DIRAC File Catalog

A.Tsaregorodtsev,

CPPM, Marseille

- The DIRAC project has as a goal to provide a full middleware stack to build distributed computing systems

  - With the possibility to easily integrate third party services

- File Catalog is a mandatory part of any distributed Data Management system

- The grid de facto standard LFC catalog has certain limitations in functionality and performance

- Rich experience acquired with the LHCb Data Management system

- The DIRAC File Catalog subproject was launched 3 years ago

2

- DFC is fully built in the DISET framework

- Part of the DIRAC set of services
  - Coupled to Configuration and Monitoring services
  - MySQL backend

- Accessible with a standardized FileCatalog client

- Client tools
  - command line
  - CLI
  - Python API

3
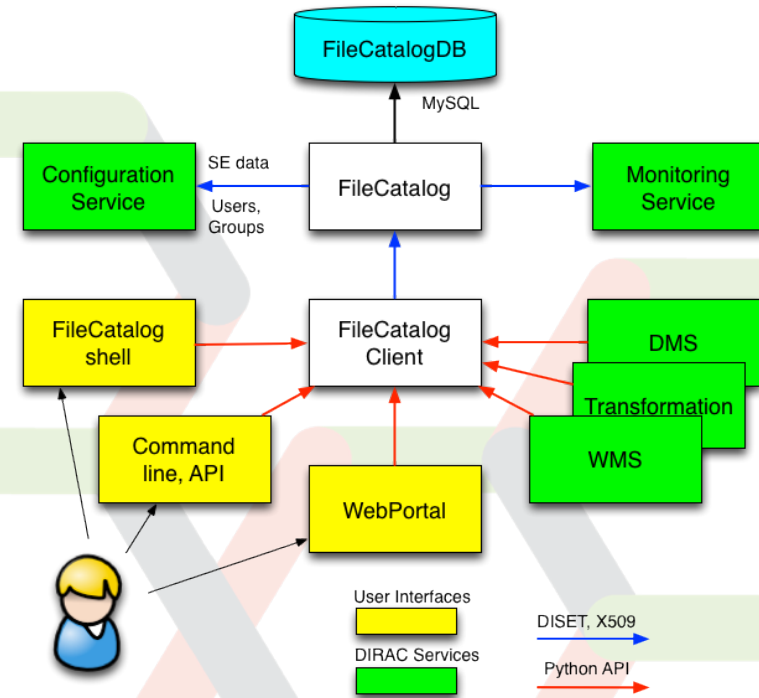
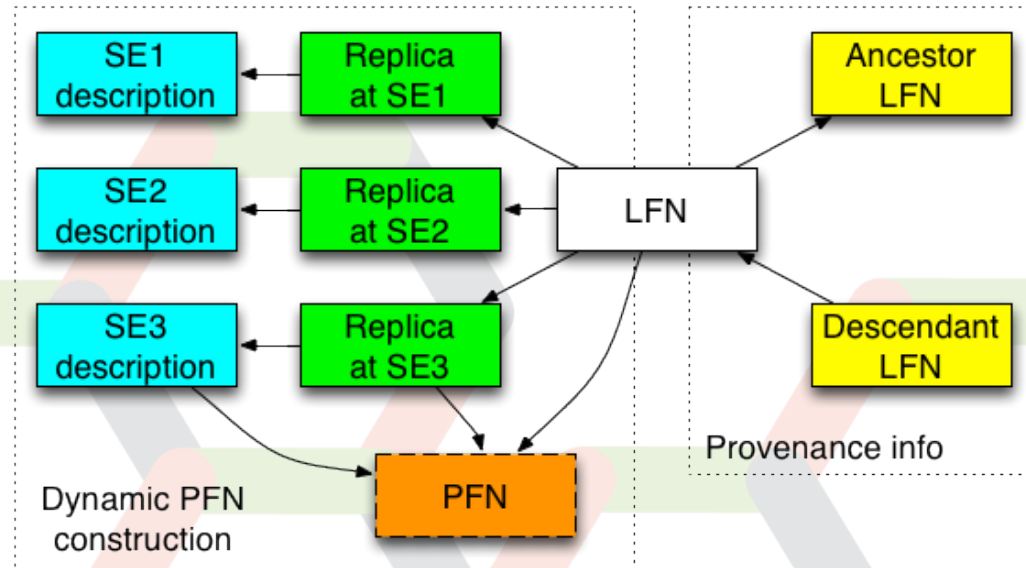# Replica Catalog

