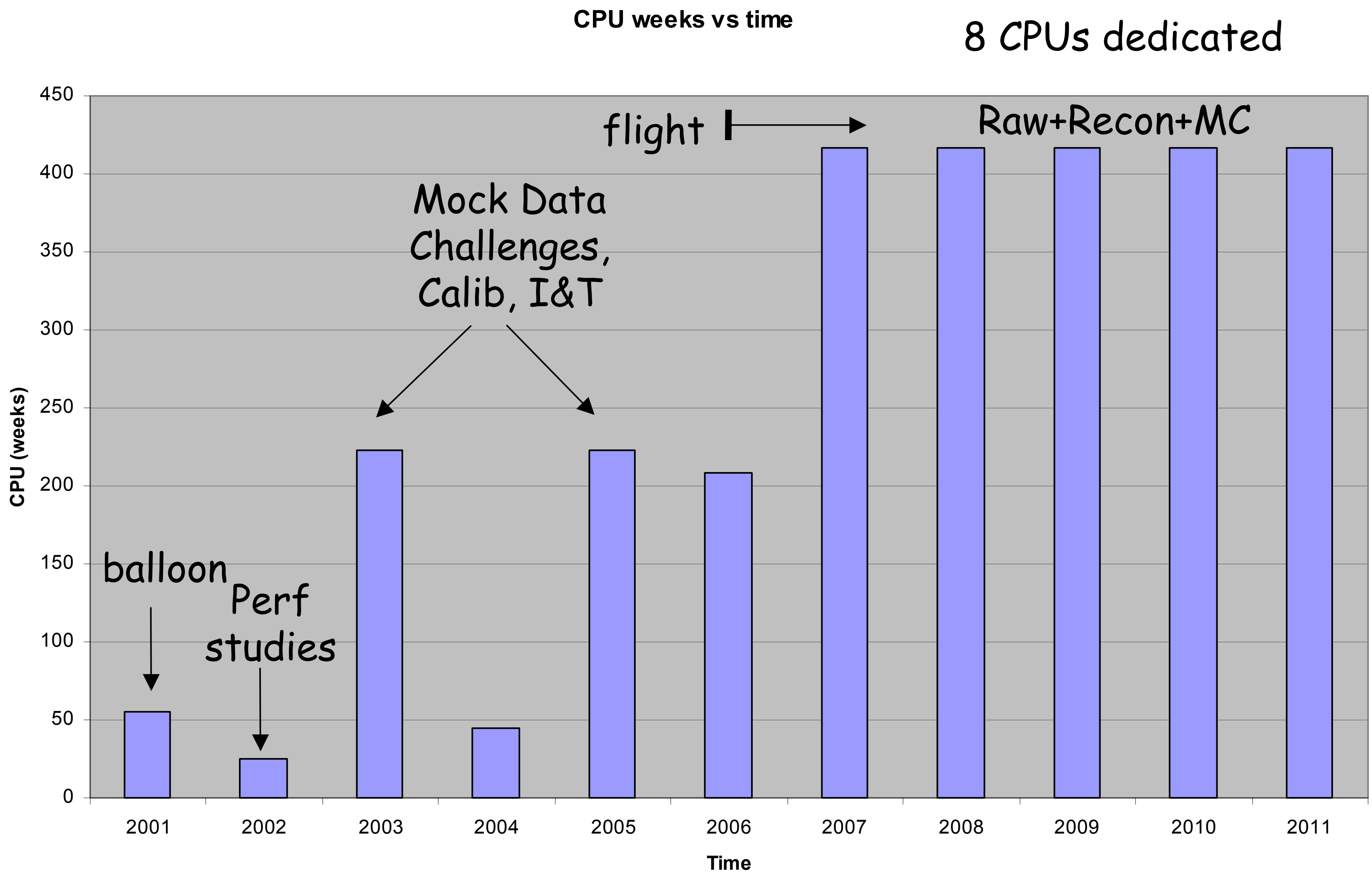
Processing Requirements - I

8.5 TB/year; < 60 TB total



(note: BABAR temp space is 10 TB)

Processing Requirements - II



(SLAC will have ~2000 CPUs for BABAR)

High-Level Science Analysis

- Analysis that follows all background rejection & PSF enhancement cuts to the data

- Inputs required:
  – Instrument response functions
  – Exposure history
  – Photons
  – Selected cosmic rays (e.g., heavies or high energy), for monitoring of calibration
  – Various astronomical catalogs, for identification of gamma-ray point sources, pulsar phase folding

High-Level Science Analysis

- Mission goals for science analysis are broad
  - Detection, localization of transient sources (AGN flares, GRBs, solar flares) with minimum latency
  - Establishment of a point-source catalog with positions, flux histories, spectra, identifications
  - Detection and characterization of pulsars, including new 'Gemingas'
  - Spectral and temporal evolution of gamma-ray bursts and solar flares (with GBM)
  - Study of production and distribution of cosmic rays via diffuse sources of gamma rays
  - Measurement of the point source fraction of the isotropic gamma-ray background
  - Special analyses (including WIMP annihilation lines)
  - ...

Infrastructure

Merriam-Webster's COLLEGIATE® DICTIONARY

Main Entry: in·fra·struc·ture
Pronunciation: 'in-fr&-"str&k-ch&r, -(")frä-
Function: *noun*
Date: 1927
1 : the underlying foundation or basic framework (as of a system or organization)
2 : the permanent installations required for military purposes
3 : the system of public works of a country, state, or region; *also* : the resources (as personnel, buildings, or equipment) required for an activity

Credit: Steve Ritz

- What falls under this topic?
  - Supported OS's
    - Windows, Linux, Solaris
  - File management – cvs
  - Code management – CMT
  - Code framework – Gaudi
  - Object I/O – Root
  - Coding rules
  - Documentation – DOxygen
  - Code Release Management
  - Event-level analysis platforms
    - Root, IDL
  - User support
    - User doc
    - Pre-built code installations

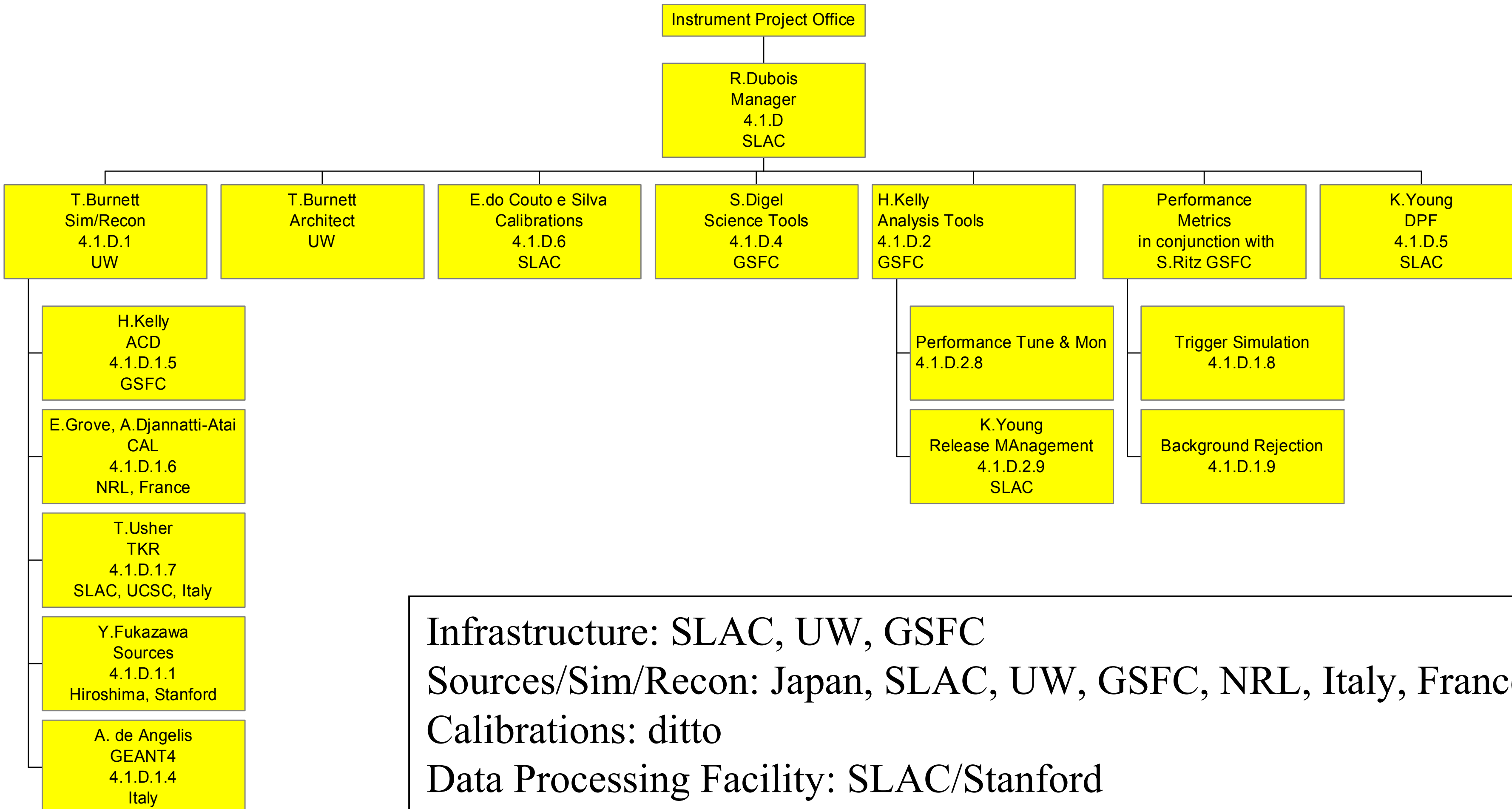
Deliverables - I

- Instrument Simulations
  - physics models of the expected particle fluxes for the flight instrument
  - geometry models of the instrument and spacecraft, and of engineering models
  - simulation of particle transport through those geometries
- Reconstruction of events
  - from data or Monte Carlo for flight instrument and engineering models.
  - includes emulating the trigger, and interpretation of the events in terms of particle content.
- Calibration algs and the machinery to store and access the calibrations, based on time of applicability
- Detailed and summary output from reconstruction
  - sufficient to understand the reconstruction process, as well as particle information including type, direction, energy and error estimates
- Data Production Facility
  - automated server to handle Level 0 to Level 1 processing
  - high level instrument diagnostics to feed back to IOC Ops in near real time
  - database cataloguing state of the server as well as of input and output datasets
  - Shared databases

Deliverables - II

- Science Tools
  - Instrument response functions
    - Effective area, energy resolution, energy redistribution, and point-spread function for all gamma-ray event types
  - Timeline (as observed)
    - Observing mode and spacecraft position & orientation as a function of time. Command and performance states.
  - Exposure history
    - Detailed timeline that includes live-time and coverage information for a grid on the sky, for rapid generation of exposure maps
  - Source catalog
    - Positions, fluxes, and uncertainties for all detected sources in the sky survey. Includes flux histories, spectral indices, and identifications
  - GRB/transient alerts
    - Most initial GRB and bright AGN flare alerts will be generated on the spacecraft; these SAS alerts will provide refined information, or for many AGN flares, the initial notification.
  - Interstellar emission model
    - The interstellar emission model is only loosely speaking a data product; it will be refined as necessary using flight data. It is essential for the production of the source catalog, and for likelihood analysis of GLAST gamma-ray data in general, so in any case it is a deliverable.
- Infrastructure
  - code architecture and coding rules
  - code development and release management tools, including code repository, code management tool, release management and verification tools
  - low-level (Level 0 and 1) analysis tools with access to data (eg ROOT & IDL) and event display for visualization
- Documentation and support of the collaboration user community for the use of the above deliverables

Organization



Infrastructure: SLAC, UW, GSFC
Sources/Sim/Recon: Japan, SLAC, UW, GSFC, NRL, Italy, France
Calibrations: ditto
Data Processing Facility: SLAC/Stanford
Science Tools: GSFC, SLAC/Stanford, collaboration

Major responsibilities agreed to with Italy & France

Group Communications

use cvs for distributed code development
Repository at SLAC

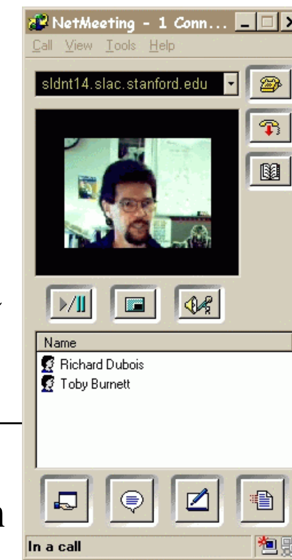
CVS Concurrent Versions System CVShome.org

VRVS Virtual Room Videoconferencing System

use web conferencing tools for meetings
Weekly General, Core & CAL; soon TKR
Used for impromptu discussions

use instant messenger tool for quicky discussions

ICQ.com Site Guide



Use MS NetMeeting for point-to-point discussions
and Windows remote debugging via desktop sharing

Software Weeks every 4 months

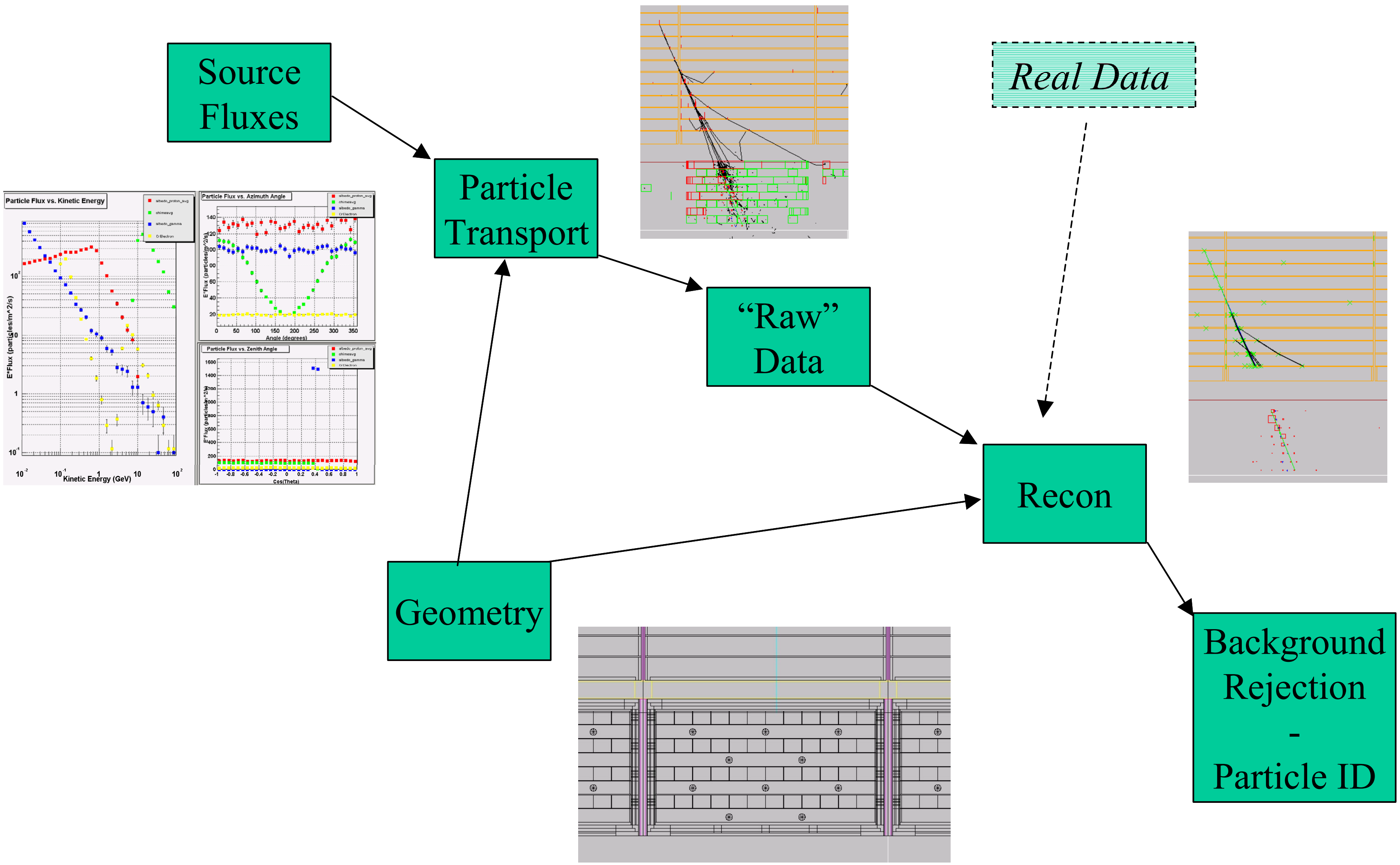
Core working meeting between Software Wks

software team is integrated
subsystem folks are matrixed between subsystem and software group
part of same development process as other components of software

Infrastructure
Sources
Simulation
Reconstruction

Presented by T. Burnett

The Processing chain



Our Products: much more than code!

- Support infrastructure, must support a variety of clients:
  - developers
  - sophisticated users
  - end users
- Elements:
  - Supported platforms & compilers
  - Development environments
  - Coding and documentation standards
  - Build tools
  - Framework
  - Analysis tools

Basic principles for technology choices

- Don't invent anything unnecessarily
- Borrow from existing solutions, experience

→ **High energy physics**

- – very similar parameters: detectors, analysis requirements, data, users
- – Pioneer was here at SLAC: the Babar experiment in mid 90's,
    - Broke with Fortran-oriented past: unix, OO C++
    - Adopted industry-standard *CVS* for version management
    - Invented package-oriented build system *SRT*
    - Developed an OO *framework* for managing processing steps
    - Successfully trained physicists to deal with new environment

Technology choices: language

- Object-oriented C++
  - Basic value of encapsulation of data now well-established
  - Build on success of Babar and all other new HEP experiments: Belle, D0, CDF, ATLAS, CMS, LHCb
  - Now a standard, most compilers approach this
  - Standard Template Library provides rich menu of algorithms and containers.
  - Required to use a framework

Technology choices: platforms

- Windows PC
  - Our preferred development environment due to rapid development made possible by Microsoft Visual C++ MSDEV
- linux
  - The preferred choice for European developers
  - Required for SLAC batch support
- solaris
  - Required for SLAC batch.

