DIRAC Data Management Framework

A. Tsaregorodtsev,
CPPM-IN2P3-CNRS

ISGC'2016, Taipei

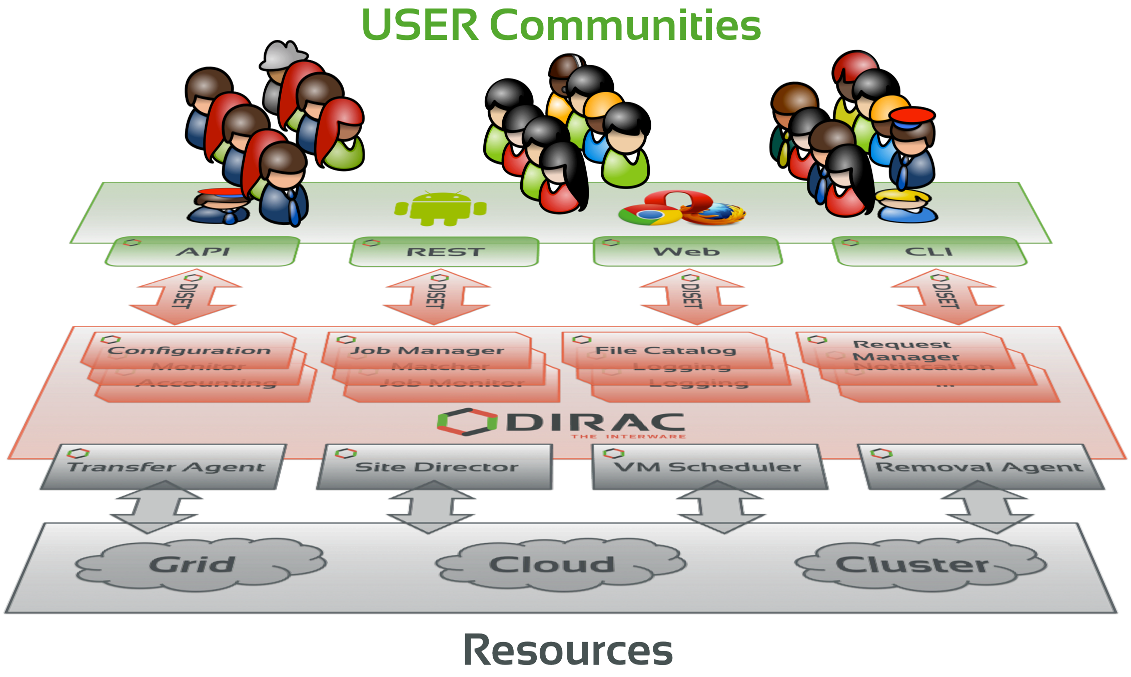
DIRAC
THE INTERWARE

Plan

- DIRAC Project quick reminder
- Data Management System problem
- DIRAC Data Management Model
- DMS Basic Components
- Managing Large Data Flows
- Conclusions

