GLAST Large Area Telescope:

Science Analysis Systems

Richard Dubois
Stanford Linear Accelerator Center
richard@slac.stanford.edu

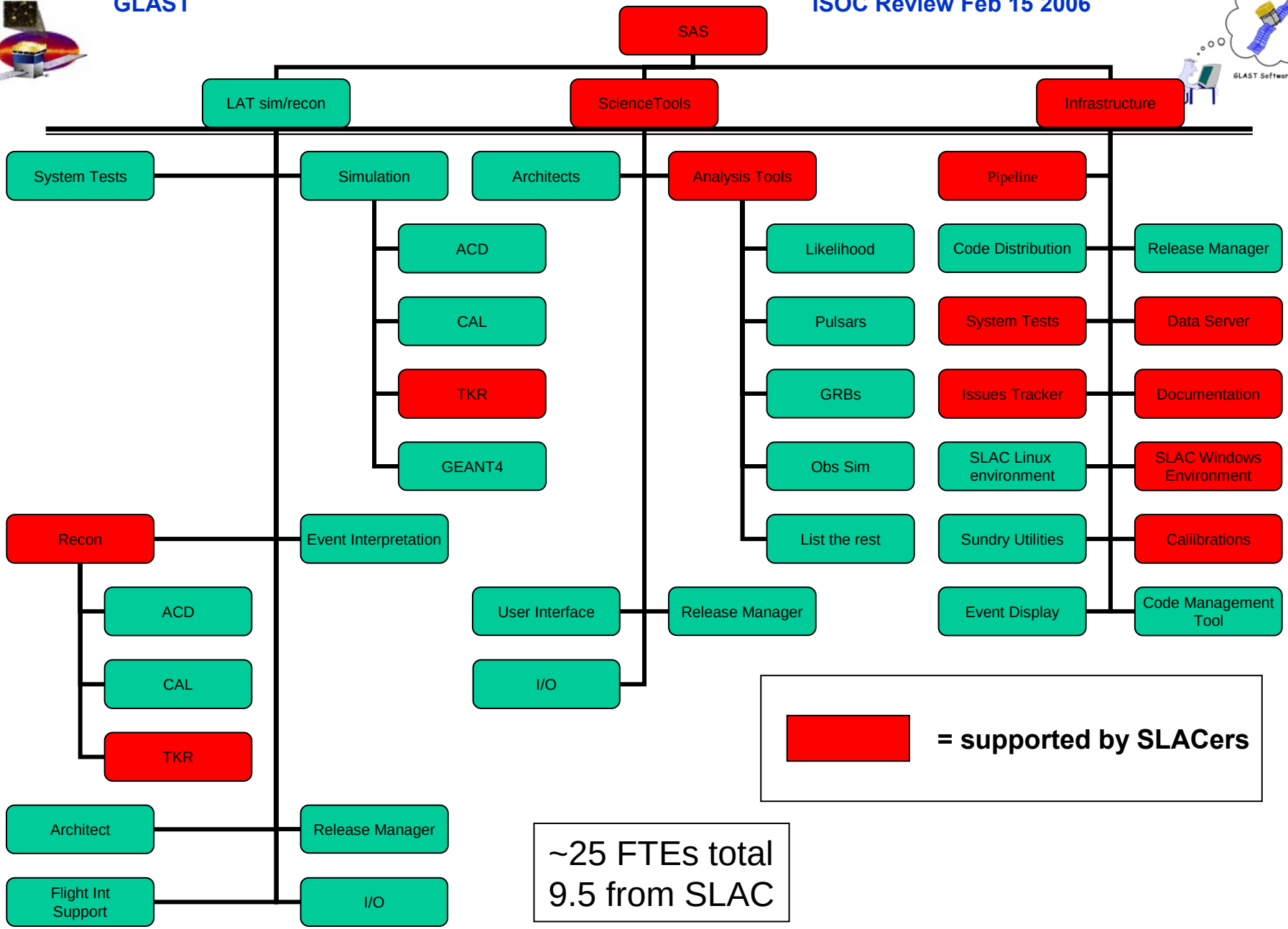
http://www-glast.stanford.edu/software

Outline

- SAS Purview
- Software Development Concept & Tools
- Reconstructing events
- High Level Science Tools
- I&T Support for Integration
- DataChallenges
- NRL & Beamtest 2006 Support
- Computing Resource Projections
- Building the ISOC Ground Operations Tools
- Major Milestones and Manpower Projections

SAS Purview

- Moving towards providing all software development for the LAT ground work
- Supports ISOC and LAT collaboration
- Support software development environment and tools
- Instrument data processing: reconstruction, calibration and simulation
- High level science tools & Quicklook
- Automated processing pipeline machinery
- Acquire and coordinate most LAT compute resources at SLAC: bulk CPU and disk usage
- Database and web development
  - System tests, Data Monitoring
  - Tools used in ISOC day-to-day handling of downlinks
- Integrated with the LAT Collaboration



~25 FTEs total
9.5 from SLAC

C++ Software Development Approach

- Enable distributed development via cvs repository @ SLAC
  - viewCvs for easy web browsing
- Extensive use of electronic communications
  - Web conferencing (VRVS), Instant Messaging (icq)
- CMT tool permits equal development on Windows and Linux
  - 'requirements' file generates MS Project or gnu Makefiles from single source
  - Superior development environment on Windows; compute cycles on linux
- documentation and coding reviews enforce coding rules
- "Continuous integration"
  - Eliminate surprises for incoming code releases
  - Build code when packages are tagged; alert owners to failures in build or running of unit tests. Results tracked in database.
  - Developing comprehensive system tests in multiple source configurations. Track results in database; web viewable.

Documentation: User Workbook

Workbook for Offline Users

HOME

Site Map

GLAST Links	SAS Software	Get Connected	Installing GLAST S/W		GLEAM	Running GLAST Applications				Advanced
			End-user	Developer		FRED	MRvcmt	ROOT	Science Tools	

ROOT: 1: Overview & Setup 2: Outputs 3: View Ntuple 4: RootTreeAnalysis 5: Accessing Data

ROOT 3 View Ntuple: View Summary Ntuple | Create Histogram | Remove this navbar

Print Version

Use Case I: Summary Ntuple

This section provides detailed procedures to open and view a summary ntuple, create TCuts, and create an ASCII file containing ntuple contents.

Open and View a Summary Ntuple

1. To download an example summary ntuple ROOT file, go to:

   ftp://ftp-glast.slac.stanford.edu/glast.u07/mcenery/systests/GlastRelease/v6r2p8/AllGamma/linux/

   Download the *AllGamma_Merit.root* file and save it in *yourWork* directory.

   Troubleshooting Tip: Make sure that, if you have not set up a permanent environment for ROOT analysis, your temporary environment is set up correctly. (Refer to Set Root Environment Variables: Linux or Windows.)

2. Start up ROOT then, in sequence, enter the following commands:

TFile f("AllGamma_Merit.root", "READ")	open the Summary Ntuple file
f.ls()	view its contents
*TTree *MeritTuple = (TTree*)f.Get("MeritTuple")*	load the summar ntuple TTree
MeritTuple->StartViewer()	start the TreeViewer

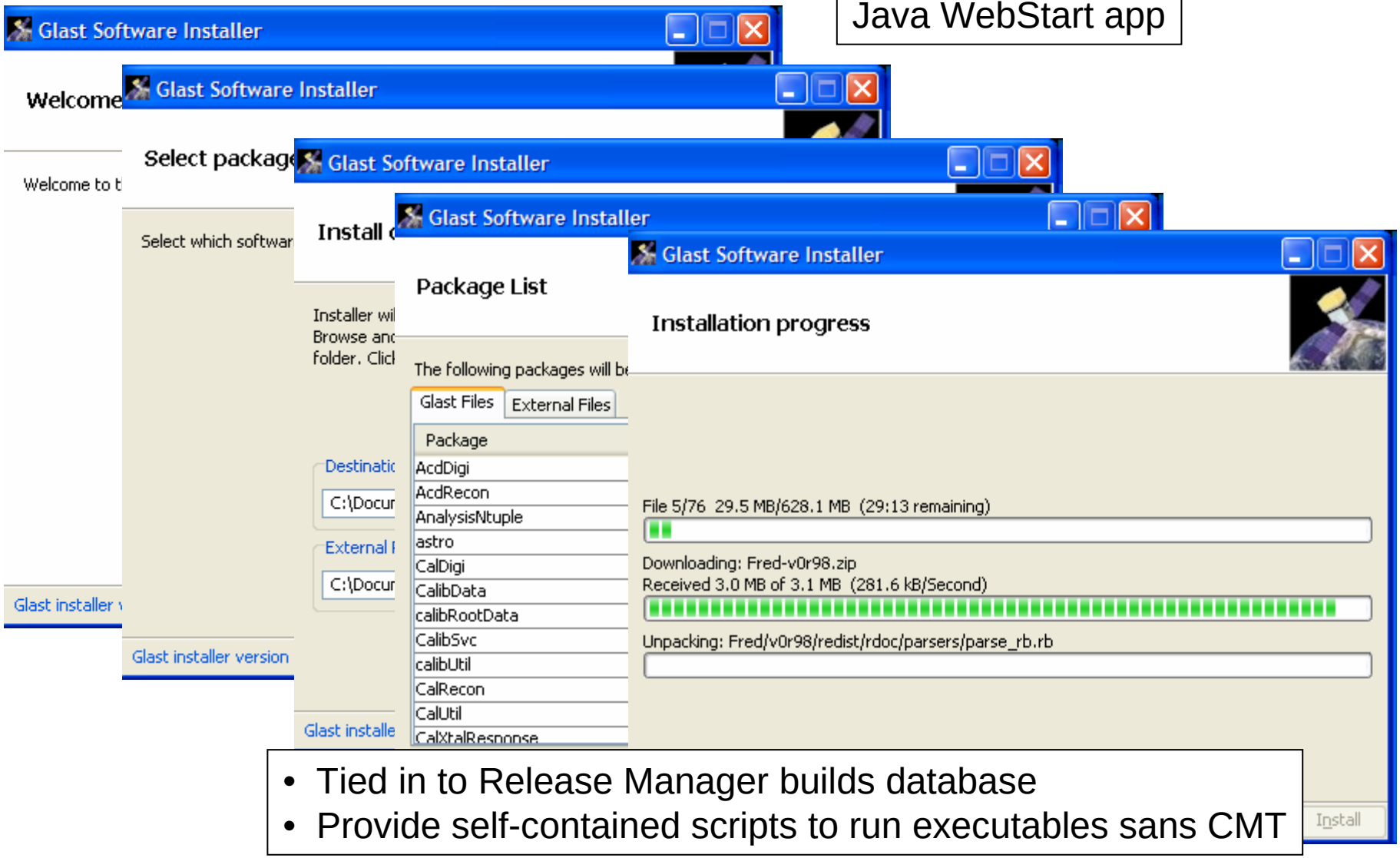
Your ROOT session should look similar to the following:

Follow on lead from SLD, BABAR, but …

- work with Tech Writer
  - skilled at extracting information from us wackos
  - worries about layout, organization
  - can write good
- we're struggling with apparent conflict of web navigation vs "printed book". Pursuing the former.

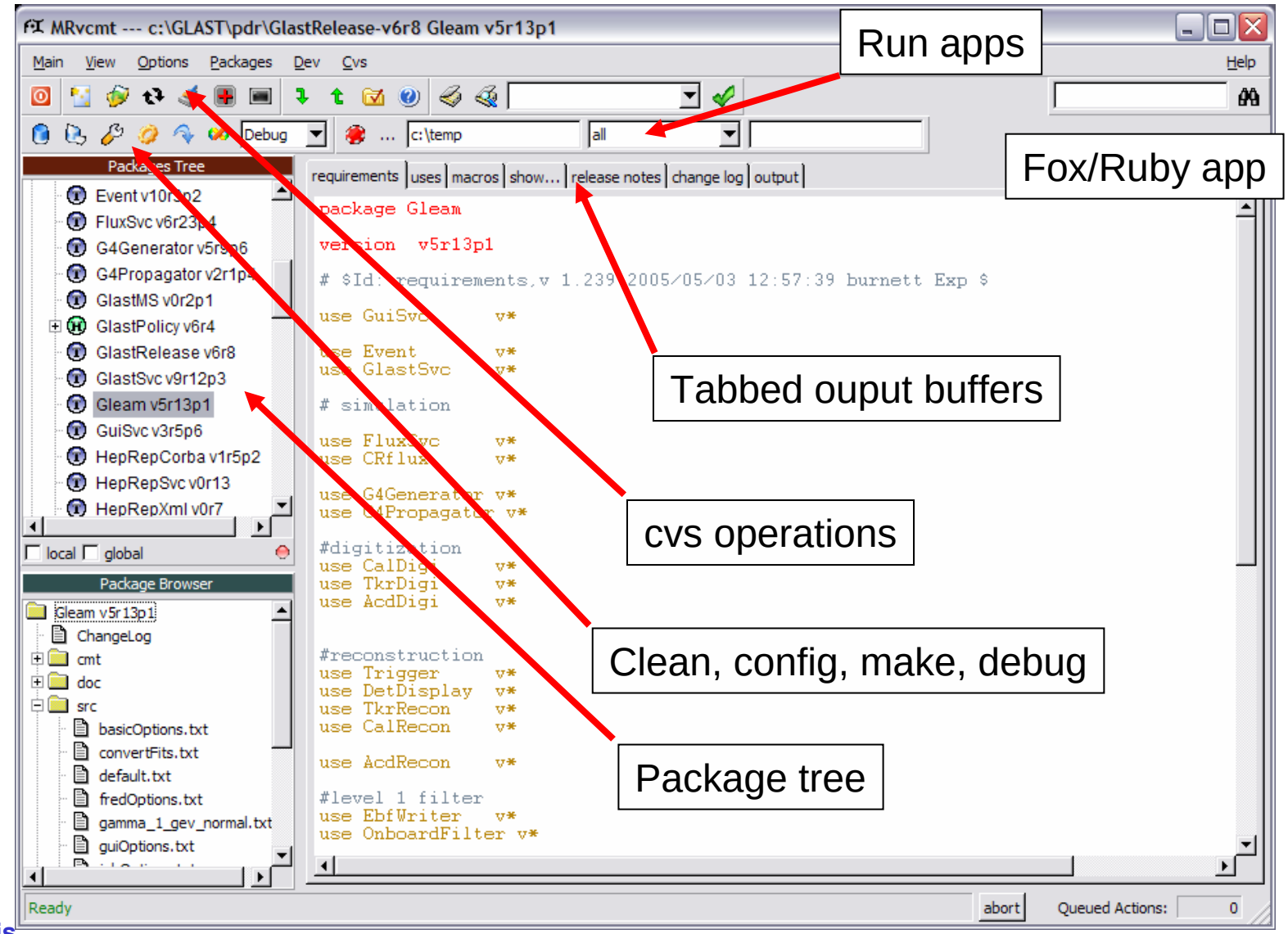
Code Distribution

Java WebStart app



- Tied in to Release Manager builds database
- Provide self-contained scripts to run executables sans CMT