Technology choices: code versioning

- CVS!
  - Concurrent Versions System, the dominant open-source network-transparent version control system.
  - Useful for everyone from individual developers to large, distributed teams:
    - Client-server access method lets developers access the latest code from anywhere there's an Internet connection.
    - Unreserved check-out model to version control avoids artificial conflicts common with the exclusive check-out model.
    - Client tools are available on all our platforms.
  - Web-based repository browser available (cvsweb)
  - Automatic conversion of CR/LF

Choices: Code management

Legacy of Babar’s SRT: building apps from *packages*

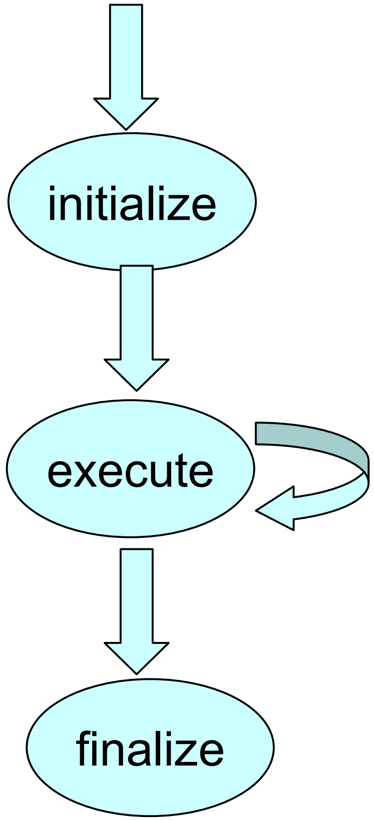
- Package: collection of source files, with public header files in a folder (usually) with the package name
- Produces a binary library and/or executable

CMT (for Code Management Tool): our choice

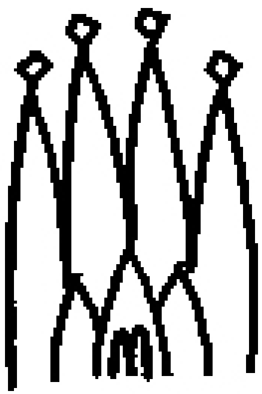
- Developed at Orsay in response to deficiencies of SRT, adopted by LHCb and ATLAS
- Supports Windows
- Clean model for package dependencies
  - Support for compile-time, link-time, and execution-time
- Configuration specified in a single file
- Includes tool to generate makefiles, or MSDEV files
- Uses CVS tags to correspond with versions

framework requirements

- Support event-oriented processing, three phases
  - initialization
  - event-loop generating or processing events
  - termination
- Define flexible way to specify processing modules to be called in the execute loop
- Provide *services*, especially for making n-tuples and histograms

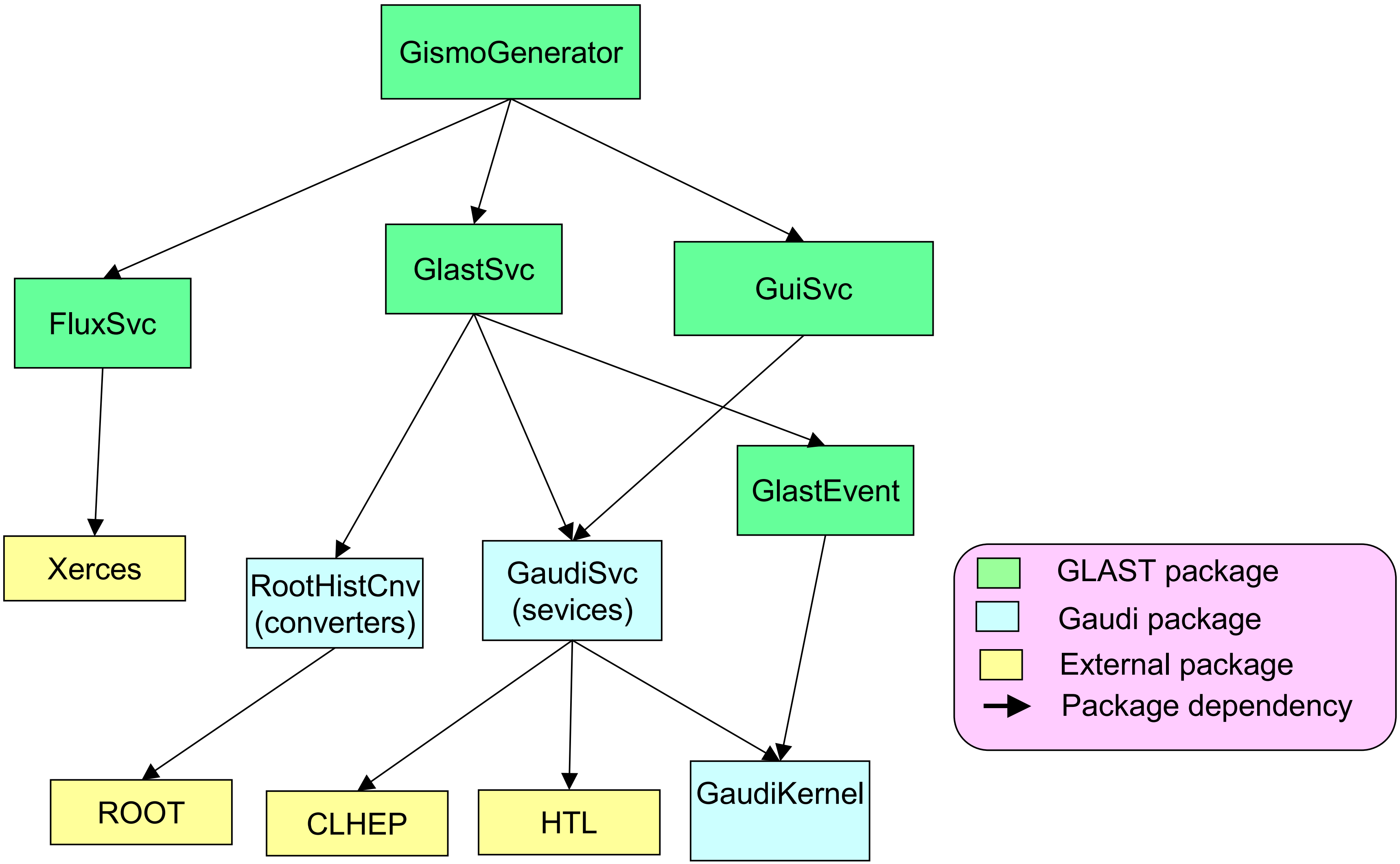


Gaudi: our framework choice



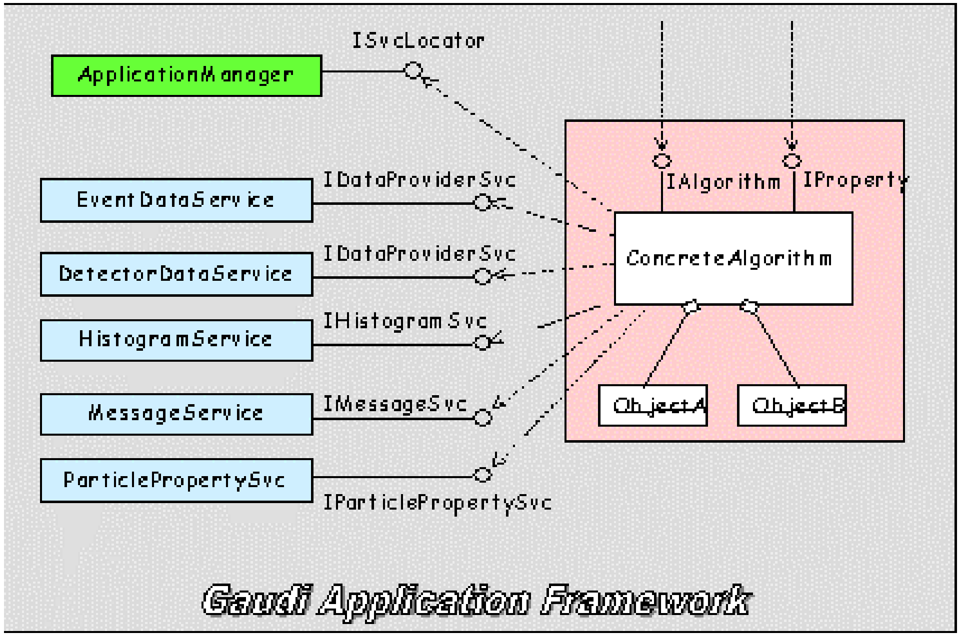
- Open source
- Stable, but active developers, in use by ATLAS, LHCb
- Very good documentation
- All code called via component interfaces:
  - Algorithm
  - Service
  - Converter
  - DataObject
- Support for shareables: all code is loaded dynamically
- Job control parameters set in job options file.

Glast/Gaudi Example Architecture

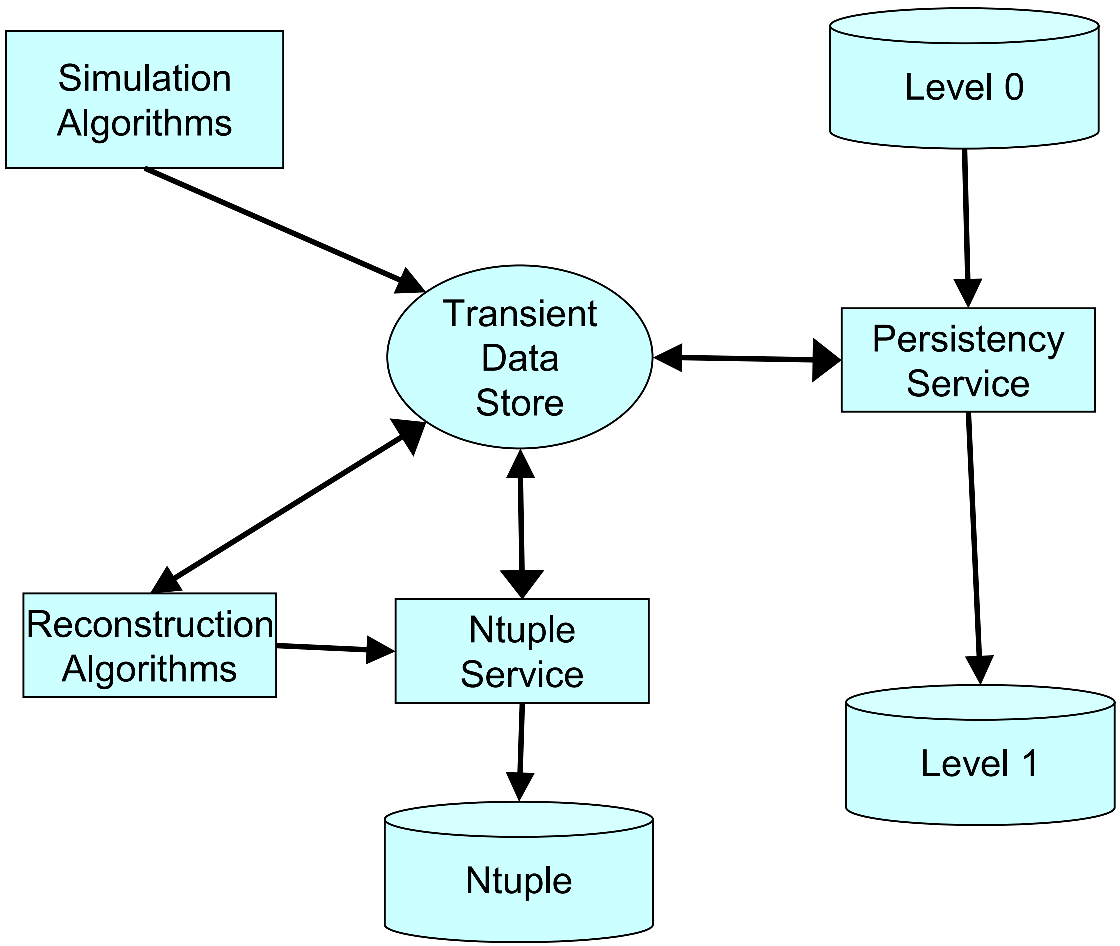


Gaudi Algorithm as a *component*

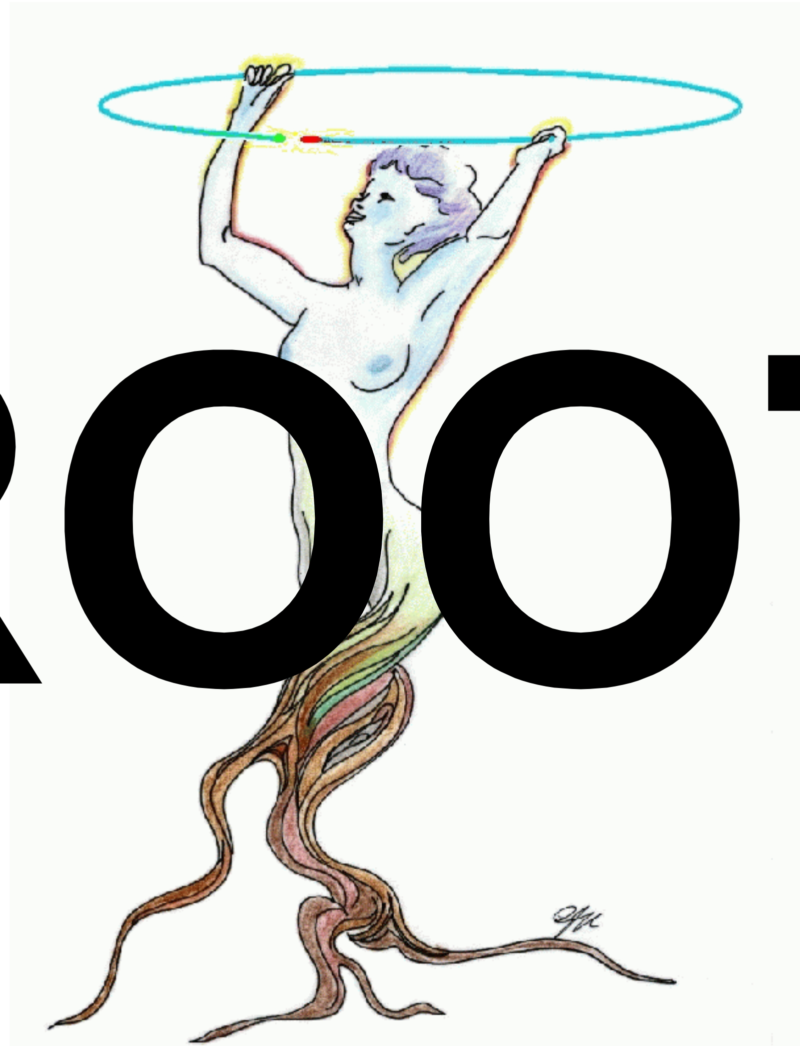
- Components are similar to Corba or COM: implement an abstract interface.
- Easy to substitute components
- Example diagram: A ConcreteAlgoritm:
  - Implements 2 interfaces
  - requests services from 6 services via abstract interfaces



Data flow in the Gaudi framework



Choice: I/O format (and Event Analysis)



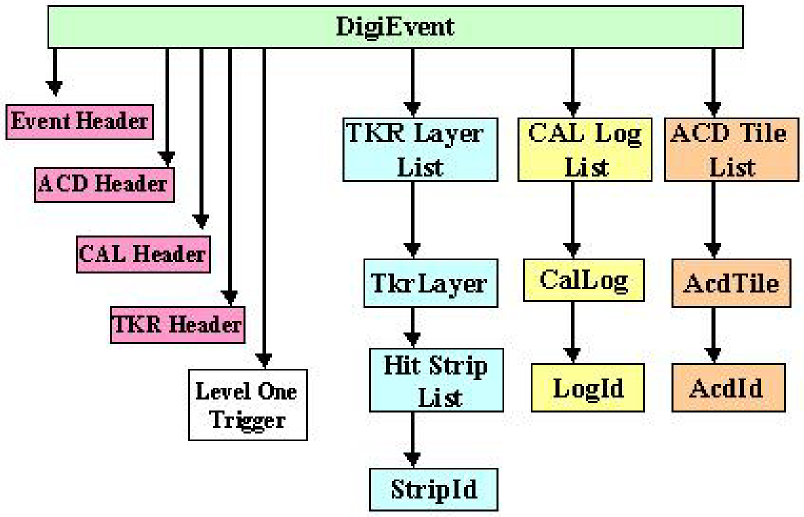
ROOT

Features of ROOT I/O

- Machine independence
  - ROOT is freely available on all of our supported platforms.
- Self-describing
  - Files created today will be readable years from now.
- Support for Object I/O
  - The detailed structure of our data is preserved for analysis.
- Schema evolution
  - Changes in our internal data structures will be tracked.
- On the fly compression
  - ROOT uses an algorithm based on gzip.
- Widespread use in the HEP community.
  - CDF at FNAL; several experiments at RHIC

Object I/O

- Detailed tree structure of data is preserved.
- Described by C++ classes
- Branched I/O
  - Reduces unnecessary I/O by reading in only desired branches.
- Summary data is available in ROOT Ntuples.



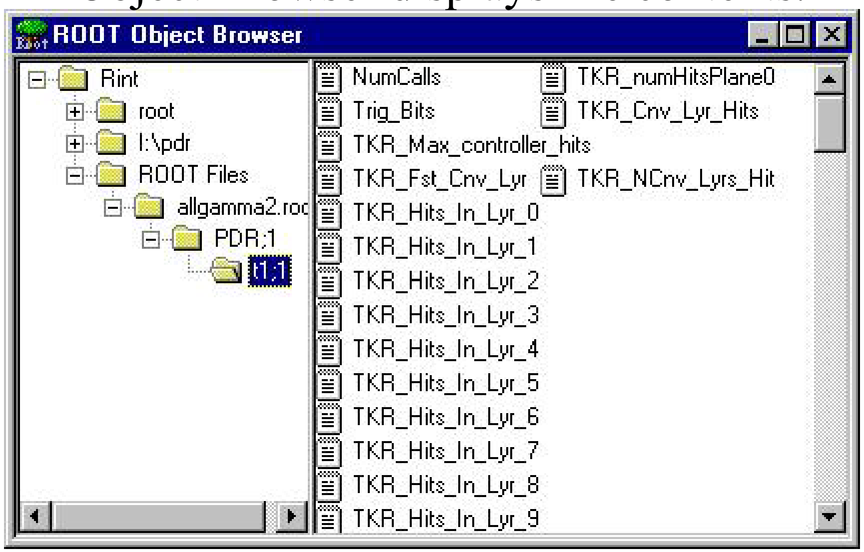
Logical structure for the raw digitization data

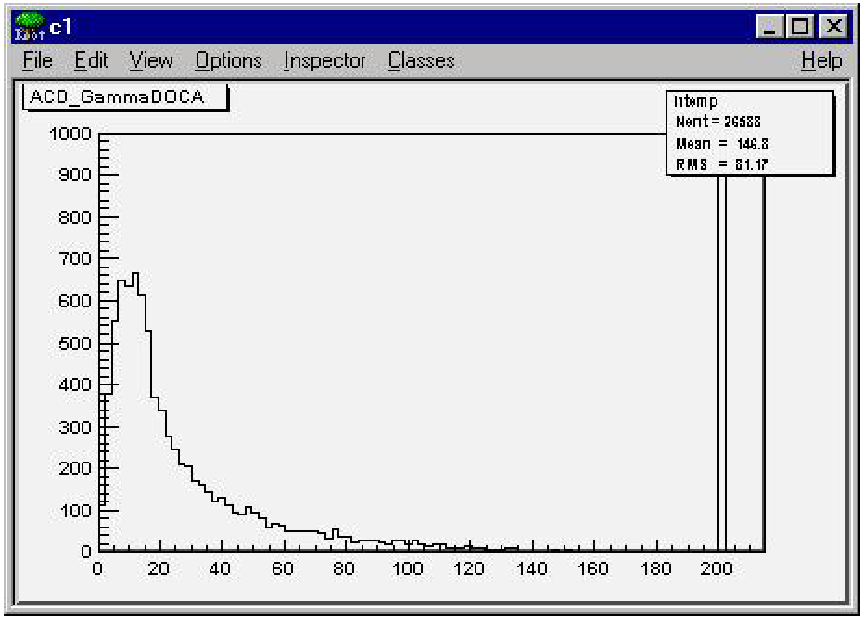
Internal structure for storage of detector data

ROOT for Event Analysis

- Supports both interactive and batch processing.
- Free and available on all supported platforms.
- Strong and growing user base.
- Histogramming, function fitting, and GUI widgets.

