GLAST Large Area Telescope:

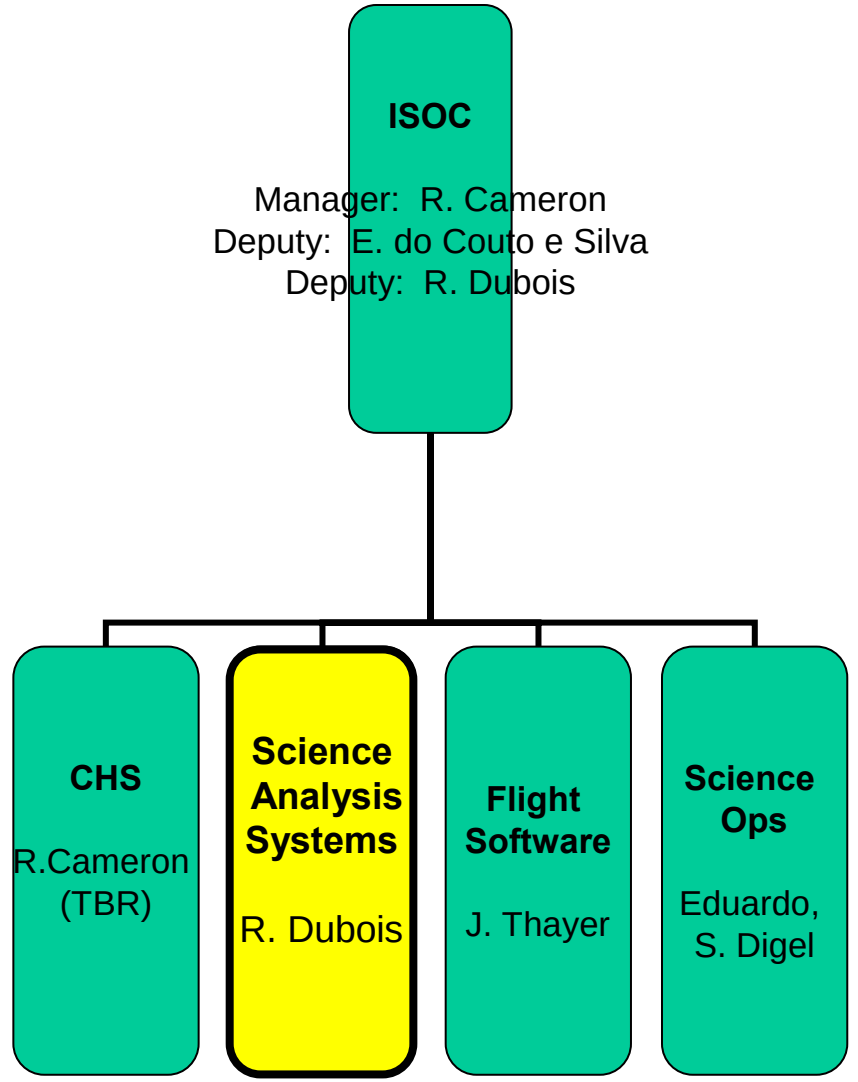
Science Analysis Systems
&
Data Challenge 2

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

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

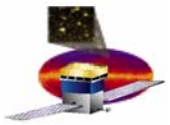
Outline

- SAS Purview
- Software Development Concept & Tools
- Reconstructing events
- High Level Science Tools
- I&T Support for Integration
- DataChallenges & DC2
- NRL & Beamtest 2006 Support
- Computing Resource Projections
- Building the ISOC Ground Operations Tools
- Major Milestones and Manpower Needs



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



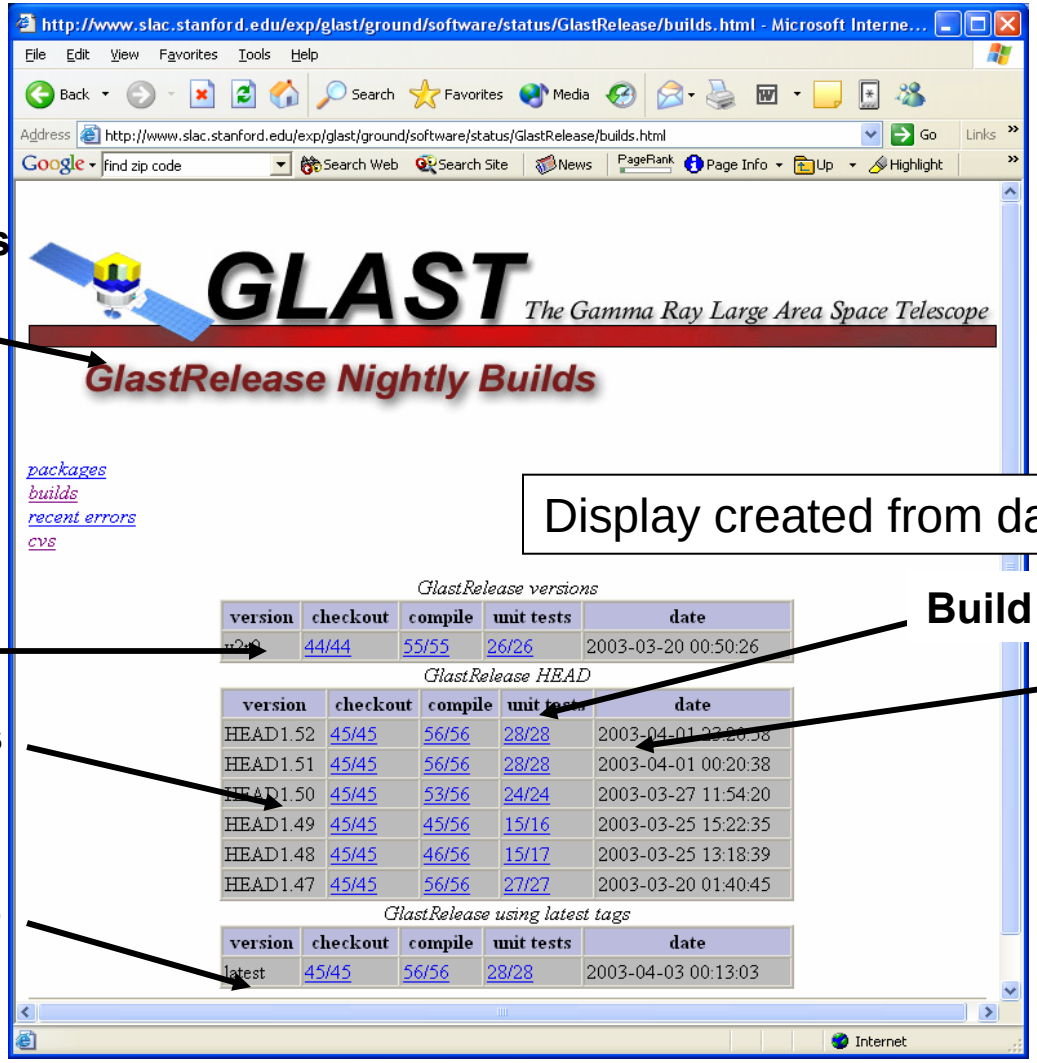
~30 FTEs total
10.5 from SLAC

C++ Software Development Approach

- Enable distributed development via cvs repository @ SLAC
- Extensive use of electronic communications
  - Web conferencing (VRVS), Instant Messaging (icq)
- Support Windows and Linux
- “Continuous integration” with code build tools
- “System Tests” for extensive code validation
- Simple installer for code, using automated build manager output
- Intuitive guis for running code
- User Workbook to support documentation for collaboration