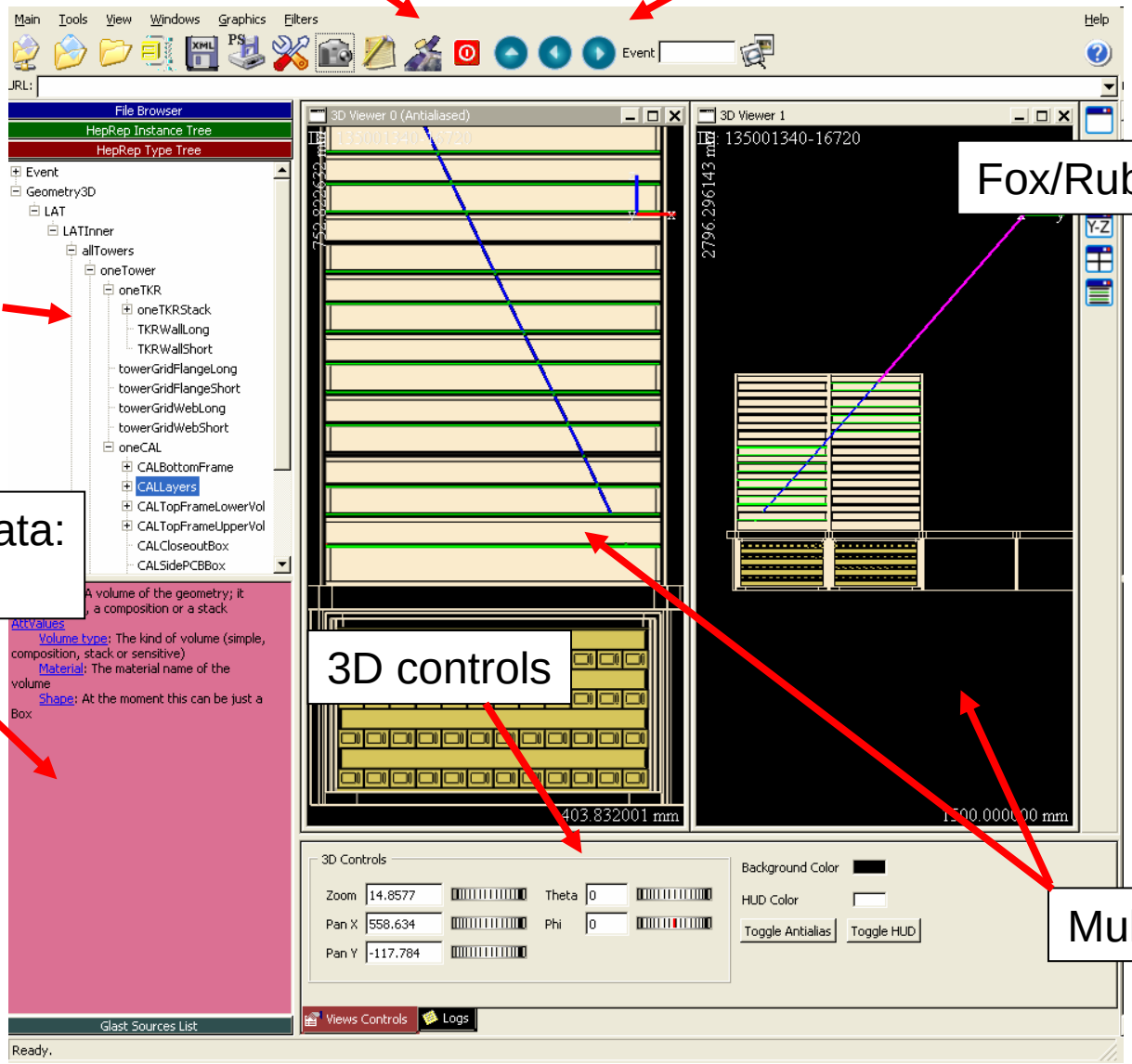
MRvcmt – gui for code development



FRED – Event Display

GLAST plugin
GlastRelease config

Event control



Fox/Ruby/C++ app

Graphics tree

Graphics metadata:
HepRep

3D controls

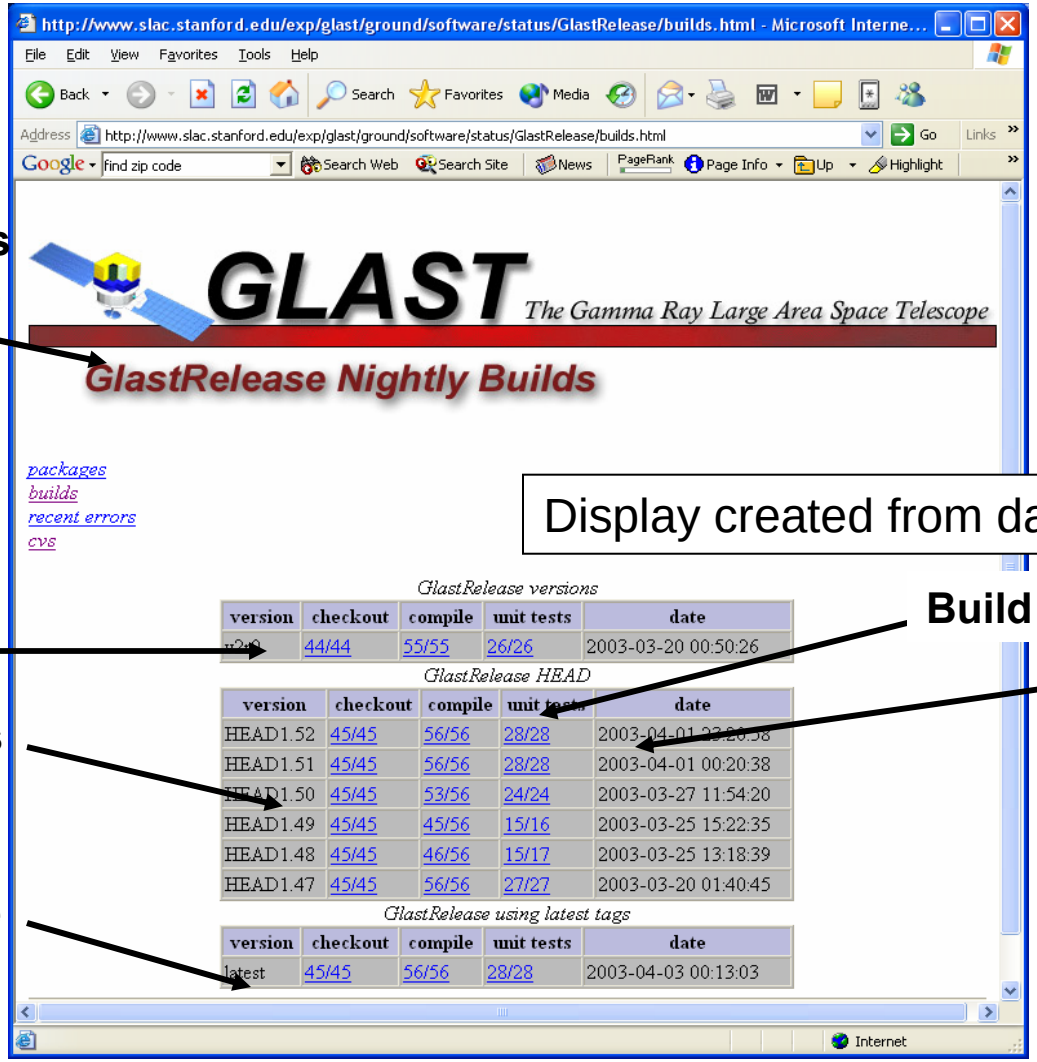
Multiple views

Issues Tracker; CCB; wiki

- User JIRA web issues tracker
  - Commercial product but affordable
  - Handles bugs, features, improvements
  - Full user/group management
  - “roadmaps” for version evolution/project management
- Change Control Board
  - Code used in pipeline – sim/recon; executive scripts; pipeline itself
  - Require documentation of all changes – preferably backed up by JIRA issues
  - Demonstration that fixes work; system tests on sim/recon
  - Using wiki tool to record actions
  - 4-person board – adjudicated by email so far
- Wiki
  - Commercial product (Atlassian – same parent as JIRA)
  - Simple web editing independent of user OS
  - Space management; same groups and users as JIRA

