Plans for data processing in spring 2020

35th B2GM: DP session
23/01/2020

Stefano Lacaprara, Marco Milesi
INFN Padova, The University of Melbourne

Summary

- Status from last B2GM
  - proc10 prediction at BPAC vs reality
  - GoodRuns: issues (runDB, DS, grid)
  - Bucket8 status
- Tools: status and desiderata
- Resources assessment for spring run
  - Both local and grid
- HLT and analysis skim integration
- Plan for future processing: 100-150 /fb by June

Stefano Lacaprara, INFN Padova

Status since last B2GM

proc10 and bucket8

Stefano Lacaprara, INFN Padova

Proc10 status

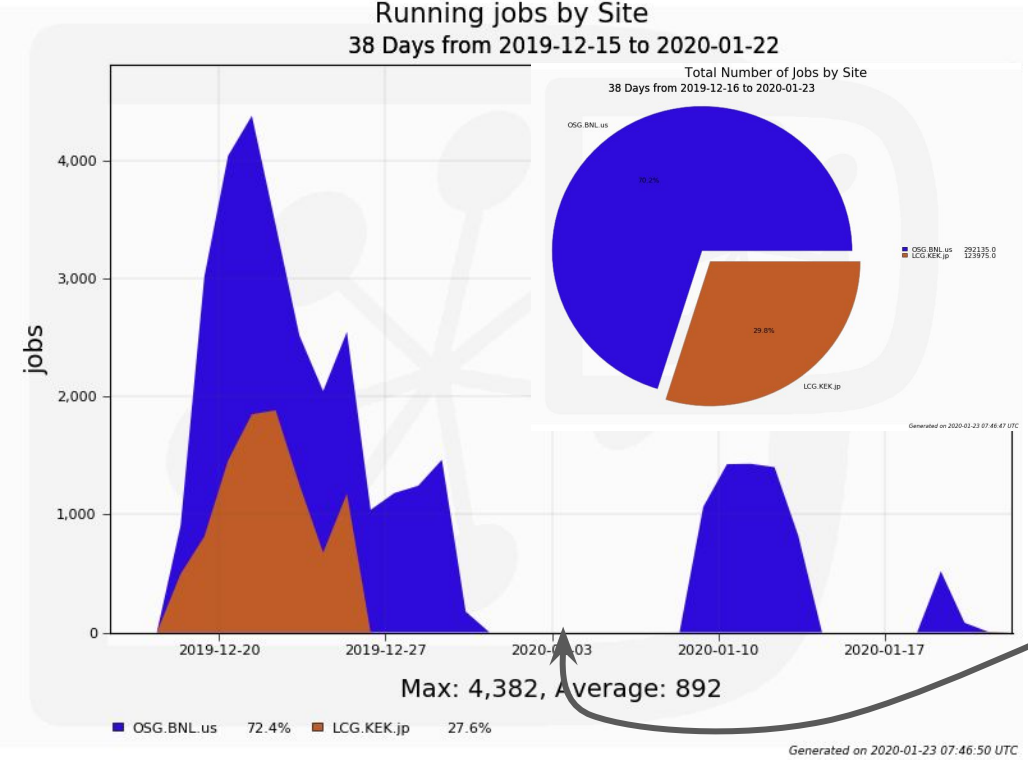
- HLT_skims **DONE ~20/12 (T_{proc}: ~2 d)**
- Local (KEKCC) All events **DONE 18/1 (T_{proc}: ~27 d).** Initial ETA **~1/1 (T_{proc}: ~10 d)**
  - Different reasons, understood and fixed (when possible):
    - b2_prod 1500->400 cores midway (now back to 1500, thanks Hara-san!)
    - Failures (due to temporary cvmfs glitch) not caught immediately.
- Good runs list provided.
  - Offline luminosity not yet available https://agira.desy.de/browse/BIIDP-2338
- All information on Confluence page:
  - https://confluence.desy.de/display/BI/Processing+2019a-b#Processing2019a-b-Processing10details
- Please report any additional issue you might find in BIIDP-2388

Proc10 on the grid

- A long and painful story.
- Multiple ProdID submitted
  - Limit set to 100 runs per ProdID
  - Additional ProdIDs submitted later due to various issues:
    - Mistake with our script for exp7 run<926
    - 4S_offres and 4S_scan runs invalidated and resubmitted with proper metadata and path
    - Few RAW files were missing on the grid for exp7 run<925

- In total 20 ProdID: 18 valid, 2 cancelled **100% DONE**
  - Exp7: 9629 9630 9631 9632 9633 9634 9863
  - Exp8 4S: 9635 9636 9637 9638 9639 9640 9641 9642 9643
  - Exp8 4S_offres: 9777
  - Exp8 4S_scan: 9776

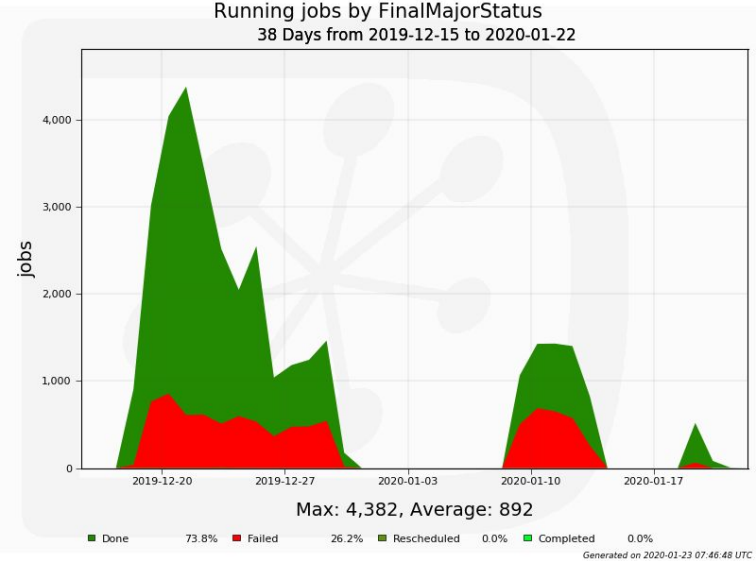
Proc10 on the grid (II)