Object Browser displays file contents.





Histograms produced at the click of a button.

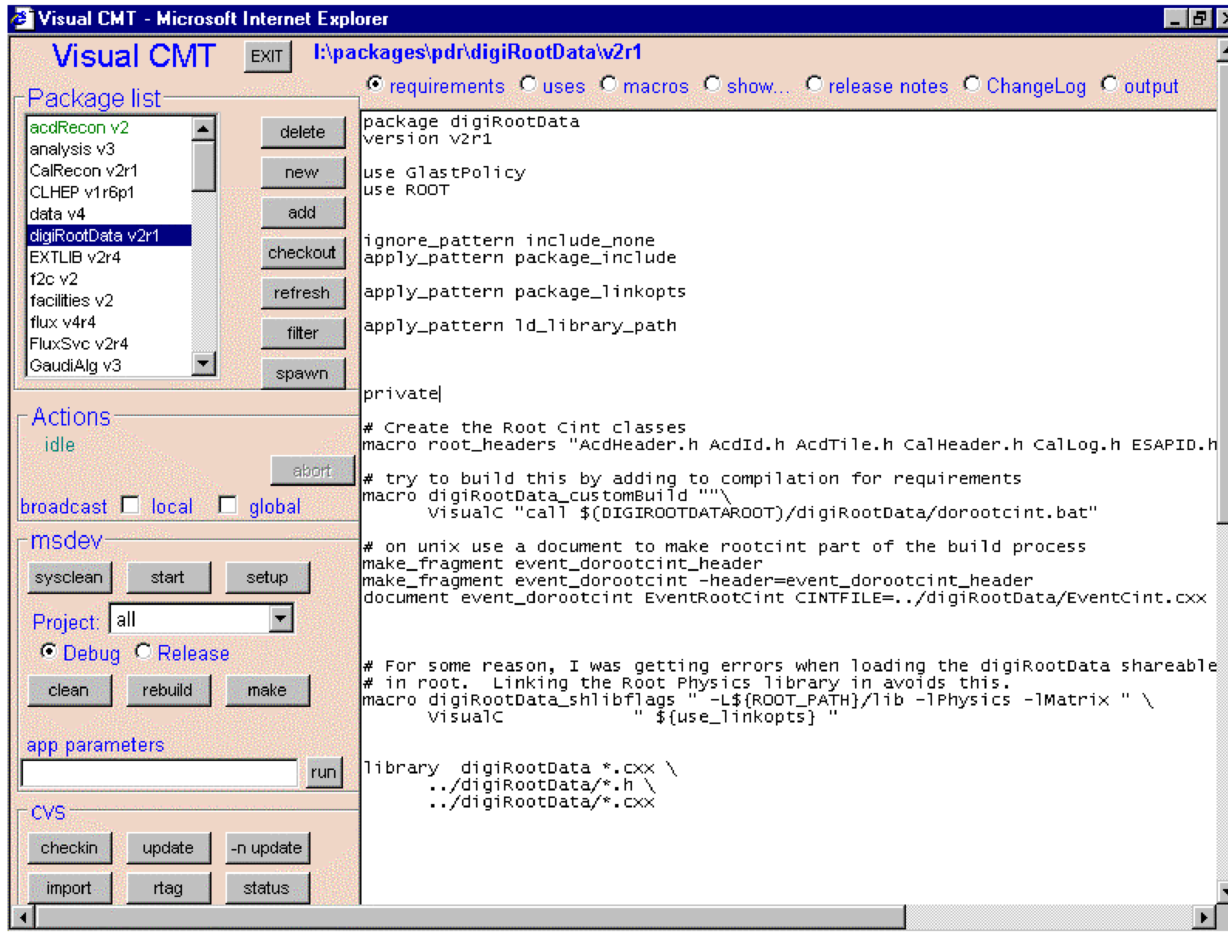
Documentation, user support

- Gaudi, CMT, cvs: user guides available
- Local guides (web-based)

Software	[Getting Started with GLAST Software (Your How-To Page)] [Web Access to CVS repository] [Using GlastSim] [Using tbsim] [Using ROOTWriter] [Using tb_recon]
Support	[Whom to Call??] [Facilities at SLAC] [UW Windows Server]
Projects	[GAUDI] [GEANT4] [PDR] [Software PDR] [Event Display]
Other Software Resources	[Italy] [UCSC TB Recon] [Goddard] [NRL Software] [Hiroshima]
Tools	[Telecon VRVS] [Using VRVS for Glast] [Instant Msg ICQ] [Using ICQ for Glast] [CVS] [Using Cvs for Glast] [CMT] [Using CMT for Glast] [Root] [Using Root for Glast] [Root at FNAL] [Creating PEGS files]

Help in the form of a GUI

- GUI interface to:
  - CMT: manage packages
  - CVS: check out, commit
  - MSDEV: build, or start its GUI
- Prototype for Windows. Plan to extend to unix.



The Coding Process

- Inline documentation: doxygen
  - Each package must have a *mainpage.h* to introduce the purpose, provide direct links to top-level classes
- Coding rules
  - Avoid potentially bad constructions
  - Maintain some uniformity
- Testing
  - Each package defines test programs
- Reviews
  - Periodic reviews of code for design, adherence to reviews

package flux
v4r5
This package contains all code to generate particles for GLAST simulation. The primary interface is via a FluxMgr object.
A list of possible sources, with details on implementation, is in the file xml/source_libarary.xml
All calculation of spectra is done in Spectrum objects,

Managed setups for developers

University of Washington Terminal Server

- Uses Windows 2000 Terminal Server: free clients available for any Windows operating system
- Complete environment available for users, including VCMT, cvs, ssh, msdev

SLAC unix

- Standard group .cshrc
- releases automatically available

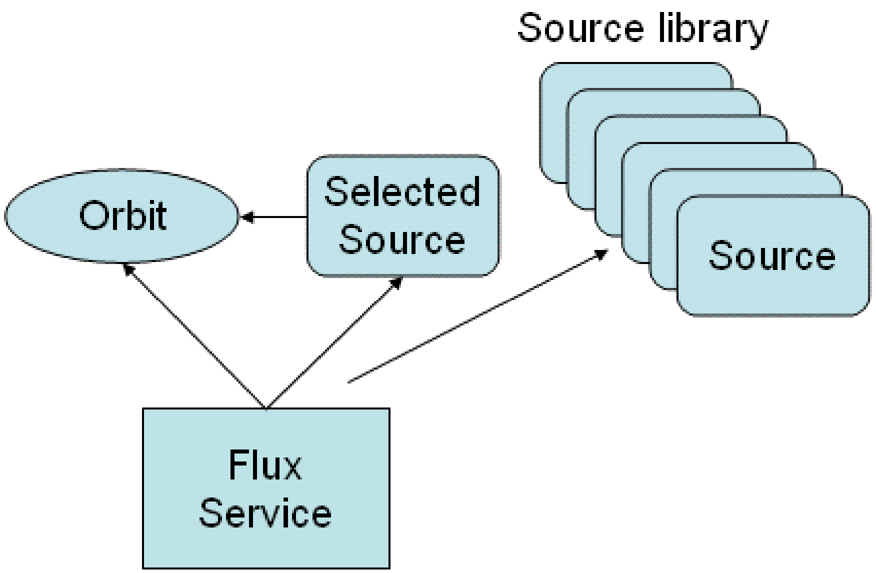
Both: plan to implement automatic build facilities for

•overnight builds of HEAD versions

•on demand builds of specified packages

Sources: Incident Flux

- Service to provide incoming particles for simulation
- Types that must be available:
  - Primary and secondary Galactic Cosmic Rays: protons and electrons
  - Albedo gammas
  - gammas for testing resolution
  - Galactic gamma point sources
  - Galactic diffuse sources
- distributions of energy spectra
- angles with respect to:
  - local zenith
  - spacecraft
  - galaxy



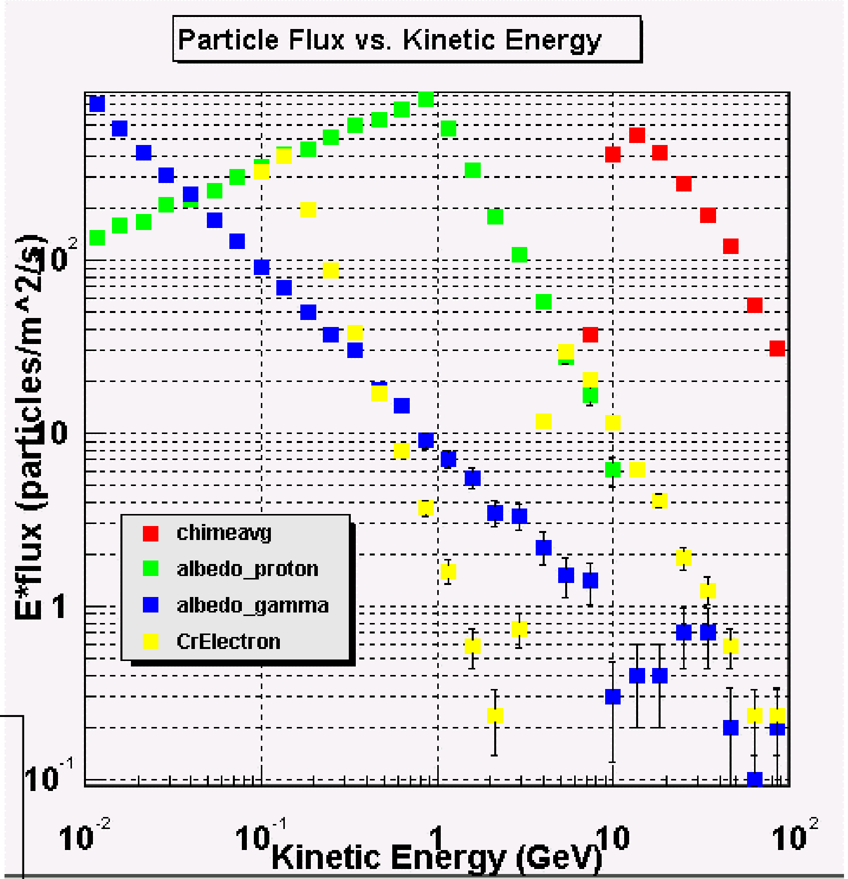
Flux Service:

- Selects from library (XML spec)
- Manages orbital parameters
- Returns particles generated by selected source

Selected Source: return particles depending on orbit

Rootplot: A useful utility to study sources

- Plot at right generated by a utilty program in the flux package.
- Can choose any combination of sources described in the XML file, and generate distributions of energies and angles that would be provided to the service.

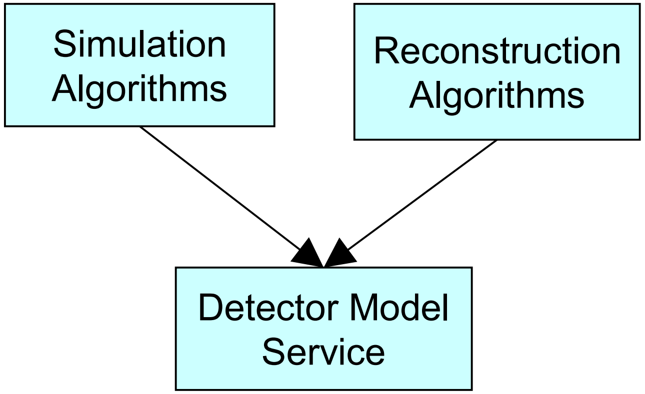


Plot of the energy spectra for various components of a proposed background mixture,

including: :

- chimeavg, representing a average rate for the CHIME model of primary proton cosmic rays;
- albedo_proton, the spectrum of albedo and reentrant protons corresponding to recent measurements;
- albedo_gamma, secondary gammas from the horizon, and
- CrElectron, a mixture of primary and secondary electrons and positrons. The abscissa is the kinetic energy of the particles (gamma, proton, or electron) in GeV, and the ordinate the flux times energy integrated over angles, in particles/(m2 s).

Detector Model

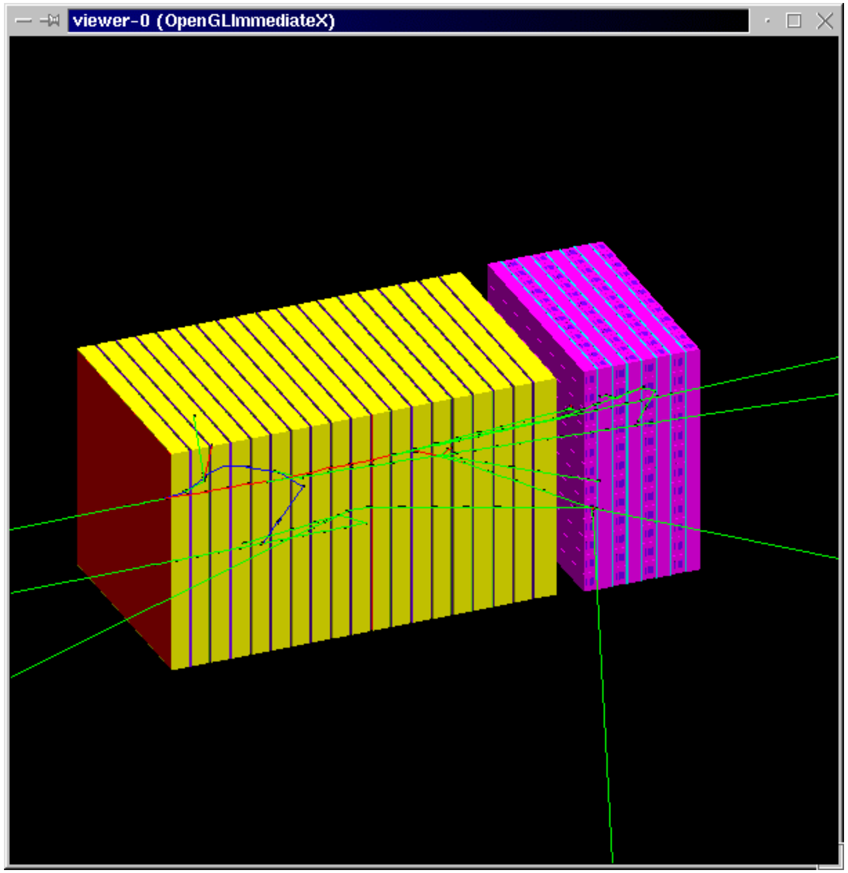


Detector Model:

- **Hierarchical geometry, each volume with unique ID**
- **Materials**
- **Sensitive detectors**

Implemente as a service, provides description of GLAST to clients with perhaps different needs.

Data in XML format



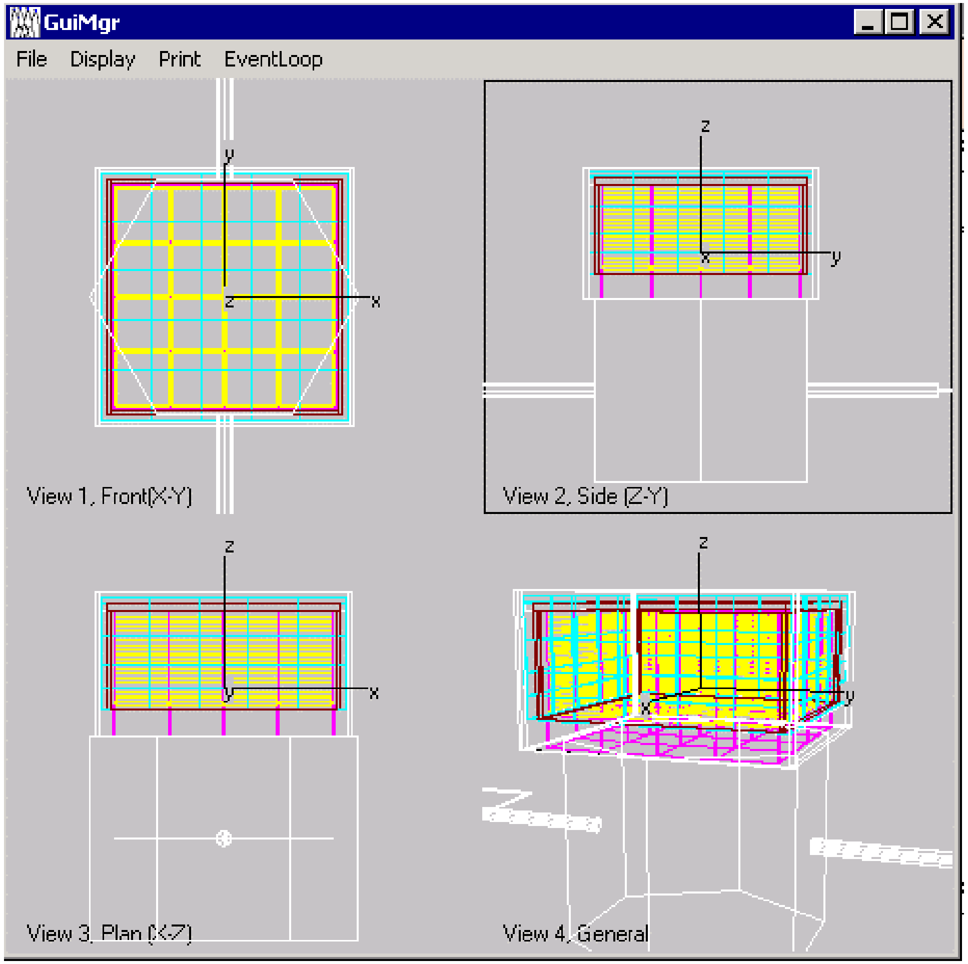
Geant4 demonstration

Event and Detector display

Interactive 3-D display is vital

GUI also can control processing

(This is a prototype: plans to adopt a HEP standard instead.)