Code Builds

Performing builds for Science Tools also



Display created from database query

Build status

Unit test status

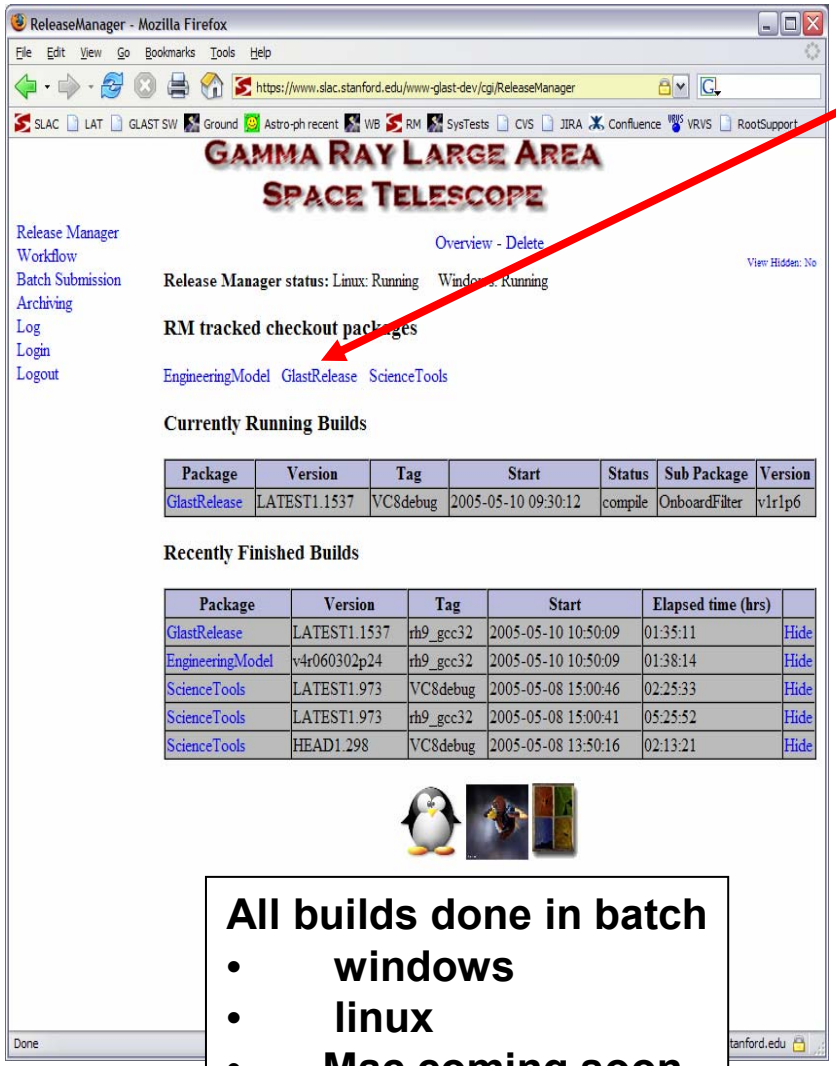
Past release

Release in progress

Future release

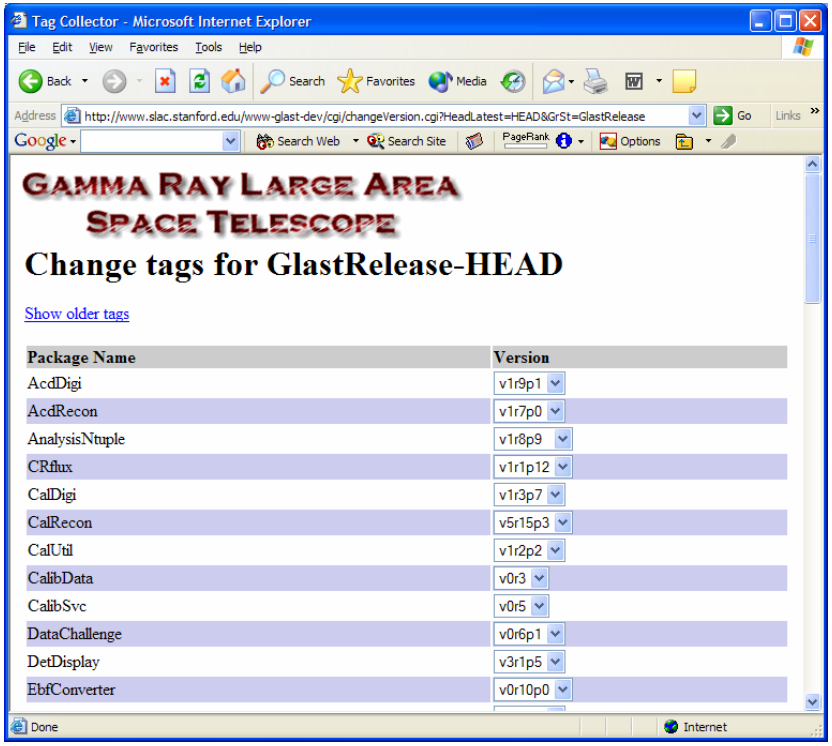
Alex

More Code Builds



Multiple packages being tracked

Web tag collector



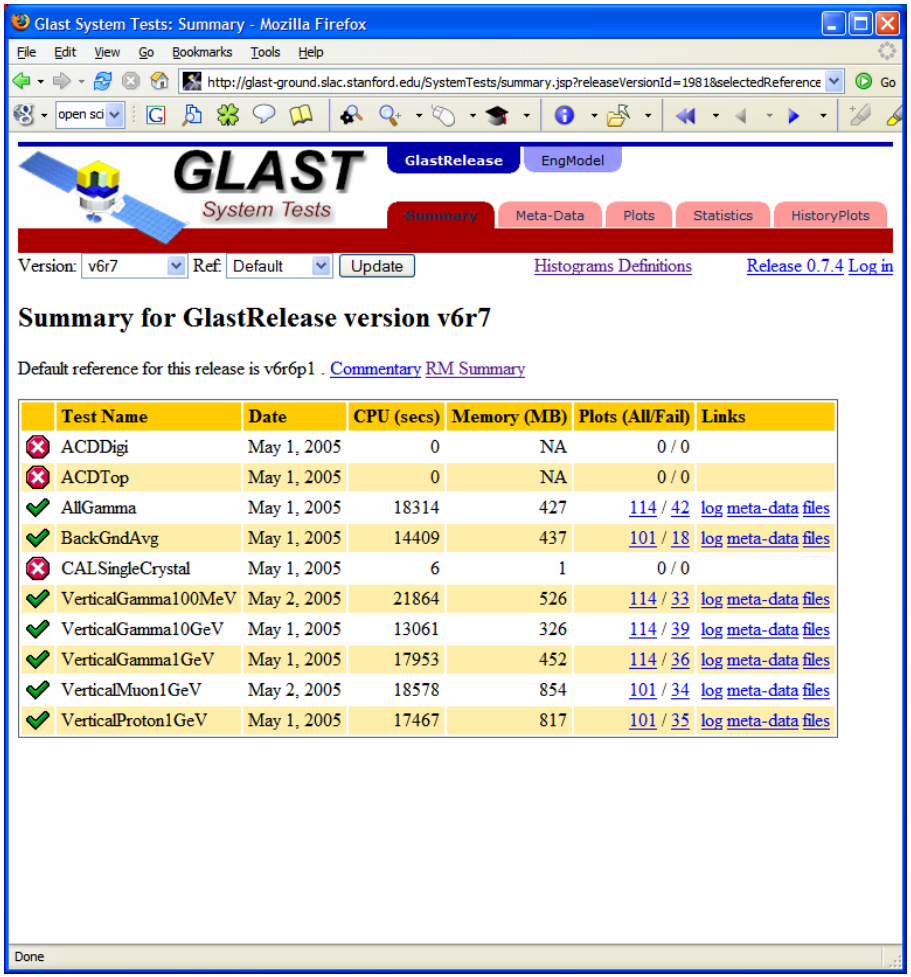
All builds done in batch

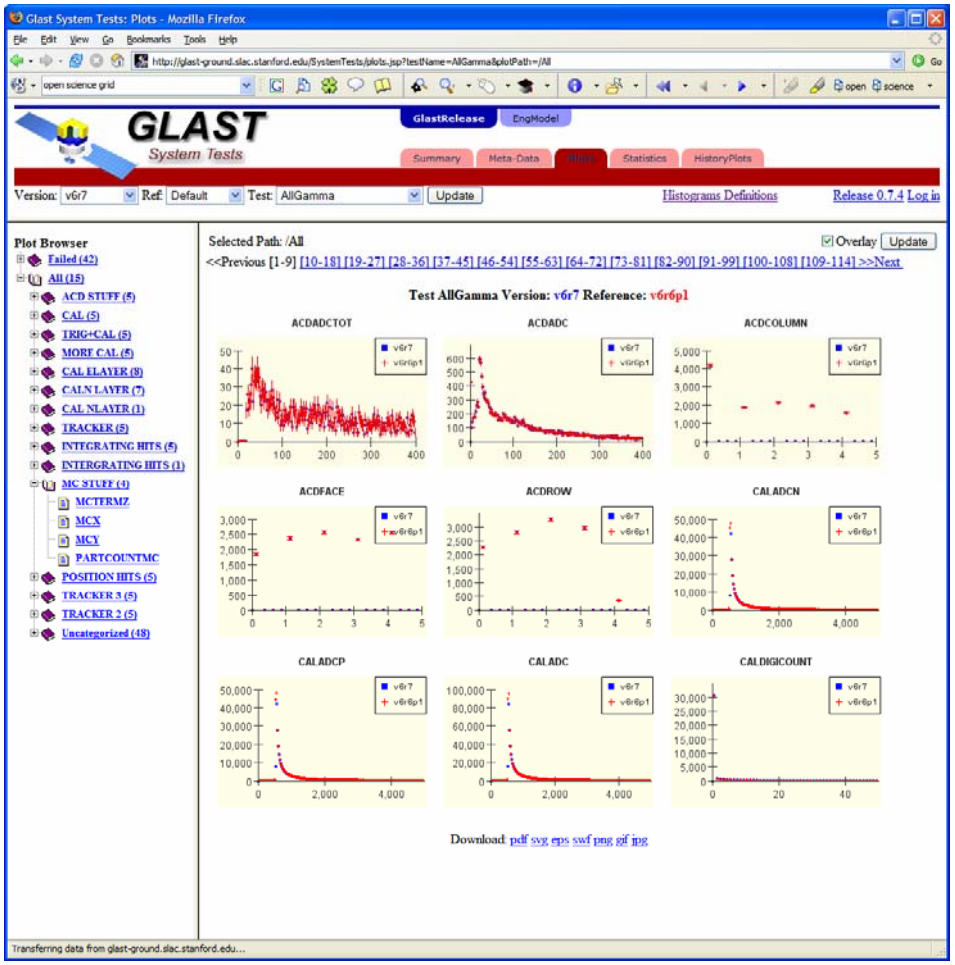
- **windows**
- **linux**
- **Mac coming soon**

System Tests

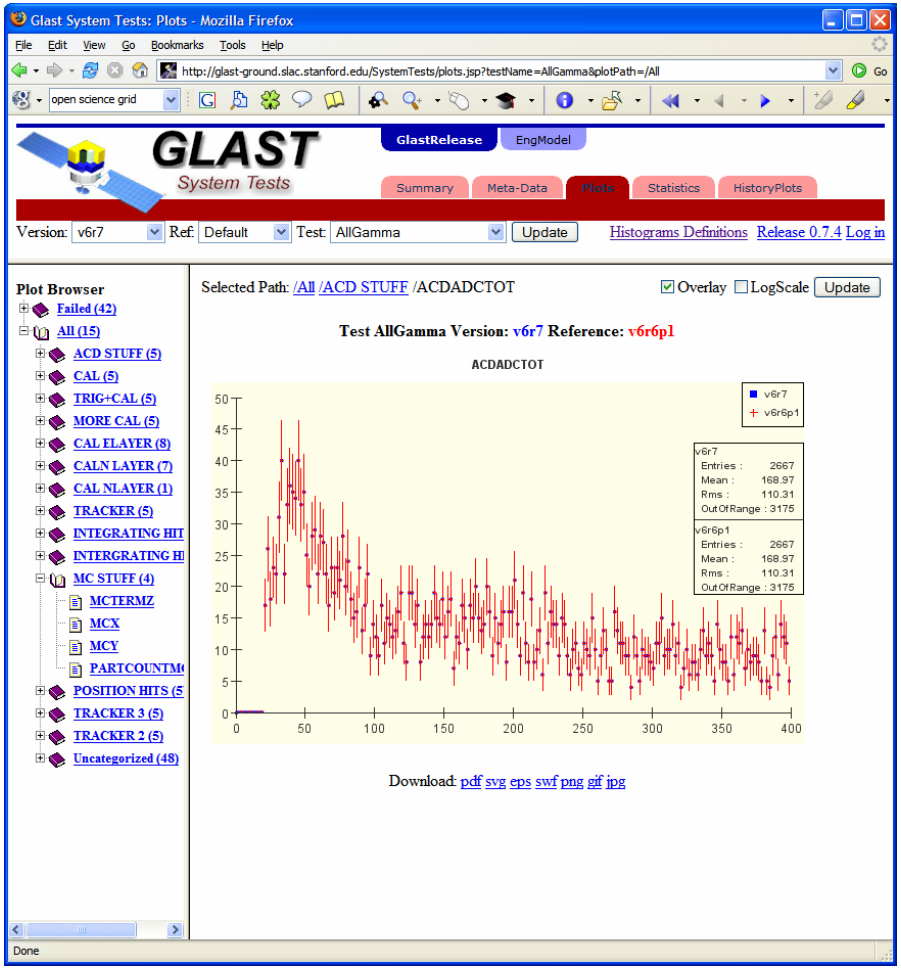
- Goals
  - Provides mechanism for validating:
    - Software releases (now)
    - Data quality (after launch)
  - Run (automatically) after each software release
    - Compares plots to references and flags problems
- Web based access to system tests results from any platform
  - No software install needed
  - Accesses data from combination of
    - Oracle database tables
    - Root files
  - Implemented using JAIDA, xrootd, JSP, Tomcat

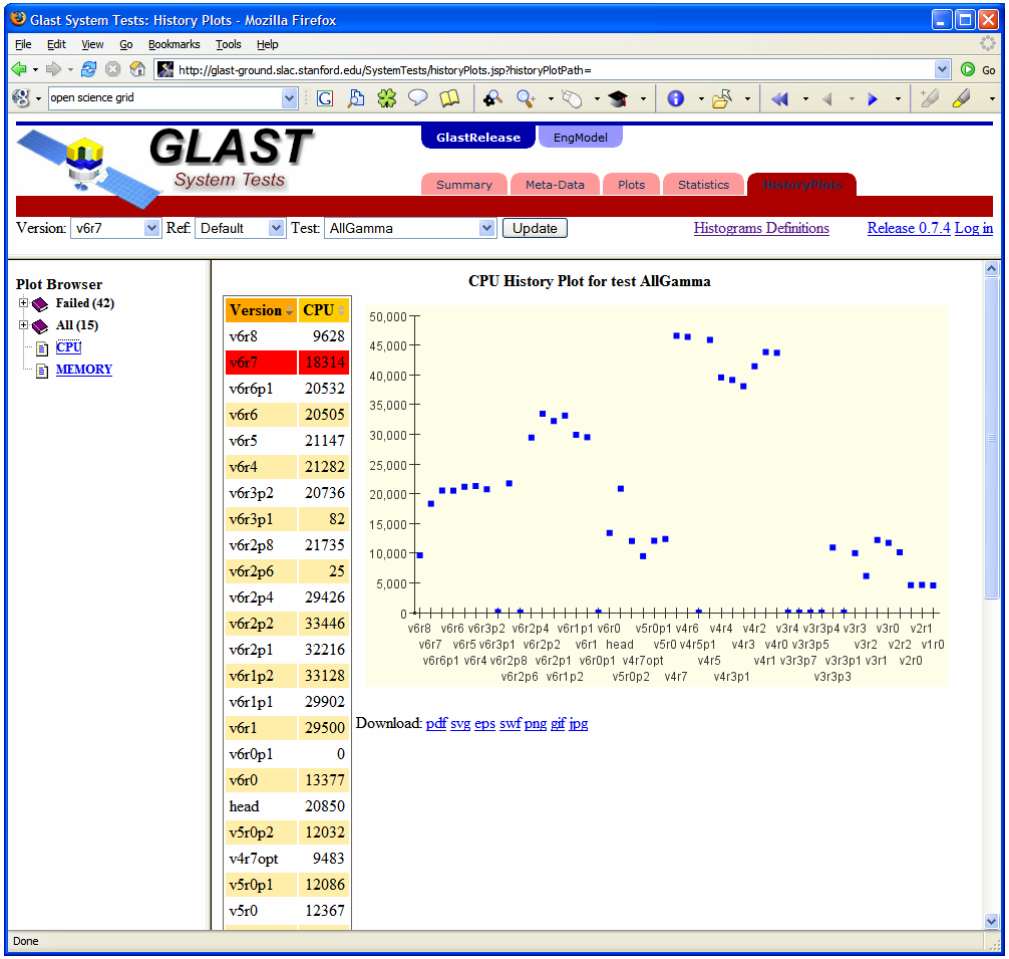
System Tests





System Tests





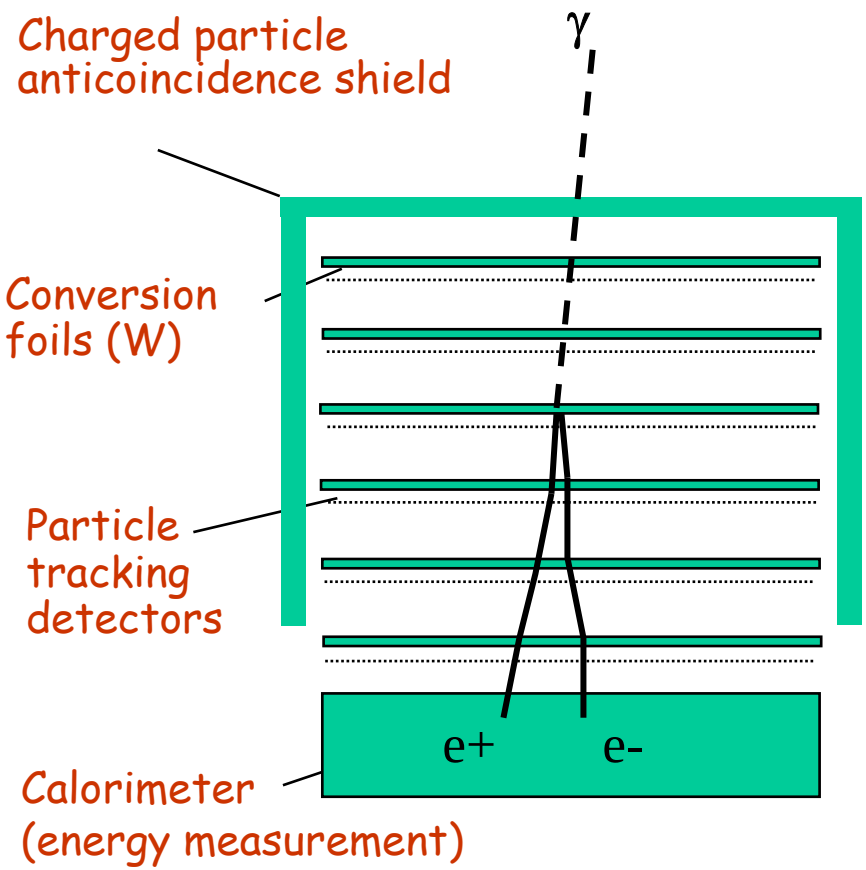
Sim/Recon Toolkit

Package	Description	Provider	Status
ACD, CAL, TKR Recon	Data reconstruction	LAT	90% done In use
ACD, CAL, TKR Sim	Instrument sim	LAT	95% done In use
GEANT4	Particle transport sim	G4 worldwide collaboration	In use
xml	Parameters	World standard	In use
Root 4.02.00	C++ object I/O	HEP standard	In use
Gaudi	Code skeleton	CERN standard	In use
doxygen	Code doc tool	World standard	In use
Visual C++/gnu	Development envs	World standards	In use
CMT	Code mgmt tool	HEP standard	In use
ViewCvs	cvs web viewer	World standard	In use
cvs	File version mgmt	World standard	In use