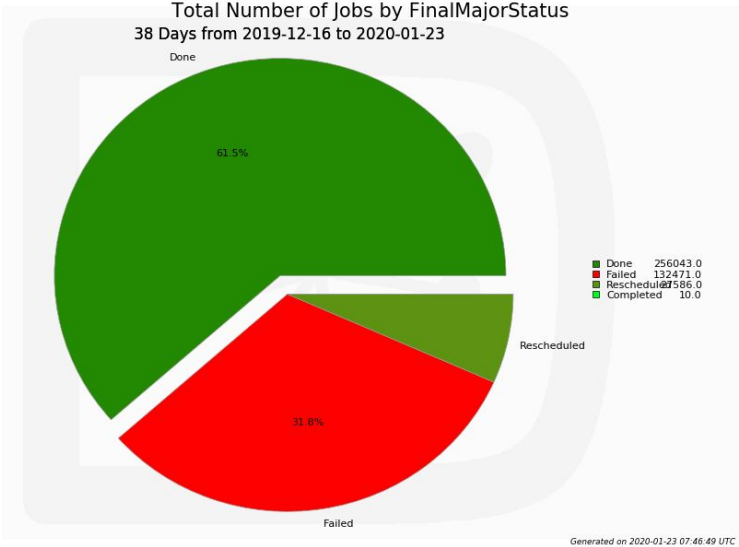


Only RawProcessing jobs shown, don't know how to show merge jobs just for this campaign (and not from MC13 also)

- A very nice start (both BNL and KEKCC) 70-30
  - Up to **4.3k jobs running**
- Then several issues at BNL
  - Jobs seen as stalled
  - Long ticket BIIDCO-2194
  - Problem not understood
    - Possibly related to SL6/SL7 watchdog
    - Still investigating
- Prod w/o progress for several days
  - Last peak is the very last processing for run <926

Failure rate: all failures at BNL.





- Initially issues with a WN at BNL (cvmsfs) fixed
- Two crashes in basf2: input files removed and tickets opened
- Then tons of stalled jobs killed and resubmitted (automatically by DIRAC)
  - 30% of total jobs

Bucket 8 processing

- Calibration (including cdst) by AirFlow (Umberto/David)
  - **cdst processing will start today: initial test during last night**
    - **(yes, there are people working even after B2 party)**
- Final processing as usual:
  - First HLT_SKIM (including hlt_hadrons) at KEKCC
    - Might consider to **run first hlt_hadrons**,
    - and then the others (**bhabha, gammagamma, mumu2trk**) to finish sooner
  - Process all events on the grid (+ KEKCC backup processing?)
    - Grid processing will start in parallel with hlt_skims at KEKCC
- Timescale: **L(exp10)=4 fb-1**
  - Hlt_skims 0.3 day per fb^{-1} => **1.5 days**
    - We will know from cdst processing for calibration
  - All events local: x12 (based on proc10 statistics) => **2 weeks**
    - Do we want to do this? [well, we probably will anyway, if no clash for LSF occupancy]
  - All events on the grid (based on bucket7) : **~1 week + merging delays**
    - Provided we don't face same issues as for proc10…+ N_{days} contingency?

Tools: status and needs

Grid monitoring tools

- DIRAC/gb2_prod have good monitoring capabilities, but we'd like to have:
  - Progress (eg running jobs vs time) for a given campaign for all jobs **RawProc** and **Merge**
    - Now we don't know how to distinguish merge jobs for Data or MC
  - Command line tools:
    - gb2_prod_status not so useful if something is not right
    - gb2_prod_summary give a lot of informations, which are not easy to read
      - In particular when jobs are resubmitted due to grid-related failure
    - We have a dedicated tool which parses the output and provide easier to read status *per run*
      - Would like something similar "natively" in gb2_prod_tools (no parsing!)

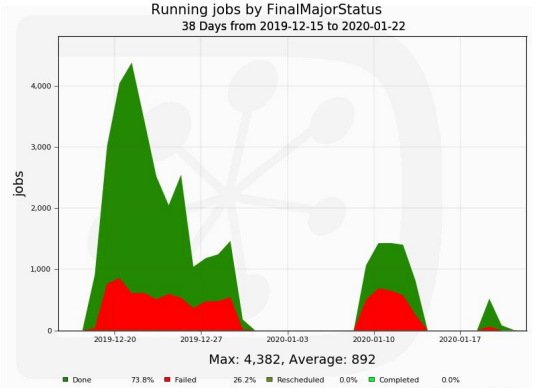
```
Run: 1834. ProdID: 9777
Tasks in Fabrication 'RawProcessing' (TransfoID: 105455):
N(Done) = 426/426
Tasks in Fabrication 'Merge' (TransfoID: 105456)
N(Done) = 43/43

Run: 1835. ProdID: 9777
Tasks in Fabrication 'RawProcessing' (TransfoID: 105457):
N(Done) = 982/982
Tasks in Fabrication 'Merge' (TransfoID: 105458)
N(Done) = 99/99


================================================================================================================================================================
SUMMARY of ProdIDs:
ProdID: 9776, Runs: [1025,1026,1027,1028,1029,1030,1031]
ProdID: 9777, Runs: [1703,1704,1707,1715,1716,1720,1726,1729,1730,1731,1736,1737,1738,1739,1765,1766,1767,1770,1772,1777,1778,1783,1788,1789,1790,1791,1792,1793,1794,1796,1797,1799,
1800,1801,1802,1803,1804,1806,1807,1808,1809,1810,1811,1812,1814,1815,1816,1817,1818,1819,1820,1821,1822,1823,1826,1827,1828,1829,1830,1833,1834,1835]
Tasks in Fabrications 'RawProcessing':
N(NoSub) = 0/31225, N(ToSub) = 0/31225, N(Wait) = 0/31225, N(Run) = 0/31225, N(WMSDone) = 0/31225, N(OnDDM) = 0/31225, N(Done) = 31225/31225, N(UnresolvedFailed) = 0/31225
Tasks in Fabrications 'Merge':
N(NoSub) = 1/3158, N(ToSub) = 0/3158, N(Wait) = 17/3158, N(Run) = 0/3158, N(WMSDone) = 0/3158, N(OnDDM) = 0/3158, N(Done) = 3140/3158
================================================================================================================================================================
```

