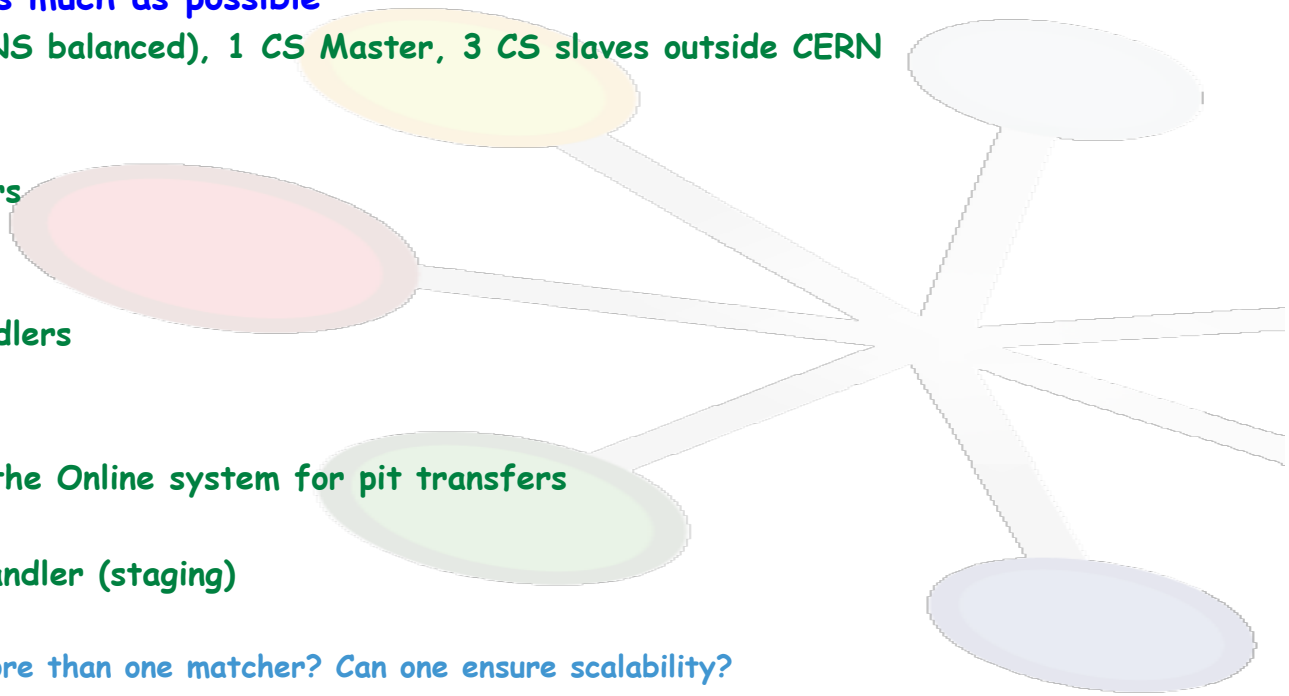
LHCb and DIRAC

Philippe Charpentier
CERN

Infrastructure for LHCb

- Moved LHCbDIRAC services and agents to a smaller set of larger VMs
  - 11 VMs with 16 processors, 32 GB of RAM
    - 2 VMs with 2 processors, 4 GB RAM
    - 1 VM with special services for BOINC
  - Redundancy for services as much as possible
    - 4 CS slaves at CERN (DNS balanced), 1 CS Master, 3 CS slaves outside CERN
    - 1 proxy manager
    - 4 DFC instances
    - 4 Transformation handlers
    - 4 Bookkeeping managers
    - 4 sets of job optimizers
    - 2 Jobs state update handlers
    - 2 Job managers
    - 2 Data store services
    - 2 RMS managers + 1 in the Online system for pit transfers
    - 4 RSS handlers
    - 1 StorageManagement handler (staging)
    - 1 Matcher
      - Is it possible to run more than one matcher? Can one ensure scalability?
- Move to using Mesos for service deployment (see Chris' talk tomorrow)
  - Already in place for the certification setup