Code Builds

Performing builds for Science Tools also



Display created from database query

Past release

Release in progress

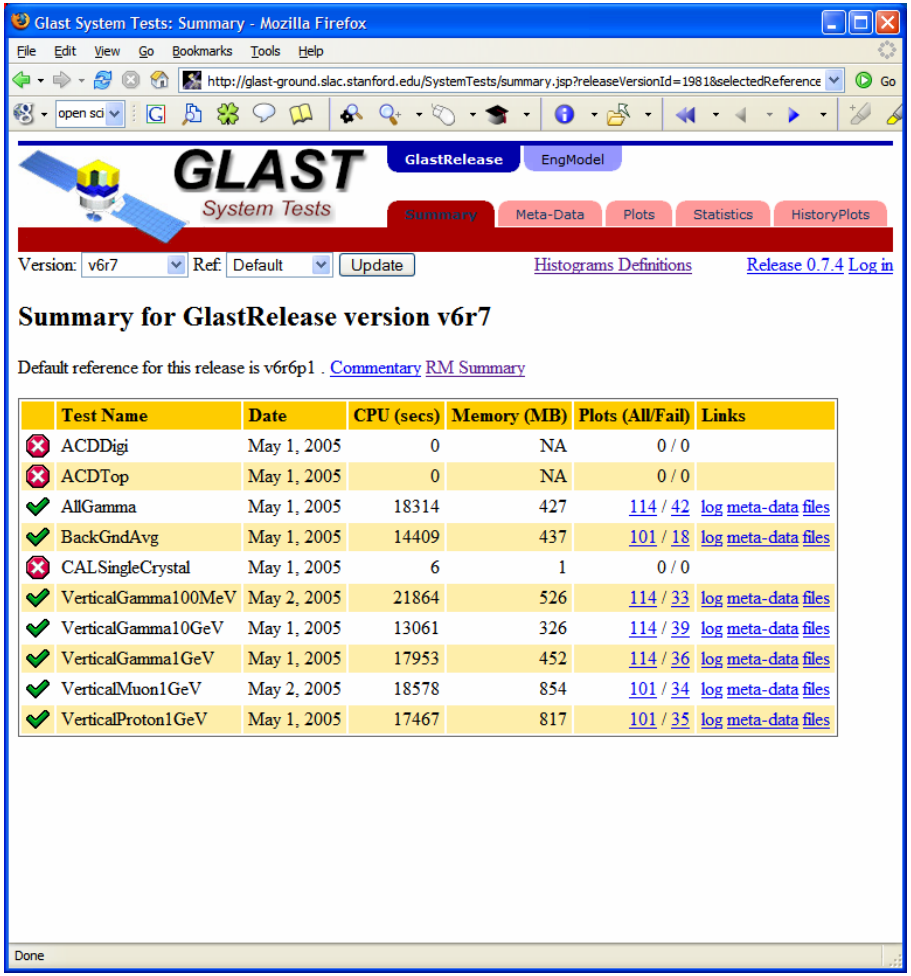
Future release

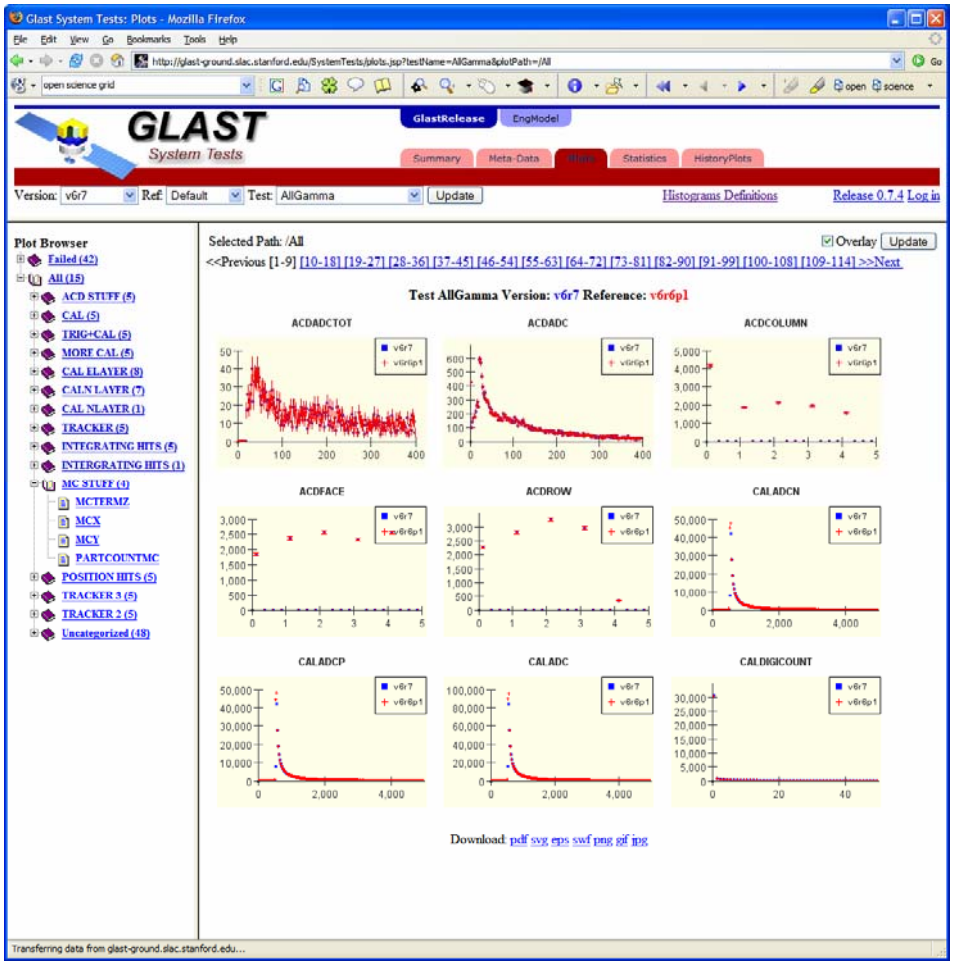
Build status

Unit test status

Alex

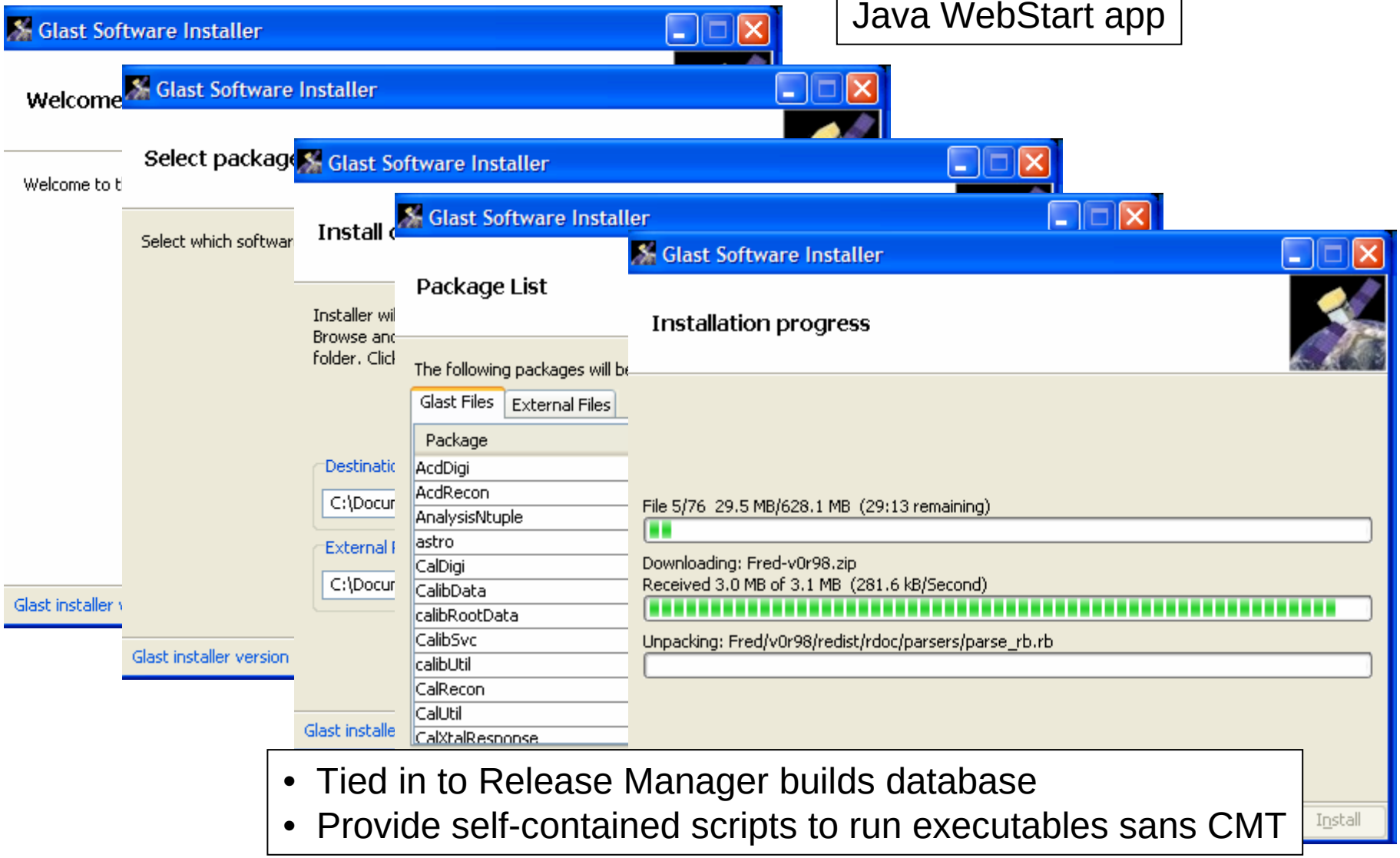
System Tests





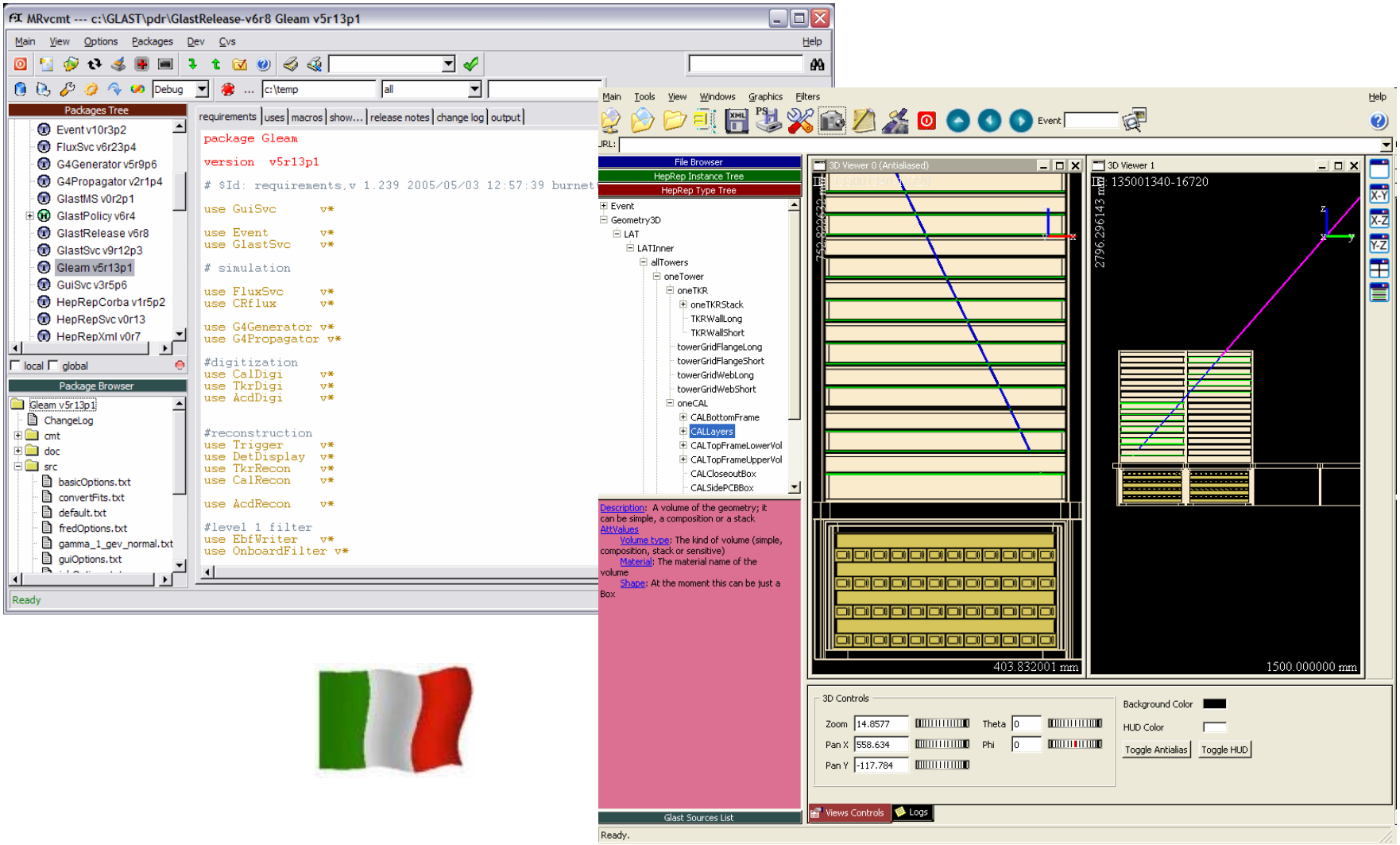
Code Distribution

Java WebStart app

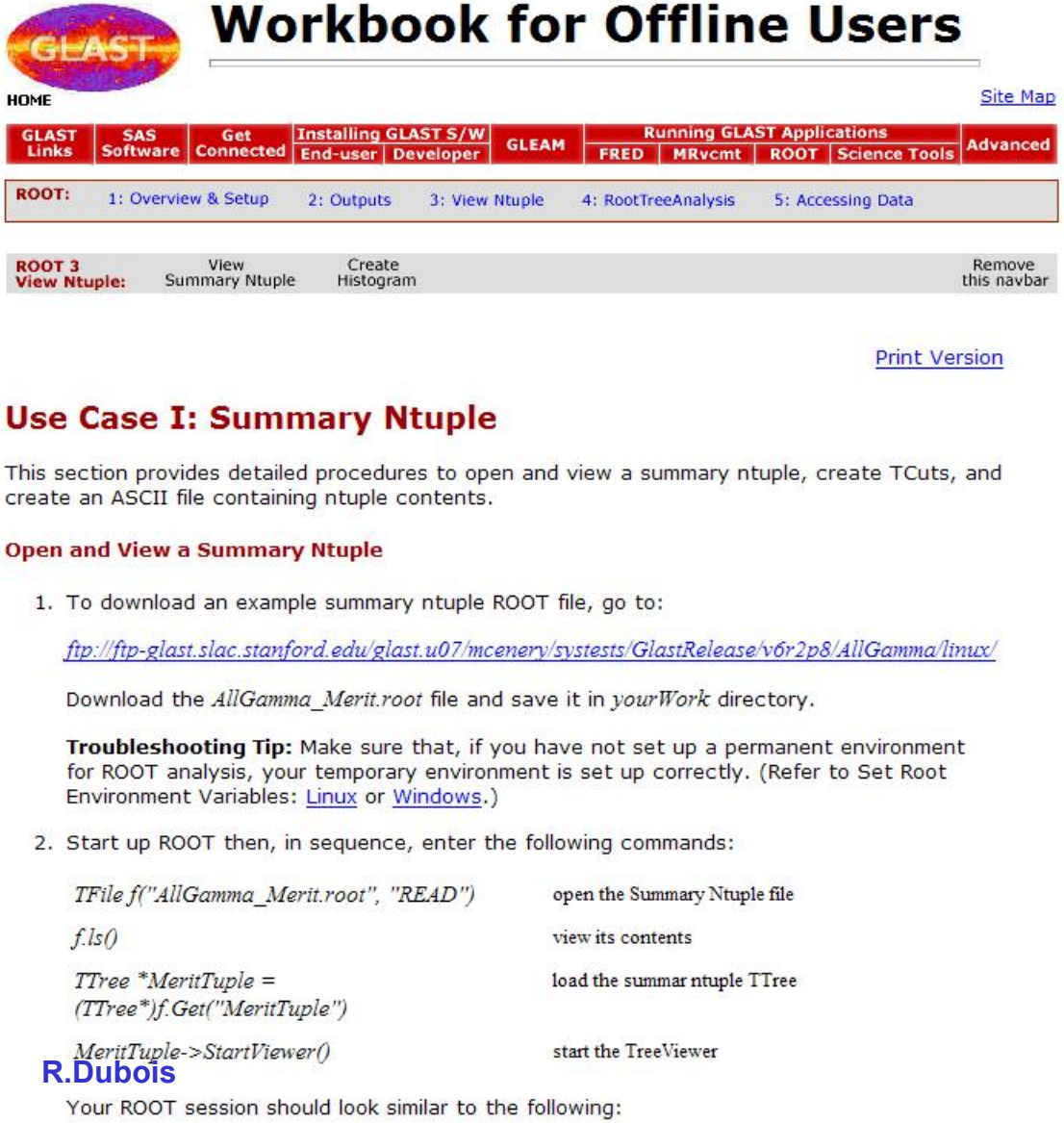


- Tied in to Release Manager builds database
- Provide self-contained scripts to run executables sans CMT

Code Development GUIs



Documentation: User Workbook



Workbook for Offline Users

HOME Site Map

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

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 …

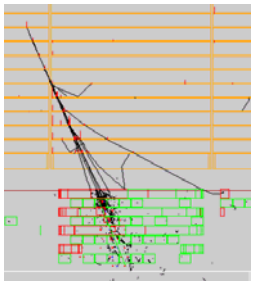
- work with Tech Writer
  - skilled at extracting information from us wackos
  - worries about layout, organization
  - can write good
- we're struggling with apparent conflict of web navigation vs "printed book". Pursuing the former.

Sim/Recon Toolkit

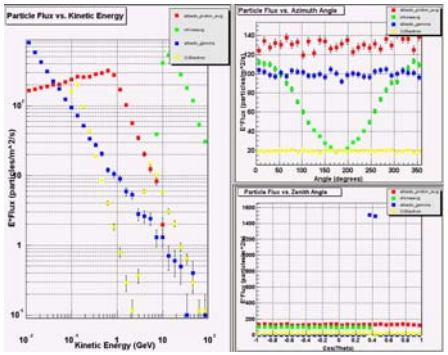
Package	Description	Provider	Status
ACD, CAL, TKR Recon	Data reconstruction	LAT	90% done In use