Data Shift for data processing

- DP managers now babysit grid processing: this will continue, but we also have dedicated Data Processing shifters, that can do part of the monitoring
  - Now DP shifter is monitoring sites w/o particular emphasis on Data Processing.
  - Eg: a problem of failed jobs at BNL is treated as a problem in any other random site
- TODO:
  - Better communication with DP shifters
    - Eg: keep https://confluence.desy.de/display/BI/Computing+OperationStatus up to date
  - Provide RawProcessing oriented view in DIRAC
    - Eg: show status of all job in the current campaign: waiting, running, done, failed, etc
  - Define action (open jira, ggus tickets, mail relevant people)
  - Issues during RawProcessing should escalate properly.



Proc10 on Dataset Searcher

- Upload of LPN on DS last step of grid processing
- No major issue
  - Procedure well documented and smooth.
- So far we are uploading all runs, since we are processing all runs
  - But we need to filter only the good runs
  - It would be nice to interface DS with RunDB to do this automatically
    - And have RunDB as single authoritative source of Good/Bad runs
  - For the time being, we have a script which does this
    - Provide a list of LPN for good runs to be fed to gbasf2 `--input_dslist`
- Eventually the purging of bad runs can be done automatically by DS querying RunDB

RunRegistry RunDB

- From DP point of view can be used for:
  - **Good/Bad runs**
  - **Offline Luminosity per run**
  - Both are processing dependent
    - proc10 bad run can be good for proc11
- What if some jobs of a run failed (eg basf2 crash on m/N files)?
  - Process a run. Input 100’000 events, processed 90’000 (10k jobs crashed)
    - we lose 10% of events (so luminosity) because of DP.
    - **Rest of run is good!**
  - We already have # of triggers per run
  - **If we put #of events successfully processed by DP, we keep track of DP efficiency**
    - Also this is processing dependent
- DP need fields in RunDB which are processing dependent.
  - Also proper API to upload values
- Technical discussion with Martin et al later today or tomorrow (tbc)

Martin’s talk later

GoodRuns - BadRuns on the grid

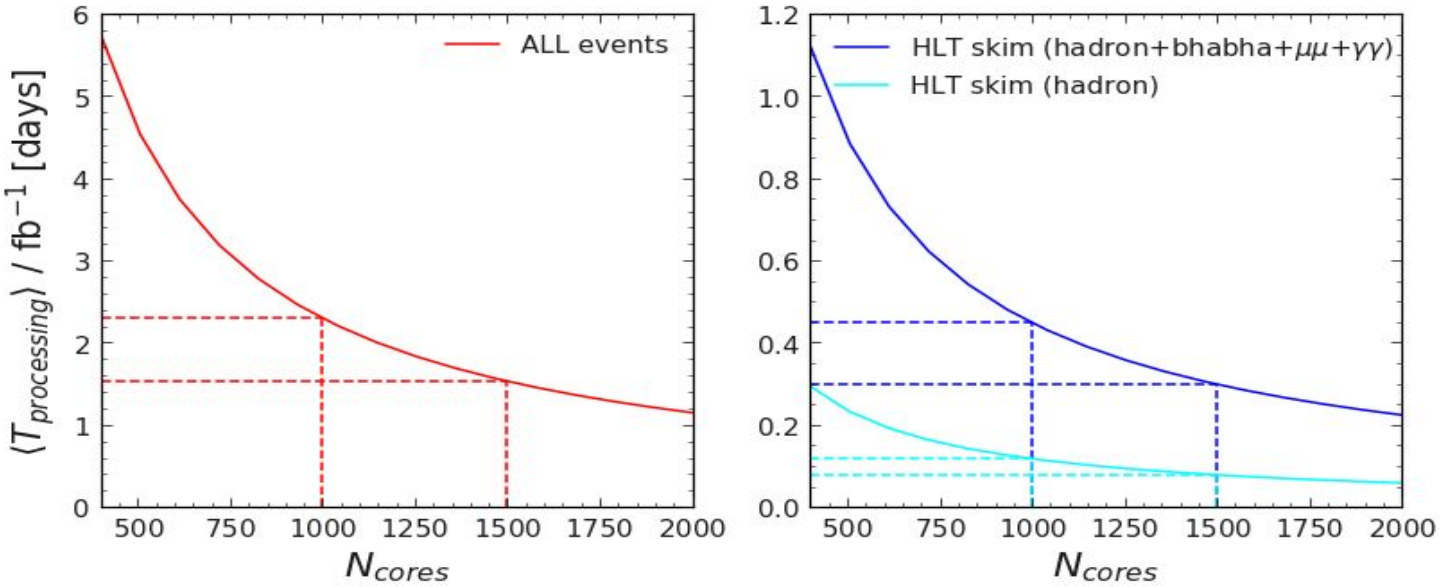
`gbasf --input_dslist GoodRunLPNList.txt …` **works BUT**

