Distributed Data Management in LHCb

Ph. Charpentier
CERN-LHCb

Data Management model: physics analysis

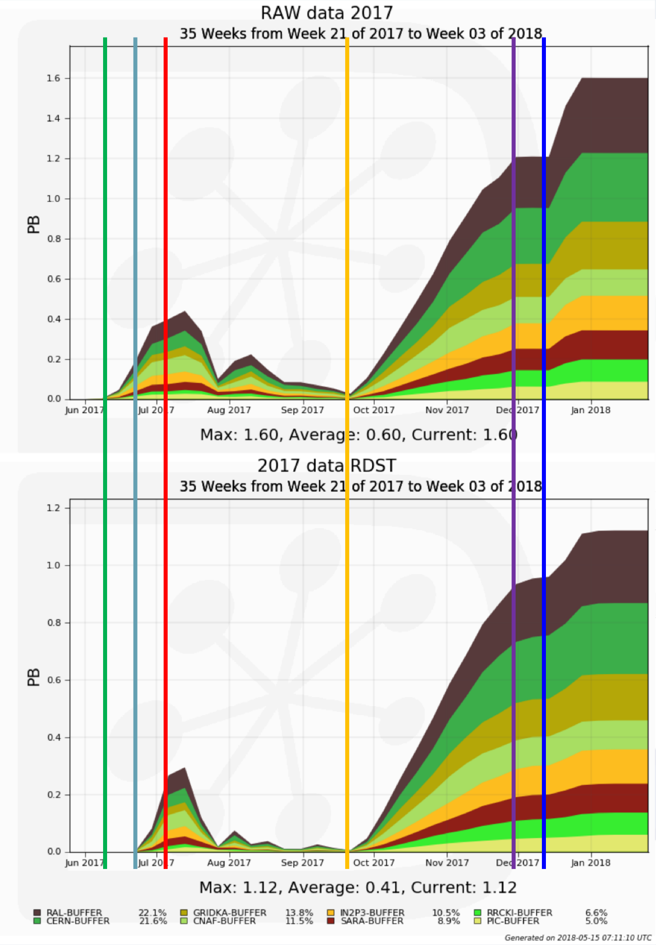
- Physics data: real data (μ)DST
  - Merge into ~5GB files: keep run granularity
  - Each run has its destination site: upload to disk at destination
  - Replicate to a second disk SE + one "archive" tape SE
- MC physics data (μ)DST
  - Small files uploaded with geographical mapping to disk @ Tier1
  - Files merged (5GB) at Tier1
  - Replicate to a second disk SE + one "archive" tape SE
- Selecting second disk SE
  - Randomly selected using free disk space at site as a weight
  - Advantage: no risk to fill up a site with replication
- For real data: keep only last processing pass
  - Sometimes last 2 processing passes
  - Additionally some incremental processing passes (fixing and adding selections)
- Use popularity for retiring datasets (real data and MC)

Data Management policy

- All production data are centrally managed
  - No user / site / group is allowed to replicate production datasets
  - Users are handling their own private data
  - Working groups also have dedicated storage, but WG productions are regular productions, i.e. centrally managed
- Duties of the production manager
  - RAW data replication and removal from disk after processing
  - RDST and FULL.DST (reco output) replication to tape / disk
  - Stripping output replication
  - MC output replication
  - Obsolete or bad data removal (complete removal or only disk removal)
  - RAW and RDST pre-staging prior to re-stripping campaigns
    - In conjunction with removal after processing
- Mostly handled by one person
  - Was me for long, now Chris

Real data reconstruction and first stripping

- Start of data taking
  - Replicating RAW data to buffer
- Start reconstruction
  - RDST to Buffer
- Start stripping
  - Removing RAW and RDST from Buffer
- Stop removing RAW+RDST from buffer after stripping
- End of data taking
- Stage first part of the year (but CNAF)



