EXPERIENCE FROM DATA PROCESSING

MC, REPROCESSING, TRAINS

MC Production

some of this requests / comments are already in JIRA but some maybe missing

Request Interface Functionality :

- Request Placing :

  https://prodtask-dev.cern.ch/prodtask/request_create

  It would need some cosmetic changes and features :

  - Improve notification mail
  - Implementation of the default notification list for requests
  - Possibility to add more than one address in CC
  - Even though it is not clear how widely it will be used, we need a JIRA ticket for the Request
  - After second "Submit" the modification should be disabled for non MC managers (also only managers should be able to modify the "manager" field)

MC Production

- Request Approval :

  https://prodtask-dev.cern.ch/prodtask/inputlist_with_request/XXXX/

  - *MC Pattern* should be extended to include "project_mode" configuration
  - Finding inputs needs to be optimized for large requests, we have not yet hit a problem here but soon will happen
  - It is a bit confusing still the steps to perform to submit requests (evgen, fullchain). It would need to be simplified.

- Request Monitoring

  https://prodtask-dev.cern.ch/prodtask/inputlist_with_request/XXXX/

  and

  https://prodtask-dev.cern.ch/prodtask/task_table/

  - Would like to know from the request page list which request need attention and which not without having to look at the request details. Now it is possible to check details by click but they would need to be improved or modified.

MC Production

Request Handling :

- Additions implementations to improve usability :
- Delete slices that had problems or are not needed
- Clone implemented but something like fix will be handy.
- Full Request Clone (needed for physics validation tasks)

Request Post-processing :

https://prodtask-dev.cern.ch/prodtask/task_table/

- "Finish ASAP" will save some mails to experts

Task/Job Monitoring :

http://bigpanda.cern.ch/task/4XXXXXX/

and

http://bigpanda.cern.ch/job?pandaid=XXXXXXXXXX

- Really improved, improvements would come from better error reporting (dedicated JIRA XXXX)

Mail Notifications :

- Working, now we need to pay more attention to them. We could promote "*failed to refine task*" to the Alarm mail.

MC Production

Processing types and ProdSys-2 :

Most standard configuration Full Sim MC12 tested :

Event Generation :

- All configurations working in prodsys tested
- Creation of tarball needed to create the production e-tag (done using scripts). We could maintain this a bit longer even if prodys is gone but not desired.
- We need to commission the new transform Generate_tf (not working either in prodsys). Likely will be needed for MC15 but it is not yet fully clear when JOs will be available.

HITS Merging :

- JEDI Merging : Not completely tested for MC Production (my bad).

MC Production

Standard Configurations :

We have to review again all possible configurations. This is the status of the last ones :

Type	Full Sim	Fast Sim
MC11	ok	ok?
MC12	ok (benchmark)	ok? (to be checked)
MC14	Fixed on Friday	Not yet available

MC Production

Special Configurations :

- Overlay : Needs absolutely FullChain transform. **It needs a person available to develop it!**
- FTK : Needs implementation, uses special inputs types that need to be defined in ProdSys-2. Just recently implemented in our scripts, I can help now to implement it in ProdSys-2. We can live without FullChain, but will make processing easier.
- Physics Validation : Not really a special configuration but it has special needs including the new transform for merging.
- Cosmics reconstruction (even now we do not have a proper workflow with our scripts).
- Other historically forgotten chains …

Reprocessing

Request definition :

Ensure all transforms and workflows needed for campaigns done in 2012 can also be submitted to ProdSys2: Next week we will do this.

- already known NTUPMerge_tf.py is not yet validated/debugged, is needed (given "automatic" merging)?

- Automatic and custom merging (in JEDI?), followed by clean-up (deletion)
- Ability to use dataset pattern as input, e.g. physics containers, COMA periods
- Append and adjust an existing request (merging, a new reco step, ...)
- Start new request by cloning an existing request (and then modify details / tags)
- Exclusion pattern to mask known problematic files / LB
  - Rucio and JEDI using GoodRunLists in production, to avoid bad Lumi Blocks ?
  - Also: what about those that *can* be processed, but will not be used for analysis - save resources ?

Reprocessing

Request definition (II) :

- Preservation of lumi-blocks in merging steps (ESD, AOD, TAG formats..) - metadata from Rucio?
- Ensure all "project_mode" options available individual options: no longer use LIST files fraction_ESD, cmtconfig, splitjobs, lumblock, diskcount, ...)
- Define input:tag:output formats, e.g. merging only for AOD even if (when available a "Merging_tf.py") transform also does ESD

Reprocessing

Request / task management :

- Simple way to abort individual jobs, tasks, group of tasks or whole request
- Resubmit jobs within tasks, so we don't have to unecessarily redo 1000s of jobs and waste resources
- Ability to increase memory requirement for a job or all jobs in a task
- Easy way to download RAW files used on jobs that fail
- Creation of JIRA tickets for (groups of) tasks

Reprocessing

Monitoring (for BigPanda)

- Ensure Reprocessing Monitor functionality integrated into BigPanda

  https://atlas.web.cern.ch/Atlas/GROUPS/DATAPREPARATION/Reprocessing/monitor/
- Automatic statistics needs catching (algorithm in athena, selecting error message,..) in log files:
  - Display statistics on TASK page of failed jobs
  - Display statistics on JOB page of failed jobs
- List of jobs with maximum number of attempts per task
- Last update in job count (jobs done, running, etc)

Dataset handling

- Automatic deletion of unmerged datasets ?
- Ability to recover lost data from any step in a repro campaign

Reprocessing

In addition to submitting our slice tests in ProdSys2, to be done:

- Need to test tag creation in AMI rather than Panda interface
- Need full commissioning plan for reprocessing, to test system thoroughly in 3 (6) month timescale
- Look at how MC production are using spreadsheets, see if it is relevant to data reprocessing

additionally from Paul

"Overlaps with other feedback expected to be 100% !"

- LB boundary-awareness seemed to be a pain sometimes in Run 1, the 1k vs 10k problem, NTUP_COMMON running limited to 5000 events which then broke metadata. Are we sure we have a robust system that can handle this? LB-awareness is most important for DataPrep. Upgraded LB-accounting should help but this is still being worked on so this is one for next S&C week.
- Merging (internal) - how does JEDI decide on merging parameters? Particularly for derivations (outputs can vary by orders of magnitude depending on the derivation type) this will need to be fairly dynamic - using the pilot (?) may be ok, but this will also need to be LB-aware - don't split complete LBs - ever !
- Number of retries - set at 25 for reprocessing due to long tails, but such a large number can just mean delaying spotting a problem.
- Related - better error codes (on "us" to define) - can we use error codes better in monitoring and automated responses?
- Request - can we predict the time to finish based on the rate of progress ? Particularly useful for large campaigns like reprocessing. Accuracy is a different question!