- Proc10 LPN good run list contains 17271 jobs LPNs (for 983 runs)
  - ~170k jobs in total, merge factor 10, 2’779’433’521 events, Size: ~5 TB
  - Average file size: 300 MB (small!).
    - Side remarks: Need to move to merge by target size asap
- Anyway: if we pass full LPN list to gbasf2 -> **17k jobs**
  - If we want to group jobs together (eg **-n 10**) same jobs can request input files from different runs, which are not guaranteed to be in the same SE.
  - **Input file match fails, job does not start.**
- The problem would be exactly the same if DS would be already integrated with RunsDB and provide good runs.
  - So, now only solution is to have many **983 projects** (one per run)
    - or **17k short jobs** (1 project)
- **Issue: dataset is defined now as a run. Ok for MC (1 run per ProdID), not for Data**
  - What if I want to process all runs of a campaign?
    - At most 1 jobs per run (ok), but smart job grouping is needed

Spring run 2020a: resources

Local KEKCC Resources

Used proc10 statistics to get a metric of "**processing CPU time per fb^{-1}**"



Prediction (e.g.):

Bucket 8 (~4 1/fb) HLT skim production: ~1.5 d (N_{cores}=1500), ~2 d (N_{cores}=1000), ~4.5 d (N_{cores}=400)

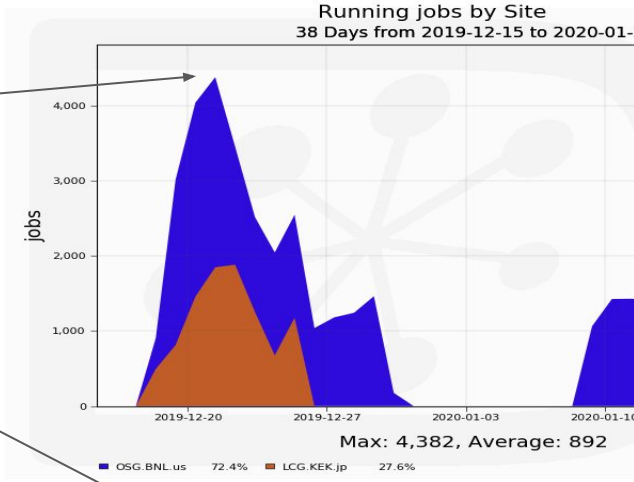
(all events production): ~6 d (N_{cores}=1500), ~10 d (N_{cores}=1000), ~24 d (N_{cores}=400)

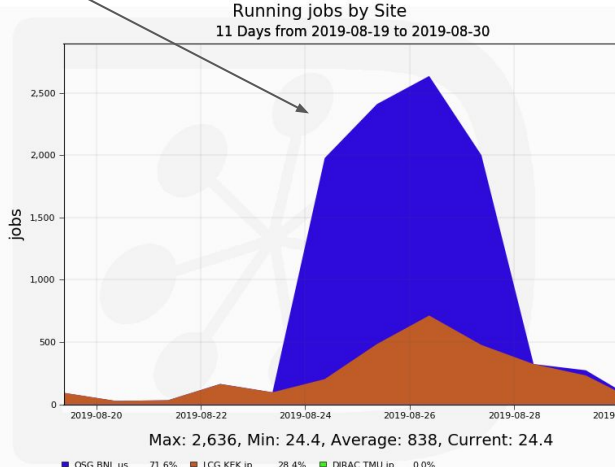
Local Resources

- Plan is to do at KEKCC only cdst production for calibration
  - Current estimate is ~0.3 days per 1/fb with 1500 cores for 4 hlt_skims streams
  - We will have 1500 until July 20th
- From Ijima'san plenary talk:
  - **Data quality values available run by run to the CR shifters**
    - **mirabelle → dqm**
- This requires some kind of express (?) reco (like the Unofficial we had so far)
  - Full dataset? Or some sampling?
  - HLT_hadron (1.5% of data) -> cdst/mdst? -> offline skims or more?
  - To be performed at KEKCC (or at BNL?)
- If we aim to 100/fb in 100 days, 1/fb per day (on average):
  - ~200 cores needed for continuous hadron processing
- Need to integrate offline-skims to mdst to fast processing

Grid resources for 100/fb

- Now Data processing runs at BNL and KEKCC
- Proc10 had max 4.3k jobs in parallel
  - Bucket7 had ~2 k jobs (~all at BNL)
- Processing all events:
  - Bucket7 2.9/fb in 1 week
  - Rescale: 6/fb per week of processing
- **To process 100/fb we need ~17 weeks**
  - If we process only selected HLT_skims (say ~30%)
    - **5 weeks** plus contingency
  - If we process only HLT_hadron (10%)
    - **2 weeks**
- The actual time to get all process DONE (and so available to user) will be longer: merge!
  - Will we have more resources by summer?