RAW data 2017
35 Weeks from Week 21 of 2017 to Week 03 of 2018

Max: 1.60, Average: 0.60, Current: 1.60

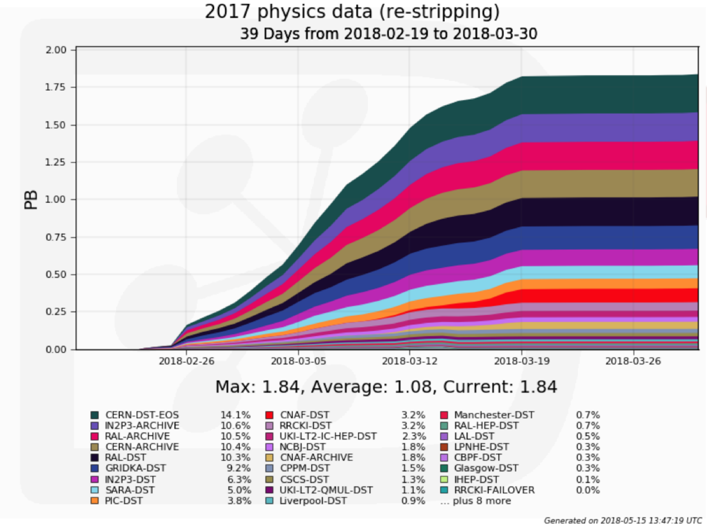
2017 data RDST
35 Weeks from Week 21 of 2017 to Week 03 of 2018