GLAST Reconstruction

Anatomy of a "Typical" Event

Pair production is the dominant photon interaction in our energy range



- **Reconstruction Goals:**
  - **Incident Gamma Direction and Energy**
  - **Reject Backgrounds**
- **Incident Gamma converts in the tracker**
  - **In particular, conversion occurs in one of the converter foils – ie at a well defined location**
- **Resulting electron-positron pair range out of tracker (TKR)…**
  - **No magnetic field, tracks are "straight lines"**
  - **Resulting two tracks "point" back to incident Gamma**
- **And into the CsI Calorimeter (CAL)**
  - **Measures total energy of electron-positron pair**
  - **= Gamma energy**
- **Surrounding Anti-Coincidence Detector (ACD) vetoes any wayward charged particles**

GLAST Reconstruction

What makes it challenging…

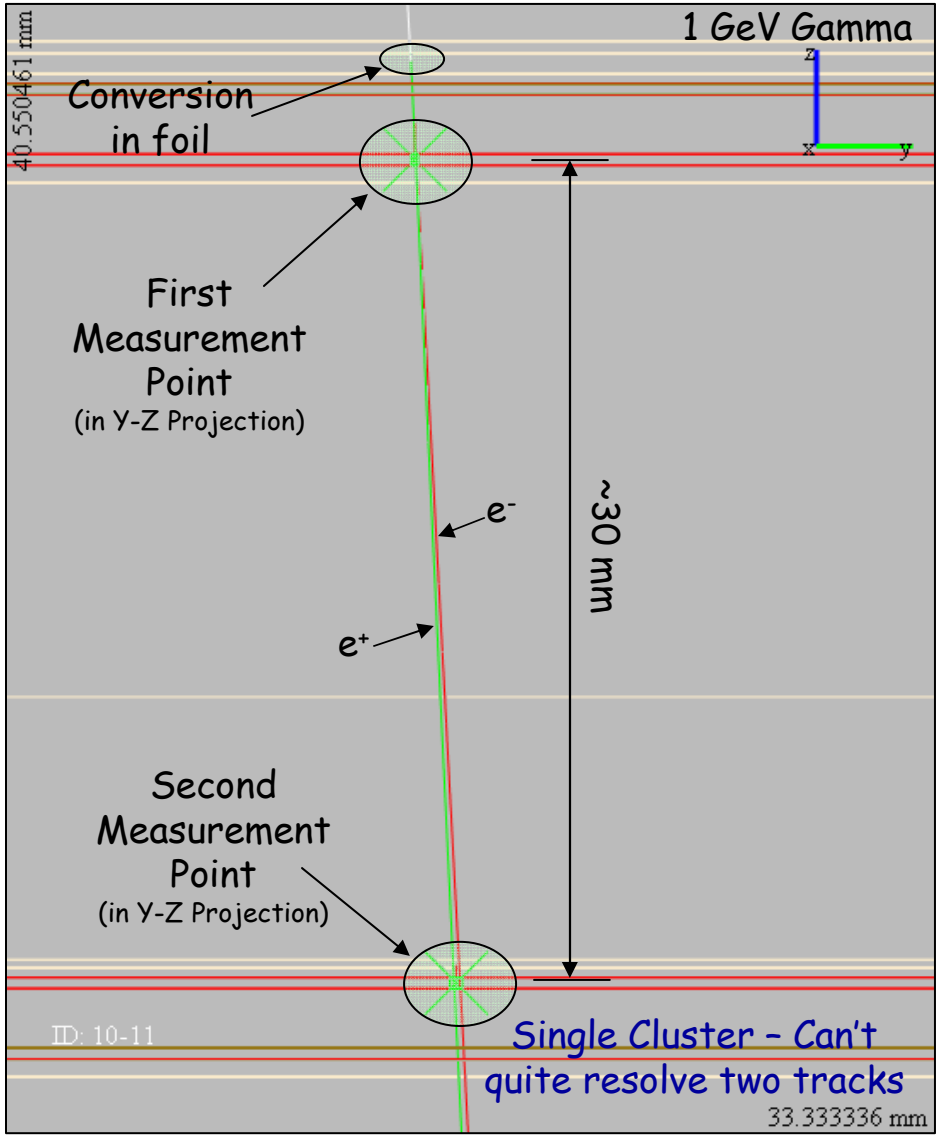
- **Track Opening Angle ~0**
  - **Resolve**
    ~ 2 * 228 um / 30 mm = ~15 mr

    Strip Pitch

    ~ Tray Spacing

    < ~50 MeV photons to resolve tracks without "help"

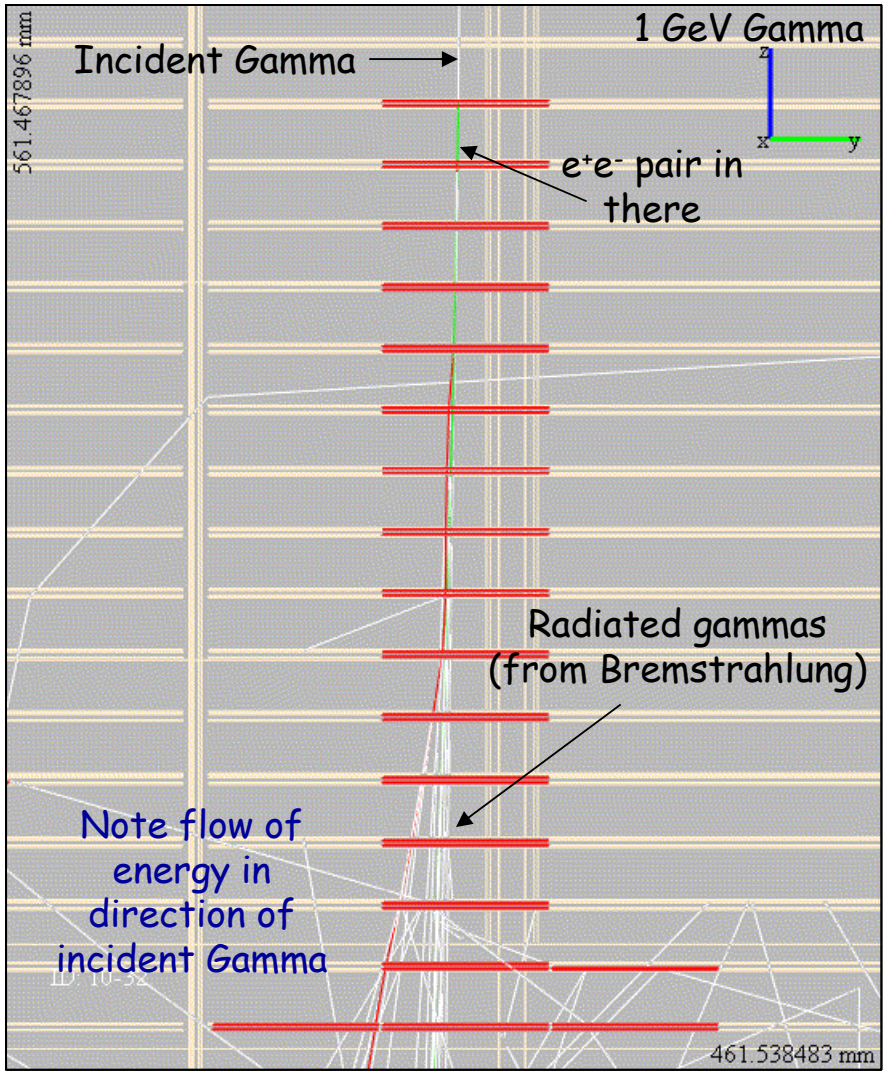
- **Looking for "v"s may not be the correct strategy for gamma direction reconstruction**
  - **Well… see next slides…**



GLAST Reconstruction

What makes it challenging…

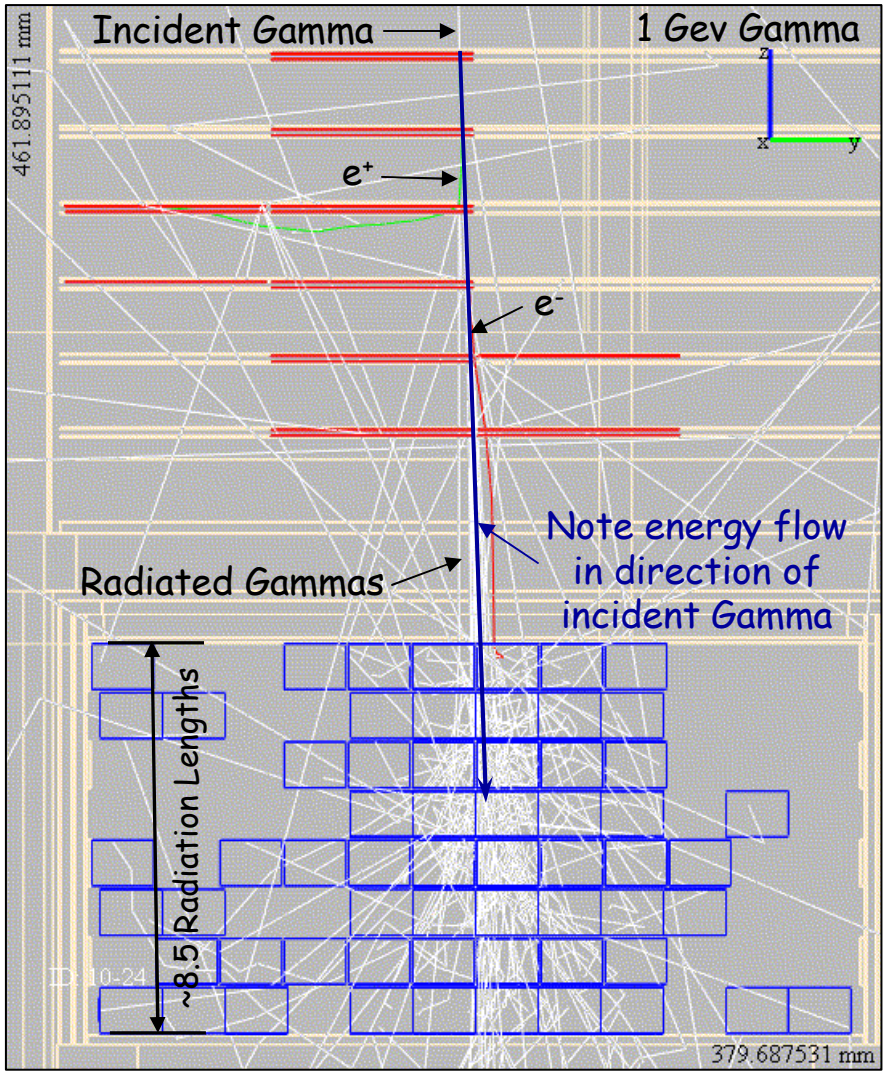
- **Tracker has a lot of material**
  - Actual tracker is ~ .3 rl
    - Could live with this…
  - Converter foils are ~ 1.1 rl
    - Love them: convert gamma
    - Hate them: tracking electrons
  - Total ~ 1.4 rl
    - For particles traversing active area of tracker
    - Does not include walls between towers, etc.
- **Issues to deal with**
  - Gammas can (and do) convert outside the foils
  - e^+e^- pair interact with tracker
    - Multiple scatter
    - Primary e^+ or e^- can stop in the tracker
    - e^+ and e^- radiate energy
    - etc.