Other changes

- Using ElasticSearch for WMS history
  - MySQL accounting discontinued
  - Much faster plot generation
  - No data storage problem
    - We considered keeping only 1 month, but have not cleaned yet
- Improved FTS system
  - 2 FTS agents dedicated to different transfers
    - Transformation requests
    - Failover requests (i.e. linked to jobs)
      - To get a better failover response while heavy transfers are ongoing
- 3 Transformation agents + 3 workflow agents
  - Partitioning the TS to ensure scalability
    - Use transformation type for sharing the work
  - TS agents: real data, merging and DMS transformations
  - Workflow agents: real data, MC simulation, MC not simulation (reconstruction, stripping)
    - Fix in DIRAC in January largely improved the submission rate

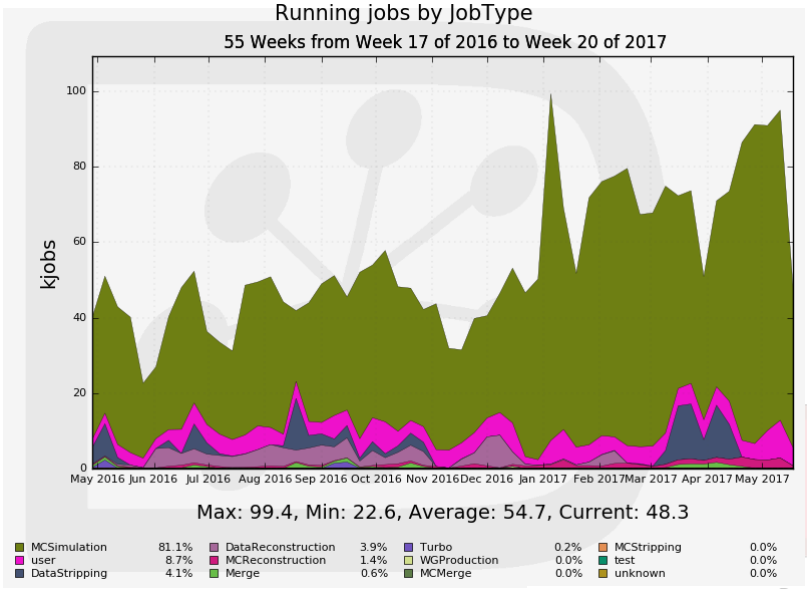
Data Management evolution

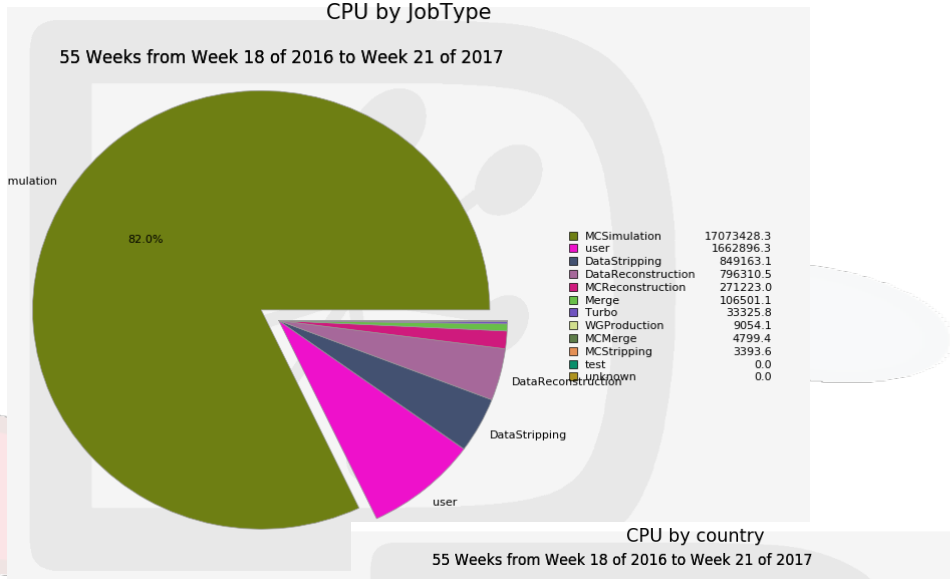
- Still mostly SRM-based
- Start using xroot protocol plugin in StorageElement
  - In production for CERN EOS in read mode
  - Will move to all SEs in read mode
    - Warning: accessURL will no longer test that the file exists!
      - We developed a set of scripts for checking files at SEs (in LHCbDIRAC, can partly be moved to DIRAC)
  - Write mode being tested
    - First tests look OK, but xrootd libraries need a fix (only works at EOS!)
    - Only possible if a physical SE endpoint serves a single service class (i.e. not disk AND tape)
- Enhanced DM scripts, in LHCb still now
  - Assumption that LFN starts with /<vo>/ and no other appearance of it
    - Not true for all Vos, otherwise could be moved to DIRAC
- Next step:
  - Whenever possible avoid SRM in FTS transfers (xroot or gsiftp)
    - Same limitation as write mode above, but can be used for the source, and SRM for destination, requires careful testing…