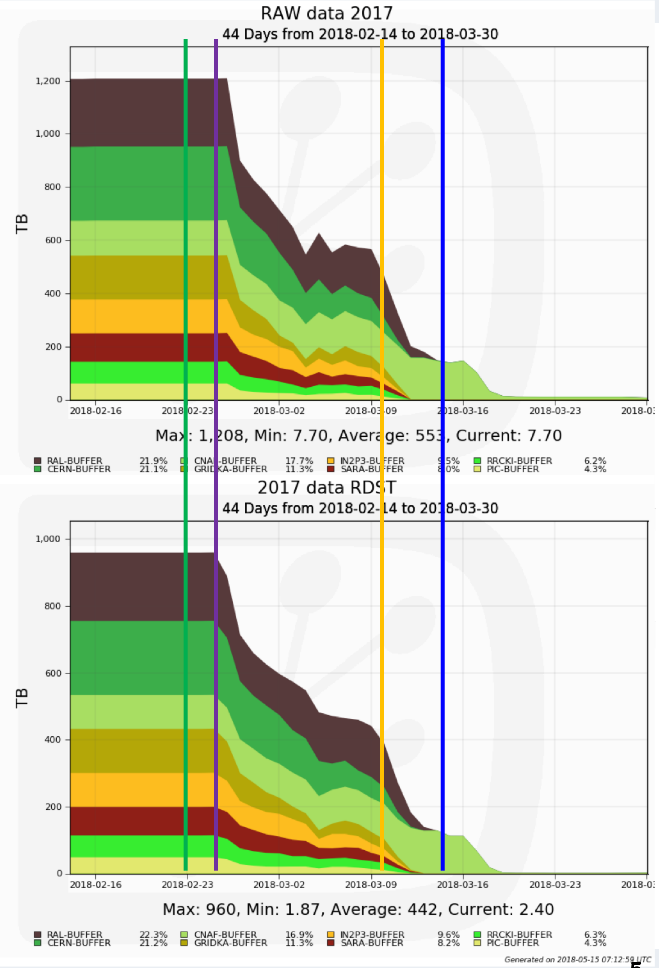
Max: 1.12, Average: 0.41, Current: 1.12

Generated on 2018-05-15 07:11:10 UTC

Real data re-stripping

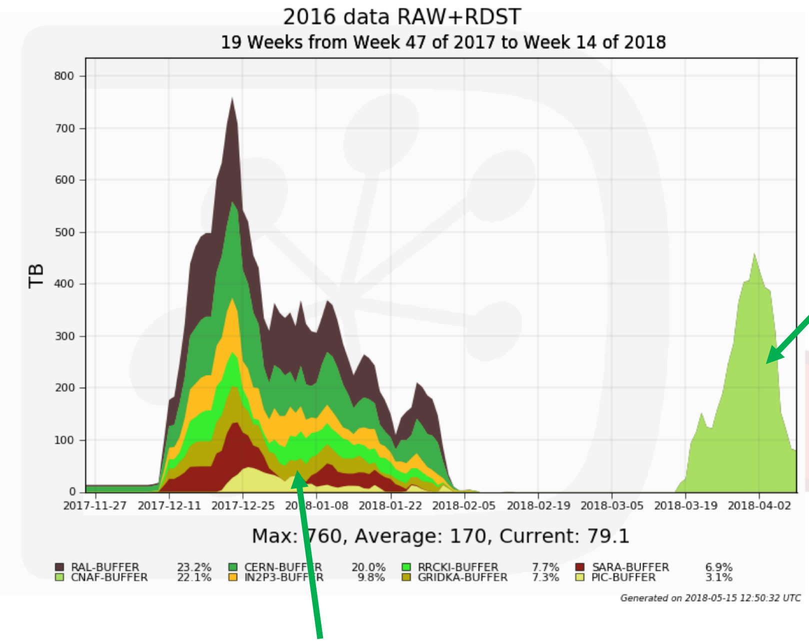
- Start re-stripping
- Start removal of RAW+RDST
- Start staging at CNAF (after flood)
- Start CNAF processing





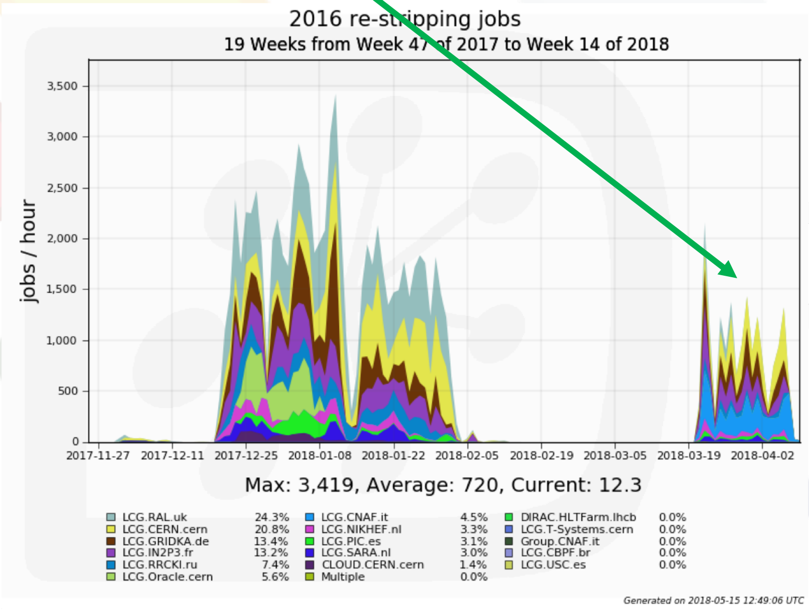
- □ Physics data replication

2016 re-stripping campaign



No files from CNAF

Files at CNAF
with mesh processing



Some technical details: Transformation plugins

- All DM (and WMS) operations are handled through the TransformationSystem
  - Specific set of LHCb plugins
- Plugins used to create replication tasks
  - RAWReplication
    - Replicates RAW files to a Tier1 tape SE and to a disk buffer either at that Tier1 or at CERN
    - A run (1 hour of data taking) is assigned to a site
  - ReplicateWithAncestors
    - Replicates files from tape to disk buffer as well as their ancestors (useful for staging RAW+RDST data prior to re-stripping)
  - LHCbDSTBroadcast
    - Replicates files (grouped by runs) to a given number of disk SEs (current default is one more) and a number of archive SE (current default is 1)
  - LHCbMCDSTBroadcastRandom
    - Replicates files randomly to a number of additional disk SEs (default: 1) and a number of archive SEs (default: 1)
  - ReplicateDataset[ToRunDestination] / ArchiveDataset
    - Creates additional replicas on specific SEs (disk or archive)