- ## Standard Replica Catalog functionality
  - Optimized for bulk queries
- ## On the fly PFN construction
  - Small database footprint
  - Pattern used in LHCb
- ## Ancestor-descendent relations
  - Basic provenance information
  - Possibility to select ancestors in a given generations

- Efficient Storage Usage reports
- Using no special prefilled tables
  - Exact snapshot of the current DFC status
  - No overhead when adding new files and replicas
  - Slow
    - In the case of LHCb (~8M files, ~12M replicas, 1.5M directories)
      - ☐ Up to 20 secs for "heavy" users
      - ☐ Up to 2 mins for the overall report
- Using special prefilled tables
  - Updated at each new file or replica insertion
    - More efficient with bulk insertion
  - Instant reports for any directory
  - Possibility of instant "*du*" command

```
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 community data

▸ Per user data

▸ "Logical" storage

  ▸ LFNs, sum of the LFN sizes

▸ "Physical" storage

  ▸ Physical replicas, total volume per Storage Element

▸ 6

Data Management components

- For DIRAC users the use of any Storage Element or File Catalog is transparent

  - Community choice which components to use

  - Several File Catalogs can be used in parallel

    - Complementary functionality

    - Redundancy

- Users see depending on the DIRAC Configuration

  - Logical Storage Elements

   - e.g. DIRAC-USER, M3PEC-disk

  - Logical File Catalog
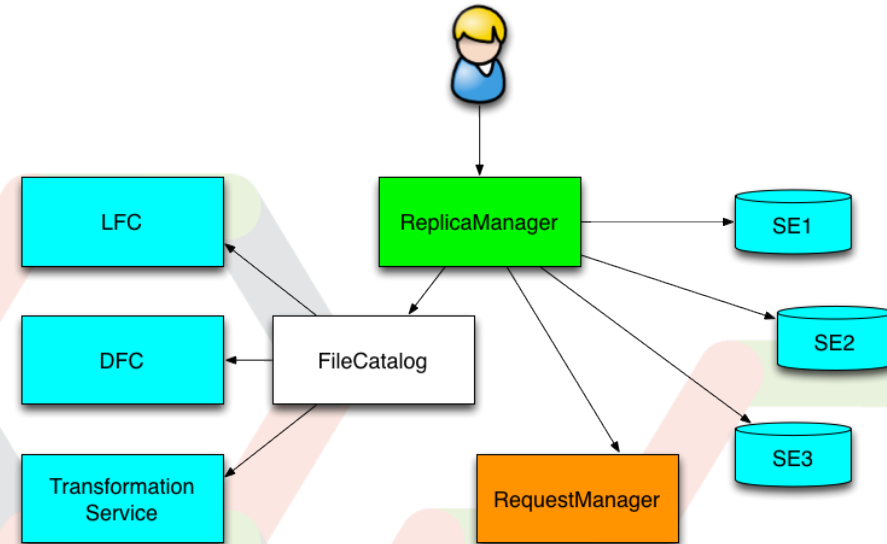
- Using LFC and DFC in parallel is perfectly possible

  - Provided the use of the DIRAC file naming conventions

- If the LFC part is not following the DIRAC file naming conventions

  - The whole PFN for replicas must be registered

    - Reducing the DFC efficiency

- Several communities are willing to evaluate the DFC usage as a possible LFC replacement

  - Biomed, Auger, …

  - Unclear LFC prospects due to the end of the EMI project

Combined use with LFC

- The LHCb case
  - LFC is the main replica catalog
    - Central instance at CERN
  - DFC is alternative write-only catalog
    - Kept in sync with LFC via a common FileCatalog client
    - Synchronization ensured by the failover mechanism
    - Can replace the actual heavy StorageUsage service
  - Deployment plan
    - Installing an empty DFC service
    - Starting to put new data in both catalogs
      - ☐ Ignoring errors due to orphan replicas
    - Copying the existing LFC data to DFC in parallel

File Catalog Metadata

- Similar functionality with the AMGA metadata service

  - But coupled with the replica catalog to boost efficiency

- Metadata can be associated with each directory as key:value pairs to describe its contents

  - Int, Float, String, DateTime value types

- Some metadata variables can be declared indices

  - Those can be used for data selections

- Subdirectories are inheriting the metadata of their parents

- Data selection with metadata queries. Example:

