# Gestion des données LHCb : illustration avec DIRAC


A.Tsaregorodtsev,
CPPM, Marseille

Réunion France-Grilles/LCG-France,
27 novembre, Lyon

# Outline

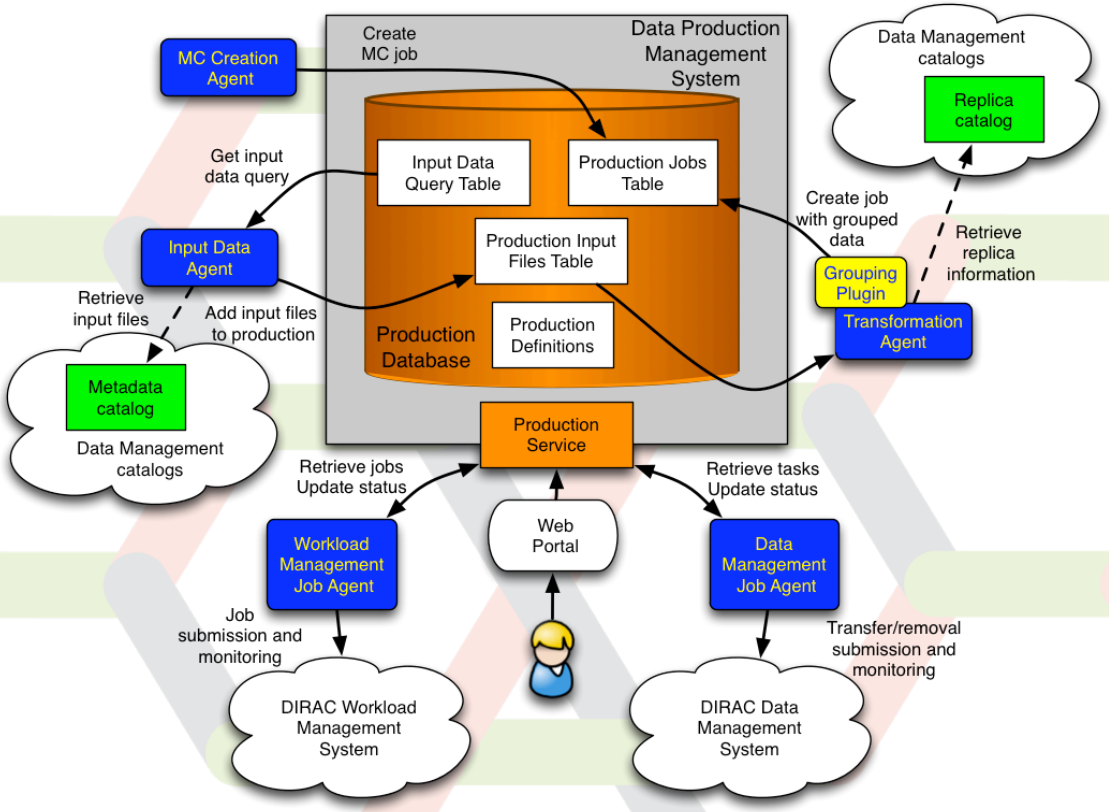
- Data placement, removal
  - Popularity
- Data integrity
- Data policies
- Conclusions

# Data placement

- General LHCb data placement rules:
  - Real data: one archive and 4 disk replicas
    - Archive: write once read never
  - MC: one archive and 3 disk replicas
  - In both cases one of the disk replicas is at CERN (EOS)
  - Includes derived data ( DSTs )
- Automated data replication using DIRAC Transformation and Request Management Systems
  - Manual or data driven
  - Asynchronous