LHCb plugins (cont'd)

- Plugins used to create removal tasks
  - RemoveReplicasWhenProcessed
    - Creates tasks to remove files from buffer after they have been processed
  - RemoveReplicasWithAncestors
    - Creates tasks to remove files from buffer after they have been processed and remove their ancestors as well (useful after stripping)
  - RemoveDatasetFromDisk
    - Creates tasks to remove all disk replicas, provided there is a tape replica (useful to remove unpopular / useless datasets)
  - DestroyDataset
    - Removes all replicas of all files

- DM tasks are transformed into requests in the RMS
  - Limit the number of files per task / request
  - DM operations executed by the RequestExecutingAgent (+FTSAgent for replication)

Plugin parameters

- Many plugin parameters are configurable
  - Decreasing order of precedence:
    - Default in code (usually None)
    - Global defaults in CS (`/Operations/TransformationPlugins` section)
    - Plugin-level defaults (`/Operations/TransformationPlugins/<plugin>` section)
    - Transformation level values (additional transformation parameters)
  - Examples:
    - `DestinationSEs` / `NumberOfReplicas` for replication plugins / transformations
    - `FromSEs` for removal plugins / transformations
  - Pre-staging throttling
    - Limit the total number of files pre-staged for a given directory
    - Uses the current occupancy (from StorageUsage DB) and number of files in the pipeline
    - Limit per site computed using site shares
    - Watermark limit on each SE (typically 30-40 TB free space)
    - Allows to launch pre-staging with many files and let the system throttle
      - New files are replicated when old files are removed (timescale of days)

Feeding the TransformationSystem

- LHCb uses the Bookkeeping system (BK) to add files to the TS
  - Usually expressed as a BK "path"
    - Can also be an explicit production number
  - Example: `/LHCb/Collision16//RealData/Reco16/Stripping28r1//BHADRON.MDST`
  - Possibly add data quality criteria
    - E.g. avoid data flagged as BAD

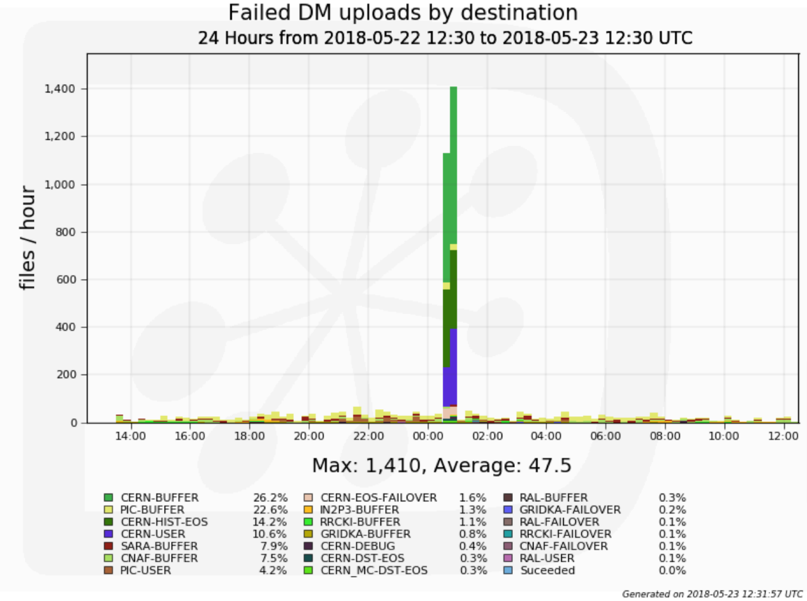
- BK query associated to a given transformation
  - Incremental periodic queries, add files to transformation
  - Daily full query (for safety / recovery of failures)

- Files can also be added "manually" to transformations
  - Good practice to not mix with BK query!