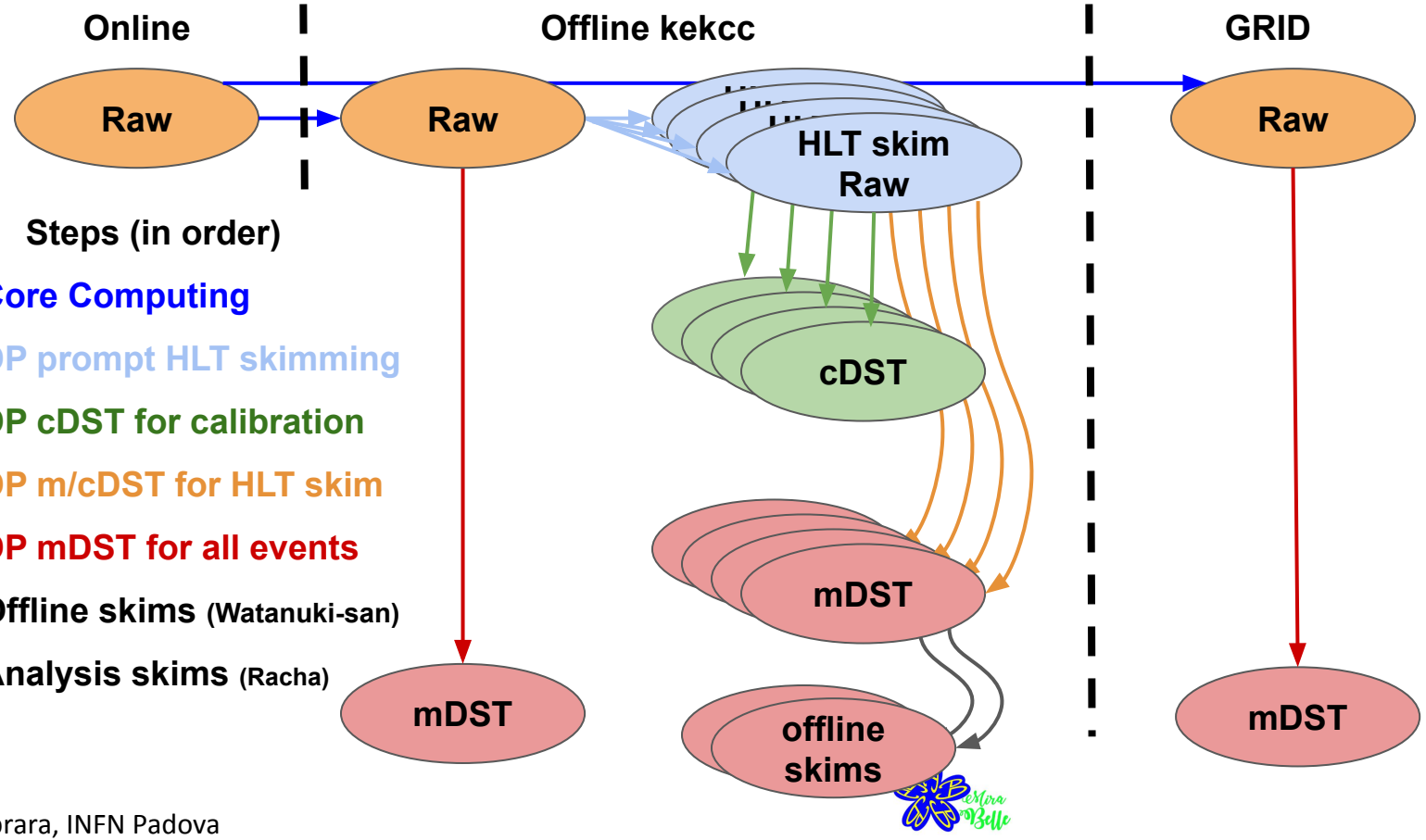


HLT and analysis skim

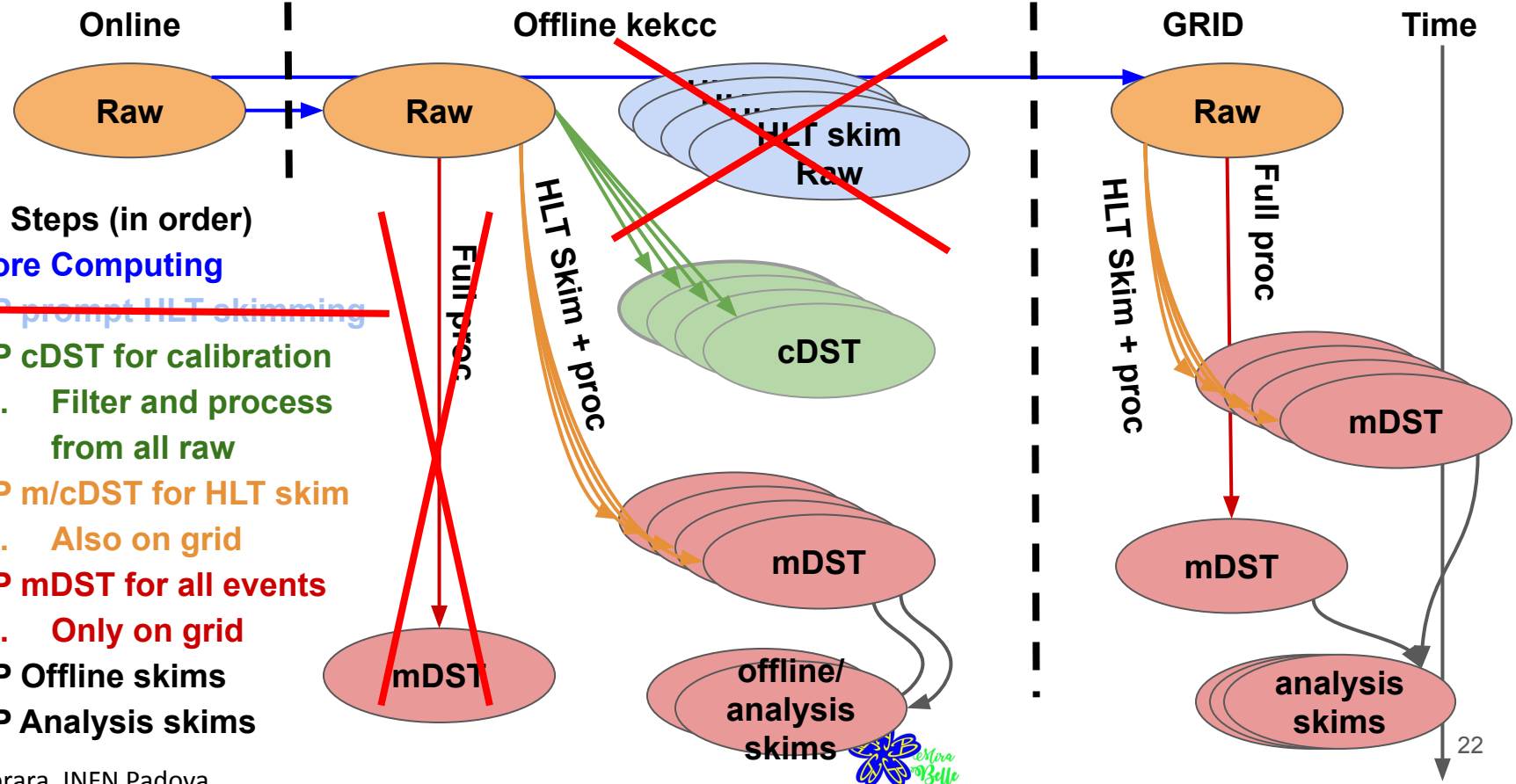
HLT and analysis skim in mdst processing