ACD, CAL, TKR Sim	Instrument sim	LAT	95% done In use
GEANT4	Particle transport sim	G4 worldwide collaboration	In use
xml	Parameters	World standard	In use
Root 4.02.00	C++ object I/O	HEP standard	In use
Gaudi	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

Instrument Simulation and Reconstruction

3 GeV gamma interaction



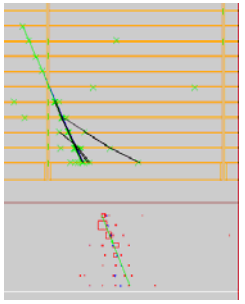
Source Fluxes



Particle Transport

"Raw" Data

Instrument data

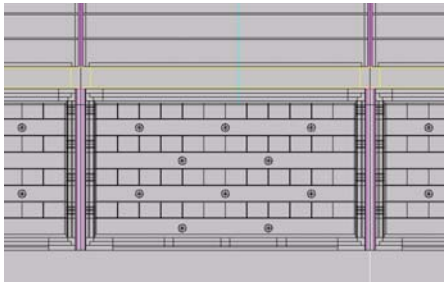


3 GeV gamma recon

Recon

Background Rejection - Particle ID

Geometry



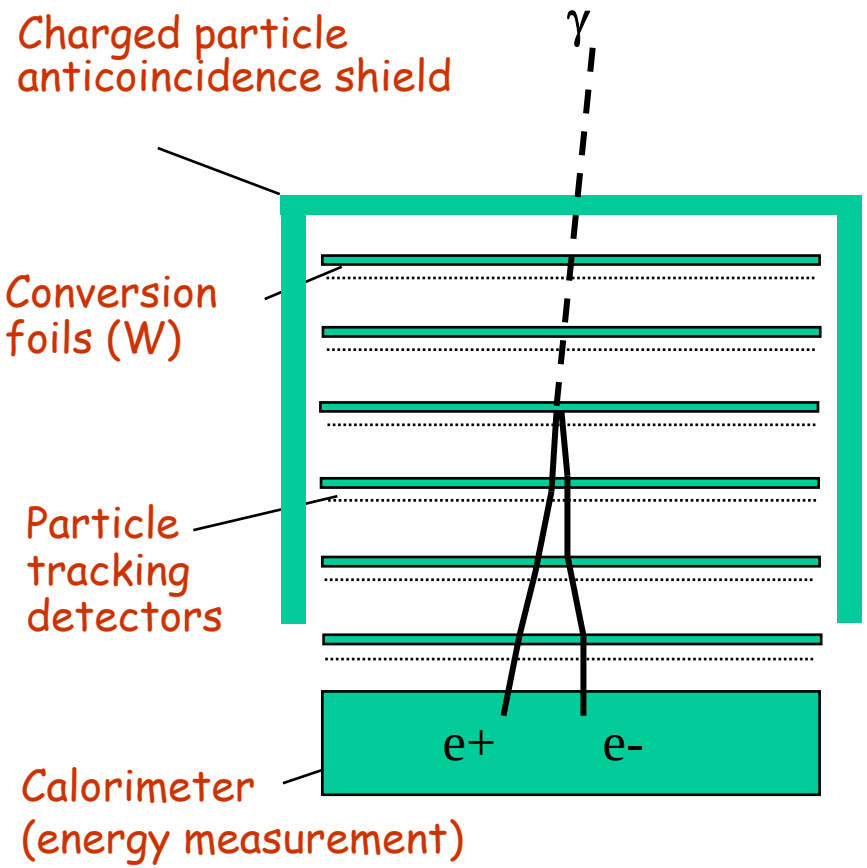
CAL Detail

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

GLAST Reconstruction

Anatomy of a "Typical" Event

Pair production is the dominant photon interaction in our energy range

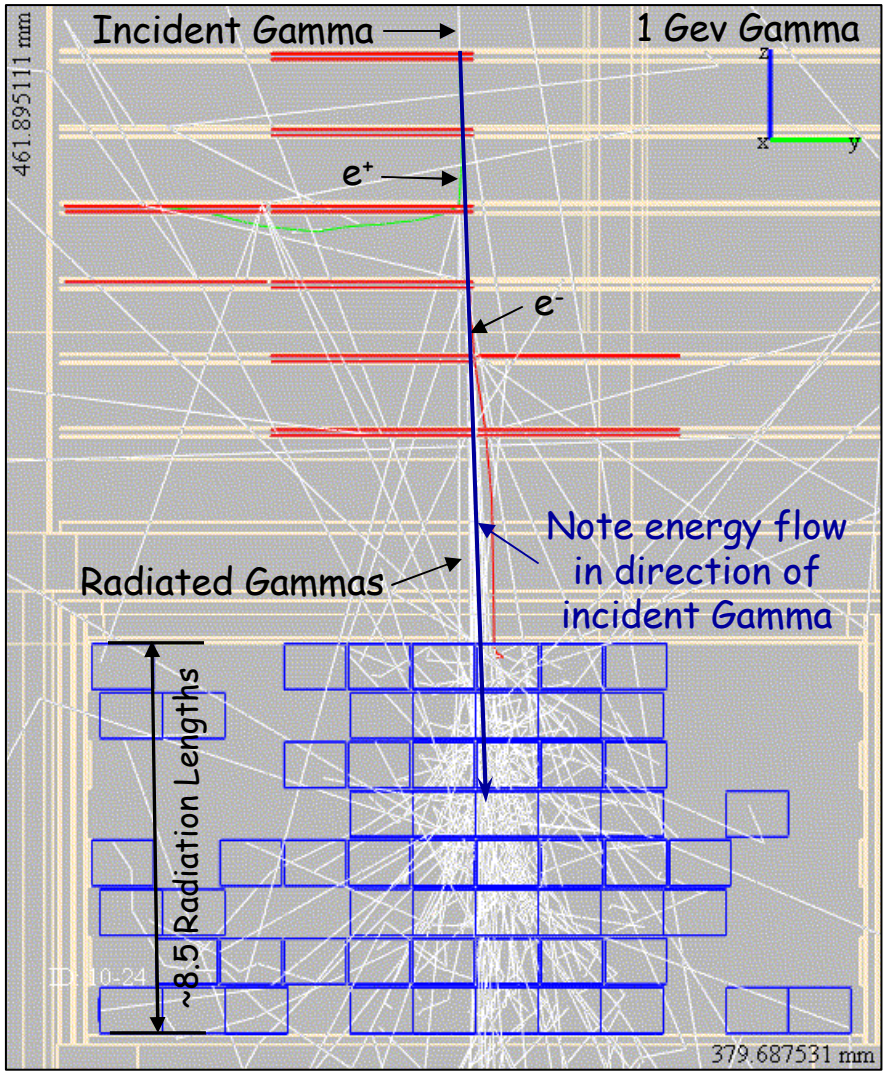


- **Reconstruction Goals:**
  - **Incident Gamma Direction and Energy**
  - **Reject Backgrounds**
- **Incident Gamma converts in the tracker**
  - **In particular, conversion occurs in one of the converter foils – ie at a well defined location**
- **Resulting electron-positron pair range out of tracker (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