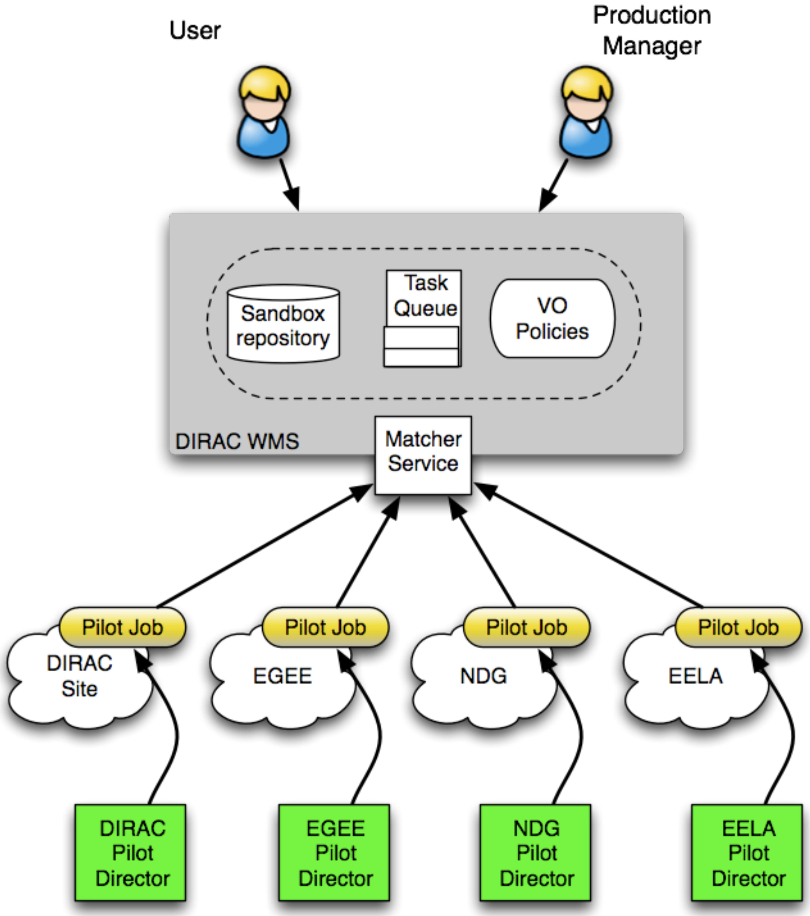
Interware

- DIRAC provides all the necessary components to build ad-hoc grid infrastructures **interconnecting** computing resources of different types, allowing **interoperability** and simplifying **interfaces**. This allows to speak about the DIRAC ***interware***.



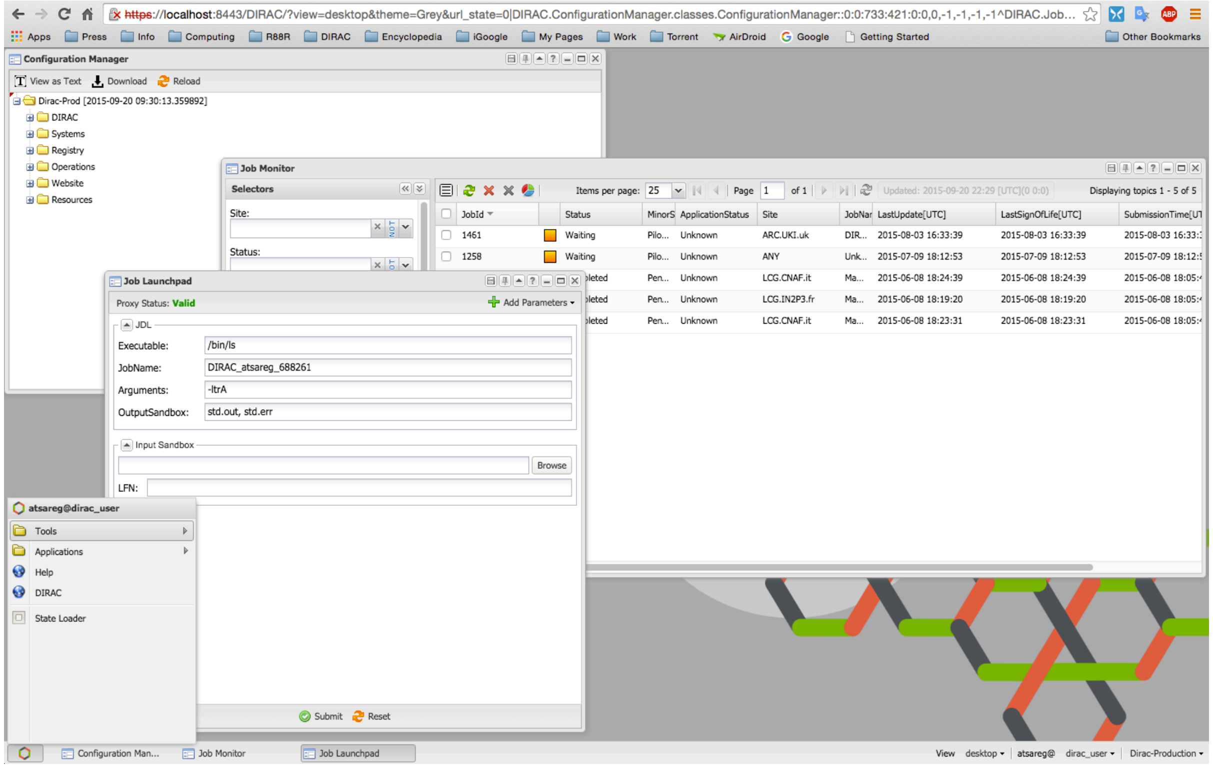
Interware

- Pilot based Workload Management provides abstraction of Computing Resources
  - Allows to combine heterogeneous resources in a transparent way
- Similar patterns are applied also for the Data Management System of DIRAC