Data upload failover

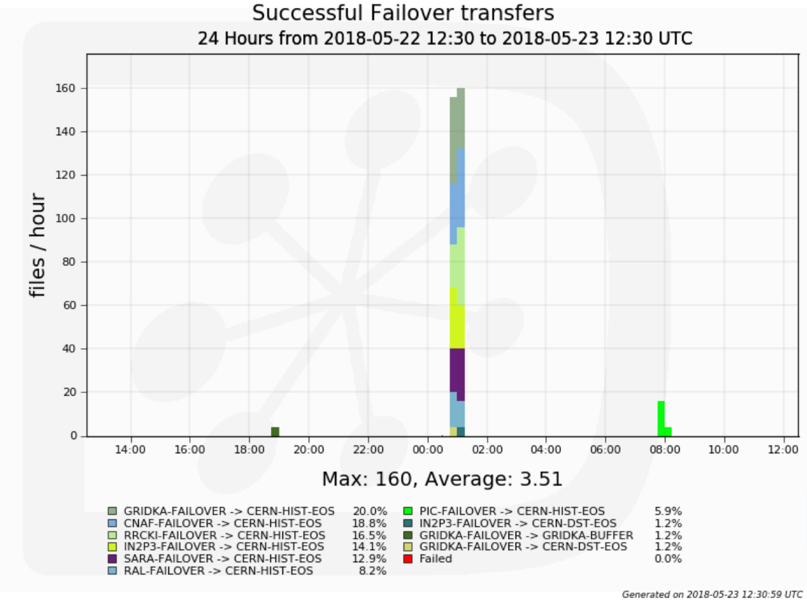
- If destination SE is banned, or if transfer fails
  - Upload the file to any of our “failover SEs” (one per Tier1)
  - Create an RMS request to transfer back the file to its final destination
- (Almost) no loss of output data
  - Unless there is a connectivity problem on the WN site
  - If this happens the job is Failed
- Other failover
  - If registration fails, a request is set (possibly for specific catalogs)
- If there is a request, the jobs remains in an intermediate status
  - Completed / Pending requests
  - This is a very bad name and should be changed to ??? (Completing?)
  - The RMS callback will eventually set the job to Done when the request is successful
- FTS is used for failover transfers

