Flight I&T Support

March 17, 2005
From Our Friends in I&T

- I&T Analysis meeting tomorrow 8 AM PST
- I&T Analysis Workshop
  Was last week – you already missed it…
- SVAC Code Review…
- When does data taking start?
EngineeringModel

- Latest EM tag: v4r060302p4
  - Based on GR v6r3p2
  - System Tests underway
  - Has new CalRecon, new LDF, stores lengths of all diagnostic & error contributions, latest calibUtil (I promise)
<table>
<thead>
<tr>
<th>What</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error bits: OR for TEM, trgParityError, and packet error</td>
<td>DONE</td>
</tr>
<tr>
<td>Handle full Error Contribution</td>
<td>Coming soon</td>
</tr>
<tr>
<td>New CAL code</td>
<td>DONE – but waiting for patches for 4 bump problem</td>
</tr>
<tr>
<td>Update LdfReader with more error checking</td>
<td>Somewhat done</td>
</tr>
<tr>
<td>New TKR Recon</td>
<td>DONE</td>
</tr>
<tr>
<td>LDF upgrade to v05-05-00 – updates to GEM</td>
<td>DONE</td>
</tr>
<tr>
<td>UDF handling</td>
<td>Thinking about processing</td>
</tr>
<tr>
<td>TAG</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>v4r060204p3</td>
<td>First version based on GR v6r2p1 – new TkrRecon</td>
</tr>
<tr>
<td>v4r060208p2</td>
<td>New TKR calibration code</td>
</tr>
<tr>
<td>v4r060302p4</td>
<td>New CAL – but not fix to 4 bumps, new LDF, stores diag&amp;err lengths,trgParityError</td>
</tr>
<tr>
<td>v4r060302p5</td>
<td>Fix to CAL for 4 bumps</td>
</tr>
<tr>
<td>v4r0604p0</td>
<td>Error Contribution</td>
</tr>
</tbody>
</table>
CAL Update

- See CAL Report
- New CAL Code fix for 4 bumps will be made available when the tags are out and included in a tested GR tag.
TKR Update

See TKR Report
1. Work on badStrips.dtd and tkrCalibration.dtd, up through yesterday (Tuesday). They are now, we believe, in final form, tagged in calibUtil v1r3p3

2. Enhanced xml --> ROOT translator for tracker calibrations to handle charge scale as well as ToT gain. Broke the latter slightly in the process, but believe it is now ok again. Am poised to try it on a charge scale file, but haven't had a chance, due to activity on 1. and on..

3. The mysterious tracker calibration-fetching failure. Anders' job was successfully getting timestamp from data and using it to determine and fetch correct CAL calibrations of several types, but could not do the same for Tracker. Turned out that the problem was in the sequencing in job options -- tracker stuff was being called too early, before event was read in and hence before event timestamp was good.

Need to finish up work on charge scale calibration:
- test translating program
- define calibration TDS classes for charge scale (CalibData package)
- write and test ROOT converter (CalibSvc package)

Since this isn't very different from ToT gain calibration, it should go reasonably smoothly and quickly.
We decided earlier this week to advise I&T not to use foreign datasets for their data processing initially, so Warren converted all of his pipelines (8 total) to the old (non foreign-dataset) configuration, and we have all of them loaded into the prod and dev servers. They have all been tested with "dry runs". Minor problems were found and fixed, and Warren reported that the pipeline is configured and ready to start taking data.

We will need to be available to help with general pipeline maintenance and provide general support when asked for day and swing shift. Dan and I should provide I&T with contact information so that they have a "support line" they can use when necessary.

The upshot is that pipeline is ready to go for I&T to start taking data this afternoon.

there is now a "Dataset Catalog" browser on the glast-ground web site that makes it easy to locate and download datasets from the pipeline
We will create a new Jira category for operations (“Pipeline I&T Ops” or similar)

svac folks will insert issues (ie they will filter problems that are pipeline related)

we will be watching during day and swing. If problems are critical they will be so marked in Jira and we will try to handle them as soon as we can. Non-critical during swing shift waits until morning. Matt will check in with svac in the mornings; Dan in the afternoons.

Warren will test the foreign datasets handling in the next couple of days

Matt and Dan both claim to have completed their respective contributions to this feature. They both need further testing to prove their stability, but Matt's is looking good so far, and Dan’s should require only additional constraints in stored procedure queries if it has any bugs.
He would like task name versioning (ie separate task name and version into separate db columns and teach createRun.pl etc about them) available by sometime next week

- I believe this will be possible by next week.
  - It will require 3 additional integer columns to the task table (version "v", revision "r", patch "p") and Matt will have to add support to his XML up/down-loader to take a version string in the form vVrRpP and split it into these numeric columns. (It must be split so that I can sort without misordering v1, v2, and v10, for example.)
  - It will also require a few changes in the perl management and utility code, and
  - It will likely require many changes in the PL/SQL stored procedure code

Keeping a copy of all datasets is deemed a priority. A possibility is to rsync u12 to u16 while the tape archive is being completed. We need an ETA from Dan on it.

- The archiving script to do Warren's task was almost done many moons ago. I will have to revisit it after discussing the implications of weekly task-definition changes with Warren. Perhaps task versioning makes this moot. If not, I may have to rewrite portions of the code to accept arbitrary task versions.
- u12 is half full; we’ll need to be sure we can switch output locations live when it fills up and we will add Warren to the notification list for the pipeline nfs partitions.
- We will hold off CCB until real data starts happening.
- Dan also have to yank out some of the persistent DB connections -- Dan may be able to do this more efficiently with Navid's help next week, by moving from my simple LSF wrapper to his more robust implementation.
- We also need to tidy up the scheduler and LSF log files generated by the pipeline and it's processes. We should split much of the warnings and errors off to run-by-run log files. We are considering having Igor research Perl logging modules and have him clean up my terribly verbose debug print statements. Dan believes it would be a good opportunity for him to become familiar with the code in it's entirety. A better logging mechanism would serve to make Warren and the I&T folks much more self-sufficient when debugging pipeline errors. It will also help Igor, Matt, Richard, and Tony in taking on some of the trouble-shooting-support.
- Dan will meet with Matt and Igor tomorrow to brief them on the command line utilities Dan built to administer pipeline tech-support. Dan spent today cleaning up existing utility scripts, and creating a new one for failing a run that has somehow become stuck in running or finalizing status as sometimes happens.