- Now analysis is performed on hlt_skims (hadron)
- Eventually, analysis will be run on analysis skim
  - Possibly (likely?) based on hlt_hadron
- **Now ana_skim are processed independently by Racha (next talk), who submits jobs after all mdst production is done**
  - If people is supposed to use analysis skims as input (we are pushing toward this)
  - **data is fully available only when skim jobs are done**
    - Non negligible delay: we are not taking this into account now
  - Or forget about analysis_skim and use directly hadron skim
    - same problem now for continuum MC.
- The plan is to integrate analysis_skim into mdst processing to provide them together with mdst (discussed with Racha)
  - Following is what we would like to do, need to test if doable with current gb2_prod tools
    - Or need to work with DC expert to find a solution

Data Processing schema (so far)



Data Processing schema (near future)

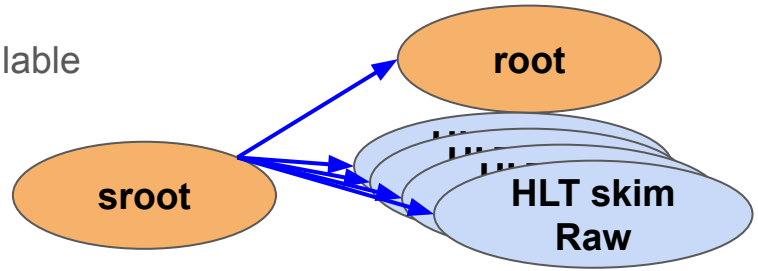


Steps (in order)

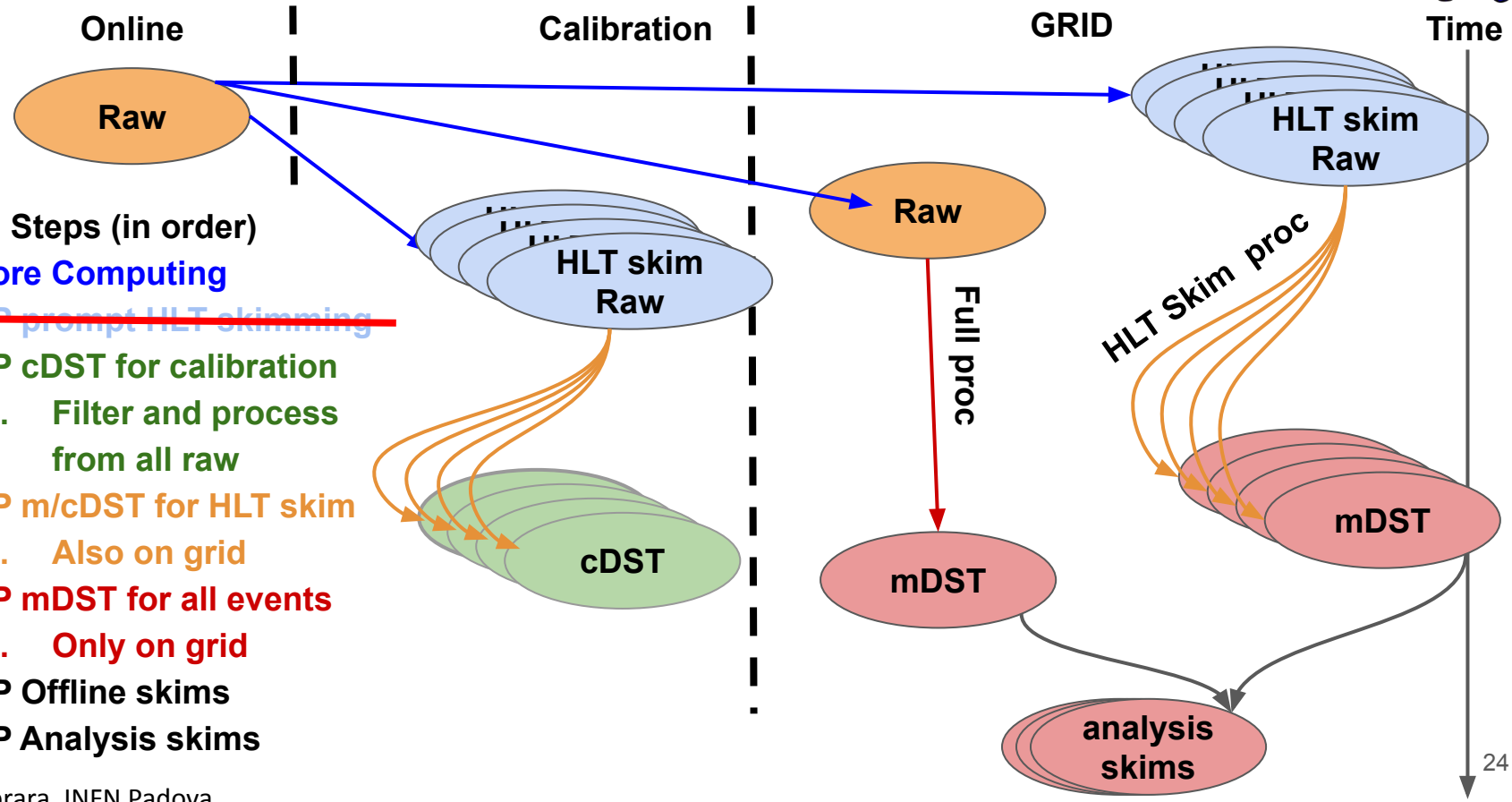
- Core Computing
- ~~DP prompt HLT skimming~~
- DP cDST for calibration
  a. Filter and process from all raw
- DP m/cDST for HLT skim
  a. Also on grid
- DP mDST for all events
  a. Only on grid
- DP Offline skims
- DP Analysis skims