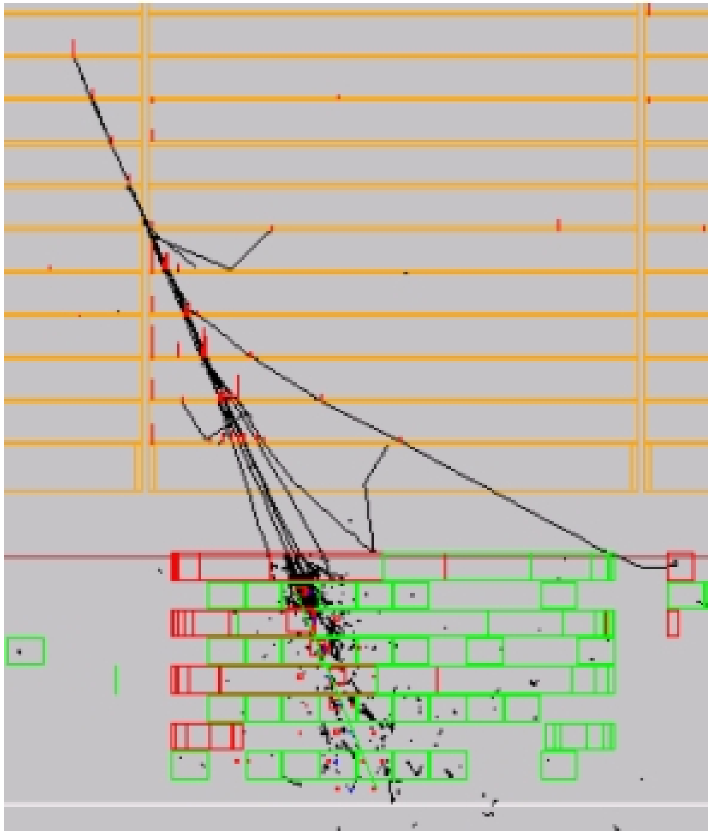


Depositing energy: bookkeeping design

- **Particles transported by the simulation deposit energy in matter by ionization loss, in many small steps**
- **Each loss is associated with the given volume, two strategies**
  - **Single-step: every step saved**
  - **Volume integrating: only keep total, perhaps in subdivisions**
- **Primary objective: create realistic detector response**
- **Secondary objective: preserve enough information about the underlying event to guide design and evaluation of reconstruction strategies**
  - **Parent particle: incoming or e+/e- from pair conversion**

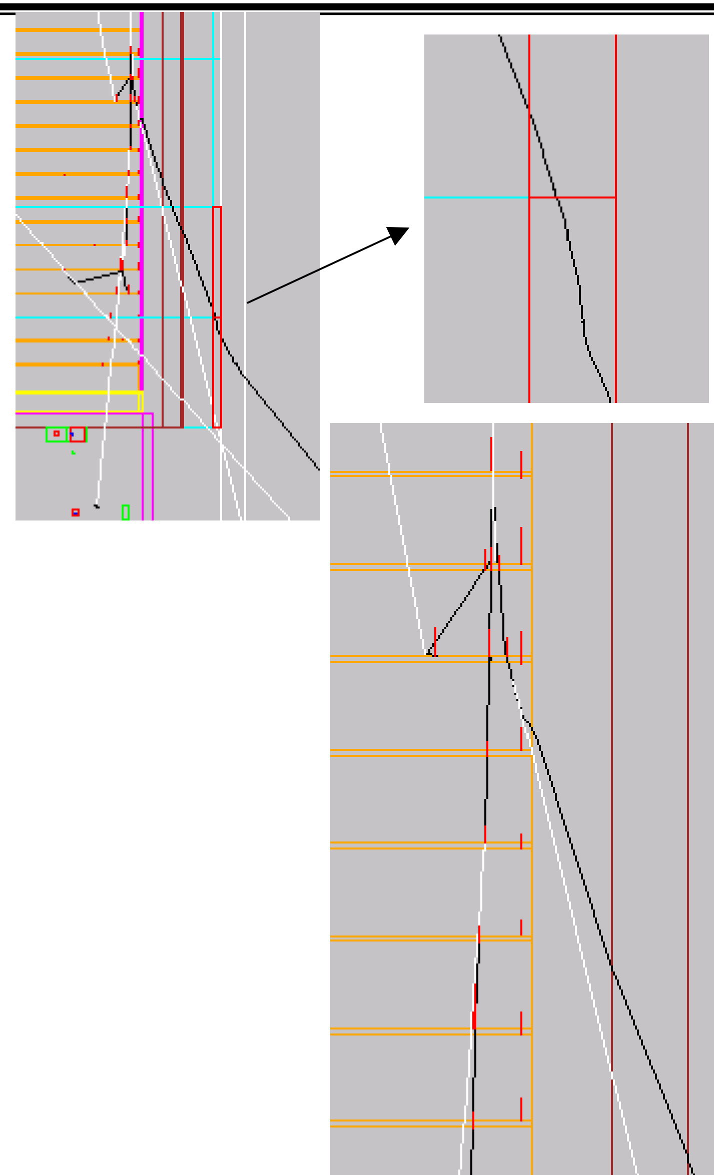
3 GeV photon interaction (charged particles shown only)

Detector responses shown



Digitization Requirements

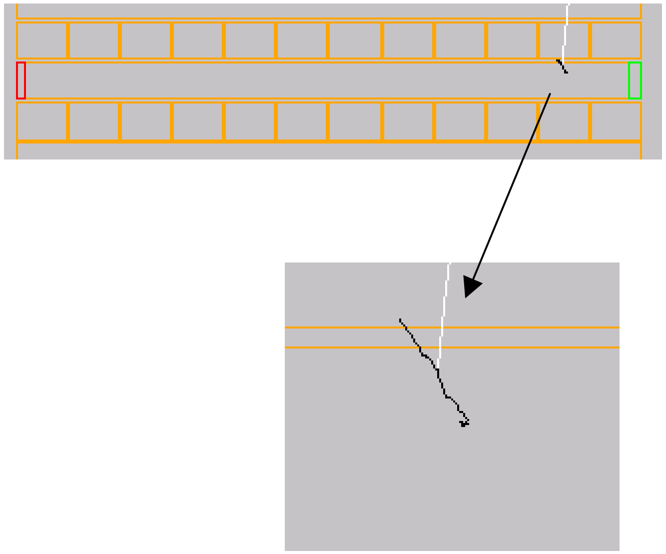
- **ACD**
  - total energy deposited
  - position of all steps and associated MC parent particle



- **TKR**
  - the dead material energy loss must be segmented at least by plane
  - Silicon treated as one volume, but complete detail of each step in the silicon.

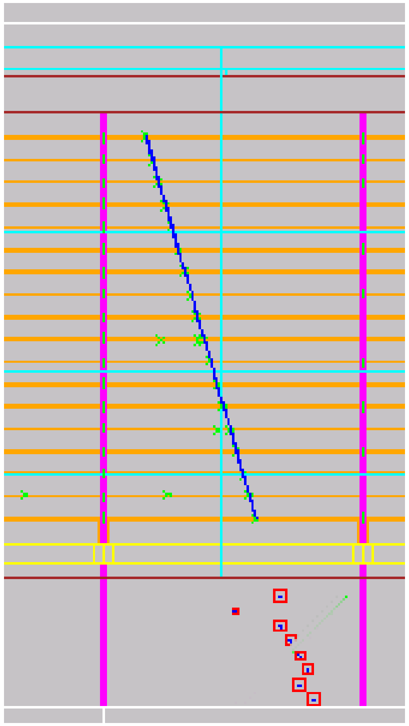
Digitization Requirements

- CAL
  - Each crystal treated as single volume
  - Impractical to save every step in a big shower
  - Accumulate energy sums in slices.
  - Also register energy sum and energy-weighted longitudinal position moments.
  - Turn the resulting info into the four PIN diode readouts



Event Reconstruction

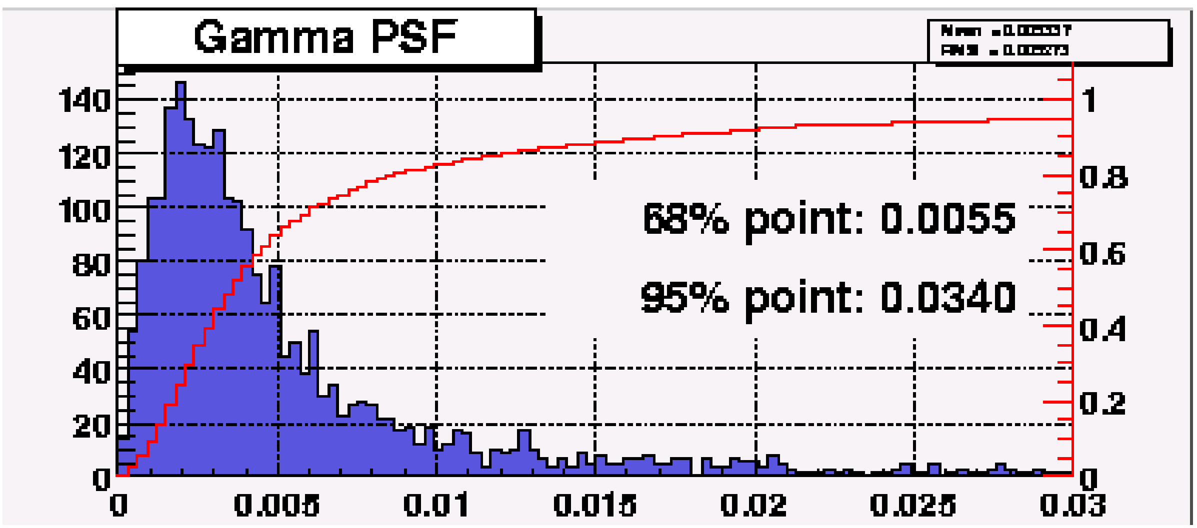
- **Sequence of operations, each implemented by one or more** Algorithms, **using TDS for communication**
  - **Trigger analysis: is there a valid trigger?**
  - **Tracker recon: patter recognition and fitting to find tracks and then photons in the tracker (uses Kalman filter)**
  - **Calorimeter recon: finds clusters to estimate energies and directions**
    - **Must deal with significant energy leakage since only 8.5 X_0 thick**
  - **Use the ACD to allow rejection of events in which a tile fired in the vicinity of a track extrapolation**
  - **Background rejection: consistency of patterns:**
    - **Hits in tracker**
    - **Shower in CAL: alignment with track, consistency with EM shower**



An easy rejection

PSF estimation from track fit

- **Requires comparison with MC truth, so at least incoming direction must be available**
- **Below is an example plot, showing flexibility of ROOT**
  - **The plot shows the need for more suppression of the tail of the distribution!**



Angle between reconstructed and incident photon (radians)
Red curve is fractional cumulative distribution, scale at right.

SAS – Calibration

Joanne Bogart, Eduardo do Couto e Silva,
Eric Grove and Leon Rochester

Calibration Goals

- Data reduction and analysis software are independent of calibration information (no “hardcoding”, software and calibration updates can happen independently)

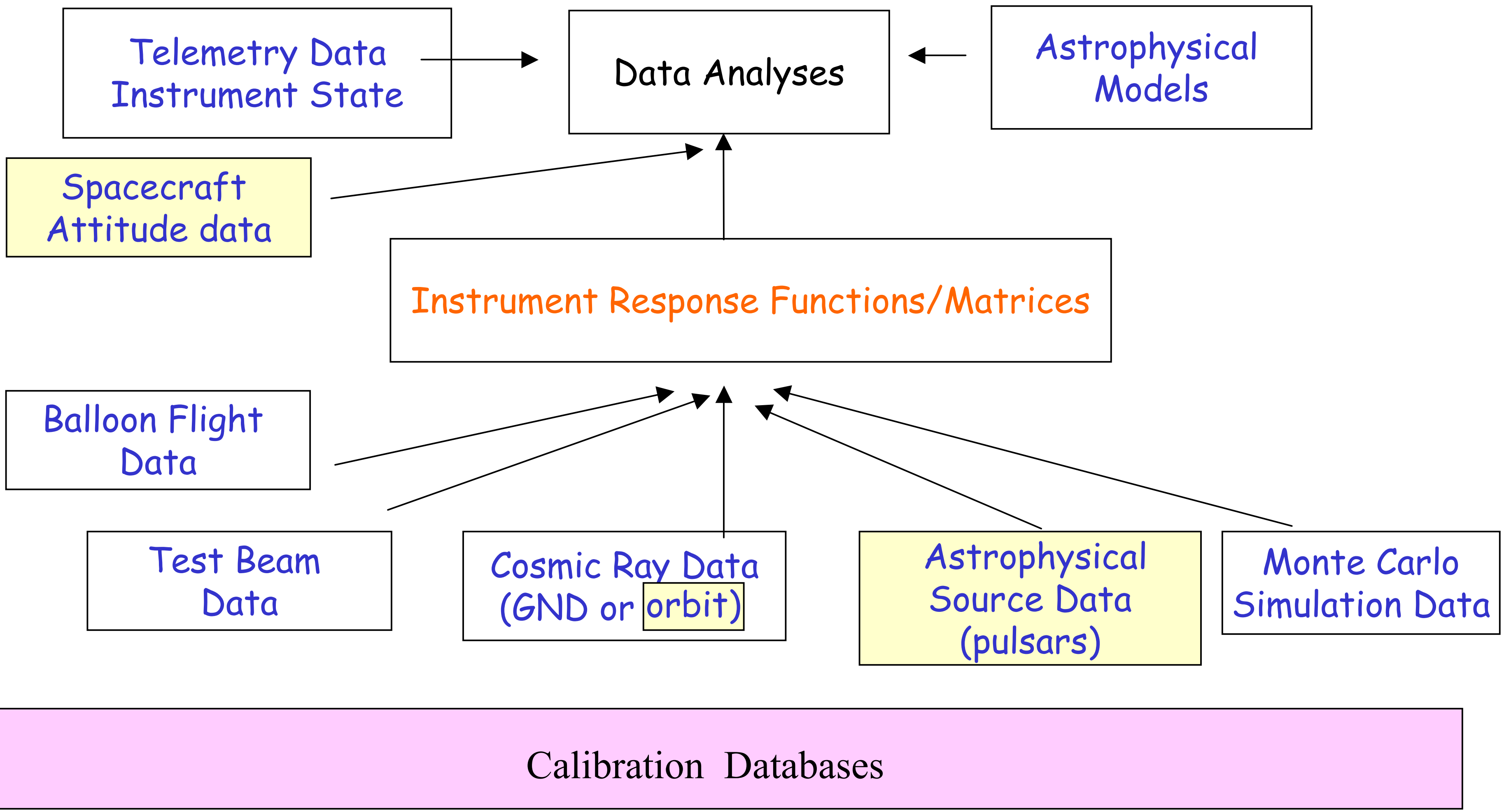
- Any existing set of calibration data can be retrieved at any time so that the impact of new calibration datasets on analysis results can be determined

- Flexible and platform independent (any user in any site can use it)

We follow the approach used by the CHANDRA mission (CALDB), which is also very similar to the one commonly used in HEP.

The calibration universe consists of data , software , database and documentation

Overview



- Time dependencies
- Environmental Dependencies
- Subdetector status

Instrument Response for γ rays

- LAT must be calibrated to give an accurate measurements of Particle Fluxes, Incident Direction, Energy Spectrum, Event Time.
- Describes the response of the instrument in the entire parameter space and depends (at least) on
  - Angles of incidence (inclination and azimuth)
  - Energy of incident particle (true or reconstructed ?)
  - Energy resolution
  - Impact point on the instrument
  - Location of photon conversion in the instrument
- The representation can be in the form or functions and/or matrices, whatever is best suited for the data analyses
- IRF's must be derived in the context of analysis cuts (tradeoffs between effective area, background rejection and PSF)
- Side notes:
  - The concept of exposure is not common in HEP.
  - The scanning mode and the complexity of the LAT instrument may complicate the definition of the response functions.

Calibration Data – Subdetectors

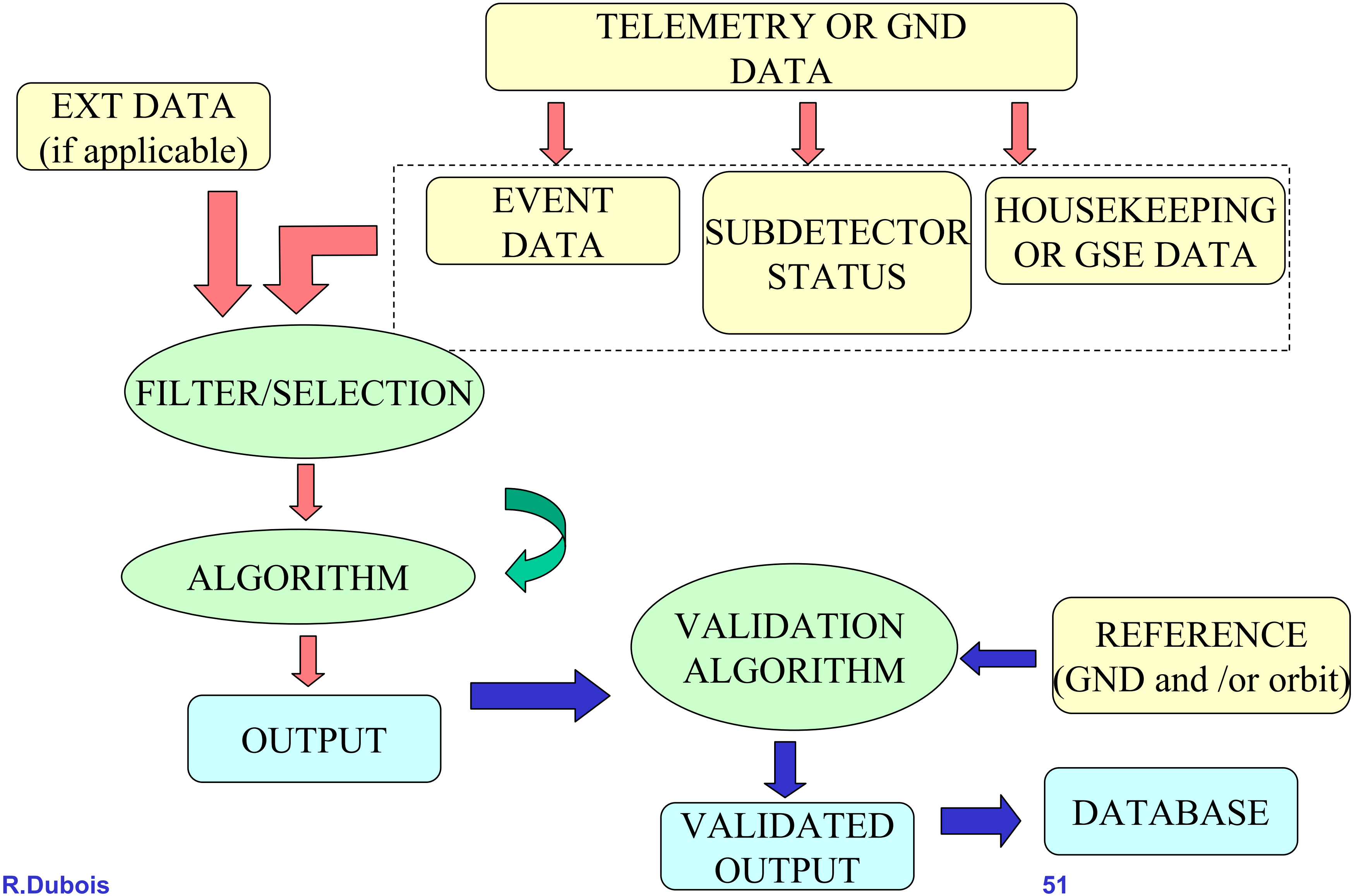
1. Which instruments need these data ? (BFEM, BTEM, LAT, etc..)
2. When during the lifetime of the project are these data needed ?
3. Which programs and or procedures (data clients) use these data ?
4. How precise (error estimates) data inputs/outputs need to be ?
5. What is the data format ?
6. How to access these data ?