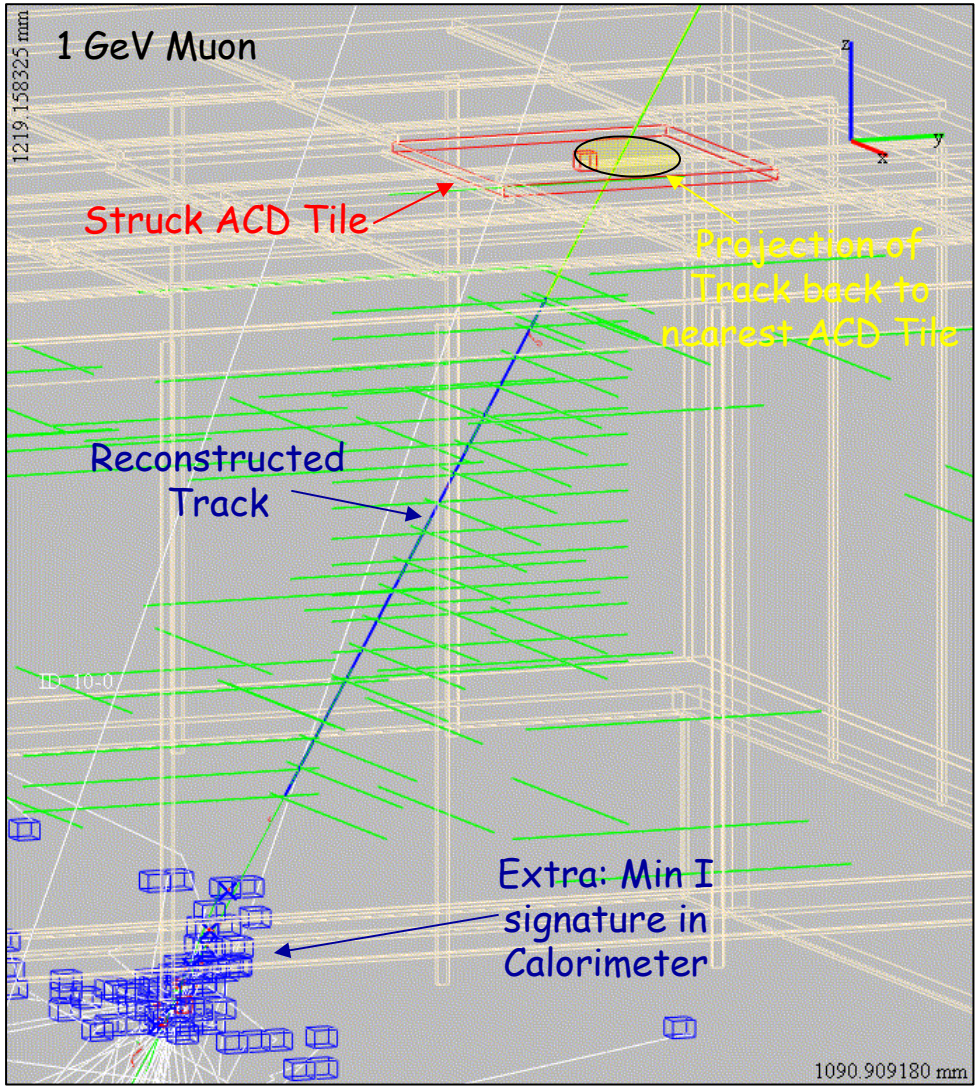
What makes it challenging…

- **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

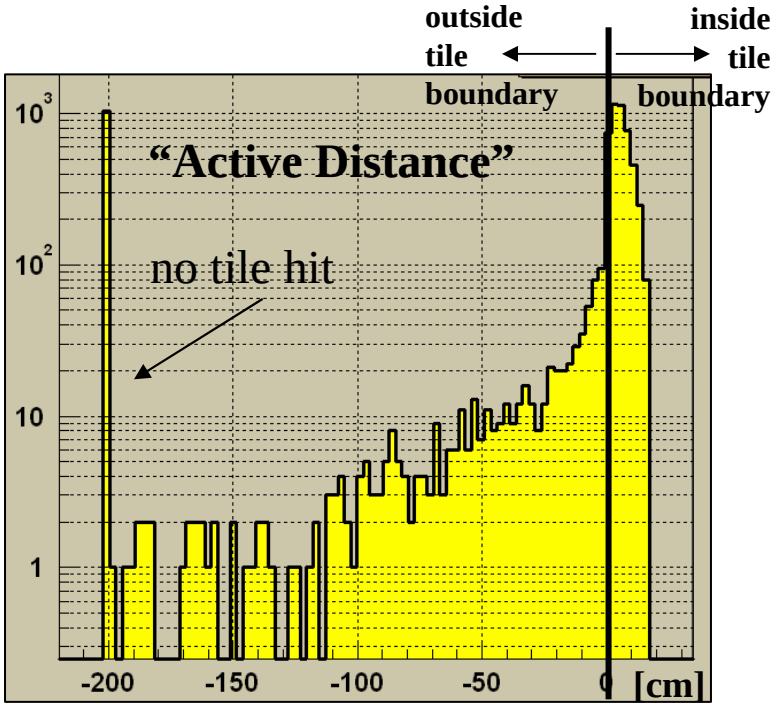


Background Rejection

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



Science Tools

- The 'Science Tools' are the high-level analysis tools for astrophysics
- 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)

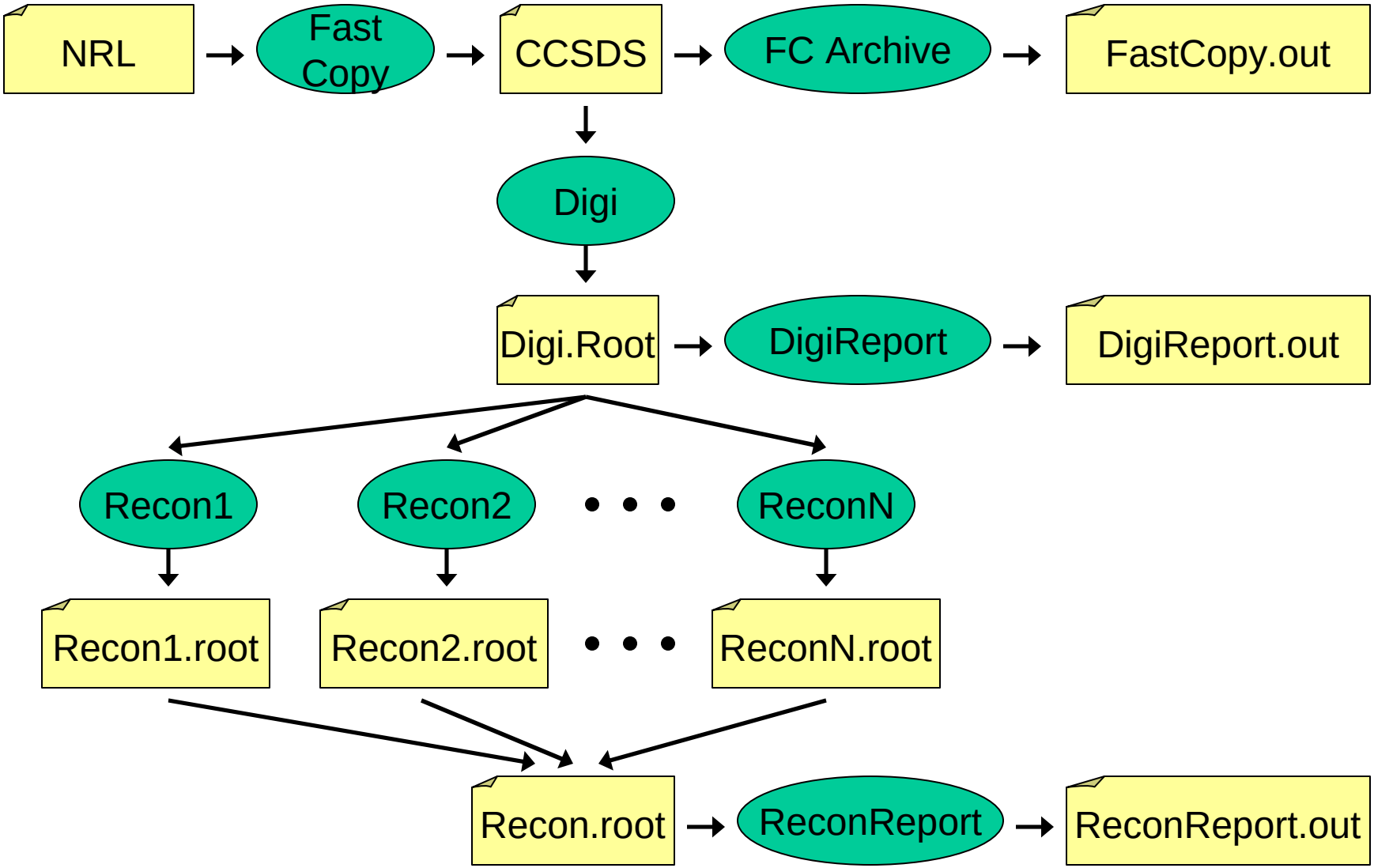
- After a period of definition and review, the tools have been developed incrementally, with the milestones for evaluation
  - Data Challenges (see later) as major milestones and 'Science Tools Checkouts' (3 so far) as intermediate ones
- The core Science Tools are
  - gtlikelihood, gtexpmap, and numerous 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.

Automated Pipeline Processing

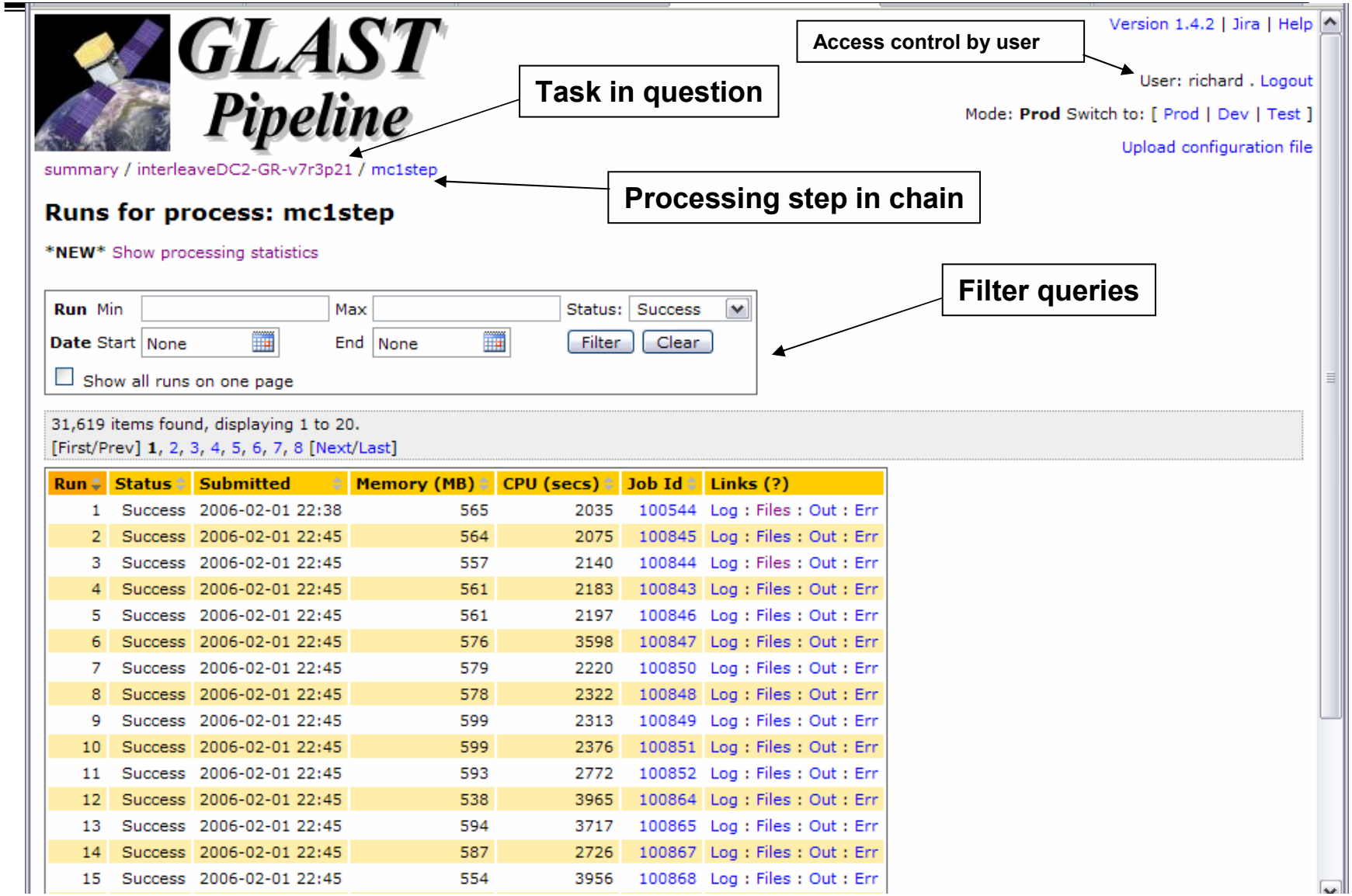
- **What is the pipeline?**
  - Envisaged as tool to provide a tree of processing on a given input dataset
  - Handle multiple "tasks" concurrently, eg LAT commissioning, DC2 Monte Carlo runs
  - Full bookkeeping to track what happened
  - Archive all files touched