HLT processing

- Goals:
  - Provide fast reconstruction up to analysis skim for physics relevant data
  - Reduce staging stress for SE to process only a fraction of staged RAW
- How:
  - Do together mdst production **and** analysis skims from hlt_hadron skim
    - Low multiplicity will need to run on all events
      - Unless a HLT_lowMult skim will become available
    - Just run udst skimming after the mdst production
  - Save multiple udst (similar to what ana_skim is doing)
  - This requires to stage full RAW file for processing
    - **Unless HLT_RAW are available on the grid**
    - Can be produced by DP
      - Stage all RAWs, low CPU, hard I/O, and write RAW_SKIM. BAD!
    - At ONLINE full RAW are already on disk, and **sroot->root** conversion can produce multiple stream:
  - To run on hlt_hadron need to stage only a fraction of RAW, not all! GOOD.



Data Processing schema (future)

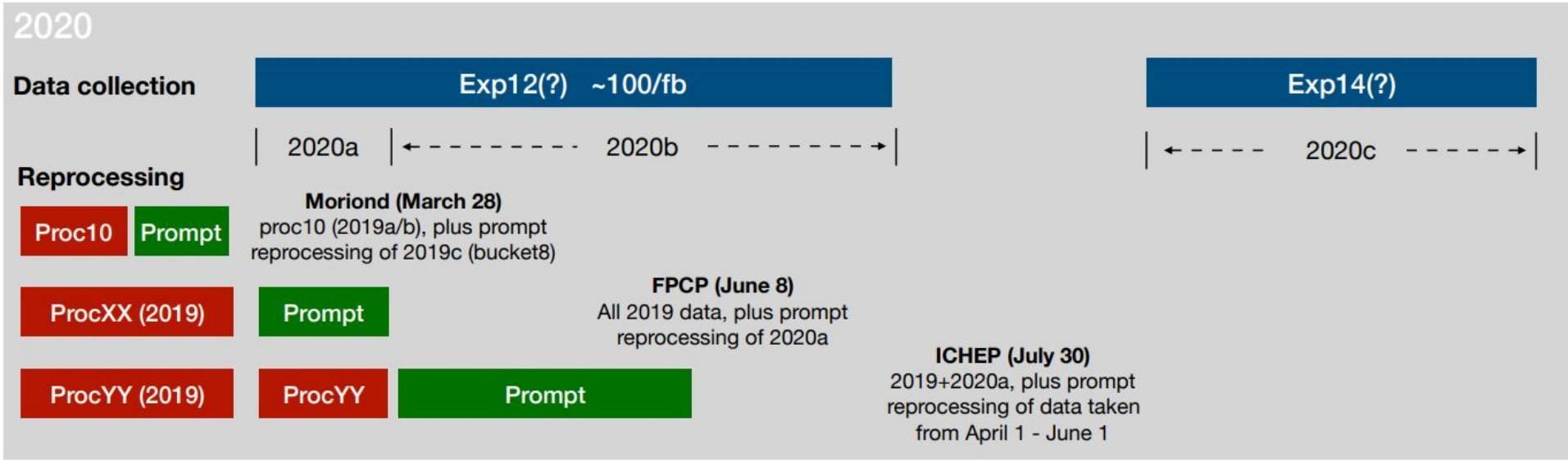


Steps (in order)

- Core Computing
- ~~DP prompt HLT skimming~~
- DP cDST for calibration
  a. Filter and process from all raw
- DP m/cDST for HLT skim
  a. Also on grid
- DP mDST for all events
  a. Only on grid
- DP Offline skims
- DP Analysis skims

Plan

Processing plans for 2020



- **Expectation for FPCP** (due May 1):
  - ProcXX (2019a-c) plus prompt reprocessing of data collected March 1 - April 1 (2020a?)
- **Expectation for ICHEP** (due June 30):
  - ProcYY (2019a-c, 2020a?) plus prompt reprocessing of data collected April 1 - June 1

- "Aggressive" datasets including prompt reprocessing of newest data can be added upon availability