Calibration Types – few examples

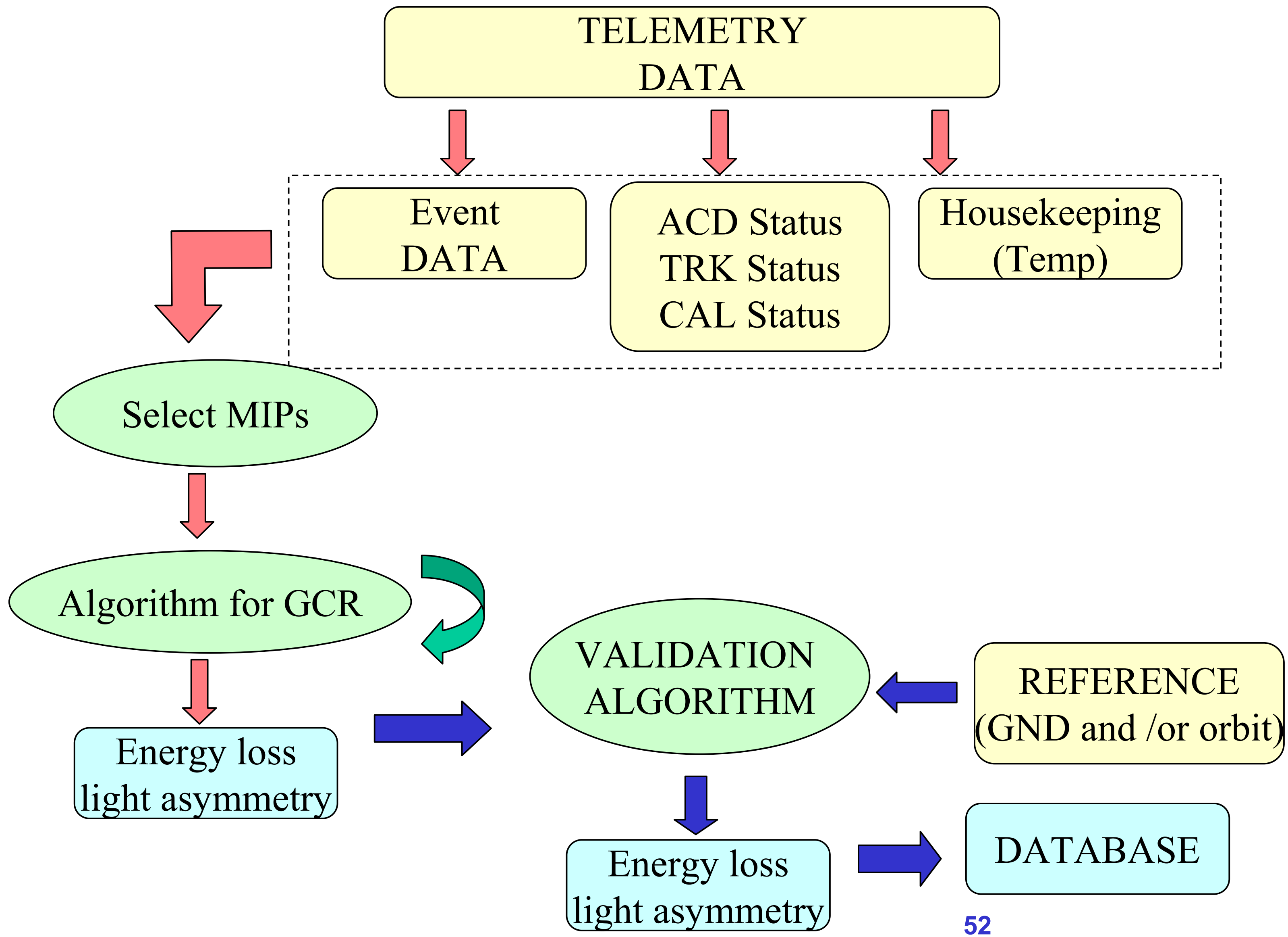
Stage 2 requires reconstruction software
Stage 1 does not require reconstruction software

TRACKER	CALORIMETER	ACD
Stage 2	**Stage 2**	**Stage 2**
Tracker alignment	Scintillation light Gain Linearities	Single hit efficiency Gain Linearities
Stage 1	**Stage 1**	**Stage 1**
Dead/noisy strips TOT versus charge	pedestals	pedestals

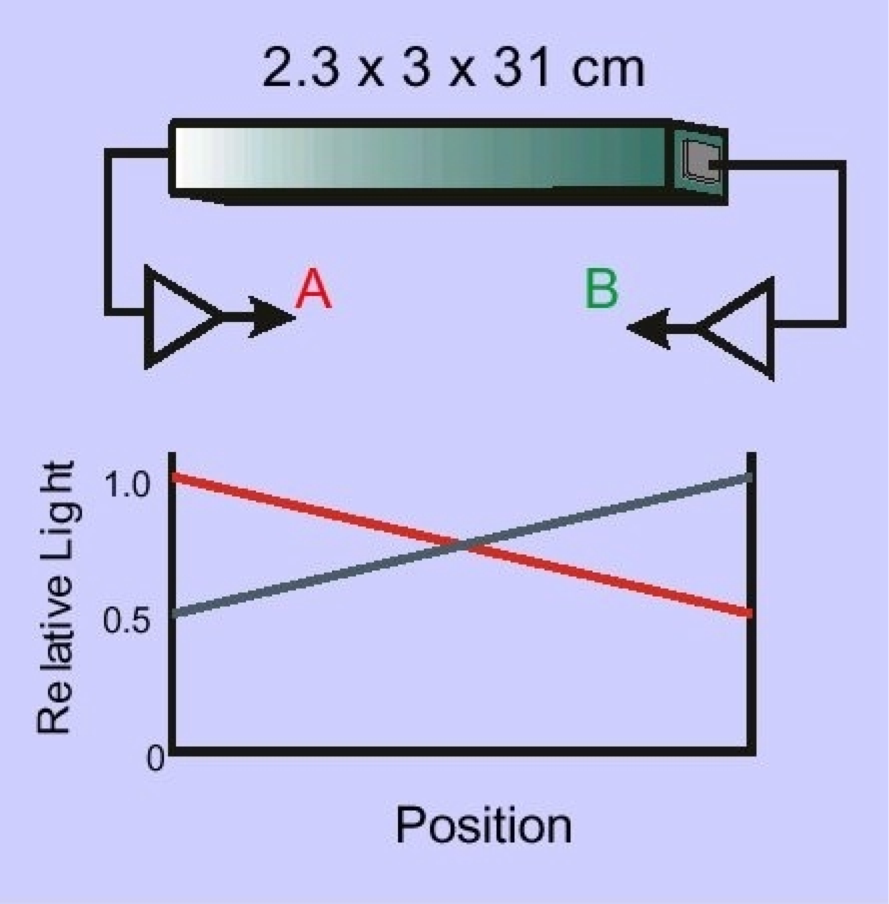
General Scheme



Light Asymmetry - Orbit



Light Asymmetry - Calorimeter



From Eric Grove

Extract Multi MIP events from telemetry

Select Galactic Cosmic Ray (GCR) candidates

Fit GCR tracks - See alignment

Extrapolate Tracks into CAL and ACD

Accept events with clean track through logs and reject glancing hits or edge events

Identify GCR charge

Identify charge changing interactions.

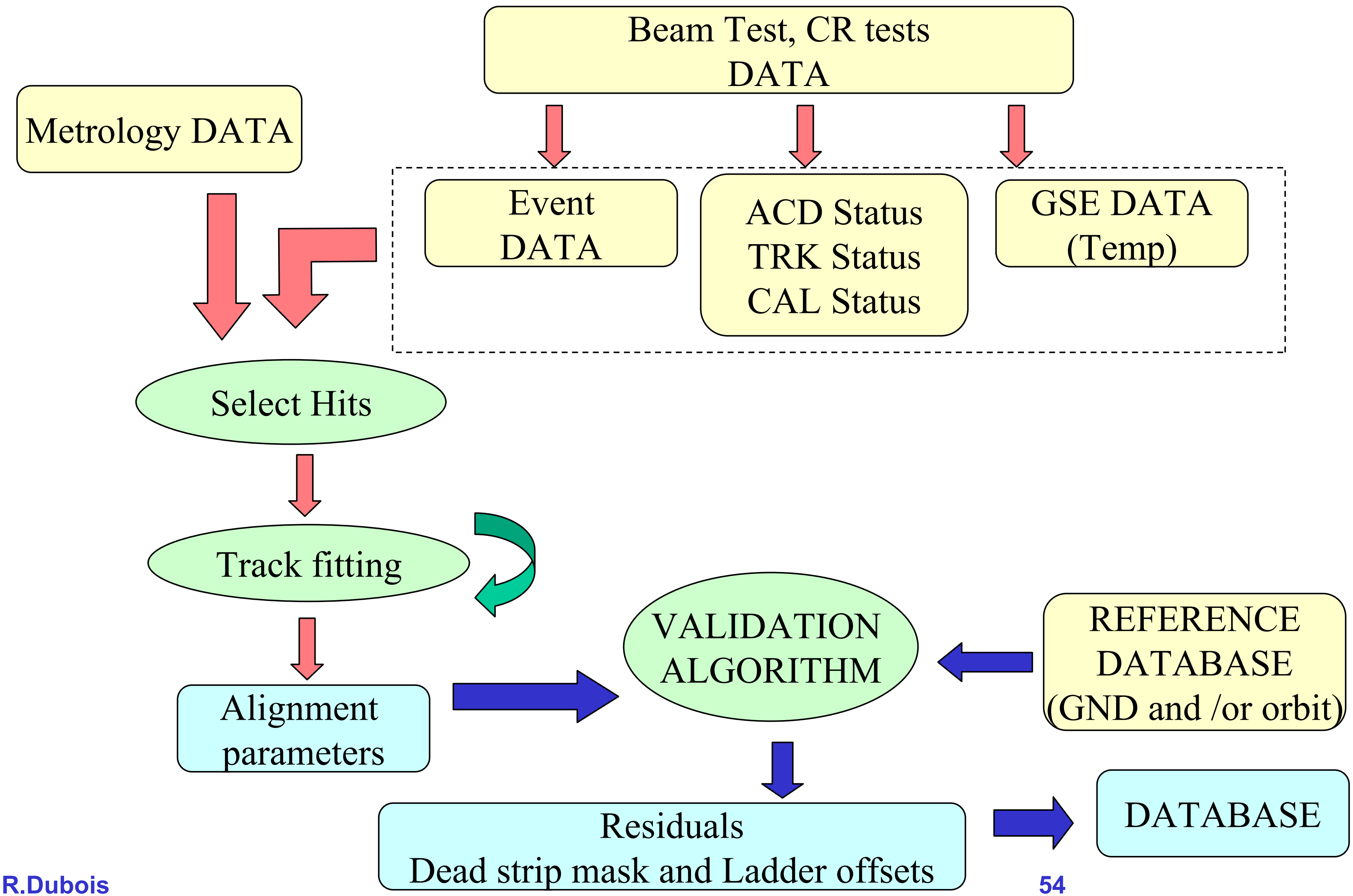
Identify mass changing interactions.

Fit dE/dx

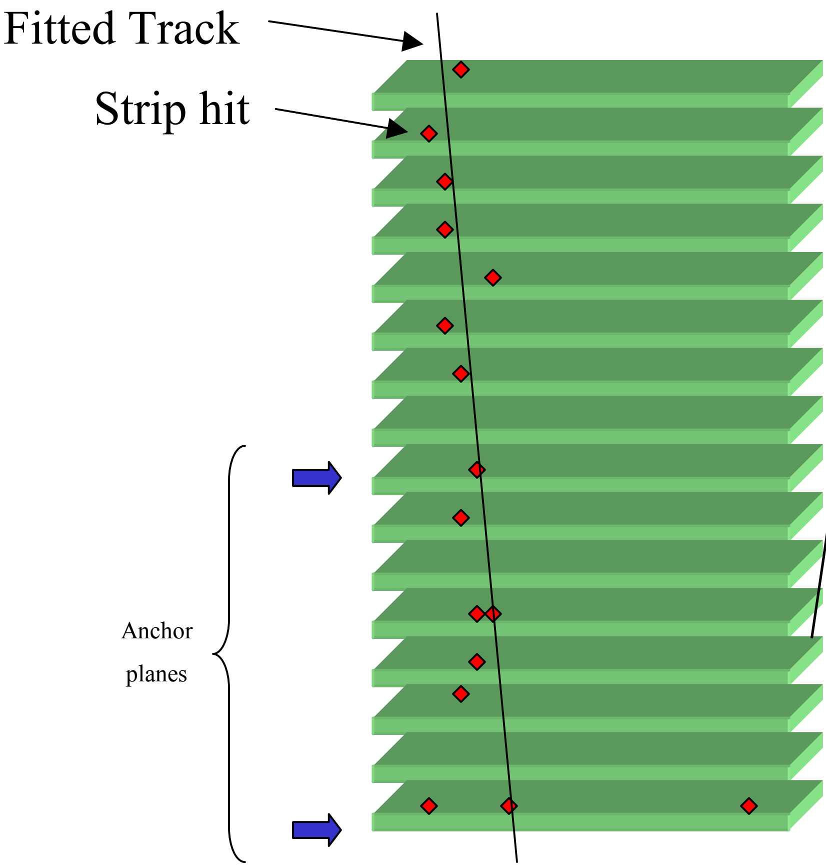
Iterate steps VI to IX until charge identification is stable

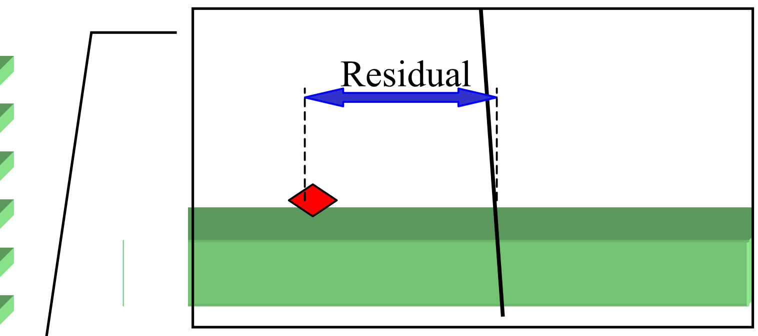
Accumulate energy loss and light asymmetry

Tracker - GND


Beam Test, CR tests
DATA
Metrology DATA
Event
DATA
ACD Status
TRK Status
CAL Status
GSE DATA
(Temp)
Select Hits
Track fitting
VALIDATION
ALGORITHM
REFERENCE
DATABASE
(GND and /or orbit)
Alignment
parameters
Residuals
Dead strip mask and Ladder offsets
DATABASE

Tracker Alignment - GND





From Jose Hernando and Ian Dobbs

I. **Read in silicon strip hits, dead channel mask and ladder offset values**

II. Create clusters from the hits

III. Calculate the position of the clusters correcting for any offsets.

IV. Find the best track (contains most hits). If multiple tracks have the same number of hits then the one with the lowest Chi-Square is selected.

V. Throw out any hits that have a large contribution to the Chi-Square value for the track.

VI. Use a least squares algorithm to fit the track.

VII. Use the hits in the best track on anchor planes to fit a track.

VIII. Calculate the residual of each cluster: residual = predicted – measured.

IX. **The sigma of residual distributions is one of the figures of merit for the alignment**

Alignment Input data

Hit Strip level (Raw Data)

Dead Strip mask

Ladder offsets

Frequently updated
- in orbit
- during LAT integration
- during TKR construction

Need a database to study time and environmental dependences using ground calibration data aiming at minimizing "debugging" time in flight

Alignment Program

When	Where	Module	# Towers	Focus	Does algorithm exist ?	How often?
1999/2000	Beam test	BTEM	1	Do we need metrology ? Are there effects on the PSF ?	Yes	Once
2001	Balloon Flight	BFEM	1	How many triggers are needed ?	Yes	Once
2002	CR tests	EM1,EM2, FMA,FMB	2	Develop automation and database concepts	Yes	TBD
2003-2004	Beam Tests CR tests	Calibration Unit	4	Inter tower alignment	NO	TBD
2005	CR tests	LAT	16	Tune algorithms and database for orbit	NO	TBD
2006-2016	Orbit	LAT	16	Operation mode	NO	TBD

Summary

This presentation

- We presented few preliminary ideas on GND and orbit calibrations.

- The LAT calibration concept is very broad it exploits an extensive GND based calibration program coupled with very sophisticated Monte Carlo simulations (typical in HEP experiments)

- Formally, calibration shows up in schedules in 2002/2003. In practice, we have already started with BTEM and BFEM exercises.

- Our early start allows us to can provide valuable input to the planning of the LAT Integration activities.