  - ```
    find . Meta1=Value1 Meta2>3 Meta2<5 Meta3=2,3,4
    ```

- File metadata can also be defined
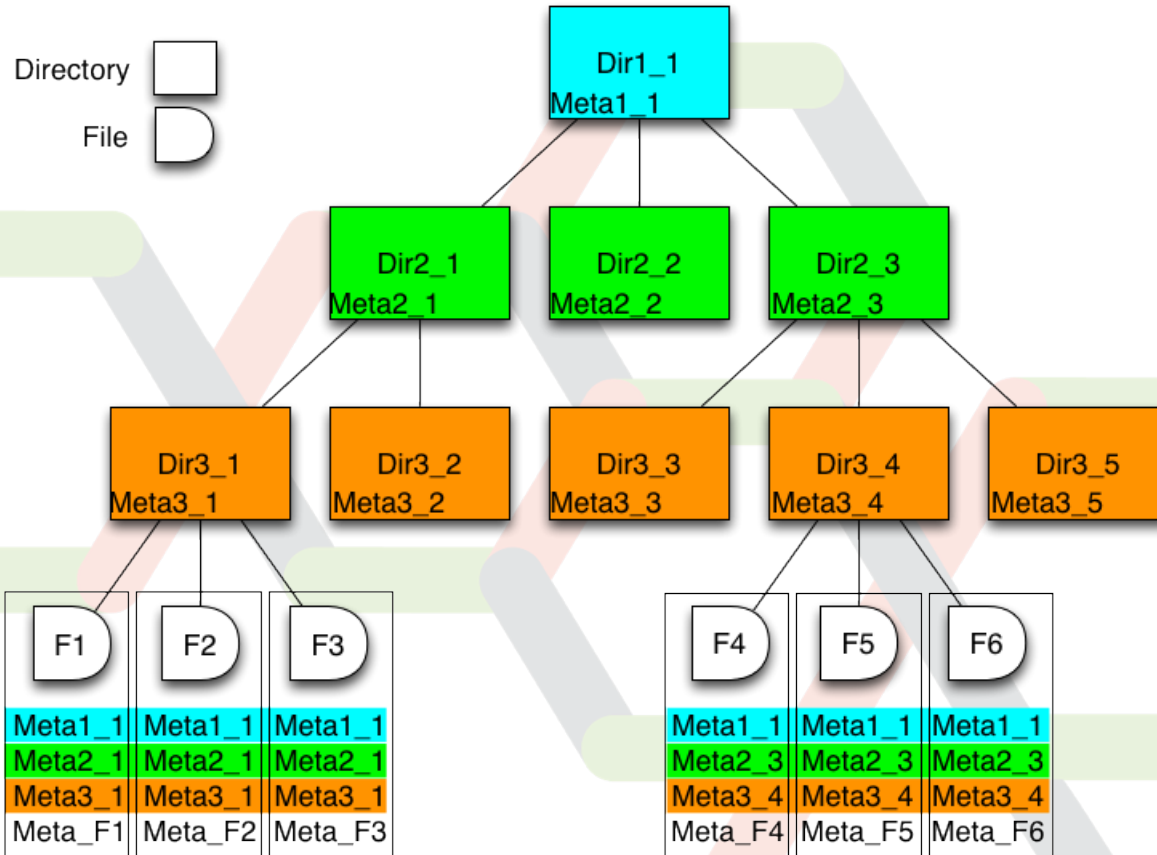