Tentative plan for Spring

- The next two major conferences are:
  - FPCP 8/7 - ICHEP 30/7
- Showing results including a large fraction of the data taken in 2020 will not be trivial;
- Counting backward: 1-2 weeks for CWR plus 2-4 weeks for RCR
  - So, data needed about 1-2 months before conference (top up possible of course)
    - Namely **mid April for FPCP**
    - Beginning of **mid June for ICHEP**
- **Proc11 in march** (?), before the arrival of large amount of new exp12 data
  - **Including exp 7+8+10**
  - **Which release? Rel5 expected April/May (maybe too late)**
- Then **prompt processing** (buckets) for **FPCP**
- And possibly a **proc12 in may** (?) for part of exp12 data for **ICHEP**
  - **Proc13** with full **exp12** (plus 7-8-10?) **later in summer**

Backup

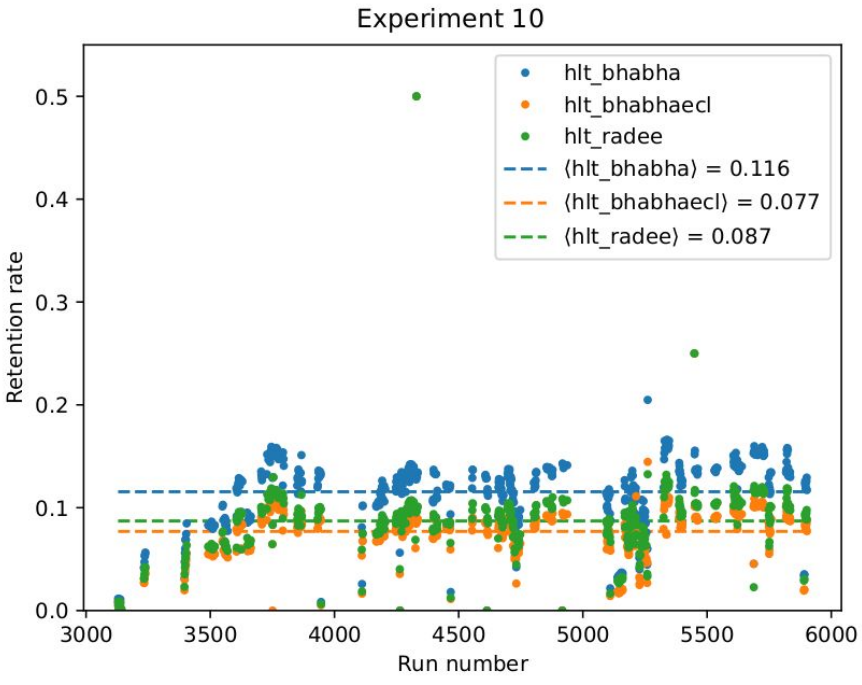
Processing Statistics - proc10, Exp8 (@KEKCC)

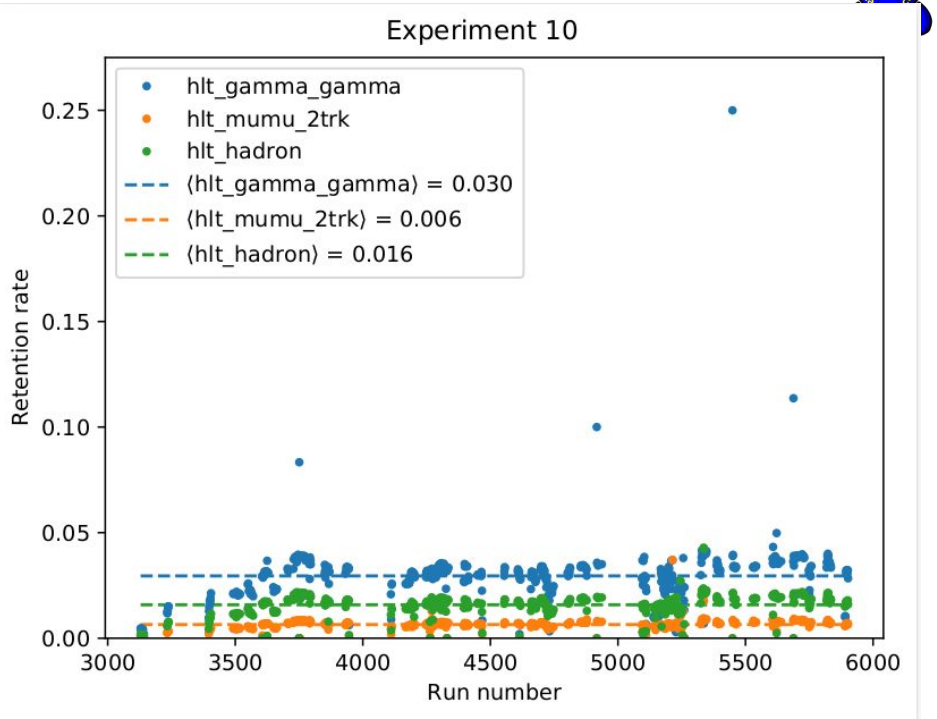
Int lumi (Exp8) = 5.8 1/fb

	ALL events	∑ HLT skims	
	mdst	mdst	cdst
<TB/fb^{-1}>	0.97	0.14	10
<N_{jobs}/fb^{-1}>	29k	4.3k (hlt_hadron: 600)	
<T_{job}>$_{CPU}$	1.9 h	Avg: 2.5 h (hlt_hadron: 4.7 h)	
<T_{job}>$_{turnaround}$	36 h	Avg: 11.6 h (hlt_hadron: 14 h)	

$$\langle T_{processing}/fb^{-1}\rangle = (1/N_{cores})*\langle T_{job}\rangle_{CPU}*\langle N_{jobs}/fb^{-1}\rangle$$

HLT retention rate





- NB: Exp10 HLT (mostly) ran in “monitoring” mode → no online event filtering, just flagging.