Summary – (ctnd)

Towards PDR

• We are currently working on the design by studying the CALDB approach from the CHANDRA mission. The following documents are now being studied:

1. HEASARC Calibration Dataset (CAL/GEN/91-001)-GEN
2. The Organization of OGIP CALDB (CAL/GEN/93-006)-GEN
3. Calibration Database User's Guide (CAL/GEN/91-002) -USR
4. BCF & CPF Calibration File Guidelines (CAL/GEN/92-003)-DEV
5. Calibration Index Files (CAL/GEN/91-008)-DEV
6. Mandatory FITS Keywords for Calibration Files (CAL/GEN/92-011)-DEV
7. The OGIP format for Effective area files (CAL/GEN/92-019)-DEV

LAT Data Processing Facility

- Automatically process Level 0 data through reconstruction (Level 1)
- Provide near real-time feedback to IOC
- Facilitate verification and generation of calibration constants
- Produce bulk Monte Carlo simulations
- Backup all data that passes through

LAT Data Processing Facility

Some Important Numbers

- Downlink rate – 300 Kb/sec ⟶ 3 GB/day ⟶1 TB/year
- Data plus generated products ~ 3 – 5 TB/year
- Over 5 years ~ 15-30 TB
- Average event rate in telemetry ~ 30 Hz (♑,background)
- Current reconstruction algorithm
  ~ 0.2 sec/event on a 400 MHz Pentium processor
- Assuming 4 GHz processors by launch ~ 0.02 sec/event
- ~ 5 processors more than adequate to keep up with incoming data as well as turning around a days downlink in ~ 4 hours
- Represents only about ~1 % of current capacity of SLAC Computing Center

LAT Data Processing Facility

- Even inflating estimates by considering re-processing data concurrently with prompt processing, a conservative estimate of resource requirements over the life of the mission is:

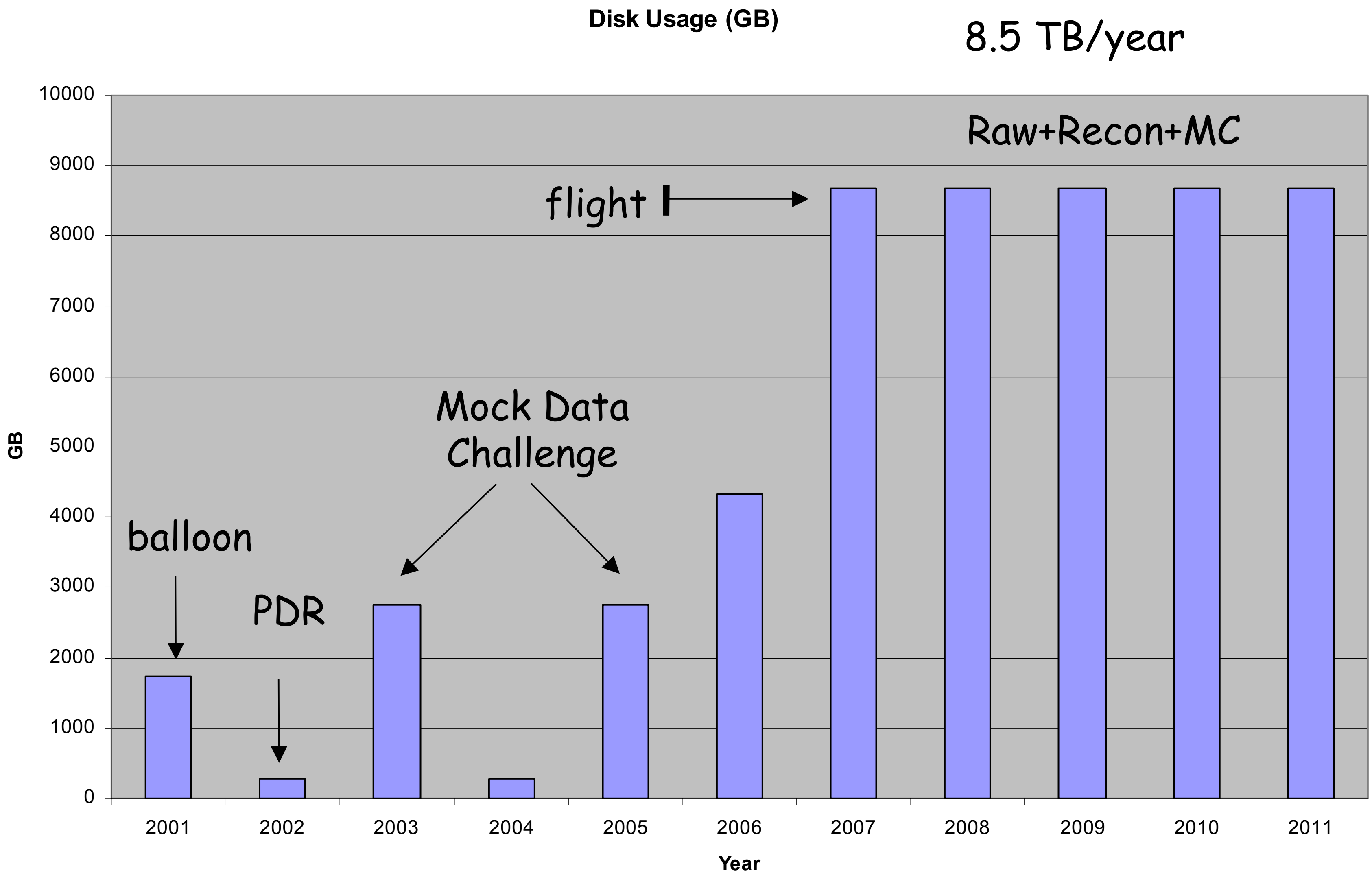
~ a few tens of processors

~ 50 TB of disk

- SLAC Computing Center is committed to providing these resources at no explicit expense to GLAST

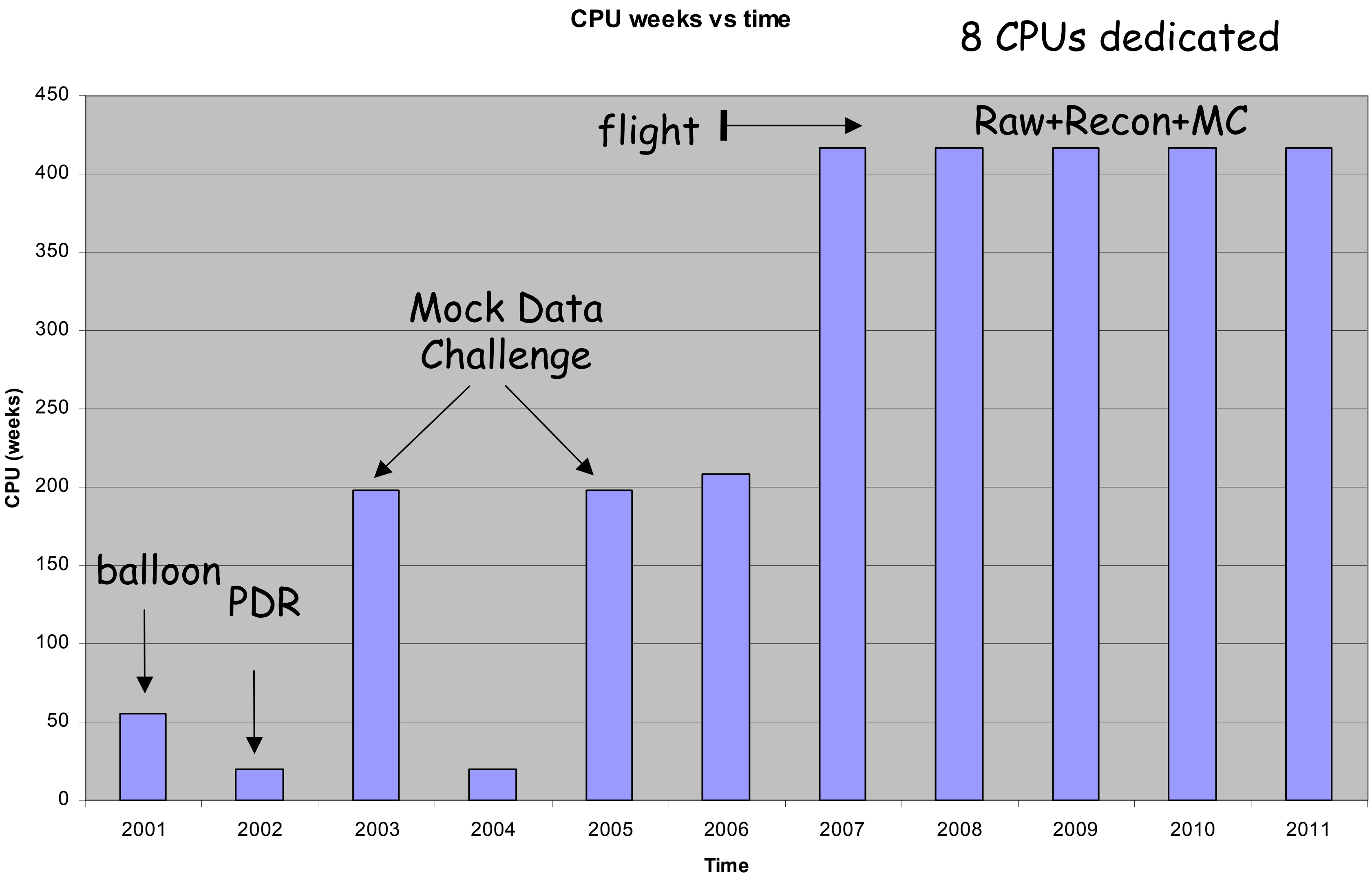
LAT Data Processing Facility

8.5 TB/year



(BABAR temp space is 10 TB)

LAT Data Processing Facility

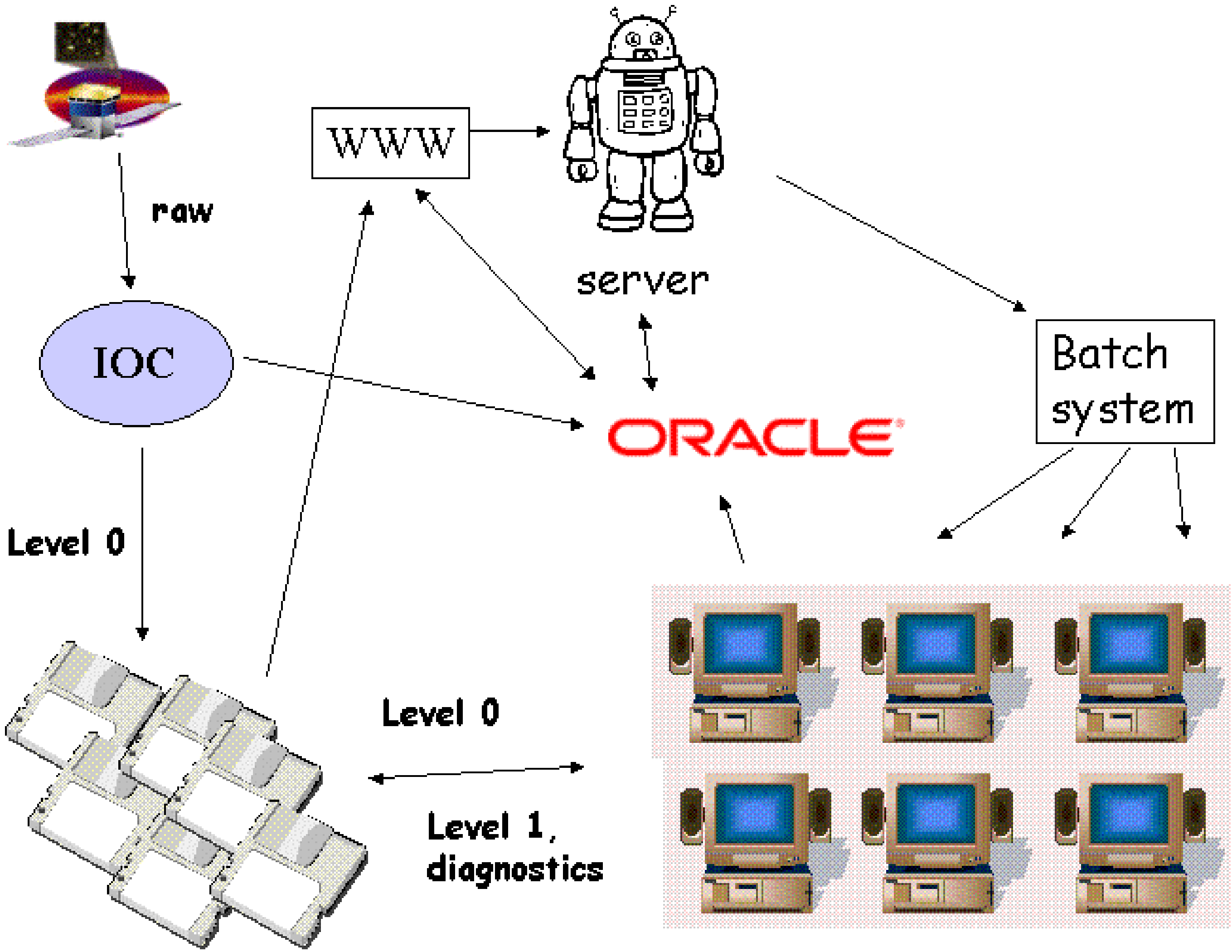


(SLAC Computing Services will have ~2000 CPUs for BABAR)

LAT Data Processing Facility

- Heart of data processing facility is a database to handle state of processing, as well as an automated server
  - File based relational database tracks state of dataset throughout lifetime in the system, from arrival at IOC or MC generation, through Level 1 output
  - Automated server will poll IOC generated database entries for new Level 0 datasets and take immediate action, as well as generate MC data, and log all actions to the database

LAT Data Processing Facility



LAT Data Processing Facility

Database

- Three categories of relational tables
  1. Dataset property description
  2. Processing status
  3. Dataset metadata

- Tables allow for grouping of similar datasets

- Current prototype based on experience with similar data pipeline used for SLD experiment at SLAC

LAT Data Processing Facility

Automated Server (Data Manager)

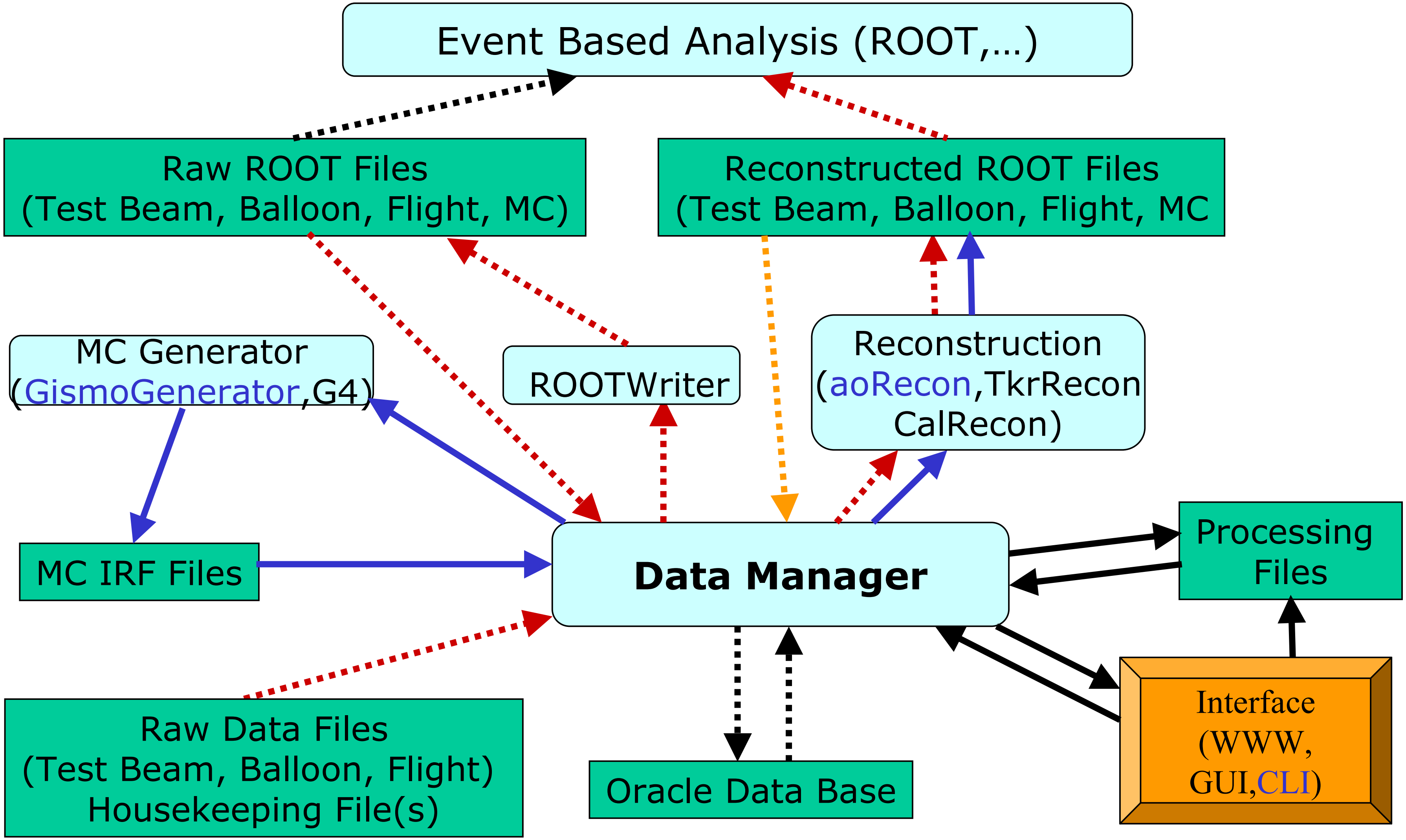
- Dispatches files in various states to appropriate processes
- Tracks state of processing for all datasets in system (completed, pending, failed, etc.) and logs this information to the database
- Provides near real-time feedback to the IOC by performing rapid, high level diagnostic analyses that integrate data from all subsystems

LAT Data Processing Facility

Automated Server (Data Manager) - cont.

- Design is simplified by having all datasets always on disk
- Utilizes load balancing LSF batch system at SLAC to dispatch processing jobs in parallel
- Provides a WWW interface for dispatching and tracking processes
- Current prototype at SLAC, written in perl, is used for processing MC runs

LAT Data Processing Facility



Science Analysis Software PDR Review:
Science Tools

S. W. Digel (GSFC/USRA)

Outline

- Design considerations
- Data flow
- Description of high-level data analysis
- High-level analysis environment
- Infrastructure of the analysis environment
  - Science databases
  - Science tools
  - Interstellar emission model
- Observation simulators

Design Considerations for Science Tools

- High-level analysis of LAT data is fundamentally model fitting
  - Driven by: limited # of γ's, modest angular resolution

⇒

- Interstellar emission model is essential
- Likelihood analysis is central

- Characterization of LAT is complicated
  - IRFs depend on energy, angles, conversion position, background cuts
  - Large FOV and scanning obs. also complicate analysis

⇒

- Acceptable degree of data binning to be studied
- Efficient, flexible exposure calculations are needed

γ-ray data access will be by region of the sky first, not time order
  - Level 1 data volume 1–2 Tb/yr

⇒

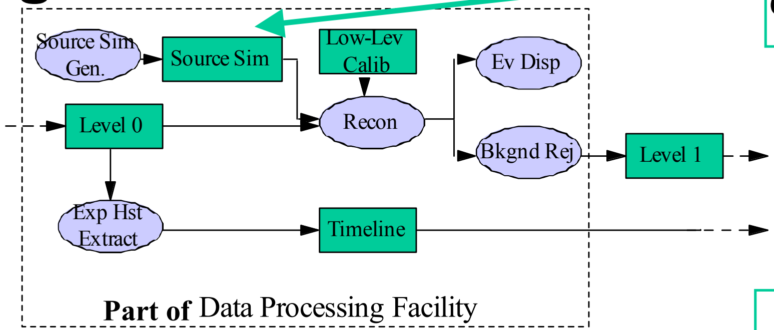
- Need efficient, flexible event selection
- Not likely to distribute all event data to LAT team/GIs

Data Flow for Science Analysis

- **Processing and databases**

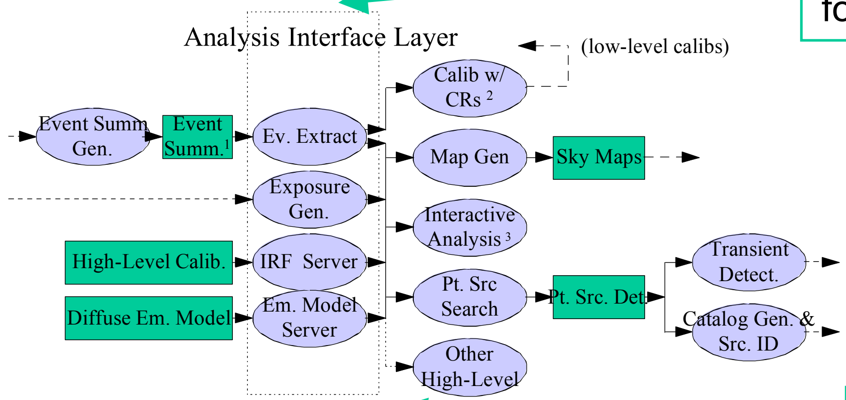
Source Sim. is a phony Level 0 database

Low-Level



Mirror sites (within LAT team and SSC) will reproduce from here forward

High-Level



[1] spatial access (celestial and instr. coords)
[2] evaluate CAL, ACD response, TKR alignment, dead strips,...
[3] spectroscopy, confused and extended sources,...