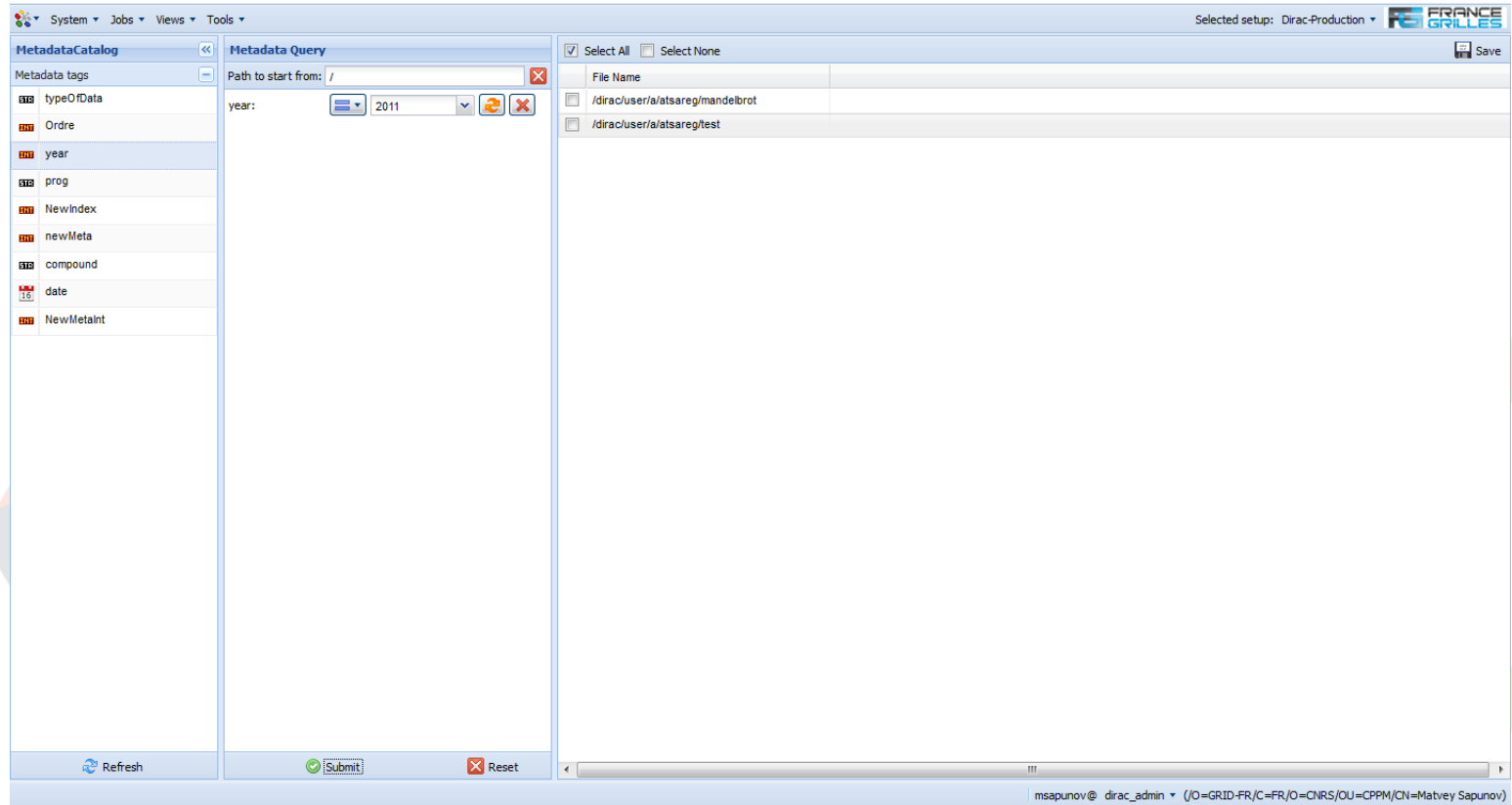
- Data search by metadata
- Basic information about data
- Data downloads
- More functionality to come