Example


Failed DM uploads by destination

24 Hours from 2018-05-22 12:30 to 2018-05-23 12:30 UTC

Generated on 2018-05-23 12:31:57 UTC


Successful Failover transfers

24 Hours from 2018-05-22 12:30 to 2018-05-23 12:30 UTC

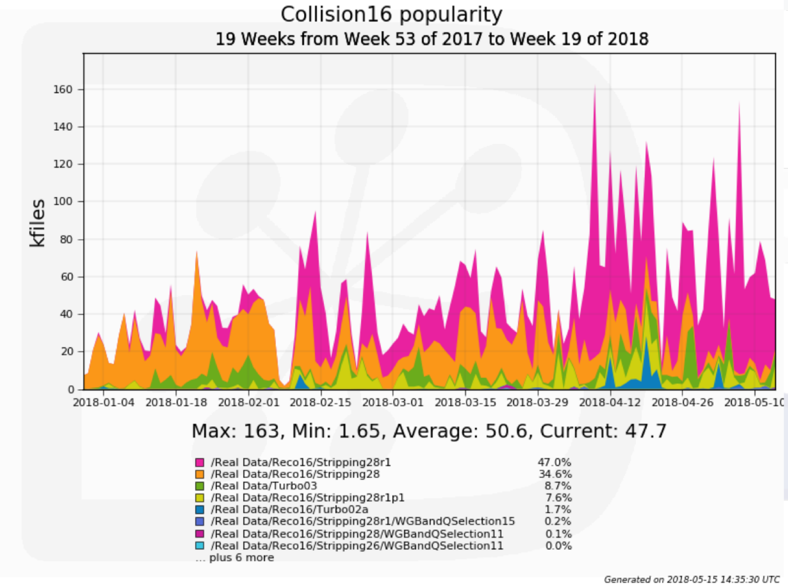
Generated on 2018-05-23 12:30:59 UTC

Storage accounting

- LHCb specific extension
- StorageUsageAgent: twice a day DFC scan to record per directory and per SE
  - Number of files
  - Size
- StorageHistoryAgent: uses output of the above but translates DFC directory into BK path
  - Records each entry into the accounting DB
  - Allows to make plots shown previously in this talk

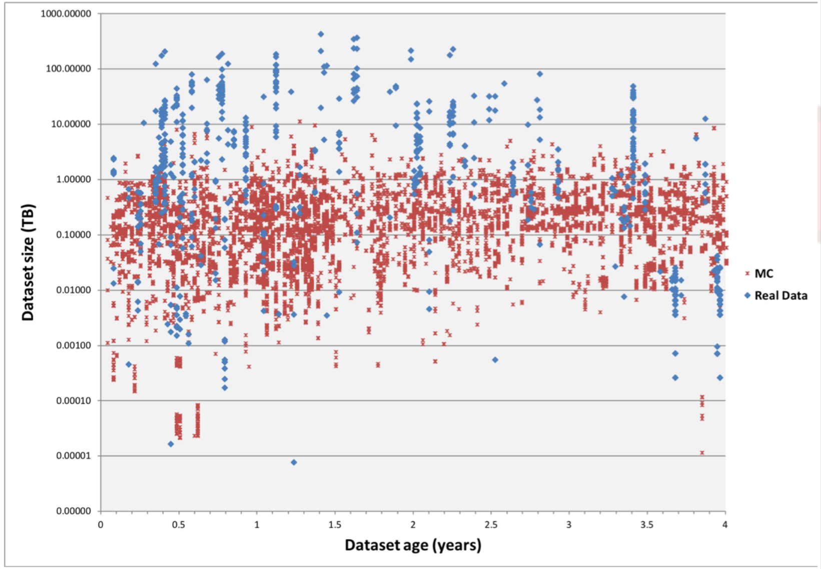
Popularity

- Each user job reports, per directory:
  - Number of files processed
  - SE used
- PopularityAgent:
  - Records in accounting DB this information, after transformation into BK path (similar to StorageHistoryAgent)
- PopularityAnalysisAgent
  - Creates huge CSV table
    - Used for manual DM checks and operations
  - May give hints for data removal
    - Based on big data algorithms
    - Developed by our Yandex colleagues