Distributed computer

- DIRAC forms an abstraction of a simple computer which has the power of thousands of CPUs and petabytes of storage behind the scene coming from various sources and various technologies (grids, clouds, etc)



JobId		Status	MinorS	ApplicationStatus	Site	JobNar	LastUpdate[UTC]	LastSignOfLife[UTC]	SubmissionTime[UT
1461	🟧	Waiting	Pilo...	Unknown	ARC.UKI.uk	DIR...	2015-08-03 16:33:39	2015-08-03 16:33:39	2015-08-03 16:33:
1258	🟧	Waiting	Pilo...	Unknown	ANY	Unk...	2015-07-09 18:12:53	2015-07-09 18:12:53	2015-07-09 18:12:
		pleted	Pen...	Unknown	LCG.CNAF.it	Ma...	2015-06-08 18:24:39	2015-06-08 18:24:39	2015-06-08 18:05:
		pleted	Pen...	Unknown	LCG.IN2P3.fr	Ma...	2015-06-08 18:19:20	2015-06-08 18:19:20	2015-06-08 18:05:
		pleted	Pen...	Unknown	LCG.CNAF.it	Ma...	2015-06-08 18:23:31	2015-06-08 18:23:31	2015-06-08 18:05:

Job Launchpad

Proxy Status: Valid

Add Parameters

JDL

Executable: /bin/ls

JobName: DIRAC_atsareg_688261

Arguments: -ltrA

OutputSandbox: std.out, std.err

Input Sandbox

Browse

LFN:

- DIRAC Web Portal is following the computer desktop paradigm
    - Natural for a non-expert user

DM Problem to solve

- Data is partitioned in files
- File replicas are distributed over a number of Storage Elements world wide