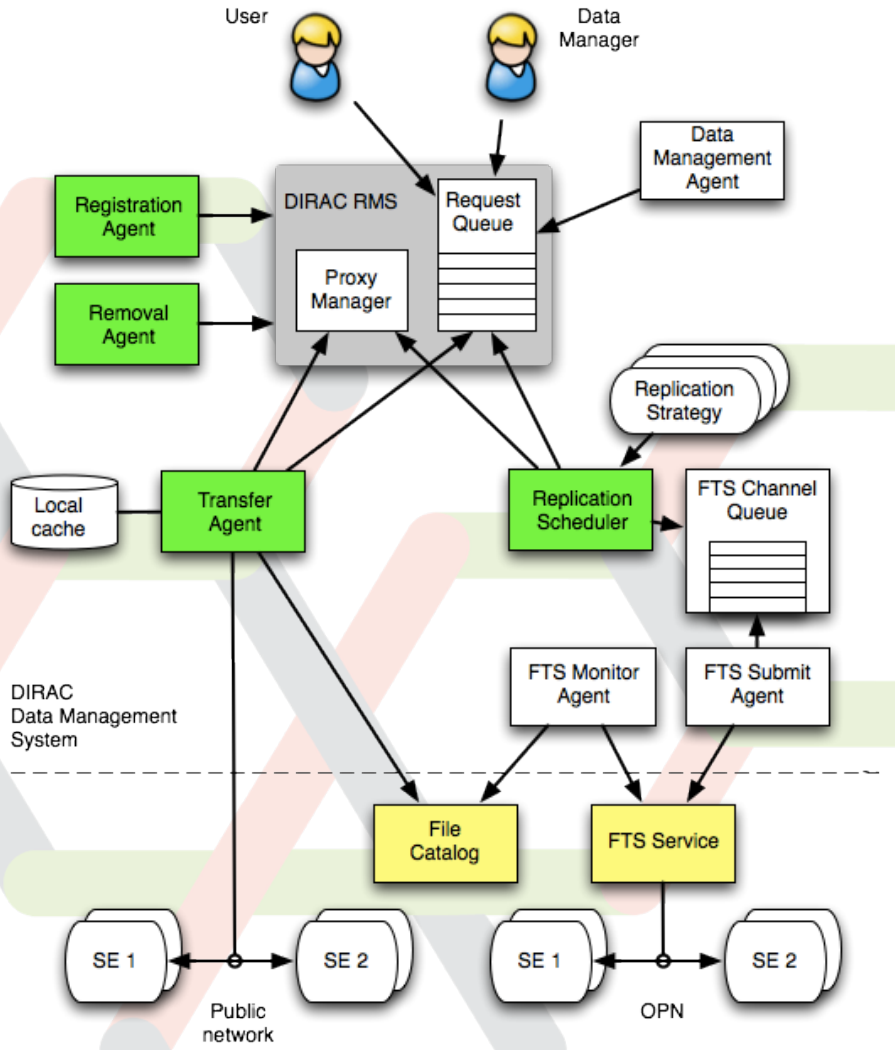
# LHCb Production system

- Based on the DIRAC Transformation System
  - Multiple extensions and custom plugins
- Data driven task generation
  - Triggered by new files registration in the catalog
- Generating both data processing and replication tasks
  - E.g. replication tasks according to the LHCb Computing Model



# Data Management

- Based on the Request Management System
  - Asynchronous data operations
  - transfers, registration, removal
  - Retries on failure
- Two complementary replication mechanisms
  - Transfer Agent
    - user data
    - public network
  - FTS service
    - Production data
    - Private FTS OPN network
    - FTS2 and FTS3