GLAST Reconstruction

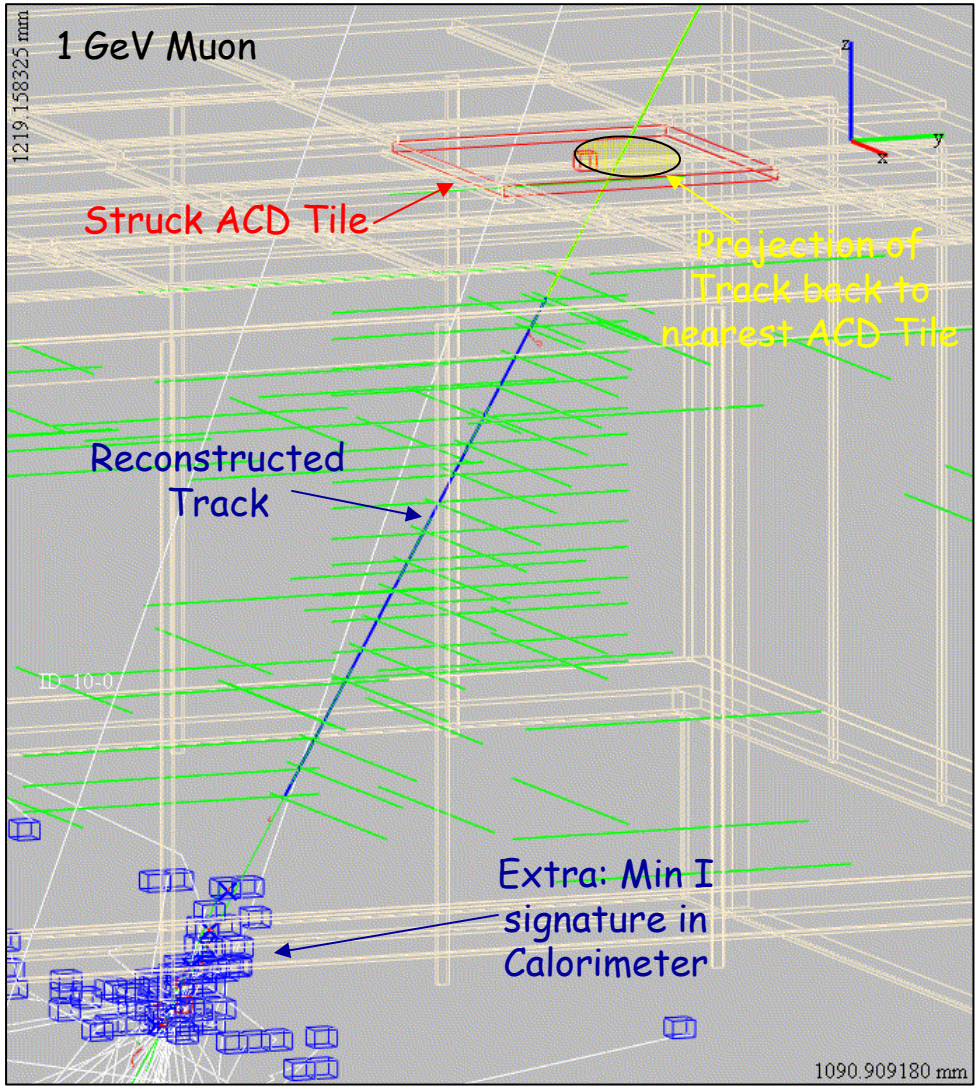
What makes it challenging…

- **Calorimeter Issues**
  - **Measure Event Energy – Not Track Energy(ies)**
    - **Don't have resolution to separate**
    - **Large fraction of measured energy from Brems**
    - **Implications for determining gamma direction when you do have two track events…**
  - **Measure Fraction of Event Energy**
    - **Energy "loss"**
      - **in tracker**
      - **Leaking out of Calorimeter**
    - **Significant contribution at**
      - **lower energies (e.g. < 1 GeV)**
      - **for conversions starting higher in the tracker**
    - **Must augment total energy determination with contribution from tracker**

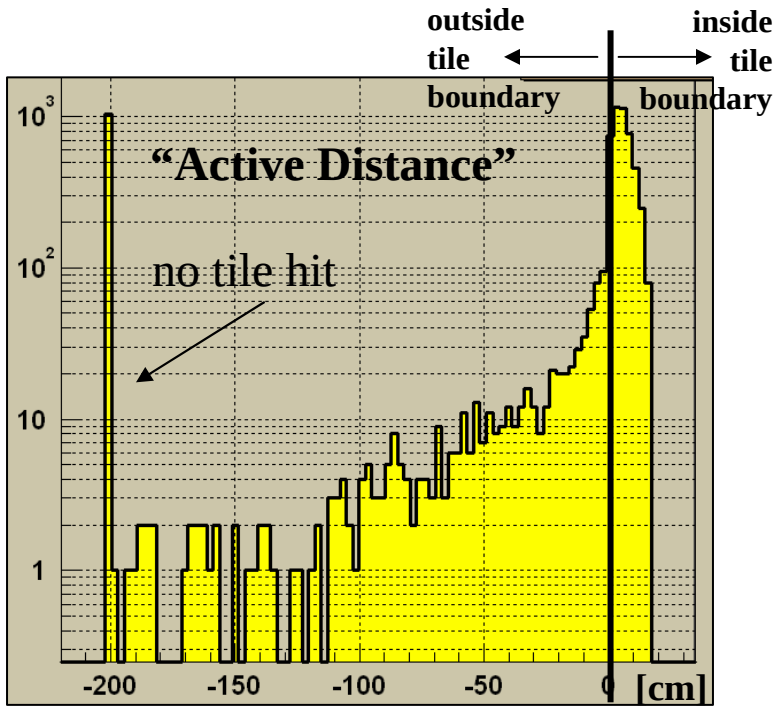


Background Rejection

Example: Charged Particles in Tracker

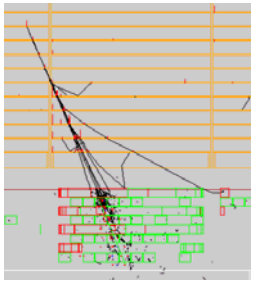


- Project Track to plane of struck tile
- Calculate distance to nearest edge
- Sign
  Positive if track projection inside the tile
  Negative if track projection outside the tile
- Reject if inside the tile

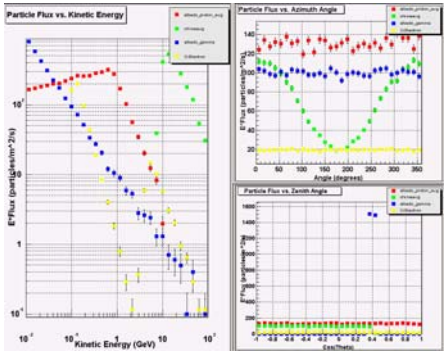


Instrument Simulation and Reconstruction

3 GeV gamma interaction



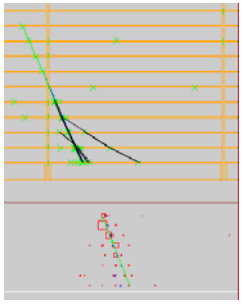
Source Fluxes



Particle Transport

"Raw" Data

Instrument data



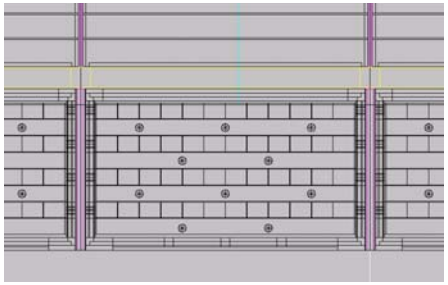
3 GeV gamma recon

Recon

Geometry

Background Rejection - Particle ID

Full geometry in xml with C++ interface
G4 discovers instrument from the xml



CAL Detail

Science Tools

- The 'Science Tools' are the high-level analysis tools for astronomy
- The core analysis tools have been defined and developed jointly with the GLAST Science Support Center (NASA/GSFC)
  - NASA staffed the GSSC early with this intent
  - These tools all adhere to the HEASARC FTOOL standards
- To the extent possible we have reused code from existing tools
  - Most notably for pulsar timing, e.g., barycenter arrival time corrections
- For source detection and characterization, the science tools use Instrument Response Functions (PSF, effective area, and energy dispersion as functions of relevant parameters), effectively abstracting the reconstruction and classification process
  - The greatest differences from the formalism for EGRET analysis is that the LAT will almost always be slewing, so that the response functions that apply to any given source also change continuously

Science Tools (2)

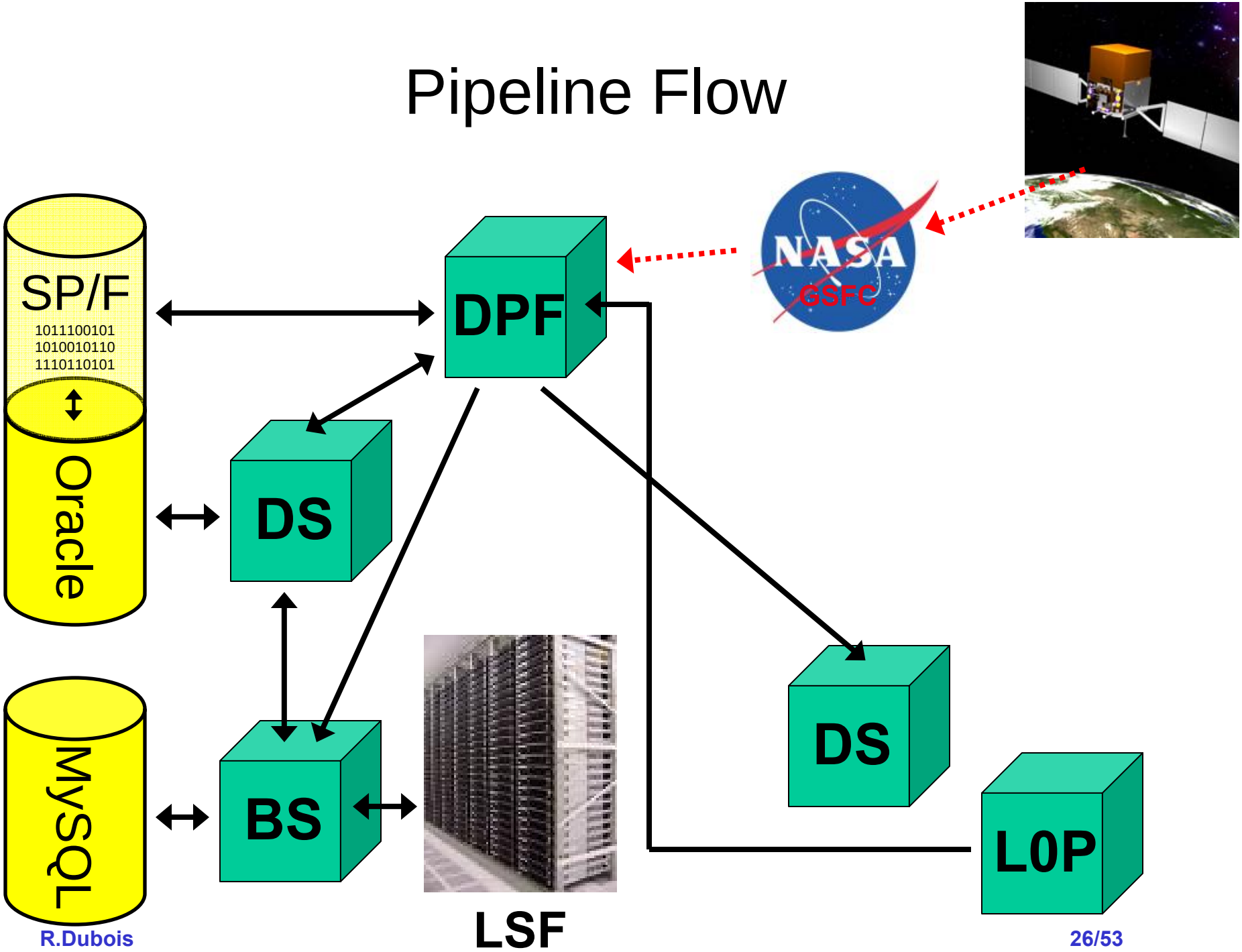
- After a period of definition and review, the tools have been developed incrementally, with the milestones for evaluation
  - Data Challenges (see later) as major milestones and 'Science Tools Checkouts' (3 so far) as intermediate ones
- The core Science Tools are
  - gtlikelihood, gtexpmap, and numerous associated utilities – for defining a model of a region of the sky and fitting it via maximizing the likelihood function
  - gtrspgen, gtbin – for generating response matrices and counts spectra for analysis of GRBs in XSPEC, including jointly with GBM data
  - gtbary, gtpphase, gtpsearch – and associated utilitites for pulsar timing, periodicity tests
  - gtobssim, gtorbsim – fast and flexible observation simulator using the IRFs, and an orbit/attitude simulator.

