# Group production wish list, missing tools, monitoring and Data reduction framework task force

*Nurcan Ozturk, Paul Laycock, Rob Henderson and James Catmore*

*ADC Technical Interchange Meeting - May 16, 2013*

# A front-end tool to the ProdSys

- Need a front-end tool to the current/new ProdSys to define/submit the group production tasks. Initial ideas:
  - a webtool to create the task submission list files automatically
  - to be used by both the group contacts and the group production managers
  - simple enough by drop-down menus to configure the parameters needed in the list files
  - when a group contact submits a production request the tool generates email notifications to the group production managers for them to check and approve the request
  - once approved the list file is automatically fed into the Panda Task Request Page for actual task submission
- Such a webtool will also be used in the next generation of group production, namely by the data reduction framework in the new model as per the AMSG report
- Alden Stradling agreed to work on such a tool, initial discussion started this week. It should now also be possible to get help/manpower from the data reduction task force just set up.
- Note that a webtool setup by Rob Henderson is currently in use which extracts info from the php read of the DPD Savannah page and creates the list files and submits the tasks. We will leverage this experience for faster progress in the new tool.

# A monitoring tool for task operations

- Need a monitoring tool to help with task operations. Initial ideas:
  - available on the ATLAS Production Task Monitor (may fit better) or Panda Monitor
  - to be used by both the group contacts and the group production managers
  - enables group contacts to set their tasks to finished/aborted state directly w/o contacting with the group production managers (when AOD->NTUP task is aborted, its NTUP merge task should also be aborted automatically)
  - should also provide the ability to kill the active jobs when needed
- Such a tool will help the whole group production team by avoiding usage of the current scripts (different versions, database passwd, etc.). Group contacts will act on their tasks faster when there are problems or more importantly when dealing with the tails to finish up production, thus will ease on the operational manpower by the group production managers.
- While the above is implemented, can the following be provided on the current Panda monitor:
  - Instead of a single task id, can a list of task id's be typed on Panda monitor to get their status, owner, etc. info displayed
  - can the associated task id's be displayed when a list of dataset names are given
  - can the job id's of active jobs be displayed when a task id is given

# Wish list for processing functionalities (1)

- For the partial processing of AOD's we need to have splitJobs=yes option to be moved from the development version to the production one, request already added to:
  - http://prodsys.blogspot.ch/2013/03/march-2013-updated-list-of-requirements.html
  - Very much needed on the verge of productions for the summer conferences
- To be able to run on group/period containers, e.g.:
  - group.phys-higgs.data12_8TeV.periodA.DESD_SGLMU.pro14.embedding-01-01-08.Ztautau_filter_FTFP_BERT_EXT0/ with filenames like: group.phys-higgs.277050_022295.EXT0._00707.AOD_EMBLHIM.pool.root
  - also on the period containers, also on a year data container (namely all 2012 data for a given stream)
- Recovery of lost output files:
  - When merged output files get lost on group disk, need to re-run the subjobs in the merge task, for that first need to unfreeze the merge dataset, then re-run the subjobs. In some (rare) cases the unmerged dataset is already deleted, so AOD->NTUP task needs to re-run first.
  - Some automation is needed in the recovery of lost output files. Can we trigger a recovery mechanism (to be set up) when the dq2 consistency service publishes the lost files? How soon the dq2 consistency service catches the lost files, in most cases the unmerged dataset is still available so it is only a matter of re-running the merge task (already mentioned in ProdSys2 TDR).

# Wish list for processing functionalities (2)

- Handling of events_per_job parameter:
  - Should be automatically configured from the parent AOD task. Was a big hassle to extract it from the database and use it in the list files during Moriond production when most reprocessed AODs had events_per_job=10k vs the usual 1k (already mentioned in ProdSys2 TDR in the meta-task concept, expect this will be handled there).
- Handling of mis-requests and non-existent samples:
  - Group contacts may request to run on the datasets they ran before or the datasets may not exist or the datasets can be empty, in this case the ProdSys sends a bulk submission error messgae w/o submitting on the rest of the datasets.
  - Though this notification is very useful, can the tasks be submitted on the valid datasets by skipping the problematic ones w/o intervention by the production managers?
- Obsoleting group production output:
  - If the MC sample is buggy, the obsoletion should be done down to the chain including the group production outputs. Discussed with MC production coordinators and agreed, as a short term solution, that group production datasets will also be deleted by the MC production managers with an email notification to group production managers.
  - In the long term the obsoletion should be done centrally and automatically trigger all the datasets in the production chain. More on Wolfgang's talk.

