

Fermi database migration plan (new server, move to 11g)

Migration from the old server to new server, 10g to 11g

Distilled Steps (Estimated 30 work days to completion, 6 calendar weeks)

1. Take outage, Migrate pipeline and datacat dev database to new server
2. Test out new server on Pipeline-II 1.4
3. Test changes on new Pipeline-II 1.5 branch
4. Take outage, perform pipeline and datacat prod TEST migration
5. Test out temporary trending database, L1Proc at scale
6. After Pipeline/datacat prod migration TEST, schedule actual migration
7. Run migrated production pipeline on temporary trending table space for a few days
8. After the trending database is migrated to the new server, update it with temporary trending values
9. Swap to permanent trending database

Detailed migration of GLAST_DP_TEST tablespace (pipeline and datacatalog tables)

1. Requires transfer of .dbf (database files) and endian change. (2 days prep, 1 day exec)
 - Targeting Pipeline/datacat dev database
 - Requires the database files to be locked, so no updates are allowed to the database
 - a. For pipeline/datacat-dev, shouldn't be more than an hour or so
 - b. For pipeline/datacat-prod, size is somewhere around 200GB I believe
 - c. Assuming 20MB/sec transfer off of glast-oracle03, this will require a 3 hour outage
2. Test current pipeline on new database at scale (3 days)
 - Verify that this will work with a new pipeline dev-test pipeline instance
 - Test that it works at scale with 10,000 simulated parallel jobs
 - Try to run an L1Proc dev job, may be a bit difficult
3. Test performance changes to database structure for pipeline (2 days, 3 days, 3 days, 5 days)
 - Make 1.5 branch of Pipeline-II
 - Enable new partitioning features available in 11g for Stream table
 - Attempt to enable those for ProcessInstance table
 - Normalize ProcessInstance table to BatchProcessInstance, ProcessInstance
 - If normalization takes too long for production tables (more than an hour), we will perform initial migration targeting pipeline 1.4 branch, and then perform migration to 1.5 online at a later date
 - Verify changes
 - Make 1.5 release of Pipeline II
 - Make migration scripts
 - Migrate SRS pipeline with scripts, verify again 1.5 release of pipeline
 - SRS and pipeline-dev should be migrated to 1.5, new tables at the end of this
4. Create Temporary Trending database tables on new server (3 days)
 - Attempt to verify L1Proc on the dev/1.5 server works with this
5. Plan outage of pipeline/datacat prod for production migration TEST
 - Again, estimated 3 hours for transfer of roughly 200GB.
 - If the database is much larger than 200GB, the outage should scale at an estimated rate of 20MB/s transfer
6. Repeat steps 1-3 for pipeline prod database migration test (3 days)
 - Verify L1Proc production will write to new temporary trending database
7. Once pipeline production migration test is verified, plan outage for actual production migration (2 days)
 - Verify TNS names update/migration
 - GLAST_DP_TEST
 - GLAST_TREND
 - others?
8. Plan migration of glast_trend
9. Execute pipeline (and datacat) production migration (2 days prep, 1 day exec)
 - Outage time should be equivalent to outage time from production migration test
 - Perform TNS names migration (glast_dp_test and @pipeline-ii, glast_trend)
10. Pipeline production is migrated by now. Should be using glast_trend temporary tablespace
 - While the pipeline is running with glast_trend temporary database, we will notify glast collab that trending info will be wonky for a few days
11. Perform migration of GLAST_TREND to a new tablespace. As it is several terabytes, this should take a day or so.
12. Once migration of GLAST_TREND is completed, update the tablespace with the new information from the temporary GLAST_TREND database.
 - Should be able to be done online, with very small if no outage time.
 - Migrate TNS name for GLAST_TREND
 - Drop temporary GLAST_TREND info
13. If Pipeline-II 1.5 Migration wasn't completed for Production pipeline, perform that.