- **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: Level 1 and 2
    - environmental testing, at Spectrum Astro, KSC
    - Data reprocessing

Sample Processing Chain



Web Monitoring of Pipeline Progress

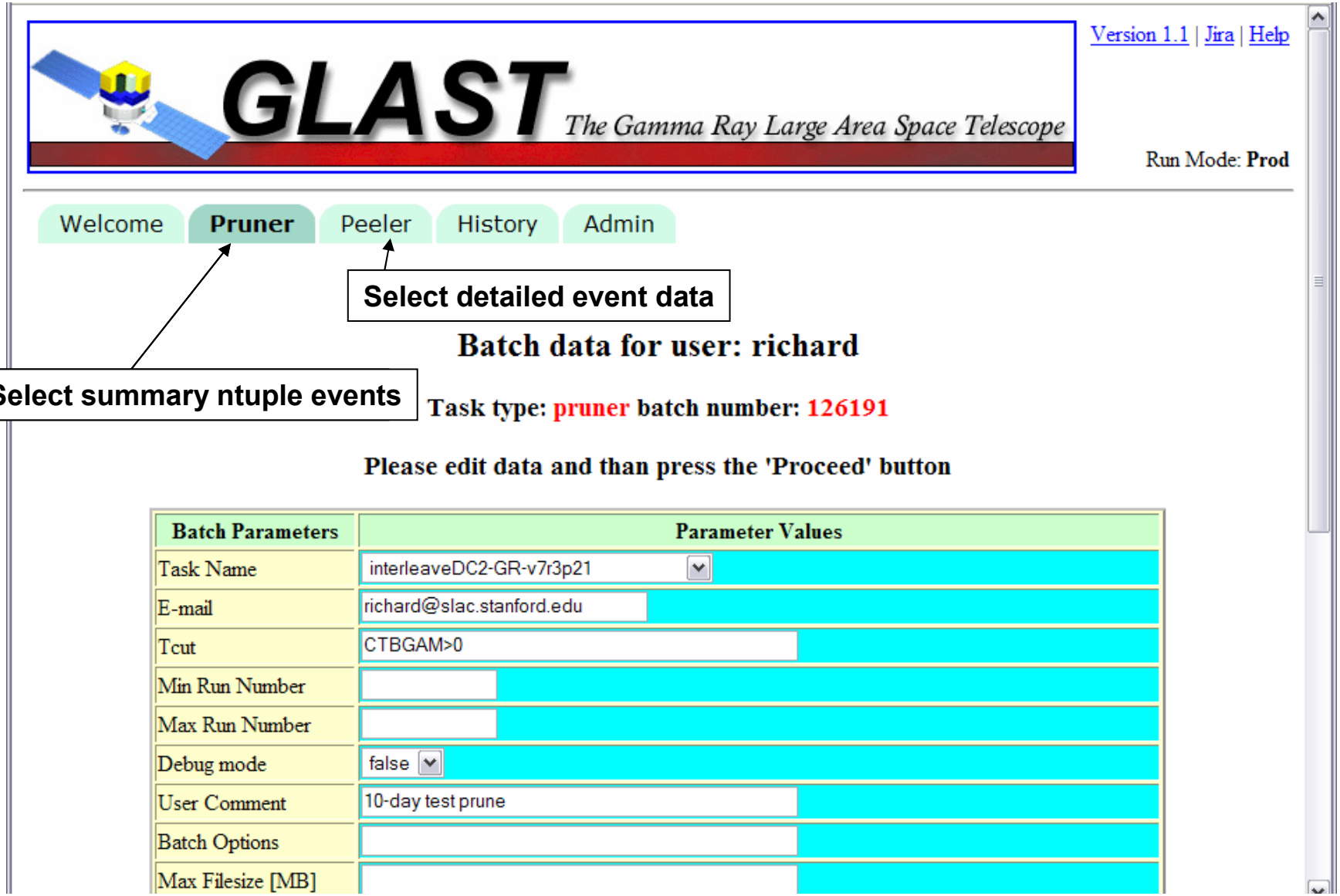


Run	Status	Submitted	Memory (MB)	CPU (secs)	Job Id	Links (?)
1	Success	2006-02-01 22:38	565	2035	100544	Log : Files : Out : Err
2	Success	2006-02-01 22:45	564	2075	100845	Log : Files : Out : Err
3	Success	2006-02-01 22:45	557	2140	100844	Log : Files : Out : Err
4	Success	2006-02-01 22:45	561	2183	100843	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 parameterized sub-tasks, with number of sub-tasks defined at runtime
    - Perhaps support conditionals 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 (to get rid of absolute paths)
  - Support reprocessing of data without redefining tasks
    - 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

Instrument Data Access



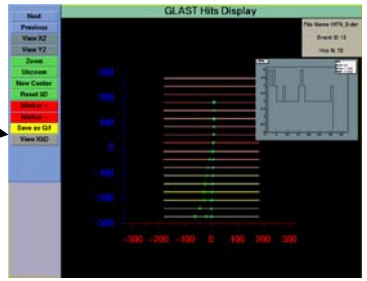
Astro Data Server

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

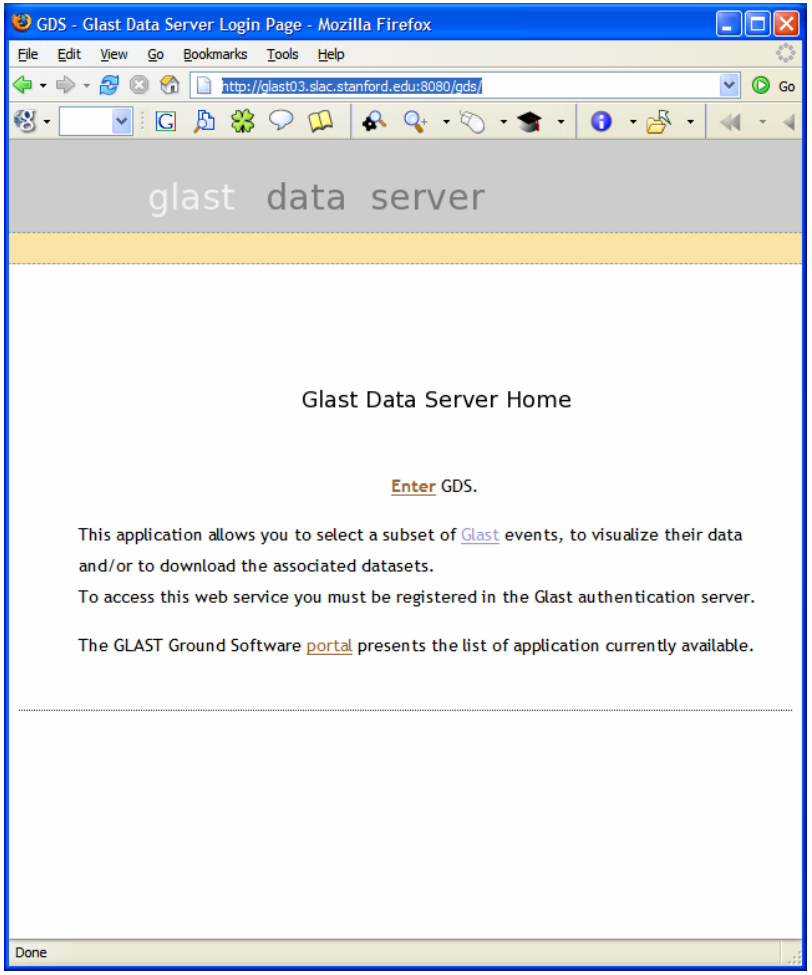
You selected 39383844 events
Change Criteria
Add “TCut”
Browse Events
Download: Event Selection
Compressed Tuple
Full Merit Tuple Full Root Tree

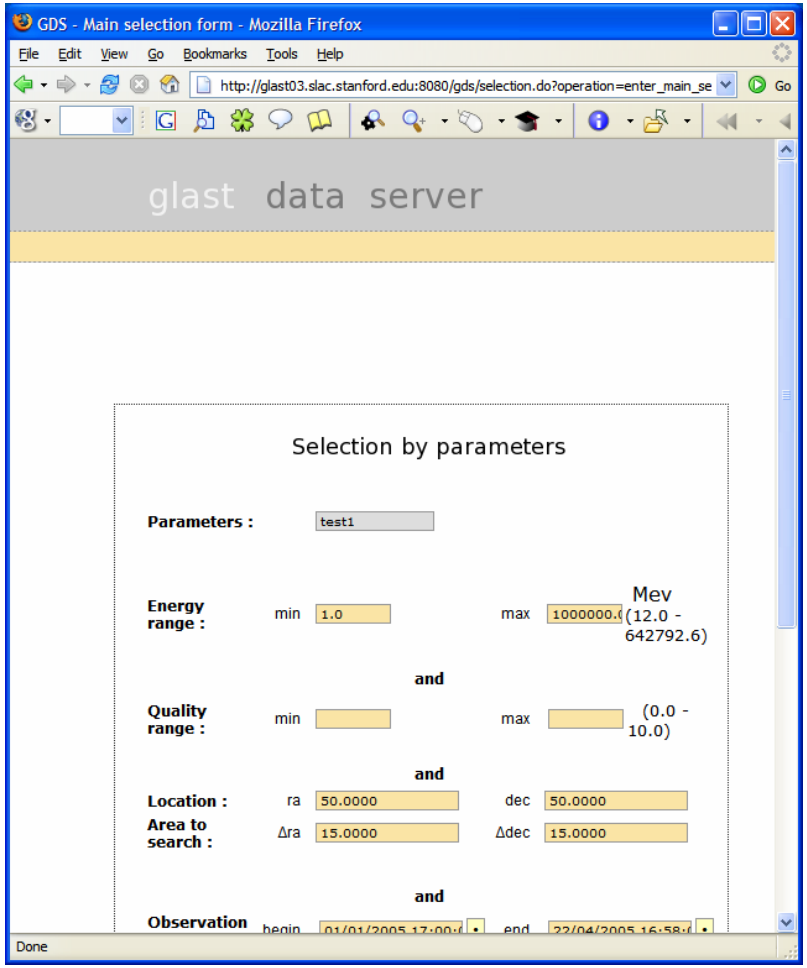
In memory meta-data
Binned by sky position, time, energy

