Notes on Tom's pipline feedback session

- Jobs disappear when they are submitted. Navid should give pipeline lsf job number when it is submitted. We should have mechanism for monitoring jobs from web.

- Express queue sometimes takes >45 minutes to start jobs. Thus Navid's cleanup jobs can be delayed for a long time.

- Sometimes we exceed 600 simultaneous connections to database. New connection pool will queue excess connections and mitigate this problem.

- Database hiccup can cause status of pipeline to become very confused. Actual problem seen was caused by two job schedulers running simultaneously on test pipeline.

- No script to nuke running jobs (in case of pending database outage).

- We should have a web interface to allow to look at batch jobs. This could be tied into pipeline if pipeline new LSF job number.

- Cannot view log file while job is running. Navid puts job file in temporary location and moves it at end of job (Why? We should ask Navid). Dan has location of log file so web interface could show it.

- Lack of timestamps (in log files). Tom suggests writing timestamp plus node to log file at every transition. Matt suggests using Perl logger for messages printed by batch system. Matt suggests replaceing print statements with C++ logger, but Richard points out Gaudi already has message service.

- Writing all log files to u05 will eventually cause NFS problems.

- We have not been able to successfully use scratch space reliably with non-Babar machines. We should raise this with SCCS. (Perhaps we can use lsf to tell us which machines have scratch space).

- Tom would like to be able to change queues for tasks.

- Would like to have a flexible way to specify file locations.

- Hard to grep log files because file names (x2000) exceed unix command line limit.

- Should not have to edit gleam wrapper script.

- Display summary of running jobs in task (e.g. cpu time used, node, time queued etc).