Metadata development

- The currently available DirectoryMetadata module
  - Subdirectory inherits the parent metadata
  - Subdirectory can not override parent metadata values
  - Simple to implement, allows for dynamic metadata optimization
  - Allow for a simple and intuitive GUI interface
  - Limiting in the description of real life cases
- The new DirectoryTagMetadata module is in the works
  - Inspired by the CTA case
  - Subdirectories can provide additional values to the parents
  - Allow for data tags – metadata with multiple values
  - The work is in progress
- Different modules can be chosen by configuration parameters of the given DFC service

DIRAC File Catalog evaluation

- ILC/CLIC Collaboration experience

  - ~1M files

  - Intensive use of metadata, provenance data

File search by metadata
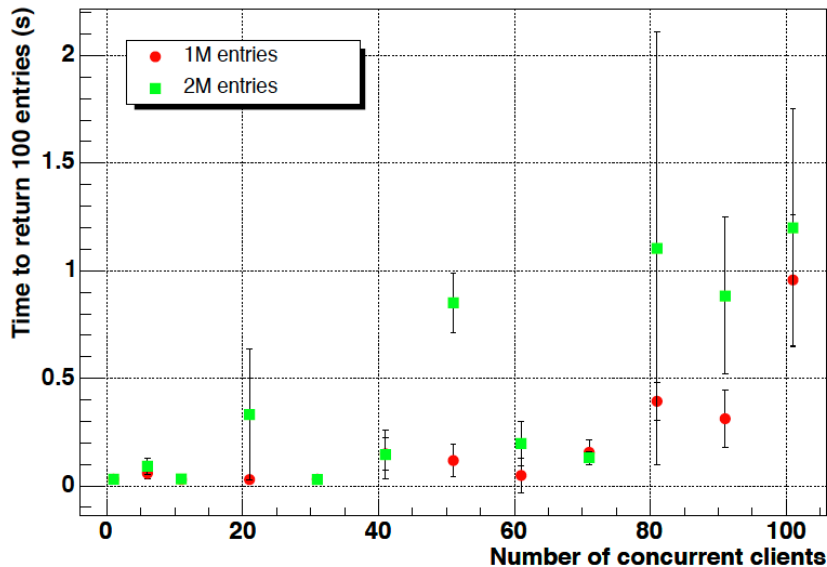
Replica information
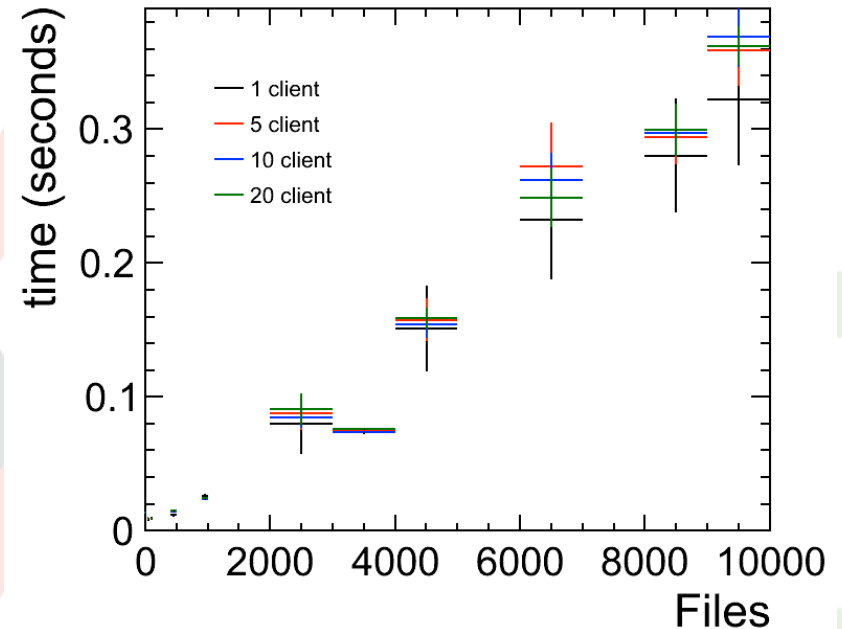
- BES Collaboration made a thorough comparison of DFC vs AMGA

  - Similar performance

  - More suitable functionality

Good example of cooperation

- The DFC development is a good example of cooperation of developers and users
  - ILC/CLIC, BES, CTA
    - Many bug reports and fixes
    - Many optimization suggestions
    - Comparison with LFC inspired many optimization ideas
  - BES
    - Comparison/competition with AMGA inspired many optimization ideas, still more to come
    - Many improvements in the catalog console interface
      - Command and data automatic completion
  - CTA
    - Many fruitful discussions on the nature of the metadata and the best way to express it in the DFC service
    - More flexible metadata schema to suit the CTA needs is in the works

- 15

- Metadata optimization
  - Metatags
  - Metaqueries
  - Query efficiency optimization
- Better Directory and File Metadata integration
  - Transparent to the user
  - Dynamically reorganized to increase efficiency
- Tighter coupling with the Transformation System
  - Possibility to register data driven operations
    - Possible now with the Transformation DB as an independent catalog
  - Basis for the Replication Service
    - Similar to the Globus Online or iRods services

16

- DFC is a service combining both Replica and Metadata Catalog functionality

- It is created based on a rich experience with the LFC and Bookkeeping service in the LHCb experiement

- It is becoming a mature service used in several projects with performance and functionality equivalent to LFC and AMGA

- More developments in managing Metadata are on the way