10's of Mbytes passed out of Analysis Interface Layer for a typical analysis: ~10^6 γ's + ~10^7 element exposure table

High-Level Analysis of LAT Data

- **Model fitting, with models of the general form**

$$I(x,y,E,t;\tilde{\alpha}) = Diffuse(x,y,E;\tilde{\alpha}_d) + \sum_i S_i(E,t;\tilde{\alpha}_i)\delta(x-x_i,y-y_i)$$

$$\tilde{\alpha} = \{\tilde{\alpha}_d, \tilde{\alpha}_i, i = 1 \ldots n\}$$

$$C(x,y,E,t;\tilde{\alpha},\tilde{\beta}) = \iint \left[\int_{t_1}^{t_2} I(x',y',E',t;\tilde{\alpha})dt \right] \varepsilon(x',y',E';\tilde{\beta}) PSF(x-x',y-y',E';\tilde{\beta}) W(E',E;\tilde{\beta}) d\Omega' dE'$$

I is the model intensity distribution, which includes diffuse emission and a number of point sources, and depends on the set of parameters α.
ε is the exposure, which has implicit zenith angle cuts used to select photons, and depends on the time range and instrumental parameters β *(e.g., inclination angle, plane of conversion)*.
C is the distribution of observed photons for the time interval $[t_1, t_2]$ and depends on α *and* β.

- **The likelihood function is used to compare observed photon distributions with a model (*C*), usually with binning of parameters, to define confidence ranges for parameters and compare models (e.g., iteratively search for point sources)**
- **So need to retrieve γ's, calculate exposure, and look up parameter dependence of instrument response functions.**

High-Level Analysis Environment

- Interactive (GUI) and batch (command line or script driven) interfaces are required depending on the analysis
  - Spectral studies, or studies of confused or extended sources will be interactive
  - Routine analysis by the LAT team, e.g., updating point source detections for newly arrived data to search for transients, will not be interactive
- Potential candidates for the analysis environment have been identified
  - Freely available and used in high-energy physics experiment or X-ray mission data analysis
  - To be discussed later as an Open Issue

Infrastructure of the Analysis Environment

- Part of implementation
  - Analysis interface layer (*)
  - Higher-level (†)
- Non-database infrastructure
  - Map generation
  - Image & plot display
  - All processing steps that produce images or tables will be able to export the results in TBD formats (on-the-fly data products)

	Database	Contents	Access Criteria	Used by
	Event	Full info. for each event, including reconstruction (Level 1 database)	Time or event number	Event Summary constructor, event display, low-level calib monitoring
*	**Event Summary**	Energy, direction (celestial and instr. coords), time, plane/tower/log layer of conversion, event id and bkgnd rej/quality flags	Energy, direction, time range, event flags, event ID	High-level map generation and analysis, CR event selection
*	**High-Level Calibration**	Instrument response functions as functions of energy, angles, plane, time, É	Energy, angles, time, ...	Exposure gen, high-level analysis
*	**Exposure History (Timeline)**	S/C position, orientation, LAT mode, and livetime for regular ~30s time intervals	Time range	Exposure gen.
	Source Sim.	Monte Carlo equivalent of Level 1 data, with truth info, and run/config. ID	As Event	Recon
†	**Point Source Catalog**	Summary of Point Source Detection, flux histories and candidate source IDs	Coordinates, spectral hardness, variability index, É	Catalog access interface
	Pulsar Ephemerides	(radio) Timing parameters for known pulsars, contemp. with GLAST mission	Pulsar name	Barycenter corrector
†	**GRB**	GRB profiles, source info.	GRB ID	Catalog access interface

High-Level Analysis Tasks (1)

- Based on scientific requirements for the LAT inferred from AO, SRD, and PDMP
- Definitions of modules motivated by discussions at LAT Software Working Group and Collaboration meetings, as well as EGRET experience within the team

Name	Function	Inputs	Outputs
Point-source detection	Analyzing a given region of the sky for point sources	Analysis interface layer	locations, fluxes, significances, spectrum or spectral hardness);
Point-source spectroscopy	Model fitting with flexible definition of spectral models; possibly developed as part of the general likelihood analysis capability described below (Extended sources and confused regions)	Analysis interface layer	Model coeffs and uncertainties
Extended sources and confused regions	ÔcustomÕ model fitting. Interactive analysis largely will be model fitting (parametric), allowing flexible specification of source Ğ multiple point sources, spectral models, arbitrary extended sources	Analysis interface layer	Model parameters, confidence ranges
Source variability	Flare detection (short term, for issuing alerts), pt. source vs. extended source determination (longer term, for quantifying variability)	Point source detection database	Flux histories, estimates of variability

High-Level Analysis Tasks (2)

- **GRB & pulsar analysis is like for point sources except for time profiles**
  - For burst profiles and phase folding & periodicity searches, want to select events by probability of association
- **Other tools (LOE, *not necessary for sci. req.*)**
  - Multi-γ events
  - Polarization
  - …

Source identification	Quantitatively defining probabilities of associations of LAT pt. srcs. with srcs. in other astronomical catalogs	Point source catalog	Point source catalog
GRB time profiles/Pulsar search event selection	Constructs time profiles for user-defined event selection criteria	Analysis interface layer (Event Summary)	Time profile histograms (perhaps normalized by IRFs, with periods outside FOV indicated), tables of events associated with a burst
Pulsar phase calculation	Assign pulsar phases to a set of photons based on the timing params for the pulsar, to allow phase-resolved analysis for most of the analysis tasks, like spectral meas., and phase binning - for histograms and maps.	Analysis interface, Pulsar Ephemerides	Phase assignments by event number
Pulsar periodicity searches	Searches for pulsations in data for a point source	Analysis interface*	Ideally, position, period, period derivative,...
High-resolution spectroscopy	for narrow-line emission at high energies	Analysis interface	Line energy, flux, or upper limits
Inflight calibration	monitoring effective area via fluxes of pulsars, monitoring PSF via phase-selected photon distributions around pulsars.	Analysis interface	Flux histories, PSF profile plots, tables

*Also may need a tool to display times when target was in FOV to select intervals with greatest continuous coverage.

Interstellar Emission Model

- Model to be developed in advance of launch
  - Won't need a 'good' one until real data arrive
  - Open issues from EGRET (e.g., GeV excess), and newly-available data relevant to gas distributions, ISRF, and cosmic-ray production and propagation mean room for improvement
- Will refine the model at least once after launch
  - Source-subtracted sky map
  - Limit the number of revisions for stability, reproduceability of results
- Model will be precomputed and tabulated for a grid of directions and energies
  - No particular advantage to on-the-fly calculations
  - Underlying components of the model (gas distributions, etc.) will be gridded anyway

Observation Simulators

- NOTE: These are not science tools per se
- *Low-level*
  - Full-blown MC through Glastsim
  - Useful for developing and testing the DPF and SAS systems
- *High-level* (or 'Fast MC')
  - Useful as a proposal preparation and observation planning tool throughout the mission
  - Depends on having an *orbit/operations simulator* with reasonable fidelity

Other Considerations

- **Import EGRET data (EGRET summary database level) into the SAS analysis environment**
  - Verify analysis by reproducing EGRET results
  - Preserve capability to analyze EGRET data
  - Exposure calculations likely to be the most tricky, but doable

Potential Backup Slides Follow -->

Exposure Calculation

- What is exposure?
  - Turns γ-ray counts into intensities
  - Depends on
    - direction on the sky (celestial coordinates)
    - inclination and azimuth (instrument coordinates)
    - range of time
    - energy
    - background rejection/PSF enhancement cuts applied (determines the Aeff that applies)
    - zenith angle cuts applied (to exclude Earth albedo gamma rays)
- Want fast calculation
  - Alternative is storing a wide variety of multidimensional arrays, or restricting flexibility of analyses

Exposure Calculation (2)

- What inputs are required for calculating exposure?
  - High-Level Calibration (Effective Area depends on energy, inclination, azimuth, and possibly time)
  - Timeline (Exposure History - position, orientation, operation mode, livetime, for regular ~30 s time steps)
  - Specification of zenith angle cuts (likely to depend on energy)
- Important point for calculating exposure:
  - High-Level calibration/zenith angle cuts can be applied at the end, after the distribution of livetime with inclination, azimuth, celestial coordinates, and zenith angle is found

Exposure Calculation (3)

- Proposed scheme for implementation of exposure calculations (C. Meetre)
  - Discretize directions on the sky *and* the viewing direction of the LAT in the *Timeline* file
    - Timeline file contains livetime, directions (i.e., grid point numbers) of viewing direction (z-axis), the x-axis, and zenith
    - Good equal area gridding schemes exist
  - $<\sim 10^4$ grid points should be plenty adequate, which implies one 10^8 element lookup table to provide the angles with respect to z-axis (inclination), x-axis (azimuth), and zenith (zenith angle); one-byte per element is entirely adequate
  - Promising and fast, even if need to interpolate to some other coordinate system at the end

Interstellar Emission Model

- A (so far) informal working group has organized to plan the interstellar emission model for LAT data analysis
  - Post-Gamma 2001, Baltimore
  - During ICRC 2001, Hamburg
- Practicality of 'factoring' the calculation
  - Defining components on a standard grid would be a starter
  - Also surveying the surveys (radio, mm, IR, …) available/anticipated for ISM & ISRF
  - Derivation of 3-dimensional distributions of ISM
  - Investigation of π^0 production function
- Model will undoubtedly be refined based on LAT observations, ideally during the first year
  - For reproduceability of analyses, want model stable, with infrequent revisions

LAT Software Quality Assurance

- Span of LAT ground software tests
  - Unit tests
  - System Tests
  - Instrument Performance Tests
  - End-to end tests – “Mock Data Challenges”

LAT Software Quality Assurance

- **Unit tests**
  - Test individual software packages via test designed by package maintainer
    - Tests have expected outcomes
    - Tests are run by release management software when maintainer "tags" package
    - Tests reside in conventional location (../test subdirectory) and have conventional names (test_PackageName)
    - Failures reported automatically to package maintainer (e.g. via email)
    - Examples:
      - Regression tests, histogram comparison tests

LAT Software Quality Assurance

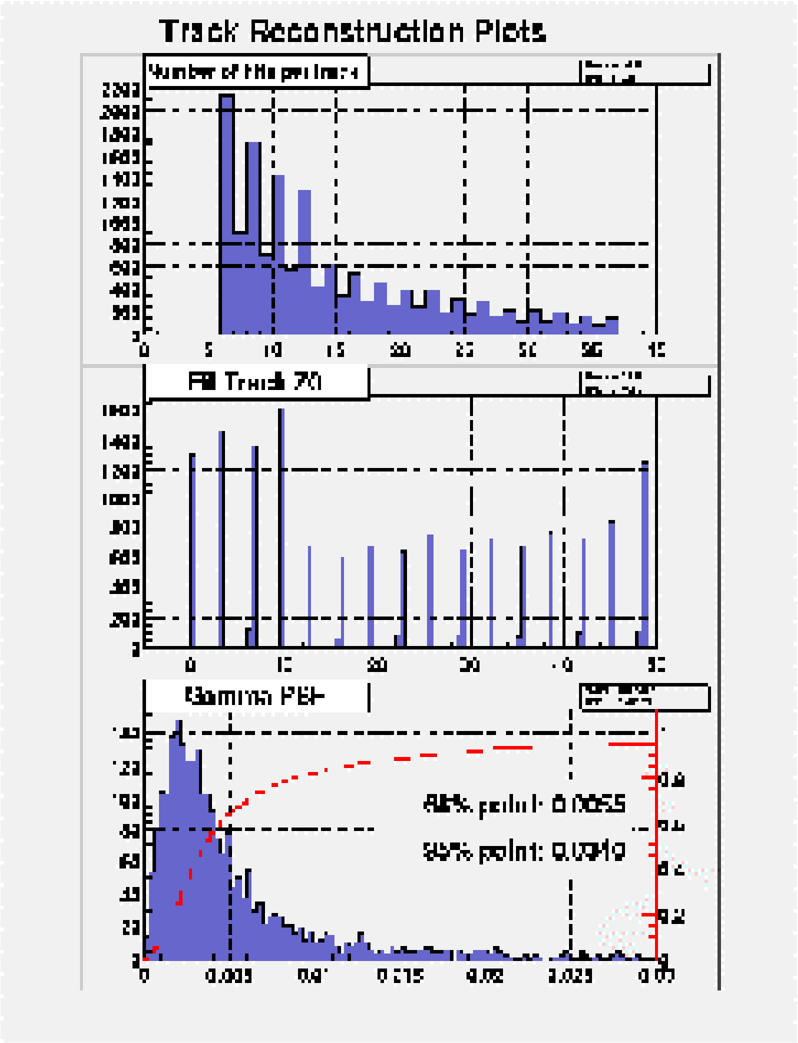
- System tests
  - Test application (checkout) packages
    - Tests are run by release management software when a release is declared
    - Tests generate diagnostics
    - Diagnostics tracked between releases and compared against standards
    - Failures reported automatically to designated list of management team members
    - Examples:
      - Regression tests, histogram comparison tests, performance tests

LAT Software Quality Assurance

- Instrument Performance tests
  - Test basic instrument performance parameters
    - Show that parameters meet LAT Performance Specification
    - Regular testing and tracking of results will allow for study of code evolution and possible large deviations from understood performance
    - In particular, examine (after background rejection and resolution cuts)
      - TKR front and back section PSF, as a function of energy and angle
      - Energy resolution on-axis and at > 60 deg. incidence, as a function of energy
      - Effective area as a function of energy and angle (and hence FOV)
      - Residual background as fraction of accepted high-lattitude diffuse flux as a function of energy

LAT Software Quality Assurance

Sample TKR reconstruction plots showing reconstructed track multiplicity, origin point of tracks, and PSF (68 % and 95 % containment) for gammas

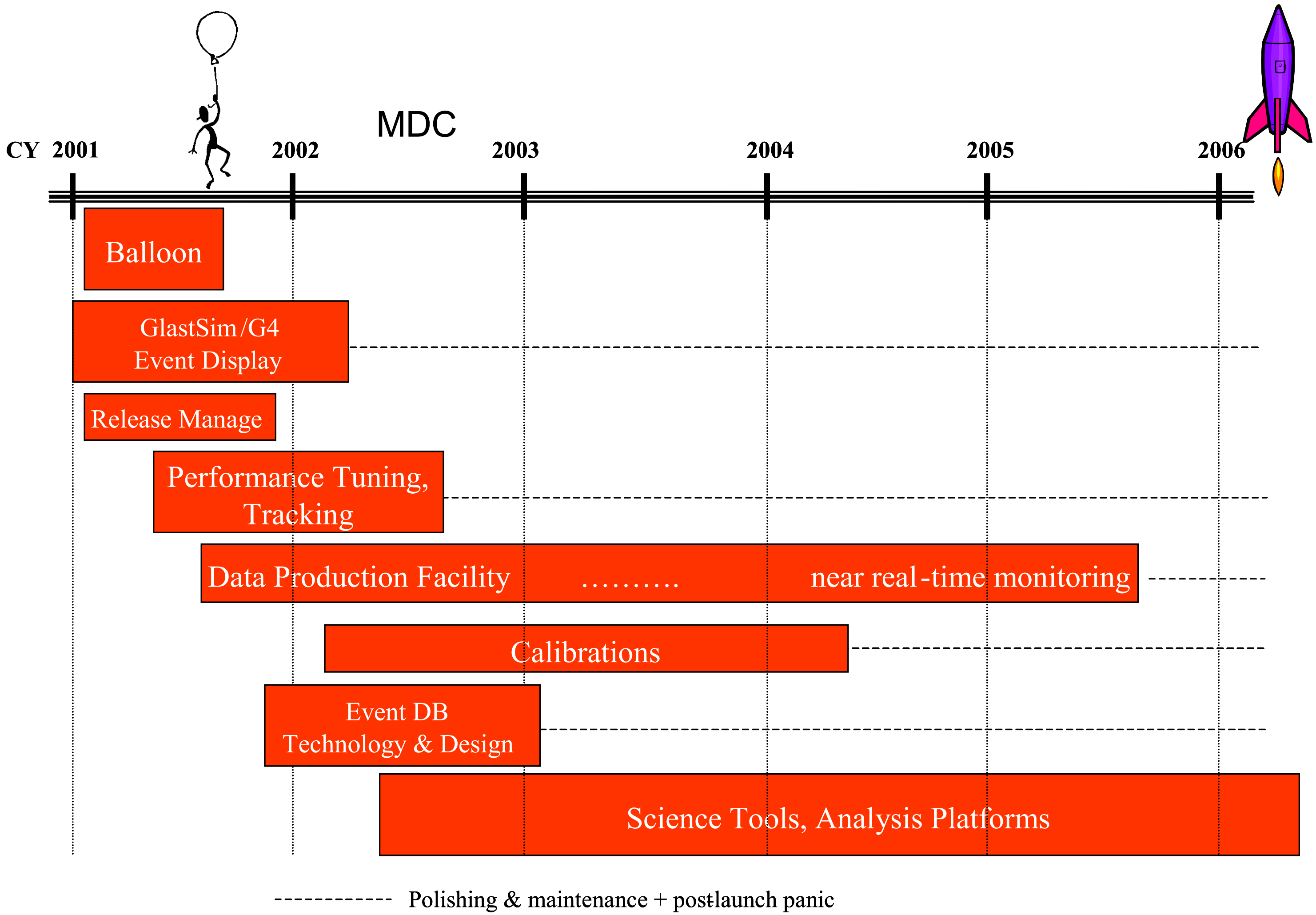


LAT Software Quality Assurance

- End to end tests – “Mock Data Challenges”
  - Large scale test of entire LAT ground software system
    - Bulk processing of simulated source raw data through Level 1 processing followed by Level 2 analysis
    - “Single blind” – those doing analysis don’t know the underlying physics – their job is to discover it
    - Large scale effort involving large fraction of collaboration and certainly Science Working Groups
    - Anticipate 2-3 Mock Data Challenges prior to launch
      - In mid 2003
      - Towards the end of 2005

Schedule, Manpower & Open Issues

Schedule


MDC
CY 2001
2002
2003
2004
2005
2006
Balloon
GlastSim/G4
Event Display
Release Manage
Performance Tuning,
Tracking
Data Production Facility
……….
near real-time monitoring
Calibrations
Event DB
Technology & Design
Science Tools, Analysis Platforms
Polishing & maintenance + postlaunch panic

Milestones

Milestone	Date
Science Analysis Software (SAS) Requirements Review	04/20/01
Start PDR Instrument Performance/Backgrounds Evaluation	05/01/01
SAS PDR	08/17/01
LAT Instrument PDR	10/29/01
Release Management & Verification in place	12/01/01
First Prototype of Data Processing Facility	2/1/02
Simulation/Reconstruction 1st iteration complete	5/1/02
SAS CDR	9/4/02
Photon Database technology in place	12/1/02
LAT Instrument CDR	12/1/02
Demonstrate First Science Tools using Database	3/1/03
First calibration algorithms in place	6/1/03
Databases shared with SSC	10/1/03
Production version of Data Processing Facility	6/1/04
All required Science Tools in place	2/1/05
End to End pre-launch test completed	12/1/05

Manpower Assumptions

Sources, Simulation & Recon

- 1 FTE GEANT4 during the development phase and 1/2 FTE ongoing (Italy)
- 1/2 FTE sources (Japan)
- 2 FTE CAL - simulation and reconstruction (NRL, France)
- 2 FTE TKR - simulation and reconstruction (SLAC + Italy)
- ½ FTE for ACD - simulation and reconstruction (GSFC)
- 1 FTE combined for Trigger, Background Rejection studies (there may be odd scientists contributing as well)

Analysis Tools & Infrastructure

- 2 FTE for tools development (GSFC)
- 1 FTE package & user support (SLAC)
- 1 FTE code release & verification (SLAC)
- 1 FTE event display – 6 mo (Italy)

Engineering Models (BTEM, BFEM, 4 Module test 2003? …)

- 1 FTE extra for duration of each test

More Manpower Assumptions

Science Software

- 3 FTEs to support 7 scientist programmers (estimated 40 MY work). Starting in mid FY '02.