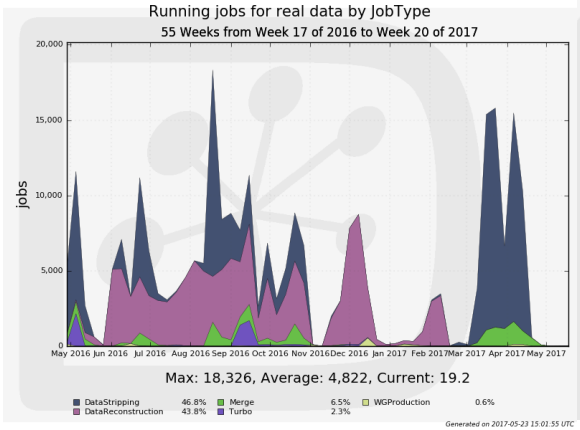
Operational optimization

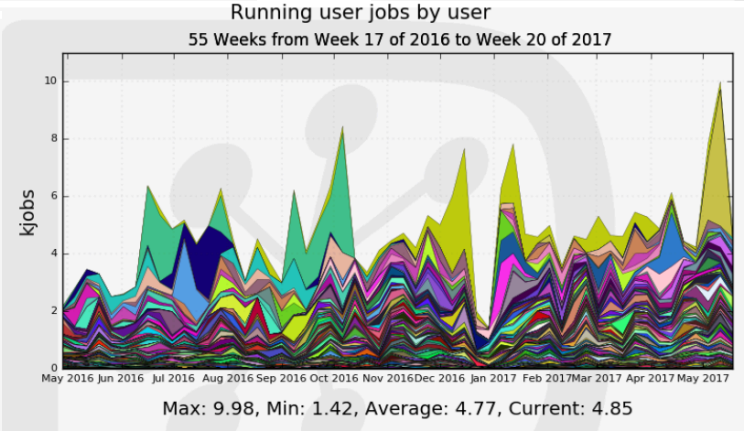
- For (re)processing data from tape
  - Pre-stage files to a disk SE first
    - Needs some preparation and enough free buffer space
    - In the TS (plugin), force to create tasks only with files from disk SE
      - If not on disk, wait for next loop
  - Launch several productions with limited (~100,000) number of input files
    - Allows parallel handling by the TS
      - TransformationAgent, WorkflowTaskAgent
      - Easier for making final checks (completeness, output data integrity)
- MC productions
  - Implement possibility to stop after current event
    - Interaction between the watchdog and the Gaudi process (signal)
    - Useful when a pilot needs to be stopped (HLT farm, VM end of life, also batch queue end)
      - Can be coupled with usage of MJF information
  - Run pure MC simulation productions first
    - Single step, most time consuming jobs
  - Further MC processing (digitization, reconstruction, trigger, reconstruction…)
    - Subsequent productions with multiple input files (replaces or reduces merging)