Pipeline Intro

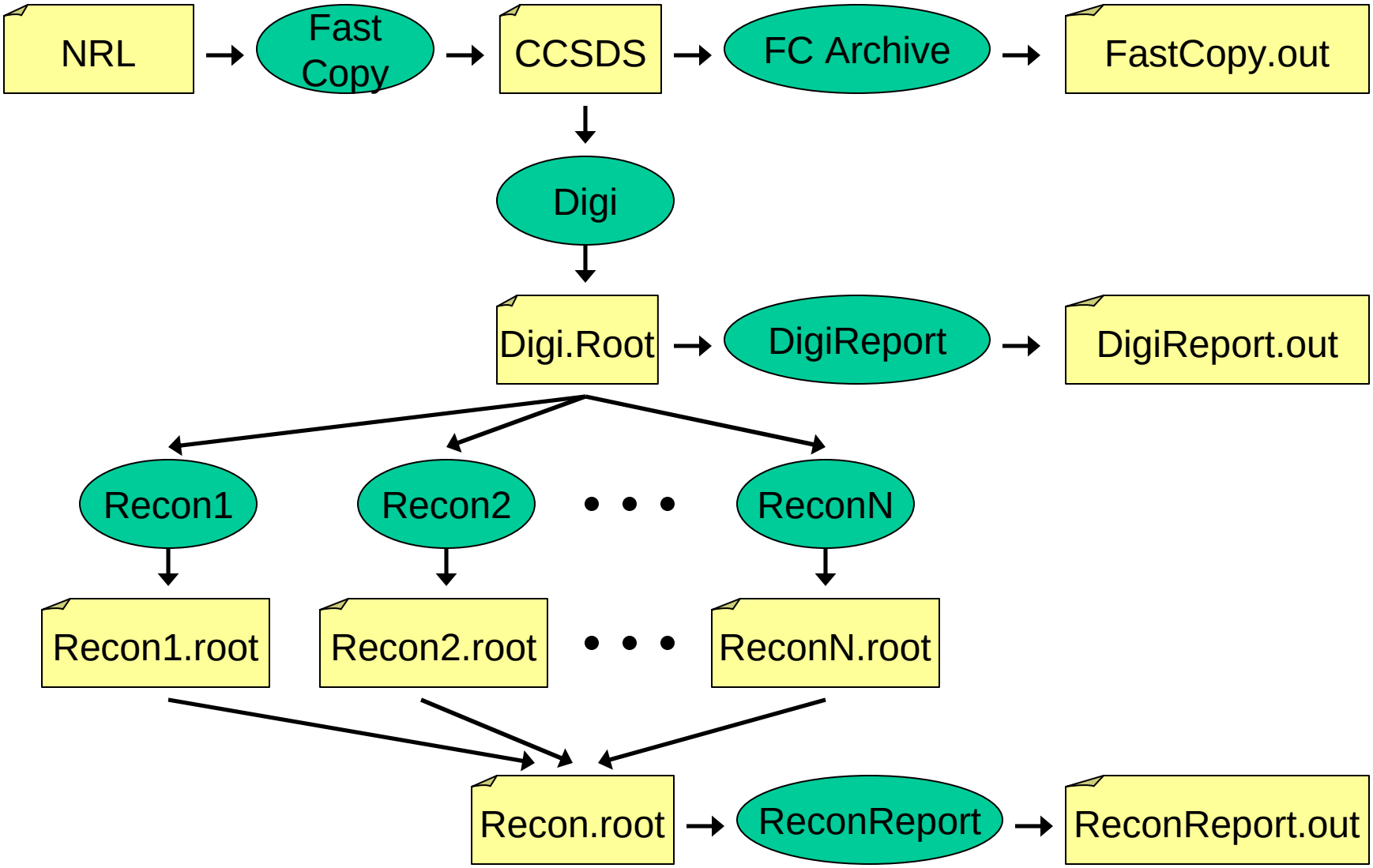
- **What is the pipeline?**
  - Envisaged as tool to provide a tree of processing on a given input dataset
  - Handle multiple “tasks” concurrently, eg LAT commissioning, DC2 Monte Carlo runs
  - Full bookkeeping to track what happened
  - Archive all files touched

- **Used by whom?**
  - Online
    - for sweeping integration data out of the clean room and to tape
    - populate eLogbook
  - SVAC (Science Verification and Calibrations)
    - for doing digi, recon
    - creating reports
    - Preparing for calibrations
  - Generic MC
    - DC2, background runs etc etc
  - ISOC (Instrument Science Operations Center)
    - Flight operations
    - environmental testing, at Spectrum Astro, KSC

Pipeline Flow



Sample Processing Chain



Current Pipeline: Major Components & Tech Used

- **RDBMS** (relational database management system)
  - Oracle
  - Contains all processing and data product history and relationships
- Data Exchange Layer
  - Oracle PL/SQL
  - Compiled SQL queries provide read/write access to tables
- DB Access Layer
  - Perl::DBI
  - Auto-Generated subroutines wrapping every public stored function and procedure
  - Provides simple, seamless DB interface to Perl Utilities
  - Also Perl classes representing each record type
- Scheduler, utilities
  - Perl
  - Higher level code to manage data and processing
  - Little dependency on actual table structure gives developer freedom to write maintainable, extensible code

Web Monitoring of Pipeline Progress

GLAST Pipeline

Version 1.4.2 | Jira | Help

Access control by user

User: richard . Logout

Mode: **Prod** Switch to: [Prod | Dev | Test]

Upload configuration file

Task in question

summary / interleaveDC2-GR-v7r3p21 / mc1step

Processing step in chain

Runs for process: mc1step

NEW Show processing statistics

Filter queries

Run Min Max Status: Success

Date Start None End None Filter Clear

Show all runs on one page

31,619 items found, displaying 1 to 20.

[First/Prev] **1**, 2, 3, 4, 5, 6, 7, 8 [Next/Last]

Run	Status	Submitted	Memory (MB)	CPU (secs)	Job Id	Links (?)
1	Success	2006-02-01 22:38	565	2035	100544	Log : Files : Out : Err
2	Success	2006-02-01 22:45	564	2075	100845	Log : Files : Out : Err
3	Success	2006-02-01 22:45	557	2140	100844	Log : Files : Out : Err
4	Success	2006-02-01 22:45	561	2183	100843	Log : Files : Out : Err
5	Success	2006-02-01 22:45	561	2197	100846	Log : Files : Out : Err
6	Success	2006-02-01 22:45	576	3598	100847	Log : Files : Out : Err
7	Success	2006-02-01 22:45	579	2220	100850	Log : Files : Out : Err
8	Success	2006-02-01 22:45	578	2322	100848	Log : Files : Out : Err
9	Success	2006-02-01 22:45	599	2313	100849	Log : Files : Out : Err
10	Success	2006-02-01 22:45	599	2376	100851	Log : Files : Out : Err
11	Success	2006-02-01 22:45	593	2772	100852	Log : Files : Out : Err
12	Success	2006-02-01 22:45	538	3965	100864	Log : Files : Out : Err
13	Success	2006-02-01 22:45	594	3717	100865	Log : Files : Out : Err
14	Success	2006-02-01 22:45	587	2726	100867	Log : Files : Out : Err
15	Success	2006-02-01 22:45	554	3956	100868	Log : Files : Out : Err

Pipeline 2

- Build on experience from #1
  - # is now robust, but lacking in areas of flexibility

- Revisited requirements:
  - Task scheduling should be more flexible that current linear chain
    - Should support parallel execution of tasks
    - Should allow dependency chain to be more general than the input file requirements
    - Should support parallel sub-tasks, with number of sub-tasks defined at runtime
    - Perhaps support conditions based on external dependencies
  - Should allow for remote submission of jobs
    - Perhaps using GRID batch submission component, or Glast specific batch submission system
    - Will need to generalize current system (e.g. get rid of absolute paths)
  - Support reprocessing of data without redefining task
    - Need way to mark Done task as "ReRunnable"
    - Need to support multiple versions of output files
  - Ability to Prioritize tasks
  - Ability to work with "disk space allocator"
  - Would be nice to set parameters (env vars) in task description
  - Would be nice to be able to pass in parameters in "createJob"
  - Ability to suspend tasks
  - Ability to kill tasks
  - Ability to throttle job submission (ie max number of jobs in queue)
  - Ability to map absolute path names to FTP path names (site specific)
  - Would be nice to remove need for "wrapper scripts"
  - Ability to specify batch options (but portability problems)

- Redesigning database schema now
- Targeting beamtest for production use

Data Server Portal

- Glast will run two data servers
  - One for the public at Goddard Space Flight Center
  - One at SLAC for Glast collaborators
- LAT Physicists will access science data via Astro Data Server
  - Pulls events associated with
    - Particular region of the sky
      - Satellite doesn't stay still so this is spread throughout data.
    - Energy range
    - Time Period
  - Removes need for users to know how/where data is stored
    - For most astrophysics measurements physicists only need to know about photon direction and efficiency, details of reconstruction/simulation are largely irrelevant
  - Should be able to download data in various formats
    - List of run/events
    - Tuples (FITS, root, possibly with choice of number of columns)
    - Full root trees
  - Should be able to browse events
    - with web based event display (WIRED)
  - Should be able to store personal favorite searches
    - Should be able to download incremental updates to data
- Expect to get 100M events/year for 10 years
  - Small compared to Babar, but we want fast turnaround

Instrument Data Access

Version 1.1 | Jira | Help

GLAST *The Gamma Ray Large Area Space Telescope*

Run Mode: **Prod**

Welcome | **Pruner** | Peeler | History | Admin

Select detailed event data

Select summary ntuple events

Batch data for user: richard

Task type: pruner batch number: 126191

Please edit data and than press the 'Proceed' button

Batch Parameters	Parameter Values
Task Name	interleaveDC2-GR-v7r3p21
E-mail	richard@slac.stanford.edu
Tcut	CTBGAM>0
Min Run Number	
Max Run Number	
Debug mode	false
User Comment	10-day test prune
Batch Options	
Max Filesize [MB]	

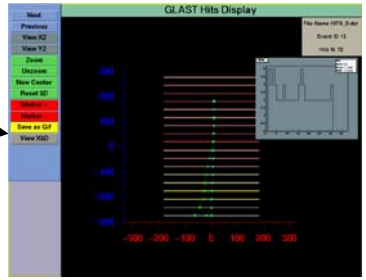
Astro Data Server

Web Form
Region in Sky:
Time Range:
Energy Range:
Gammas/Events:

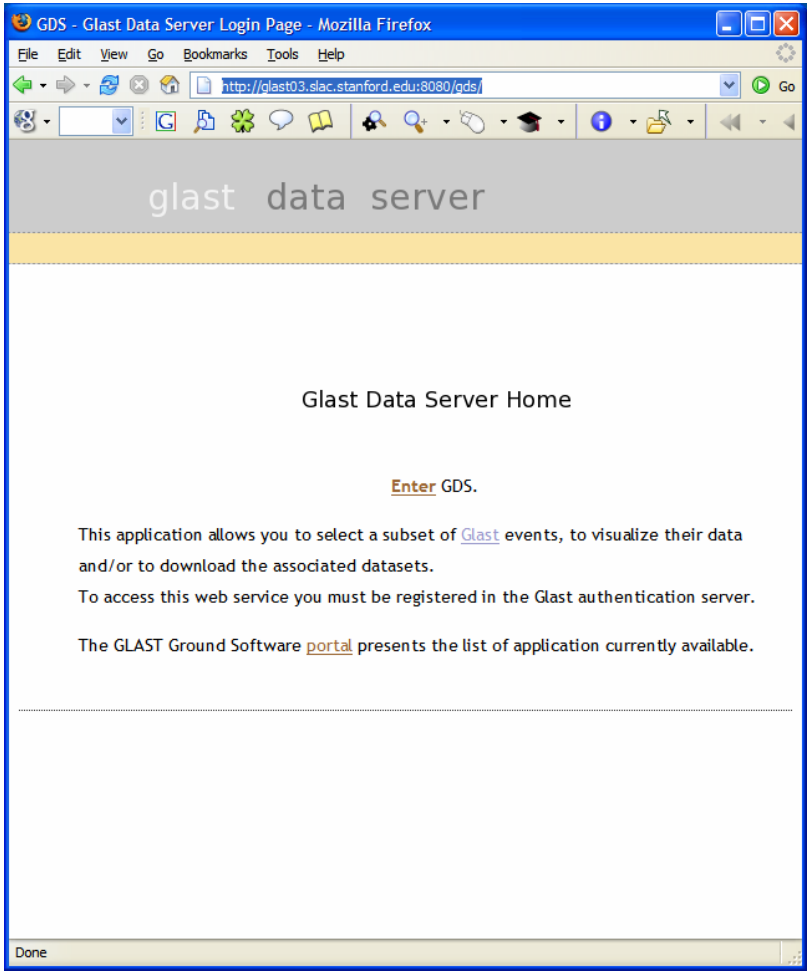
You selected 39383844 events
Change Criteria
Add “TCut”
Browse Events
Download: Event Selection
Compressed Tuple
Full Merit Tuple Full Root Tree

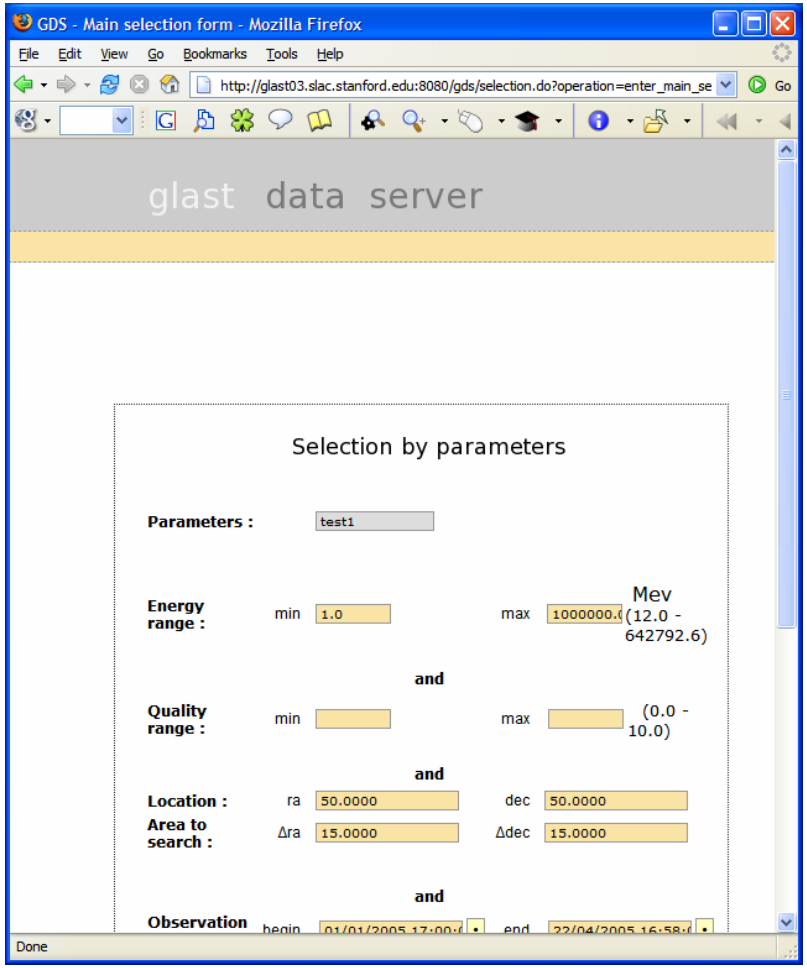
In memory meta-data
Binned by sky position, time, energy

Root Event Store



Astro Data Server





Trending Application

ISOCtrends

Version 0.6.1 | Jira | Help | Log in | Dbx Version: B0-6-1

Source: 133,134 **Archive:** Statistics **Time (UTC) Begin:** 2006-02-02 17:44:21 **End:** 2006-02-12 17:44:21 **Channel Filter:** .*LHKP0*

Change Selection ...