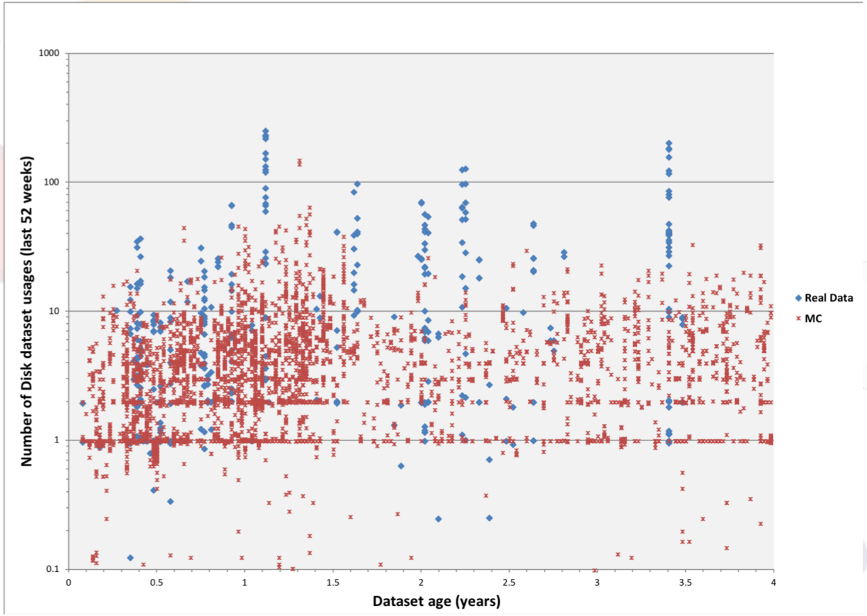


Plots from popularity analysis

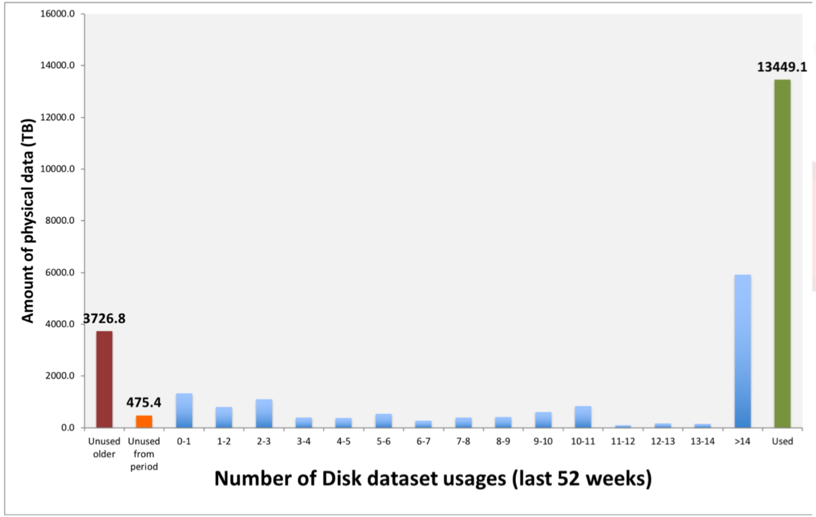
Dataset size vs age

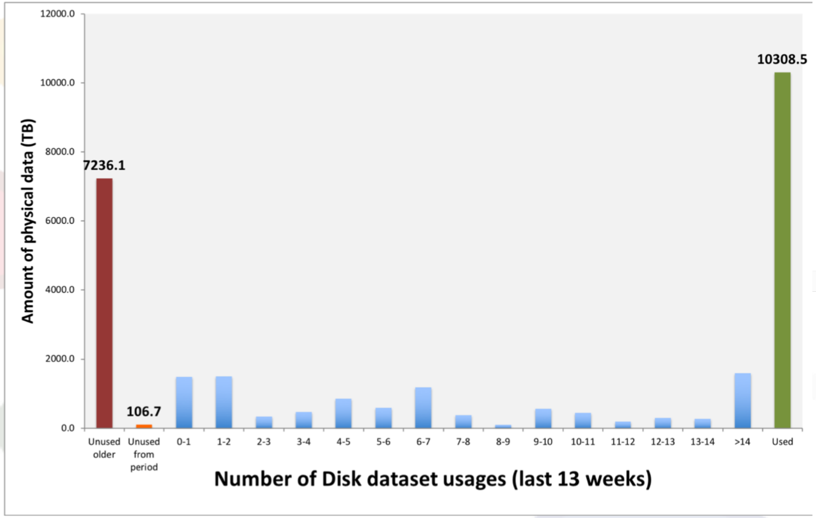


Dataset usages vs age



Popularity plots (standard for CRSG)