# Common D3PD concept

- During the Moriond production we produced:
  - 7 distinct types of DAOD
  - 41 distinct types of NTUP
- From running over:
  - Egamma 15 times
  - Muons 13 times
  - JetTauEtmiss 11 times
  - debug 4 times

Taken from the AMSG report https://cds.cern.ch/record/1543445

| Format | CPU time per event (s) | Notes |
|---|---|---|
| NTUP_PHOTON | 4.5 | includes EGAMMA |
| NTUP_BTAG* | 0.34 | BTAGSLIM and BTAGEFF |
| NTUP_JETMET | 6.9 | |
| NTUP_TAU | - | No data |
| NTUP_SUSY* | 10.2 | SUSYSKIM, SUSYBOOST and SUSY |
| NTUP_TOP* | 0.02 | TOP only |
| NTUP_SMWZ | 6.5 | |

- Way more CPU than was ever foreseen
- Suspected that much of this was repetition - reduce to a common type, NTUP_COMMON
- NTUP_COMMON: Filesize ~AOD, CPU per event ~10s/event

# Common D3PD - What can be replaced

| NTUP | COVERED? | COMMENTS |
|---|---|---|
| NTUP_PHOTON | YES | ADDED RECENTLY |
| NTUP_BTAGSLIM | NO | MAY BE IN FUTURE |
| NTUP_BTAGEFF | NO | MAY BE IN FUTURE |
| NTUP_JETMET | YES | BY DEFINITION |
| NTUP_TAU | YES | BY DEFINITION |
| NTUP_SUSY* | YES | BY DEFINITION |
| NTUP_TOP* | YES | BY DEFINITION |
| NTUP_SMWZ | YES | BY DEFINITION |
| NTUP_*HSG2 | YES | |
| NTUP_GRJETS | YES | |
| NTUP_SMQCD | YES | |
| NTUP_TRIGBJET | YES | ~150 TRIG BRANCHES |
| NTUP_SM(DILEP) | YES | SKIM/SLIM COMMON |
| NTUP_WPRIME* | YES | SKIM/SLIM COMMON |
| NTUP_ZPRIME* | YES | SKIM/SLIM COMMON |
| NTUP_EXMJ | YES | SKIM/SLIM COMMON |
| NTUP_SLIMSMQCD | YES | SKIM/SLIM COMMON |

- Essentially all AOD-input types apart from the b-tagging specific NTUPs

# Common D3PD requirements

- Pros: Save disk space (by a factor of at least 3), save CPU (factor of 5)
  - Moving targets, but factors better than Moriond is a safe statement
- Store NTUP_COMMON on datadisk instead of groupdisk
  - Free up group disk for derivatives of this
  - Have more copies of NTUP_COMMON available on the grid - how many we can have?
- Need a new group name (GR_AtlasPhysics / Common / Your suggestion)
- Production frequency of NTUP_COMMON from "all" AODs, 2 - 3 times per year assumed - is this feasible?
- Support small-scale R&D production
  - Require groups to estimate resource requirements up front - we could define limits of what's acceptable, e.g. (need feedback!):
  - Max 10s per event, not larger than AOD (for the production system)
  - Max 10M cpu hours per group for R&D productions per year (fair share of resources)
- splitJobs option is absolutely needed for NTUP_COMMON production

# Data reduction task force

- NTUP_COMMON and its predecessors were rarely used as the "final" analysis format
- Following the AMSG report, several task forces set up:
  - https://twiki.cern.ch/twiki/pub/AtlasProtected/OfflineActivityCoordinationBoard/Analysis_TaskForces.pdf
- Task Force 2 is responsible for a new "smart" skimming/slimming framework
- Can start now with NTUP_COMMON, move to its successor (the Root-readable AOD, this will remove the need for the bulk "format change" processing step) when possible (not expected before mid-2014).
- NTUP_COMMON input -> Much smaller output in NTUP (Root) format
- This "final" format may still be subject to further processing and reduction by the user of course, but it should be made once/twice per analysis team, stored on group disk?
- Significant time and effort is currently spent on users processing the group NTUPs, writing their own book-keeping and doing the job that the production system can
- Goals:
  - Coordinate this step rather than have (more) chaotic end-user production
  - Better book-keeping, more efficient use of resources
  - And we actually know and record what gets used, which can feedback into what we store in NTUP_COMMON or its successor

# Data reduction requirements

1. Group/subgroup/analysis team decide they need a derived data format. They define the skim/slim criteria - they go to their friendly DPD group contact

2. The DPD contact codes it up, probably using a webtool (to be provided by the task force members) that configures the NTUP_COMMON -> derived format job and produces jobOptions

3. A test run should produce filter efficiency, output size and processing time to estimate resource requirements - if it's not anticipated, the contact iterates with the analysis team

4. The DPD group contact can now make a request to DPD production using the front-end tool mentioned earlier

- Need to define limits / guidelines of what's reasonable per analysis team
- Prefer a model where we don't build an AtlasPhysics cache just to include the config, but the config needs to be persistified, maybe a la MC jobOptions tarball?
- DPD production manager approves/submits the request to the production system