Channels | Groups

Tree

- LHKP (628)
  - 0ACD (19)
  - 0CAL (32)
  - 0EPU (18)
  - 0GRA (9)
  - 0GRI (24)
  - 0MHP (36)
  - 0PHP (36)
  - 0RAD (28)
  - 0TEM (112)
  - 1ACD (19)
  - 1CAL (32)
  - 1EPU (18)
  - 1GRA (9)
  - 1GRI (24)
  - 1MHP (36)
  - 1PHP (36)
  - 1RAD (28)
  - 1TEM (112)
    - 033V
    - 033VST
    - 0PCT
    - 0PCTST

Selected path : /LHKP/1TEM/033VST for sources: 133,134

Plot | Tabulate | Info

☐ Show Dbx Version History Update

LHKP1TEM033VST PDU1 TEM0 3.3V digital status

TLM

mnemonic	shortDesc	longDesc	typeName	cnvType	cnvName
LHKP1TEM033VST	PDU1 TEM0 3.3V digital status	A 4 bit value representing an ADC read/evaluation status.	U12	DSC	LHKSTATUSBITS

PKT

apid↑	byte	bit	bits	typeName	FSW
642	26	0	4	U12	link
546	26	0	4	U12	link

DSC

name ↑	text	low	high	fore	back	shortDesc	longDesc
LHKSTATUSBITS	LHKREDSTAT	3	3			ADC red limit status	Status indicating an ADC red limit threshold violation.
LHKSTATUSBITS	LHKSTATMSK	2	2			Status Masked	Indicates no evalutation status because value was masked as disabled.
LHKSTATUSBITS	LHKSTATOK	0	0			Data OK status	Status indicating measurements was evaluated as OK.
LHKSTATUSBITS	LHKSTATUNDF	1	1			Status Undefined	Indicates absence of data due to acquisition timeout or other failure.

Back

Find: source Find Next Find Previous Highlight all Match case

Query and plot data from database: this example is for HSK data

I&T Support

- Two main areas of support:
  - CCB controlled sim/recon package "EngineeringModel"
    - More stable version, protected from the bleeding edge code development path. No special-purpose EM code.
    - Emphasis on real data, calibrations etc
    - ~ 0.5 FTE dedicated to this from SAS
  - Most complicated use of Pipeline
    - Most needs for Pipeline 2 are driven by lessons learned from I&T
    - In routine use by I&T for over a year now
    - I&T-specific tasks "operated" by I&T
    - Pipeline-proper maintenance by SAS

Data Challenges

- Ground software is amalgam of HEP instrument software and Astro FTOOLS
- Adopt HEP's "Data Challenges" to create a series of end-to-end studies: create a progression of ever more demanding studies
- Originated by the Mark2 experiment at SLAC while waiting for the SLC accelerator to deliver data
  - Test and oil the data analysis system from simulating the physics through full blown analyses
  - Details of physics and detector performance not revealed to the collaboration until closeout
  - Engage the collaboration and get it thinking science
- ISOC is an integral part of the collaboration
  - Exercise its part and interactions with the rest of the collaboration

Data Challenges: Three Rounds

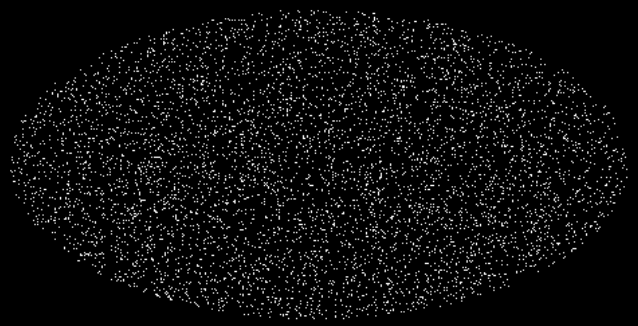
- DC1. Modest goals. Contains most essential features of a data challenge.
  - 1 simulated day all-sky survey simulation
  - find GRBs
  - recognize simple hardware problem(s)
  - a few physics surprises
  - Exercise all the components
- DC2, kickoff Mar 1. More ambitious goals. Encourage further development, based on lessons from DC1. Two simulated months.
  - DC1 +
    - Much more data
    - Backgrounds included
    - More realistic GRBs
    - Pulsars, variable AGNs
    - More and more elaborate surprises
- DC3, in CY07. Support for flight science production.

DC1 Components

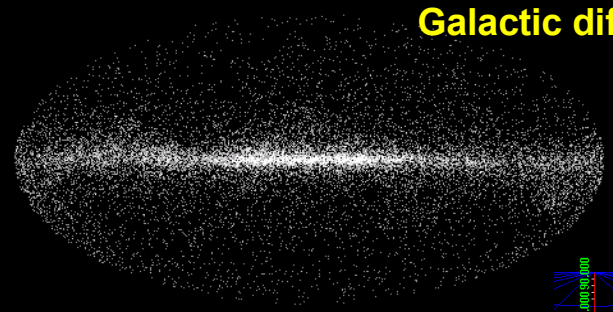
- Focal point for many threads
  - Orbit, rocking, celestial coordinates, pointing history
  - Plausible model of the sky
  - Background rejection and event selection
  - Instrument Response Functions
  - Data formats for input to high level tools
  - First look at major science tools – Likelihood, Observation Simulator
  - Generation of datasets
  - Populate and exercise data servers at SSC & LAT
  - Code distribution on windows and linux
- Involve new users from across the collaboration
- Teamwork!

The Simulated Sky

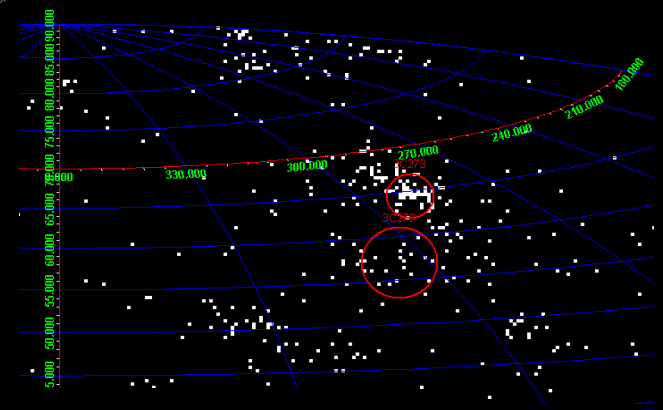
Extragalactic diffuse

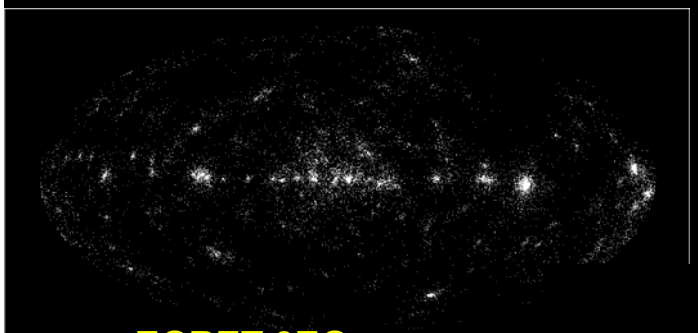


Galactic diffuse



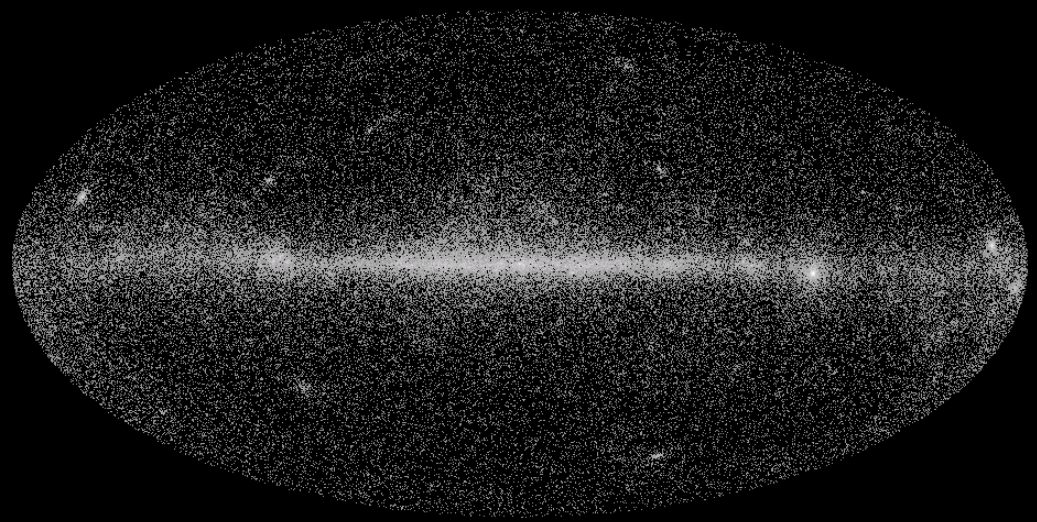
Fiddling 3C273/279





EGRET 3EG

Our Sky



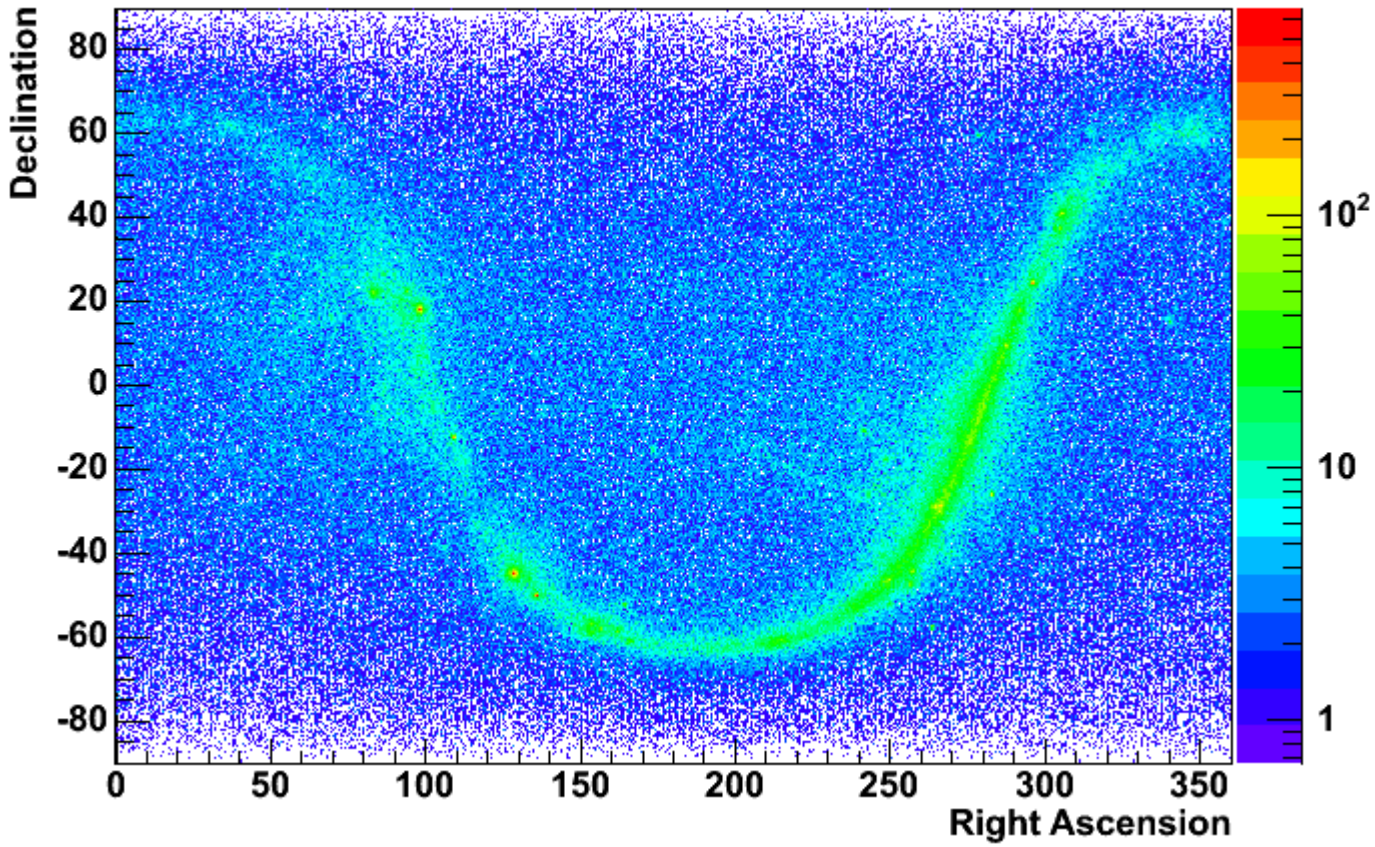
How did DC1 Go?