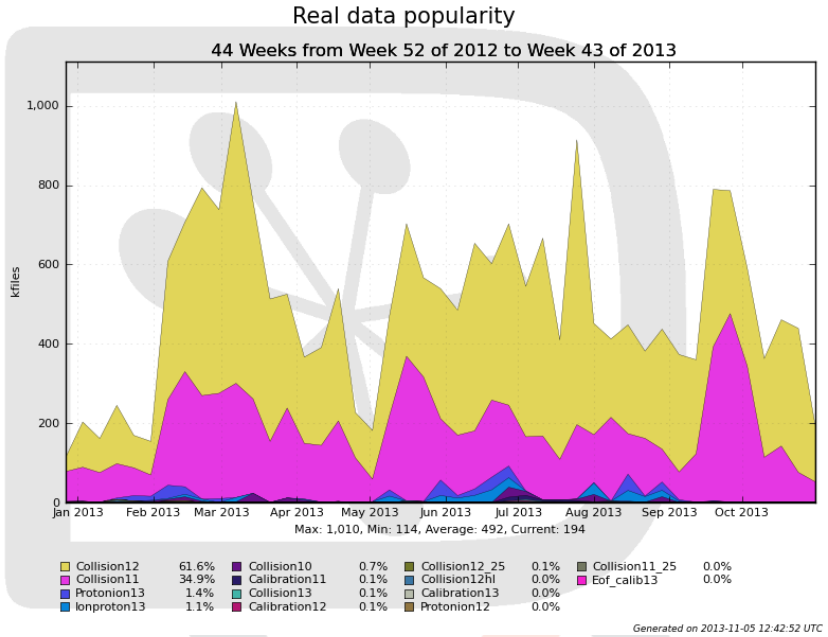
# Data popularity

- Not all datasets are used equally
  - Popularity increases while datasets are created
  - When a new dataset supersedes it, its popularity decreases
  - Some datasets are most likely read only once or twice (e.g. specific event types in MC, just for a cross-check of an analysis)
- For an efficient analysis, several replicas are necessary
  - Site downtime
  - Site overload
- Before using the Data Popularity in data placement decisions it should be measured

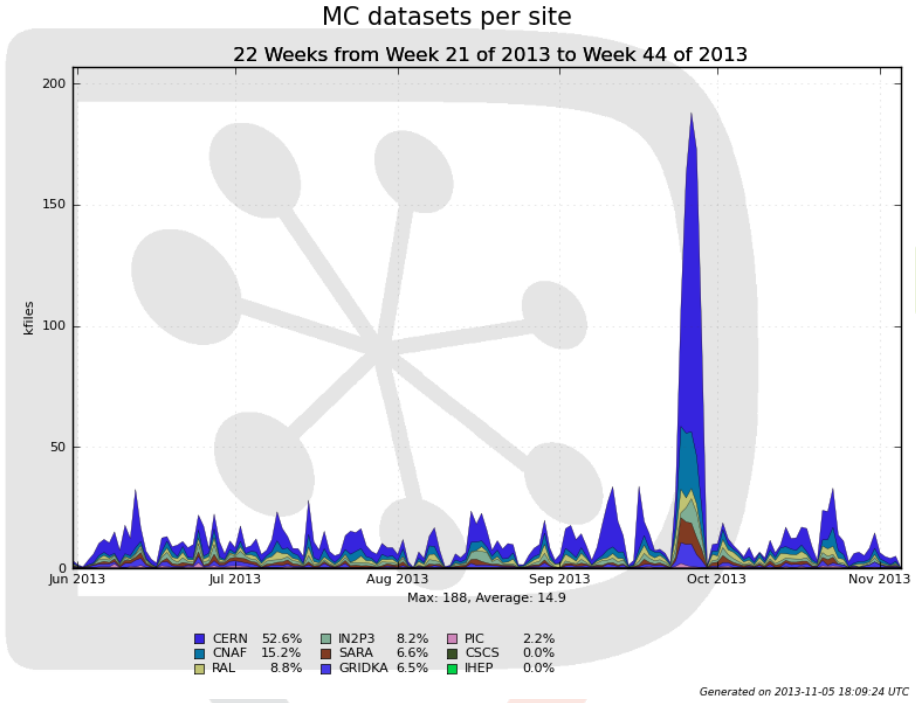
# Data popularity measurement

- Each user job reports to the central service
  - Which LFN directories are being accessed
    - How many files in each directory
  - Site name
  - Time stamp
  - Other items can be reported
    - User name
    - Job final status
- Accounting using DIRAC Accounting System
  - Accounted values per dataset
    - Number of accesses
    - Normalized number of accesses (fraction of dataset)
  - Produce popularity trend plots – number of accesses per time bin