Root Event Store

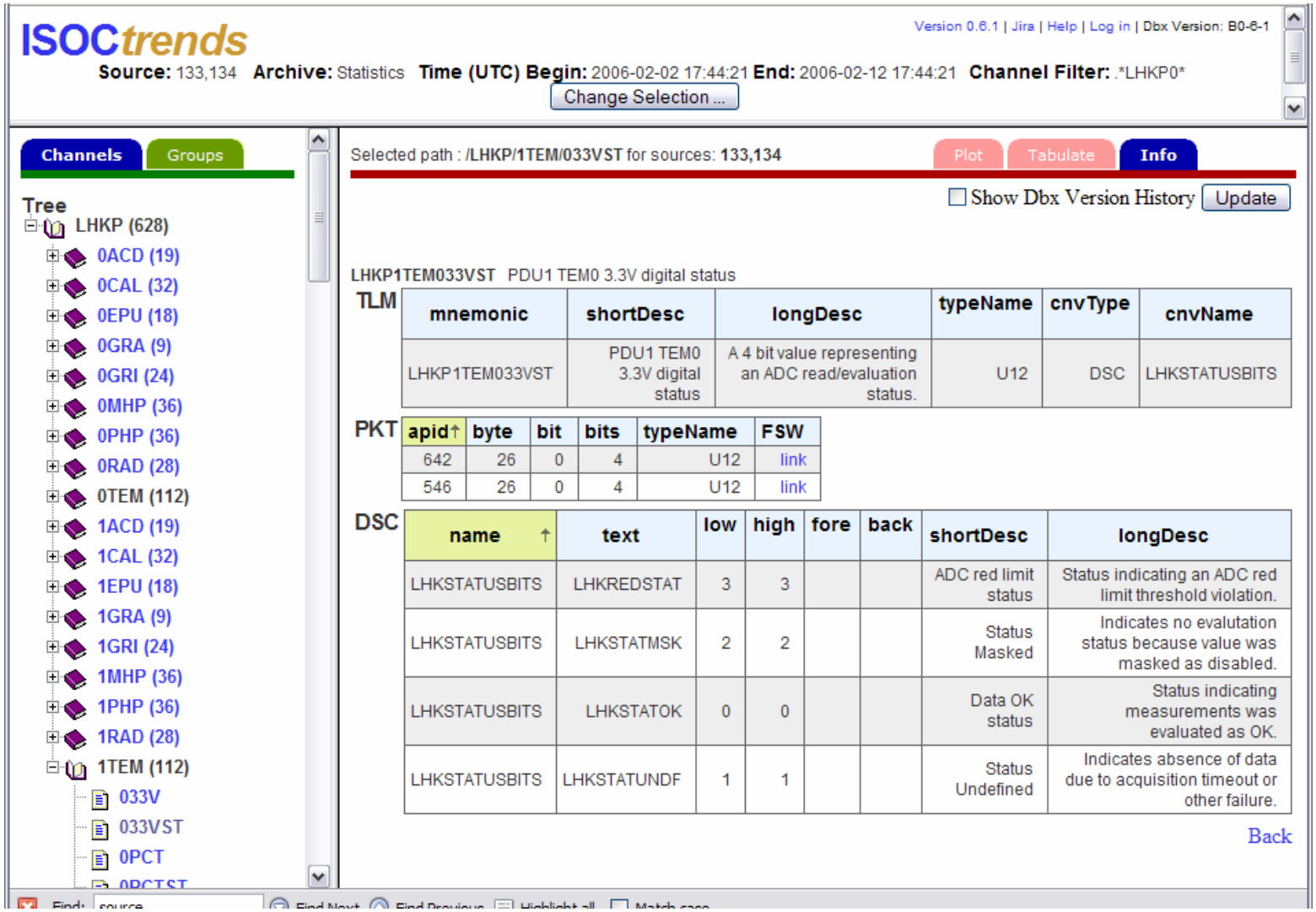


Astro Data Server





Trending Application



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 interactions with the rest of the collaboration

Data Challenges: Three Rounds

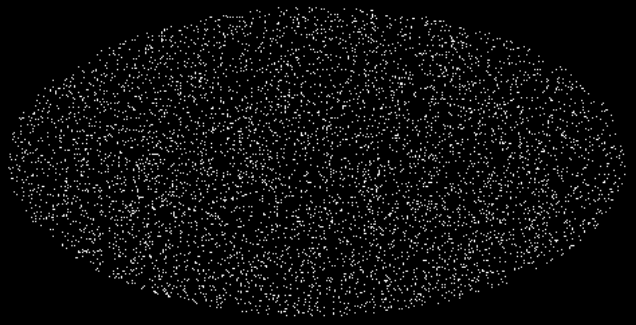
- DC1. Modest goals. Contains most essential features of a data challenge.
  - 1 simulated day all-sky survey simulation
  - find GRBs
  - recognize simple hardware problem(s)
  - a few physics surprises
  - Exercise all the components
- DC2, kickoff Mar 1. More ambitious goals. Encourage further development, 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.
  - ISOC Ops rehearsal

DC1 Components

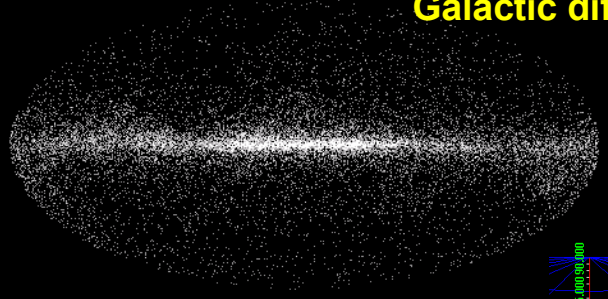
- Focal point for many threads
  - Orbit, rocking, celestial coordinates, pointing history
  - Plausible model of the sky
  - Background rejection and event selection
  - Instrument Response Functions
  - Data formats for input to high level tools
  - First look at major science tools – Likelihood, Observation 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