- A great success – cue Seth ;-)
- First time integrating the sources/simulation/reconstruction system
  - Many problems cropped up and were dealt with
  - Would not have been without this exercise
  - Were burning CDs with DC1 data the night before kickoff
- Were not able to include backgrounds in the Challenge
  - Full background rejection analysis not available
  - For 1 day, not so important
- Data distributed by CD and ftp
  - No smart servers
- Tremendous team building event!
  - 40+ attended meetings
  - All major elements of the physics were analyzed and found

Prep for DC2

- **Full background analysis this time!**
  - Tremendous collaboration effort to reduce the backgrounds to Science Requirements levels
  - Revision of background model – x4 higher than DC1 estimate
  - Detailed skymodel
    - Flaring objects; pulsars; joint GBM data(!); etc etc
    - Can't let the cat out of the bag too much
  - Mechanically a huge change from DC1
    - Have to simulate a source 10^3x signal
    - 100,000 CPU-hrs to simulate 1 day of background: 5 billion events
    - Machinery to randomly interleave that day 55 times, while simulating full rate deadtime effects
    - High-stress test of processing pipeline
      - ~400 CPUs running simultaneously for a week for the backgrounds runs
      - ~200,000 batch jobs total for DC2
    - Many scaling problems fixed

Sample of DC2 Sky Simulation

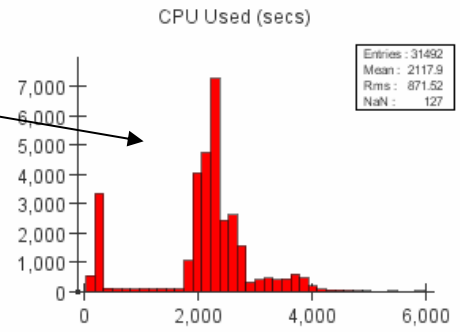


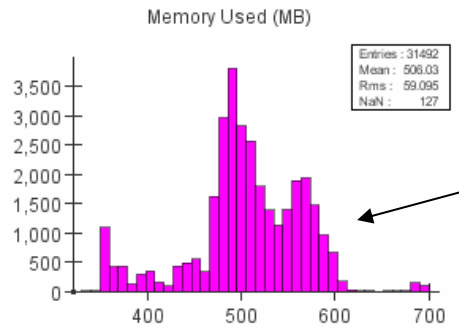
Monitoring Pipeline Throughput

Pipeline Statistics

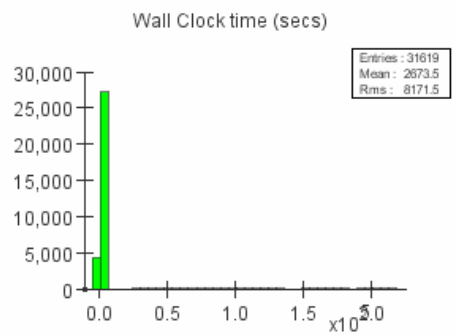
Summary | mc1step

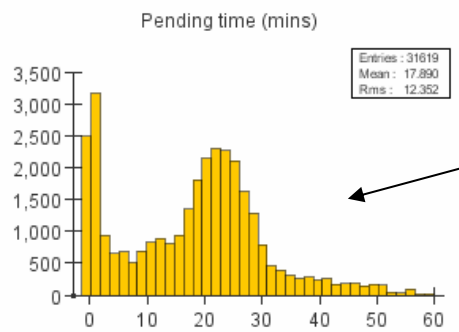
CPU time



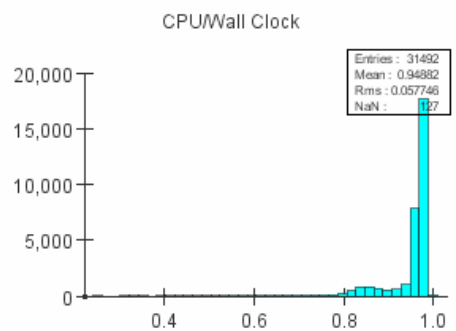


Memory





"Wait" time for jobs



Ratio wall clock to CPU

Monitoring Disk Farm via SCS Tools

sulky35.slac.stanford.edu Overview

This host is up and running.

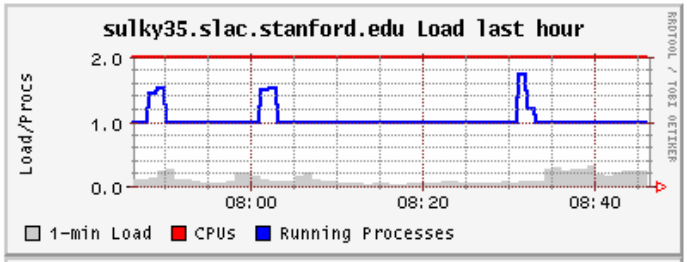
Time and String Metrics

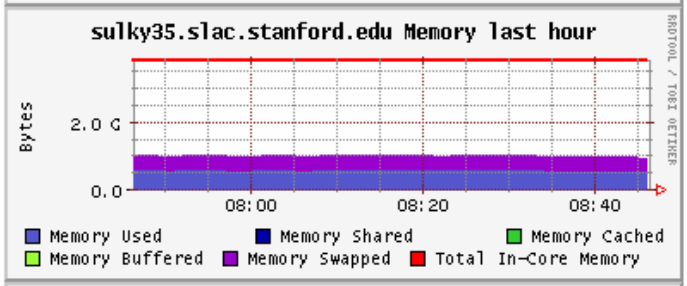
boottime	Sun, 5 Feb 2006 14:25:42 -0800
gexec	OFF
last_reported	0 days, 0:00:07
machine_type	sun4u
os_name	SunOS
os_release	5.9
uptime	6 days, 18:20:32

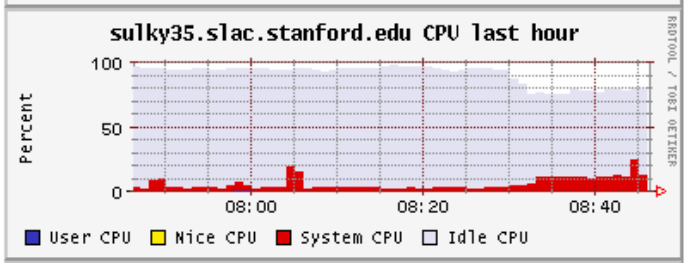
Constant Metrics

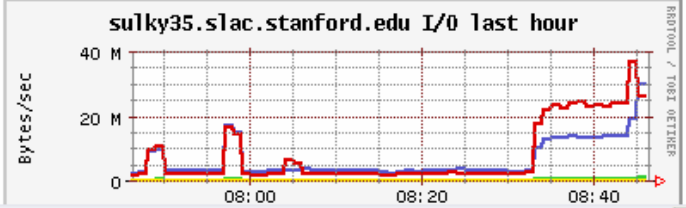
cpu_num	2 CPUs
cpu_speed	1503 MHz
mem_total	4010424 KB
swap_total	11805840 KB

Gmetrics









Post DC2

- **We will have a great dataset for future development!**
  - **55 days of simulated downlink to practise with**
  - **Simulate downlink frequency**
  - **Test Data Monitoring**
  - **Develop Quicklook**

DC3 Plans

- Planned for early calendar ’07
- Envisaged as a dry run for ISOC operations as far as data processing is concerned:
  - Calibrations
  - Instrument Diagnostics and Monitoring
  - Quicklook
  - Final pipeline
  - Product Delivery to the SSC (not by hand)
  - Most details to be worked out after DC2

NRL & Beamtest Support

- Run FastCopy to transfer files from each location to SLAC and fire up pipeline
  - Being tested now from CleanRoom Bldg 33 for FSW tests
  - Same machinery as I&T used once pipeline fired up
- NRL reuses everything created for I&T LAT commissioning
  - It all works
- Beamtest is similar, except
  - Maintain pre-FSW data formats
  - Standalone G4 simulation for beamline etc (done)
  - CU geometry (done)
  - Merge data from beamline components (awaiting decision from online on how it gets done)

Processing Numerology

- From Heather's background runs – GR v3r3p7
  - per 3-in-row (at least) triggered event

• Assume 300 Hz effective data rate, with x4 downlink increase and x3 from compression
• Assume all downlinked events kept
  •~1-2% good photons!
  • Must process all events to make selections

	Recon CPU	Merit size	MC size	Digi size	Recon size
Per event	0.15 sec	2 kB	15.2	1.3	17.1
Per day	1 kHrs	52 GB	692	42	474
Per year		18 TB	252	16	173

- Assume equal MC to data (totally arbitrary)
- L0 downlink (1.2 Mb/sec) is 5 TB/yr

~225 2004-era CPUs to turn around one downlink (≡ 5 hrs data) in one hour

Clearly we want to filter a lot of the background out very early in the process!

Resource Numerology

Dual CPU k$	Disk k$/TB	Tape k$/200 GB
2.5	4 now 2 by FY08	0.08

- CPU and disk is incremental each yr
  - roll over CPUs every 3 yrs
- only considering DATA here
- assumes archiving twice disk

	FY05	FY06	FY07	FY08
CPU	+20	+20	+20	+20
	50k	50k	50k	25k
disk	+32 TB	+40	+150	+200
	125k$	150k	600k	400k
tape	20k$	20	120	160

10% solution

	FY05	FY06	FY07	FY08
CPU	20	20	20	20
	50k	50k	50k	50k
disk	32 TB	40	40	40
	125k	150k	150k	80k
tape	20k$	20k	20k	20k
Total	195k$	220k	220k	142k

- base per flight year of L0 + all digi = ~25 TB
- then 10% of 300 Hz recon
- disk in 05-06 is for Flight Int, DC2

Current Compute Resources

In 2nd year of projected annual $300k Capital Equipment Projects
- Supplying, batch farm disk & CPU, as well as dedicated servers
- Optimize purchases based on best deals SCS can come up with

- 38 TB disk – almost entirely used up by
  - LAT Commissioning
  - DC2
  - Infrastructure needs (code builds; system tests; user disk)
- 40 TB on order for FY06

- Tremendous use of SLAC Batch farm!
  - 20 AMD Opteron dual core dual CPU boxes added to SLAC batch farm in GLAST's name
  - Have leveraged these into routine use of 400 CPUs
  - SCS wants us to use this model
    - Contribute boxes to the farm – they will 'guarantee' turnaround as if we had dedicated machines
    - The more we contribute the more assured we'll be of on-demand service

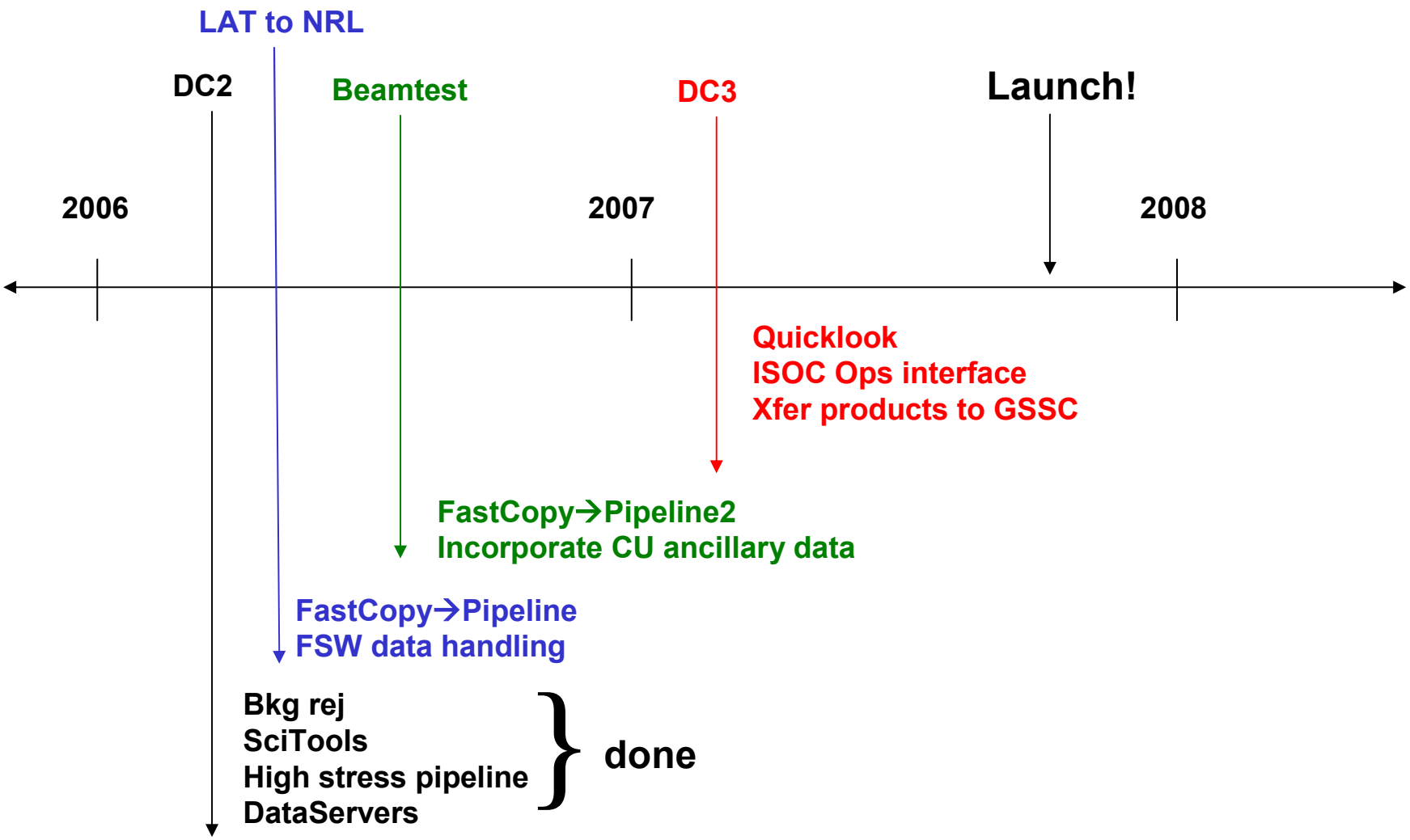
- Dedicated Oracle server and backup

- 14 special use linux servers
  - Mirrored mySql, FastCopy, Application servers, cvs, Jira/Confluence, etc etc
- 8 windows servers
  - Mirrored Web, code build servers

ISOC Development

- We have laid the groundwork and tested several components of the tools the ISOC will need:
  - Pipeline → backbone of ISOC processing operations
    - Pipeline 2 targeted for beamtest time use
  - System tests → high level data diagnostics
    - Data Monitoring targeted for beamtest use
  - Trending → use in calibrations production and monitoring
  - DataServers → access to data for follow-up examination if problems are flagged in Data Monitoring
  - Pipeline front end → web technology to provide interface for shift takers
- This is a non-trivial amount of work!

Timeline: Milestones



Manpower Projection – SLAC Only

Currently:
- 1 management
- 2 subsystem support
- 1.5 core software support
- 4.9 data handling

Starting cutting back 4 years after launch (late ’11) .

Complicated by availability/need of contributions from the collaboration.

A guess of course!