# Example popularity plots




Per dataset




Per site

# Popularity usage in LHCb

- Used in taking decisions on removing old datasets
  - Not automated
- Possible strategies
  - Regular data placement according to Computing Model
  - Reduce number of replicas for unpopular data
    - For how long data should be unpopular
  - For unused data remove all replicas ( except archive )
- Still under discussion
  - Do we want to use popularity to create more replicas ?
  - Can popularity trends help predicting access patterns in the ( near ) future ?

# Support for popularity in DFC

- The DIRAC File Catalog has an experimental plugin to support popularity data
  - Last access date
    - Per file, directory or dataset
    - Date of the last replica lookup
  - Average number of accesses in the last predefined period, e.g. last week
    - Per file, directory or dataset
    - An estimation
- This is being evaluated now
  - Can slow down considerably replica look-ups

# Data integrity in LHCb

- Checking consistency of different name spaces ( chasing "dark data" )
  - Physical storage
  - Logical name space in the Replica Catalog ( LFC )
  - Logical name space in the Bookkeeping DB
- Using dumps of the SE and LFC name space
  - Tedious operation, T1 sites produce those on LHCb request
  - Should become easier with gfal2 – capable to produce recursive SE name space reports
    - Have to see the efficiency of those reports
  - DIRAC SE and FC can provide dumps through their service interface
- Regular checks using dedicated agents
  - LHCbDIRAC extension
    - Can be moved to the core library if need would be
- IntegrityDB
  - Collecting reports on each failure to access data
  - Difficult to automate recovery measures – need for manual operations

# Storage usage monitoring

- Necessary to monitor available and used space
  - To make data placement decisions
  - To apply VO policies – quotas
- In LHCb now is based on the LFC data
  - Traversing the LFC name space to build per user, per directory, per SE storage usage reports
  - Heavy operation
    - Can take several days to perform an update
    - Heavy load on LFC
- LHCb is planning to use DFC also for storage monitoring

# Storage Usage in DFC

- The Storage Usage is built in the DFC natively
- Using special prefilled tables
  - Updated at each new file or replica insertion
    - More efficient with bulk insertion
  - Instant reports for any directory
    - User data is stored in "Home" directories which allows to follow storage consumption by users
  - Possibility of instant "*du*" command

# Storage Usage in DFC

```
FC:/> size -l /lhcb/user/a/atsareg/1
directory: /lhcb/user/a/atsareg/1
Logical Size: 134,756,846 Files: 498 Directories: 500

    StorageElement  Size          Replicas
-------------------------------------------
  1 IN2P3-USER      20,254,050    75
  2 CNAF-USER       18,363,672    68
  3 RAL-USER        16,473,294    61
  4 CERN-USER       19,443,888    72
  5 GRIDKA-USER     21,064,212    78
  6 SARA-USER       20,254,050    75
  7 PIC-USER        18,903,780    70
-------------------------------------------
    Total           134,756,946   499
Query time 0.98 sec
```

- Report of storage usage for any directory
  - Whole VO data
  - Per user data
  - “Logical” storage
    - LFNs, sum of the LFN sizes
  - “Physical” storage
    - Physical replicas, total volume per Storage Element

# Conclusions

- Data management is the most complex activity in any LHC experiment
- DIRAC provides support for the most common data management tasks ( automated, failure-safe data placement, integrity checking, storage monitoring )
- Data popularity is very popular ☺, but is not clear how it can help in smart data placement taking into account future trends
- Efficient storage monitoring is essential in applying VO policies ( quotas )