Conclusion

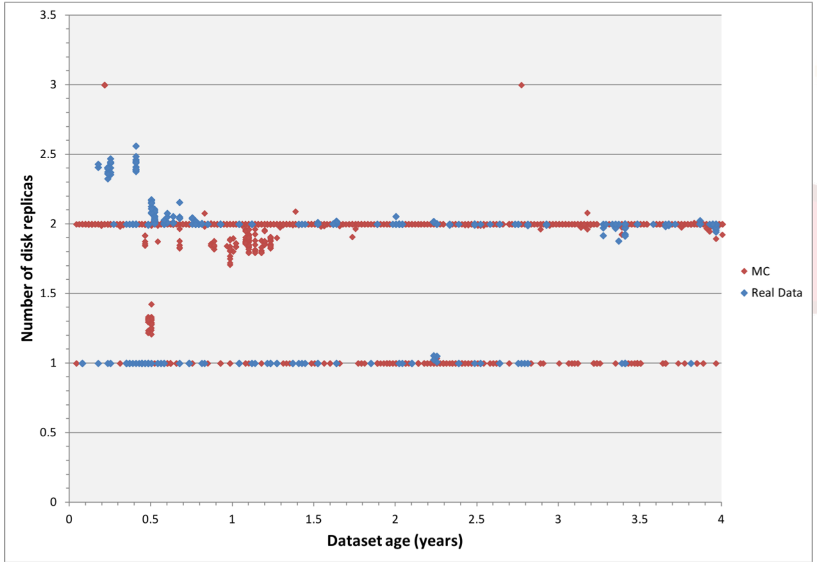
- Distributed Data management in LHCb is fully handled by LHCbDirac

- Heavy usage of the TransformationSystem + RMS + FTS
  - Input queries from the LHCb bookkeeping

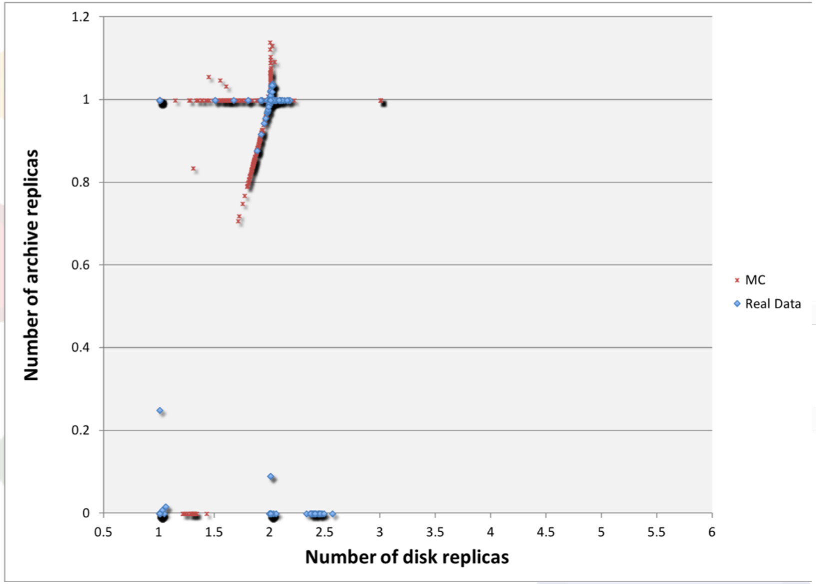
- LHCb-specific plugins developed for each purpose
  - However quite flexible (many configurable parameters)

- Re-processing campaigns quite successful using data from tape
  - Pre-staging launched a few weeks before the production
  - Automatic removal of pre-staged data after processing

Puzzled??



Number of disk replicas
3.5
3
2.5
2
1.5
1
0.5
0
0
0.5
1
1.5
2
2.5
3
3.5
4
Dataset age (years)
MC
Real Data



Number of archive replicas
1.2
1
0.8
0.6
0.4
0.2
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
5.5
6
Number of disk replicas
MC
Real Data