Data Processing Facility

- 1 FTE automated server (~1 year) – main activity in FY02, followed by a burst a year before launch
- 1 FTE instrument diagnostics – build up 1.5 yr before launch
- 1/2 FTE support of DPF pre-launch

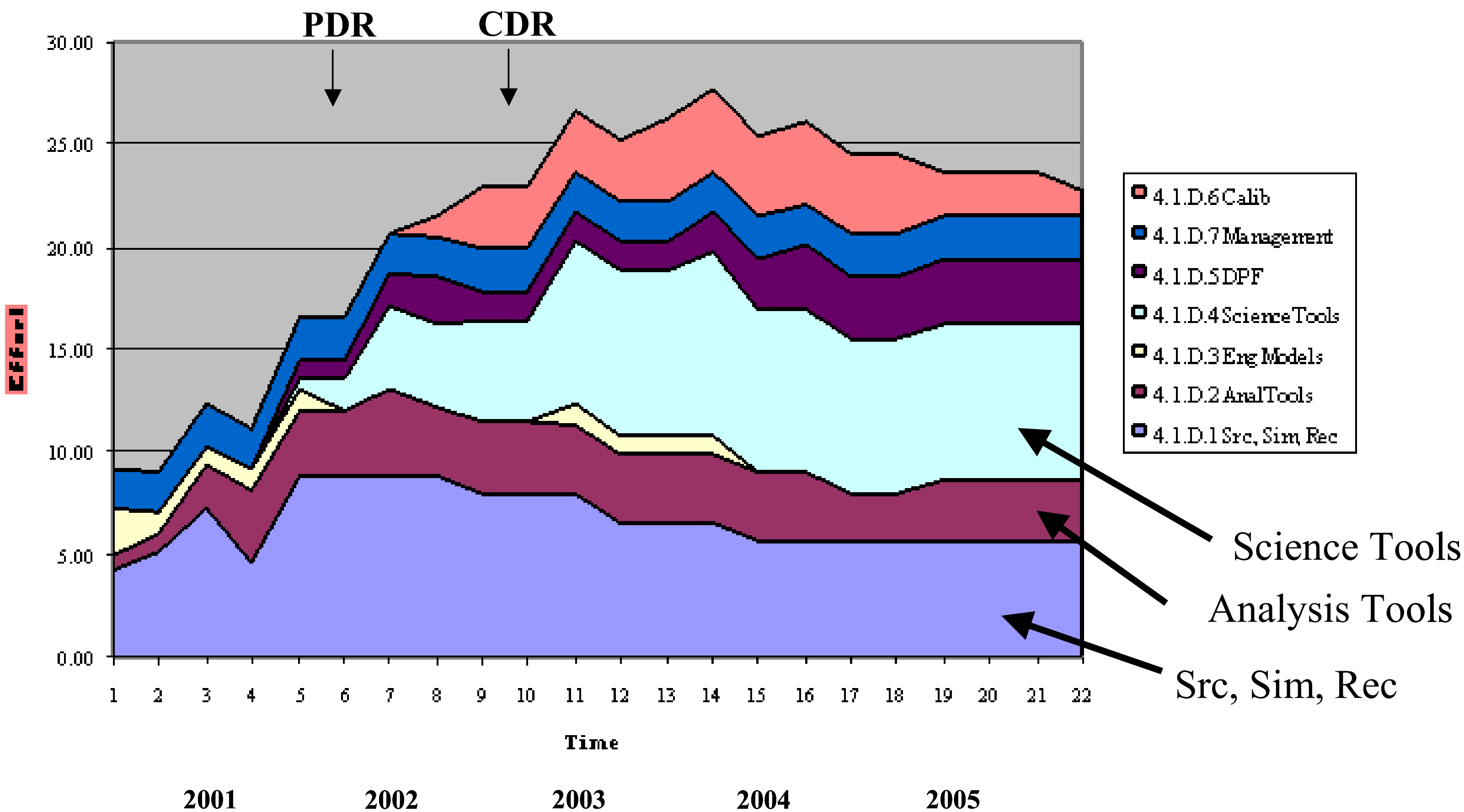
Calibration (starts 3/2002)

- 1 FTE for machinery for calibrations
- 2.5 FTE for subsystem algorithm development (perhaps more – from NRL, France & Italy)

Management

- 1 FTE code architect
- 1 FTE manager

Projected Manpower



Open Issues

- Event/photon database
  - Want to cater to most typical queries and give good response time
  - Query by (direction,time) – scanning mode not amenable to storage in files: tesselate sky, ?
  - Shared database can allow mirror sites to import data convert to other formats as they see fit
  - Trade study in 2002
- Event Analysis Platforms
  - Currently supporting 2: Root & IDL
    - Root is free HEP package targeted at event type data. Includes I/O as well as analysis (gui, histogramming, fitting, etc) features. Apparently the next big package in HEP
    - IDL is the astronomical standard. Commercial and expensive. Well suited to image data. Astronomers would die before giving it up ;-)
    - Downside is effort in maintaining 2 systems; some divisiveness.
    - Probably have no choice but to maintain two.

Open Issues (cont'd)

- Calib & I&T Support
  - Will be focusing on tracking instrument components from construction through to flight
  - Need definitions of database and analysis tools
  - Need resolution of who will actually do calibrations
  - Still under discussion in the collaboration
- User Support
  - Difficult to maintain documentation & answer user queries
  - Can be a serious drain on developers, especially during phases of active change
  - When things settle down a bit, we should prepare a "User Workbook", examples of which created by SLD and BABAR.
    - Step by step tutorials to walk users through the basics

Science Tools Issues

- **Instrument Response Functions**
  - Potentially functions of many variables:
    - A_{eff}, energy resolution, and point-spread function (PSF) could be described as functions of energy, azimuth, inclination, plane of conversion in the TKR or layer of conversion in the CAL, tower of conversion, etc
  - Different cuts will require different IRFs!
  - How much MC is required to (help) determine IRFs
- **Likelihood Options**
  - Sparse data with low resolution implies model fitting and max likelihood
    - Binned or unbinned??
    - Unbinned will be computationally intensive

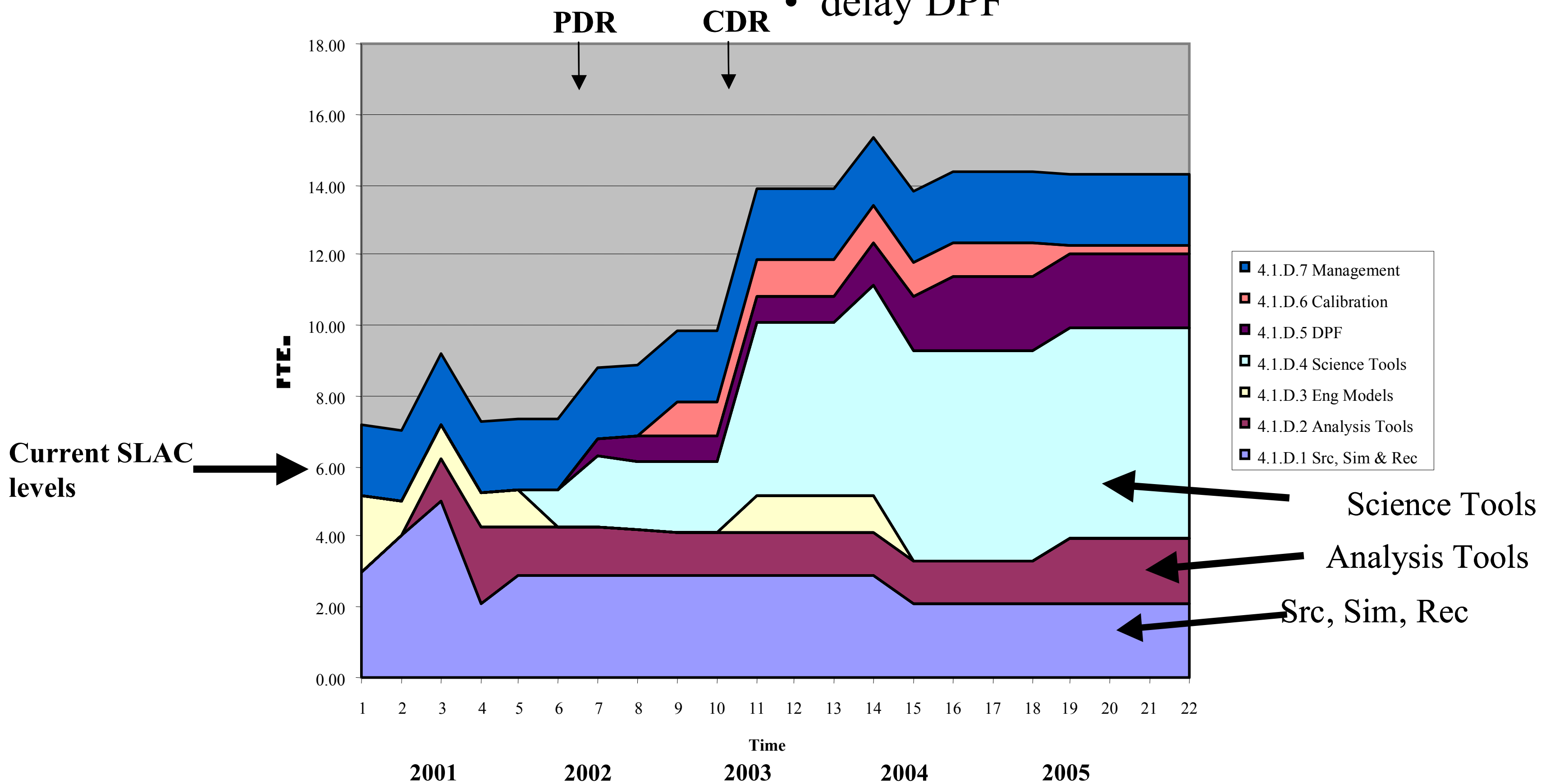
Science Tools Issues (cont'd)

- High Level Analysis Environment
  - Will need a shell &/or GUI to work within
    - Root, CIAO, ??
  - How to develop a uniform interface between the tools and the database
    - All the tools should look as similar as possible to the user
  - Will need to optimize results speed & data volume between computations on servers and delivery of data to user's machines
- Collaboration with the SSC
  - It will be the public side of GLAST
  - It will have significant resources to collaborate on the science tools: shared databases, algorithms & interfaces
  - Still in the planning & beginning staffing stage

Manpower

Basic problem will be 8 FTEs difference

- dun the collaboration for even more manpower
- stretch science analysis tools development
- collaborate with SSC
- delay DPF



Backup Slides on WBS

WBS, Schedule & Manpower – 4.1.D

4.1.D.1 SOURCES, SIMULATION & EVENT RECONSTRUCTION

- 4.1.D.1.1 SOURCES
- 4.1.D.1.2 PROTOTYPE INITIAL FRAMEWORK
- 4.1.D.1.3 GISMO SIMULATION
  - 4.1.D.1.3.1 EXISTING SIMULATION UPGRADE
  - 4.1.D.1.3.2 NEW GEOMETRY & HITS SCHEMES
  - 4.1.D.1.3.3 ONGOING SUPPORT
- 4.1.D.1.4 GEANT4 SIMULATION
  - 4.1.D.1.4.1 EXTERNAL PACKAGE REQUIREMENTS
  - 4.1.D.1.4.2 DETMODEL GEOMETRY CONVERTER
  - 4.1.D.1.4.3 GEANT4 PROTOTYPE
  - 4.1.D.1.4.4 GEANT 4 VALIDATION
  - 4.1.D.1.4.5 ONGOING SUPPORT
- 4.1.D.1.5 ACD SIMULATION
  - 4.1.D.1.5.1 EXISTING DIGITIZATION UPDATE
  - 4.1.D.1.5.2 UPGRADE FOR NEW HITS SCHEME
  - 4.1.D.1.5.3 ONGOING SUPPORT

4.1.D.1.6 CAL GEOMETRY, SIMULATION, & RECONSTRUCTION

- 4.1.D.1.6.1 GEOMETRY
- 4.1.D.1.6.2 SIMULATION
- 4.1.D.1.6.2.1 INITIAL VERSION OF SIMULATION
- 4.1.D.1.6.2.2 SIMULATION IMPROVEMENTS
- 4.1.D.1.6.3 RECONSTRUCTION
- 4.1.D.1.6.3.1 INITIAL VERSION OF RECONSTRUCTION
- 4.1.D.1.6.3.2 RECONSTRUCTION IMPROVEMENTS: SHOWER LEAKAGE CAL
- 4.1.D.1.6.3.3 ITERATIVE RECONSTRUCTION W/ TRACKER
- 4.1.D.1.6.3.4 FAILURE MODES / PERFORMANCE STATE

4.1.D.1.7 TRACKER GEOMETRY, SIMULATION & RECONSTRUCTION

- 4.1.D.1.7.1 SIMULATION IMPROVEMENTS
- 4.1.D.1.7.2 DIGITIZATION IMPROVEMENTS
- 4.1.D.1.7.3 INITIAL TRACKER RECONSTRUCTION
- 4.1.D.1.7.4 TRACKER RECONSTRUCTION REDESIGN: PATREC & FIT
- 4.1.D.1.8 TRIGGER SIMULATION
- 4.1.D.1.9 BACKGROUND REJECTION

4.1.D.1.A MAJOR RELEASES OF SIM & RECONSTRUCTION

WBS continued

4.1.D.3 ENGINEERING MODELS

4.1.D.3.1 TEST BEAM 99 SUPPORT

4.1.D.3.2 BALLOON FLIGHT SUPPORT

4.1.D.3.3 4-MODULE TEST SUPPORT

4.1.D.4 SCIENCE SOFTWARE

4.1.D.4.1 UTILITIES

4.1.D.4.2 ANALYSIS SOFTWARE

4.1.D.4.3 ANALYSIS DATABASES

4.1.D.5 DATA PROCESSING FACILITY

4.1.D.5.1 PROTOTYPE DATA MANAGER

4.1.D.5.2 AUTOMATED SERVER

4.1.D.5.3 INSTRUMENT DIAGNOSTICS

4.1.D.6 CALIBRATION

4.1.D.6.1 TOOLS FOR ACCESSING CONSTANTS

4.1.D.6.2 ACD CALIBRATION

4.1.D.6.3 CALORIMETER CALIBRATION

4.1.D.6.4 TRACKER CALIBRATION

4.1.D.6.5 HIGH LEVEL CALIBRATIONS

4.1.D.7 MANAGEMENT

4.1.D.7.1 SCIENCE ANALYSIS SOFTWARE MANAGEMENT

4.1.D.7.2 SCIENCE ANALYSIS SOFTWARE REQUIREMENTS

4.1.D.7.2.1 LEVEL 3 REQUIREMENTS

4.1.D.7.2.2 LEVEL 4 REQUIREMENTS

4.1.D.7.2.3 PDR SUPPORT

4.1.D.7.2.4 MOCK DATA CHALLENGE 1

Requirements - WBS Matrix - I

Requirement #	Description	WBS #	Description
5.1	prompt processing	4.1.D.5	Data Processing Facility
	near-real time monitoring	4.1.D.5	Data Processing Facility
	monitoring & updating calibs	4.1.D.5	Data Processing Facility
		4.1.D.6	Calibrations
	maintain state & performance tracking	4.1.D.5	Data Processing Facility
	high level science products	4.1.D.4	Science Tools
	reprocessing data	4.1.D.5	Data Processing Facility

Requirements - WBS Matrix - II

5.1	access to events	4.1.D.5	Data Processing Facility
		4.1.D.2	Analysis Tools
	perform bulk MC simulations	4.1.D.5	Data Processing Facility
	interface with mirror sites	4.1.D.5	Data Processing Facility
		4.1.D.2	Analysis Tools
	interface with SSC	4.1.D.5	Data Processing Facility
		4.1.D.2	Analysis Tools
	support engineering models	4.1.D.3	Eng Model support

Requirements - WBS Matrix - III

5.2	Code Development	4.1.D.2	Analysis Tools
5.3	Instrument Response Simulations	4.1.D.1	Sources, Simulation and Event Reconstruction
5.4	Event Reconstruction	4.1.D.1	Sources, Simulation and Event Reconstruction
5.5	Environment Logging	4.1.D.5	Data Processing Facility
5.6	Calibrations	4.1.D.6	Calibrations
		4.1.D.5	Data Processing Facility
5.7	Level 1 Processing	4.1.D.5	Data Processing Facility

Requirements - WBS Matrix - IV

5.8	Creation of High-Level Science Tools	4.1.D.4	Science Software
5.9	Analysis Platform	4.1.D.2	Analysis Tools
5.10	Longevity	4.1.D.5	Data Processing Facility
		4.1.D.2	Analysis Tools
6.1	Data Formats	4.1.D.5	Data Processing Facility
6.2	LAT Data & Alg Export	4.1.D.5	Data Processing Facility
		4.1.D.2	Analysis Tools
6.3	SSC interface	4.1.D.5	Data Processing Facility