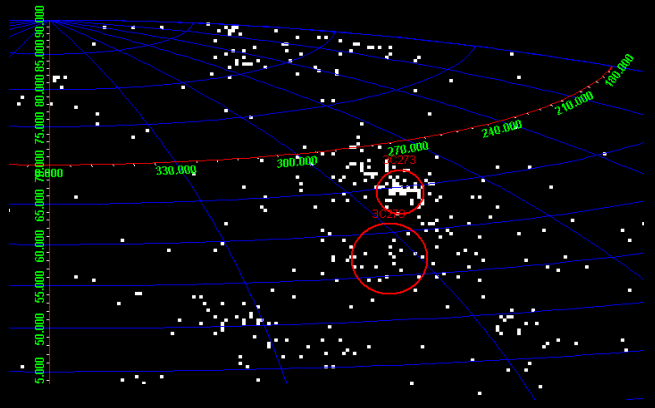
Extragalactic diffuse

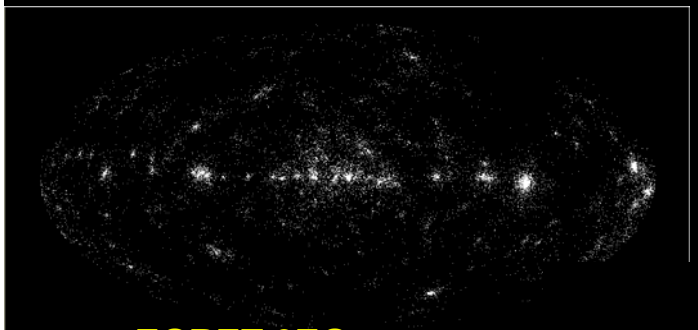


Galactic diffuse



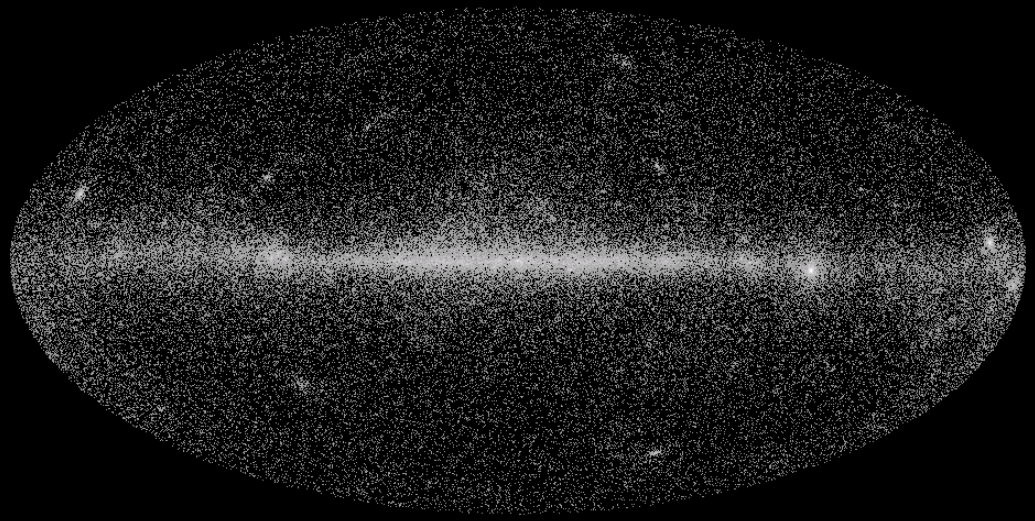
Fiddling 3C273/279





EGRET 3EG

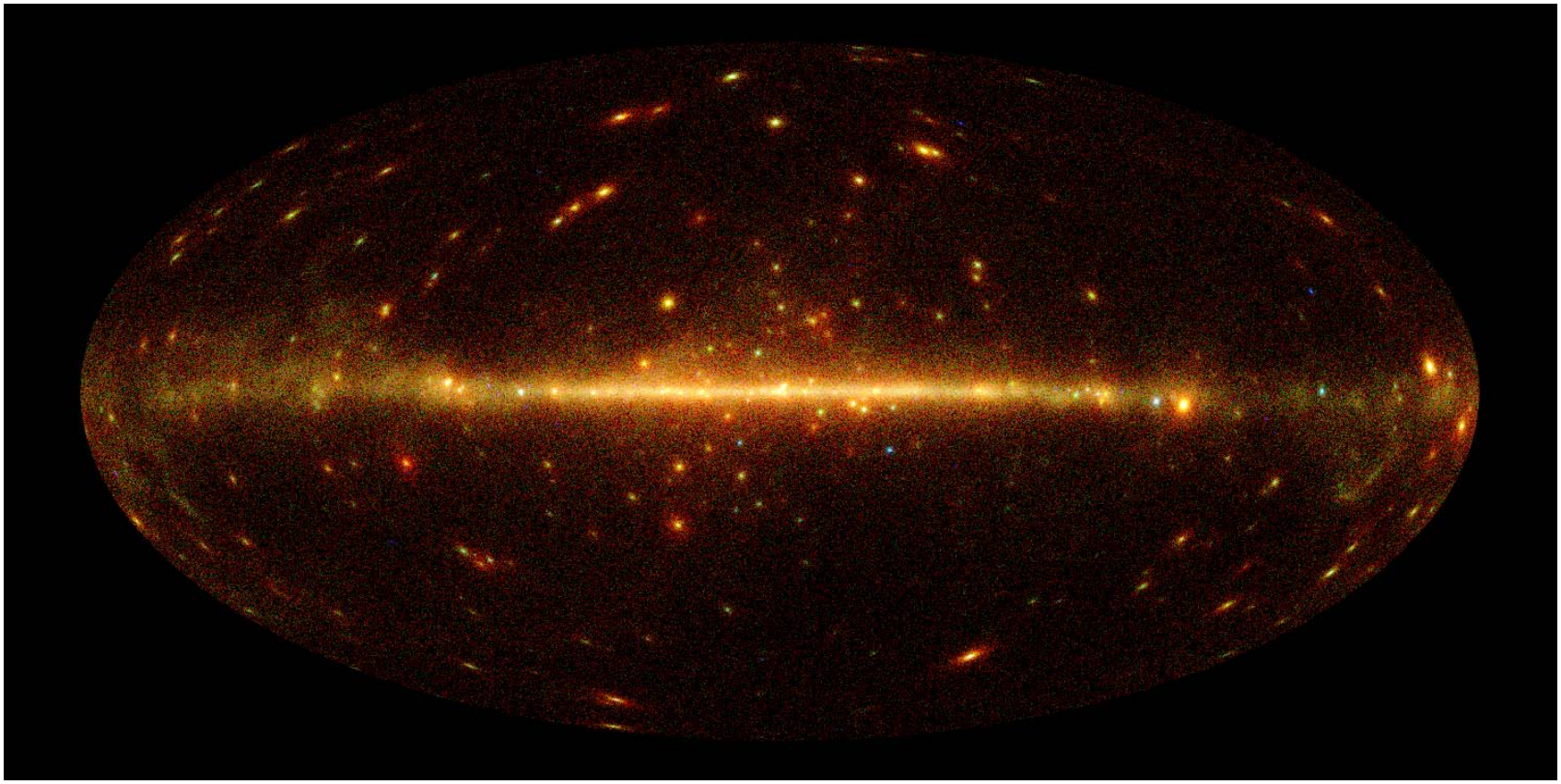
Our Sky



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
- Major rewrite of reconstruction
- Full suite of astrophysics analysis tools
- Source catalogue
- 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

DC2 Sky Simulation



Much richer sky than DC1!!

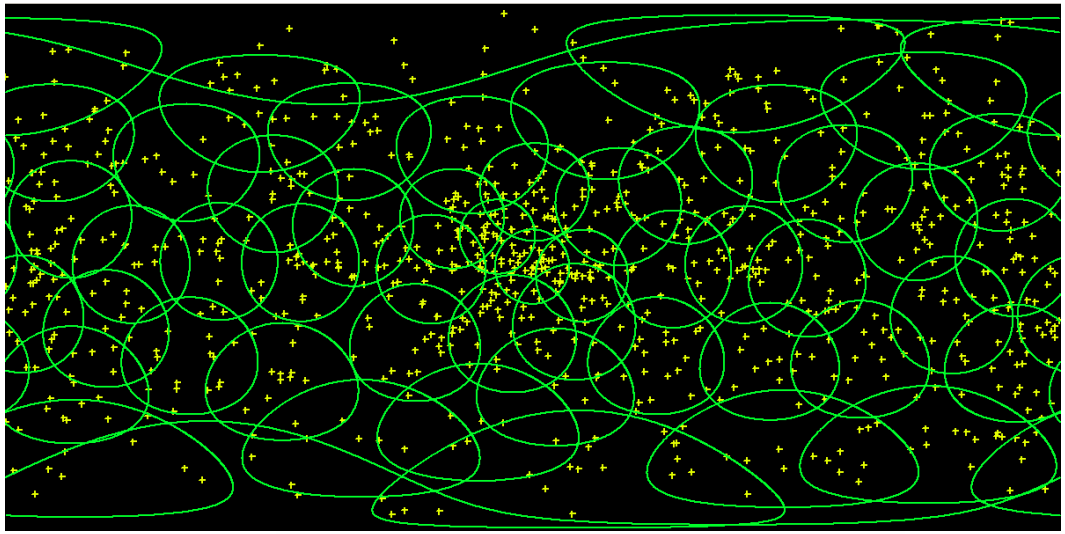
DC2 Kickoff Meeting 1-3 March

112 registered attendees!

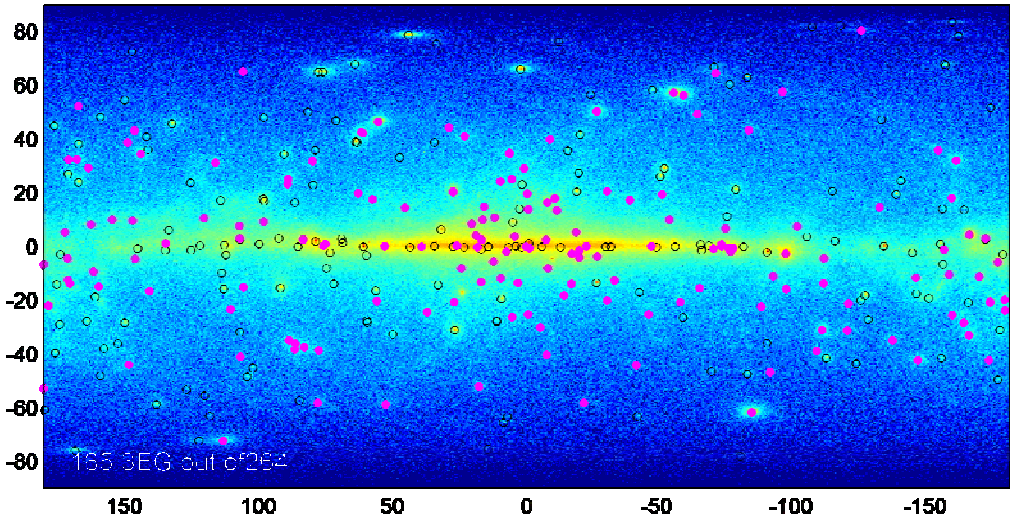
France: 13
Italy: 17
US: 71
Japan: 5
Sweden: 2
Germany: 4 (GBM)