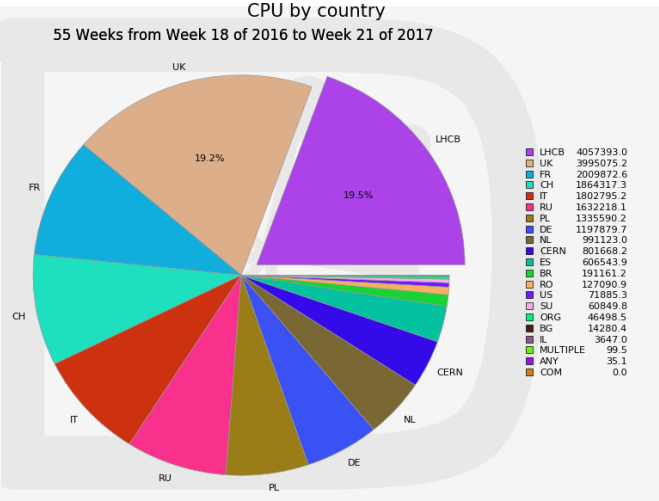
LHCb activities in the last year











Activities: comments

- Introduced 2 new "countries"
  - LHCb and CERN: country represents the funding agency, not the geographical location
- Dominated by MC simulation
  - Heavily using the HLT farm
    - Even during data taking (e.g. during p-Pb collision when HLT is less used)
    - Top CPU provided integrated over the year
    - Heavy load on CERN EOS storage as the HLT farm has no external access
- Steady user load (average ~5000 jobs)
  - 470 unique users have used LHCbDirac in the year!
- Data reconstruction and stripping campaigns
  - Takes place when processing is ready (application and calibration)
  - Main burden for data handling, but not much CPU required
  - Using non-Tier1 sites
    - "Mesh processing": sites can help each other
      - Even Tier1s can process other Tier1's data
      - Output data always uploaded depending on the run number (assigned to a site)

Possible improvements

- Less deterministic site assignment
  - Currently sites are assigned at job creation (mesh processing)
    - Impossible to react to site downtime or overload once jobs are waiting
  - What about late site binding, e.g. at matching time
    - Assign jobs to sites hosting the data
    - At matching time, check whether there are jobs for the originating site
      - If yes, get it
      - If not, use the "mesh processing" information to check other task queues
    - This would allow much more flexibility and reactivity
- Task queue agnostic pilot submission
  - Use feedback from pilots for throttling pilot submission
    - Rather than use task queue information
    - Anyway pilots don't match jobs that they have been submitted "for"
    - "VAC-like" model for site director
    - Being worked on by Andrew
  - Would allow a better pilot dissemination
    - And faster pilot submission

What is left from last year?

- Bulk submission is not yet there
  - This is a MUST: user jobs, MC productions
    - Currently ganga playing tricks with input sandbox uploaded to User storage
      - Pb if SE is overloaded or in downtime
    - Improve submission time at the client level
      - Mostly for user jobs
- Pilot filling mode is far from optimal
  - No maintenance of TimeLeft utility
    - MJF usage still very limited (also at site level)
- Multiprocessor jobs still not in use
  - Not a must as LHCb jobs are not too much memory-hungry, but still…
- Split jobs and tasks statuses
  - This is VO-dependent, but should be implemented
  - Task final status is not necessarily that of the job
    - A job may be failed but the task successful and vice-vers (more rare but still exists)
- "Completed" job status should be changed !!!

What are we expecting to come soon?

- Pilot logging is more and more eagerly expected
  - Large fraction of jobs without any pilot logfiles (not using a CE)
    - And even on CE, the lifetime of logfiles is short (a week)
- New FTS system still not commissioned
  - Hopefully will be more reliable
  - Was already developed 2 years ago, just after Ferrara WS…
- Extend multi-protocol usage
  - Still very limited now
  - Careful deployment as there may be site-related issues
- Usage of priorities for jobs should be revisited
  - Better documentation first: how does it work is far from clear
  - In LHCb we have hard time to get top priority jobs running and use MCSimulation for filling up sites
  - Need to better control user jobs and shares

Conclusions

- LHCb is running successfully a lot of workflows through LHCbDirac
- More and more new platforms are popping up
  - Usage of Pilots 3.0 (see Andrew's talk) should be generalized
- Some improvements / simplifications may help scaling
  - Late site binding
  - Vac-like pilot submission
  - Bulk job submission
- Long-standing developments should be included ASAP
  - Pilot logging
  - New FTS system
- All is a matter of lack of manpower
  - A lot of development for DIRAC and LHCbDIRAC done in LHCb
  - Most developers also participate in certification and operations
  - … and we need to keep an efficiently running system!