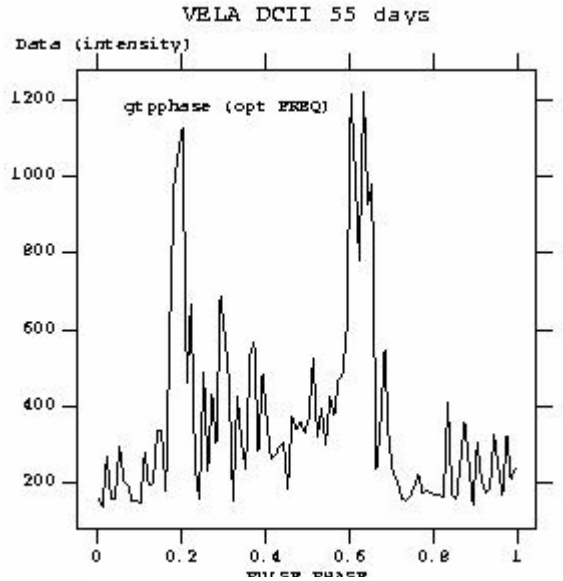
Sample studies underway on DC2 Data

Preliminary source catalogue



DC2 photon map and preliminary sources

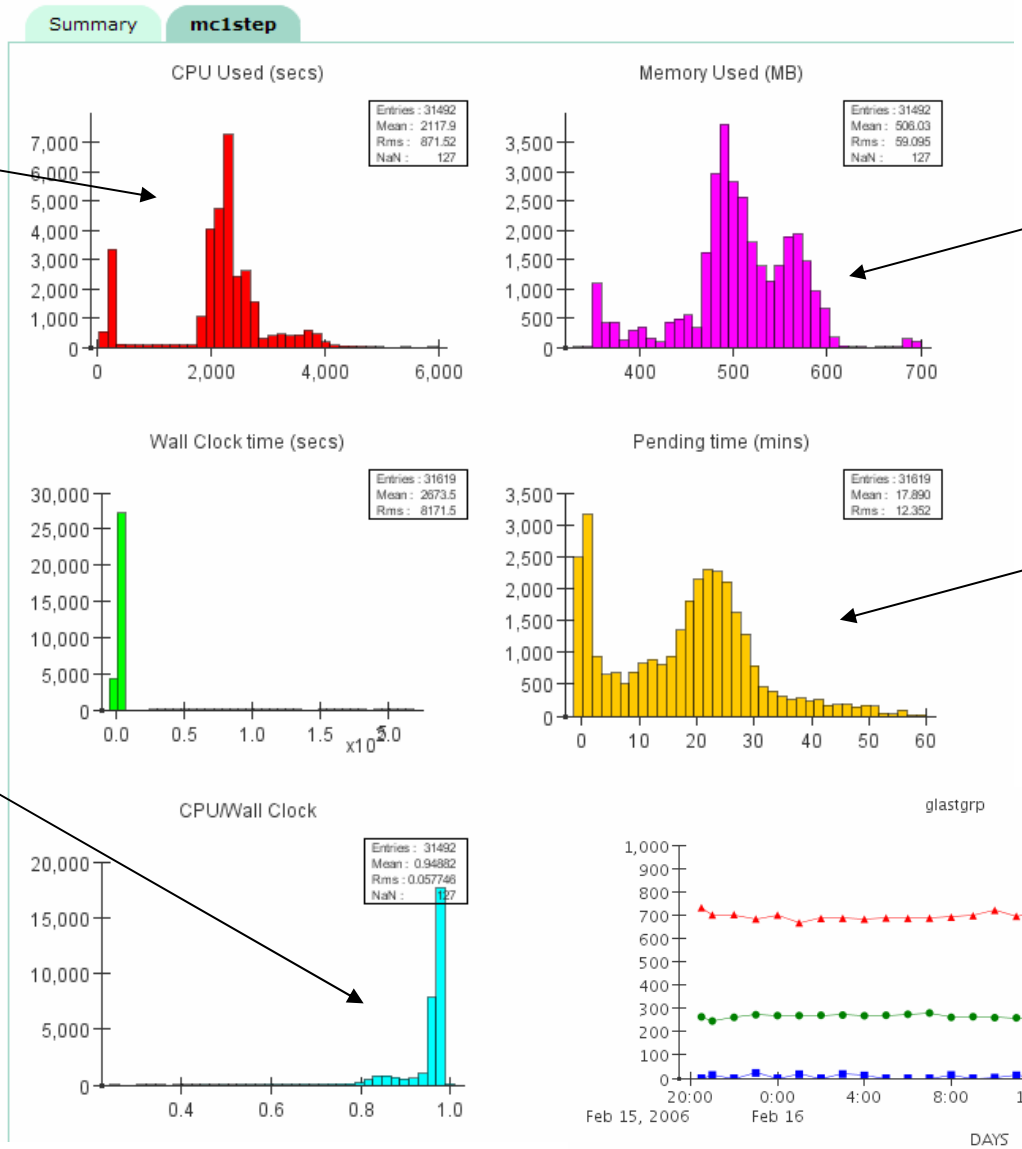




Vela pulse phase

Monitoring DC2 Pipeline Throughput

Pipeline Statistics



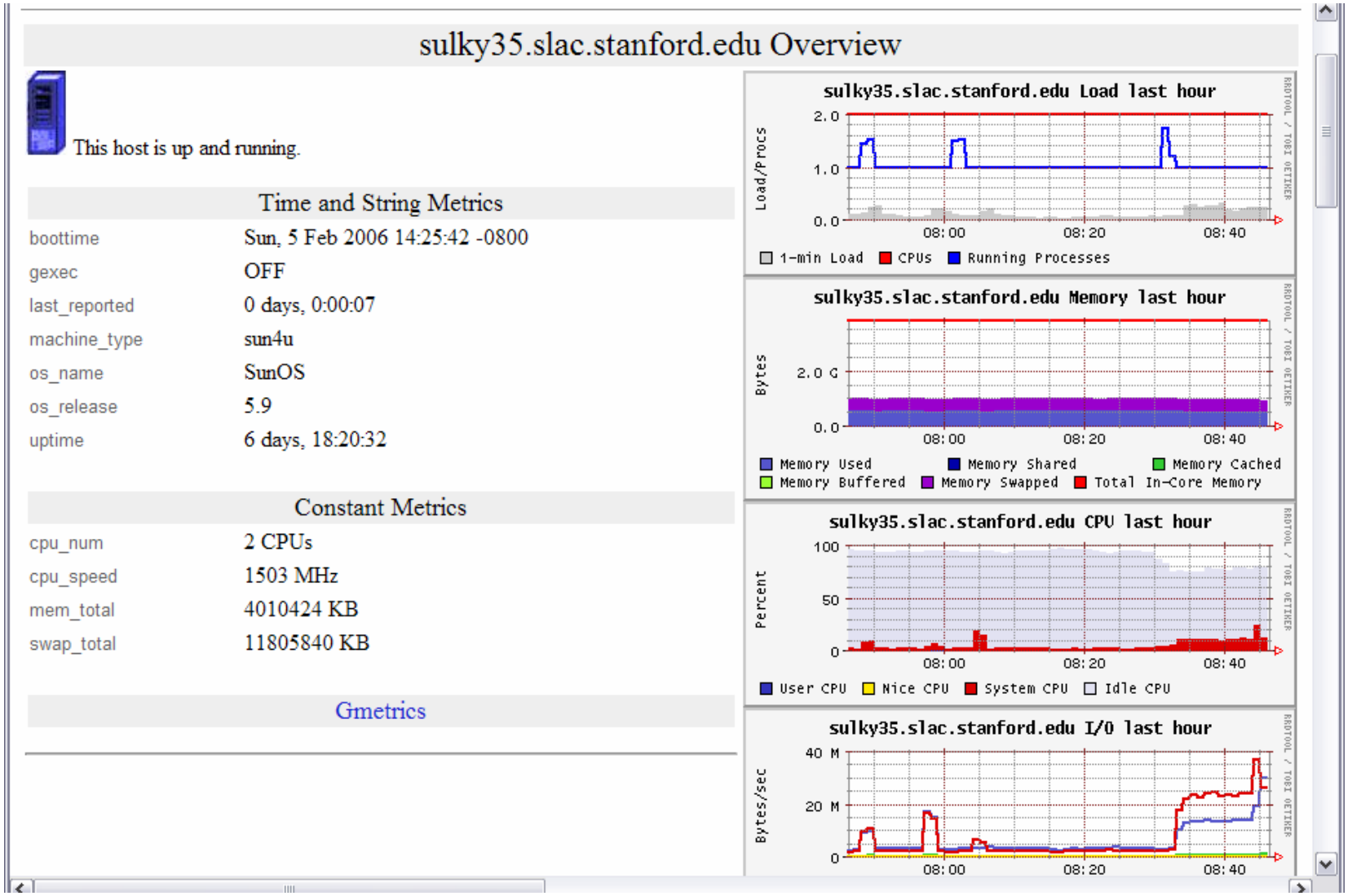
CPU time

Memory

"Wait" time for jobs

Ratio wall clock to CPU

Monitoring Disk Farm via SCS Tools



Post DC2

- **We now 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
  - Full Product Delivery to the SSC
  - Most details to be worked out after DC2

Env Test & Beamtest Support

- Run FastCopy to transfer files from each location to SLAC and fire up pipeline
  - Being used 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 DC2 background runs – GR v7r3p5
  - Events that pass Onboard Filter

- Assume 450 Hz effective data rate
- 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	0.06 sec	0.5 kB	28	1.5	8.6
Per day	650 Hrs	19 GB	1100	58	333
Per year		7 TB	252	21	121

- Assume equal MC to data (totally arbitrary)
- L0 downlink (1.2 Mb/sec) is 5 TB/yr
- ~80 2006-era CPUs to turn around one downlink (≡ 3 hrs data) in one hour

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

Resource Numerology

PC k$	Disk k$/TB	Tape k$/200 GB
3	4 2-3 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	+60	+150	+200
	125k$	220k	600k	400-600k
tape	k$		120	160

10% solution

	FY05	FY06	FY07	FY08
CPU	20	20	20	20
	60k	60k	60k	60k
disk	32 TB	60	40	40
	125k	220k	150k	80k
tape			20k	20k
Total	205k$	280k	230k	152k

Would be prudent to budget for ~60+ TB/yr and plan on continued price drops

May keep all on disk for FY07 to be cautious

We're investigating options to optimize disk vs tape with technologies developed for BABAR

Storage Options

- 10% Solution:
  - 60 TB/yr @ ~$240k for disk; ~$50k tape (or $120k if all recon kept on tape)
  - Assumes we can safely cut 90% of the background
- Full disk in ’07; 10% in ’08 and onwards (add 25 TB contingency to 150)
  - ~175 TB overall @ $700k for disk; $140k for tape
  - Free up 110 TB in ’08 which covers ’08 and ½ of ’09
- No recon on disk
  - ~30-40 TB/yr @ 120-160k$ disk; 25-35 k$ tape
  - suffer latency in retrieval when needed
    - 1 TB retrieved in 1 day currently
    - Affects reprocessing merit; event displays; calibrations
    - Is there some need we’ll find for fast turnaround that would make us regret not having things right there? Don’t know yet.
- Wrinkles
  - Background level still not well known – could be yet higher (though the downlink rate is fixed, so maybe not so bad)
  - We are optimizing data formats – could be up to 40% gain in recon size possible
  - BABAR’s xrootd system may make it much easier to settle on disk/tape splits. Under study now.
  - Price of disk will drop. How soon and how much?

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

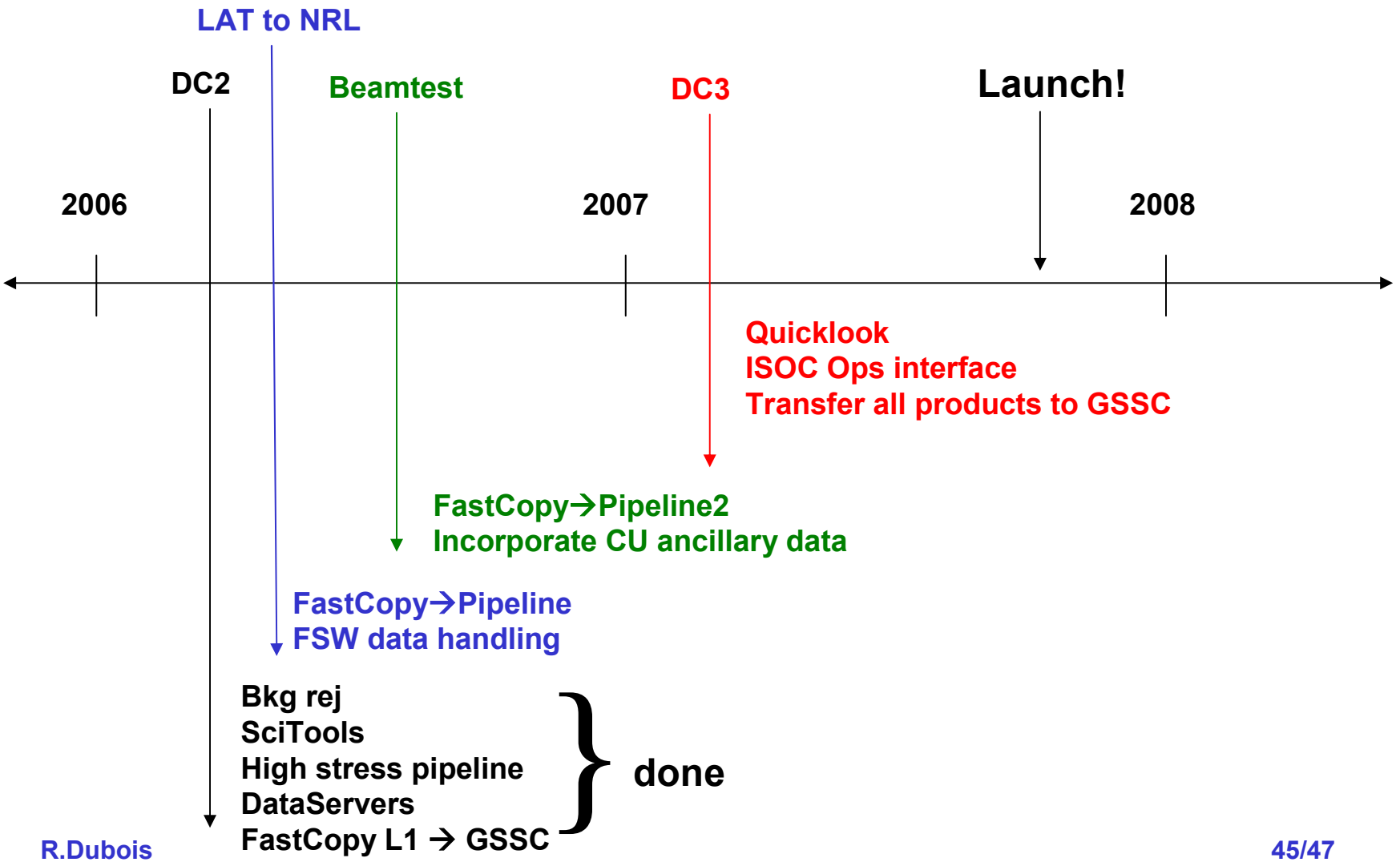
- 38 TB disk – almost entirely used up by
  - LAT Commissioning
  - DC2
  - Infrastructure needs (code builds; system tests; user disk)
- 60 TB on order for FY06

- 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

Timeline: Milestones



SAS Manpower Needs

- Already working closely with the collaboration – 20 of our ~30 FTEs come from outside SLAC!
- Could not be done without this group-wide effort
- Quite a bit of ebb and flow in who does what
- Contributions outside SAS/SLAC:
  - Italy:
    - Core tools, G4 support – 2,5 FTE
    - TKR alg development – 1 FTE
    - SciTools development – 2,5 FTE
  - France:
    - Core tools: 0,5 FTE
    - CAL sim/recon/calibs – ~5 FTE
    - Source Catalogue – 3 FTE
  - US:
    - Core tools – 3 FTE (UW, Goddard)
    - CAL sim/calibs – 2 FTE (NRL)
    - ACD – 1 FTE (Goddard, SVAC)
    - TKR – 1 FTE (UCSC)
    - Sundry contributions
- Responsibilities outside SAS/SLAC:
  - Instrument code release manager - UW
  - SciTools release manager - GSSC
  - Builds manager – Goddard
  - System tests manager – Goddard
  - Documentation - Goddard
  - Code build tool (CMT) – LLR/France
  - Build tool gui & Event display – Udine/Italy
  - Source Catalogue – CEA/France
- Responsibilities inside SAS/SLAC:
  - Overall management
  - Data handling
  - Calibrations infrastructure
  - Overall Reconstruction; TKR
- Shortfall:
  - ~ 1-2 FTE for infrastructure work
  - Hardest to get assistance here

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 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**