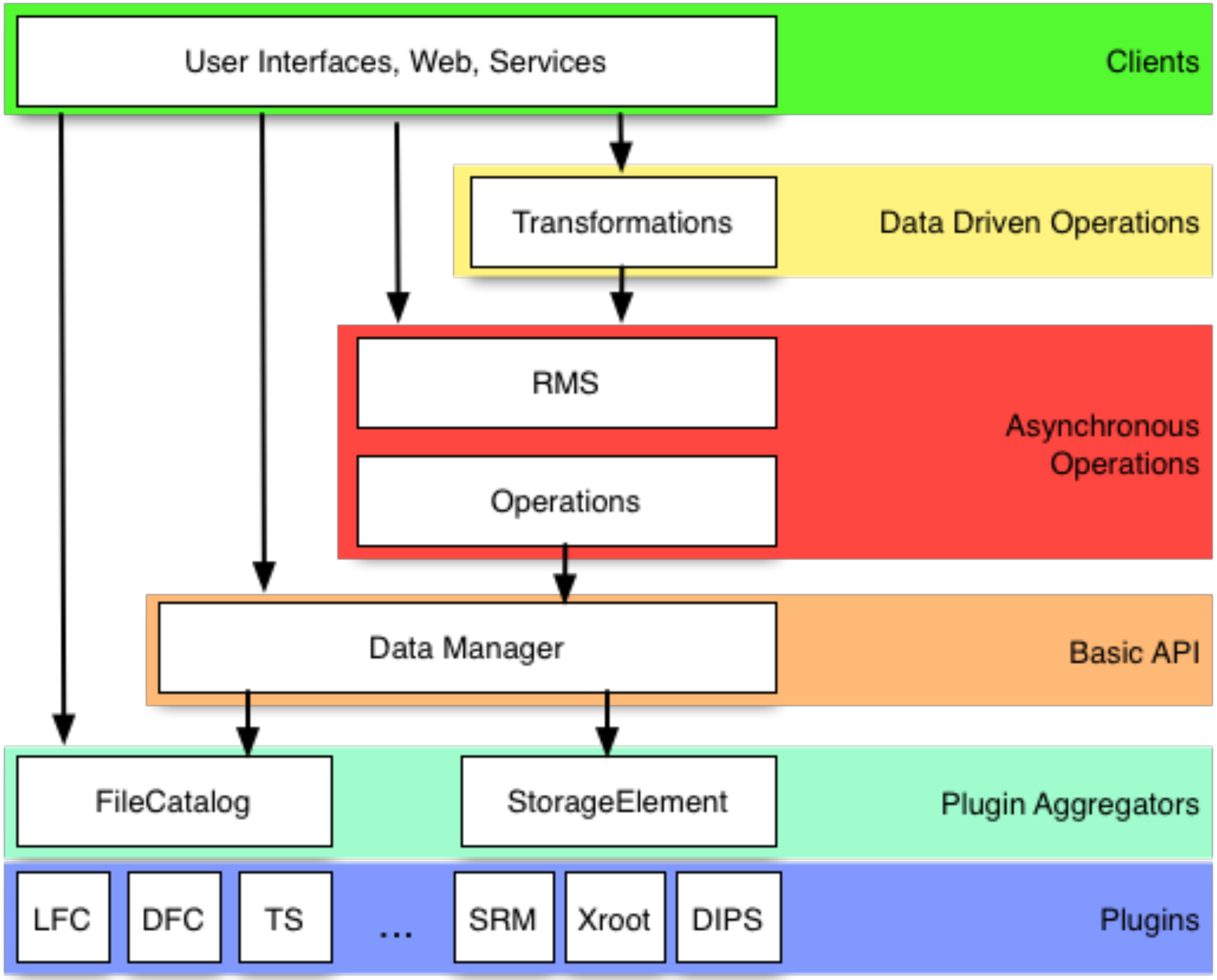
- Data Management tasks
  - Initial File upload
  - Catalog registration
  - File replication
  - File access/download
  - Integrity checking
  - File removal

- Need for transparent file access for users
- Often working with multiple (tens of thousands) files at a time
  - Make sure that ALL the elementary operations are accomplished
  - Automate recurrent operations

DM Software Stack

Services

- File Catalog
- Storage Element
- Storage Proxy
- Staging
- FTS Manager
- Data Logging
- Data Integrity



User Interfaces, Web, Services — Clients

Transformations — Data Driven Operations

RMS, Operations — Asynchronous Operations

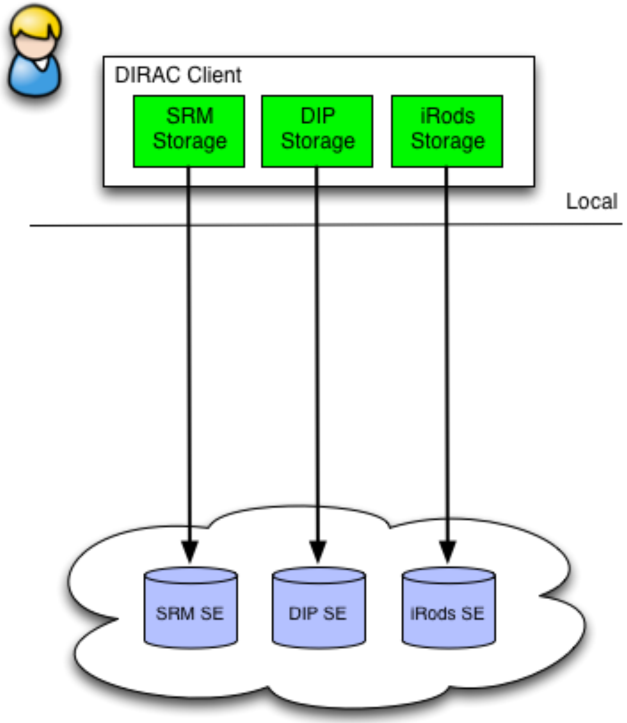
Data Manager — Basic API

FileCatalog, StorageElement — Plugin Aggregators

LFC, DFC, TS, …, SRM, Xroot, DIPS — Plugins

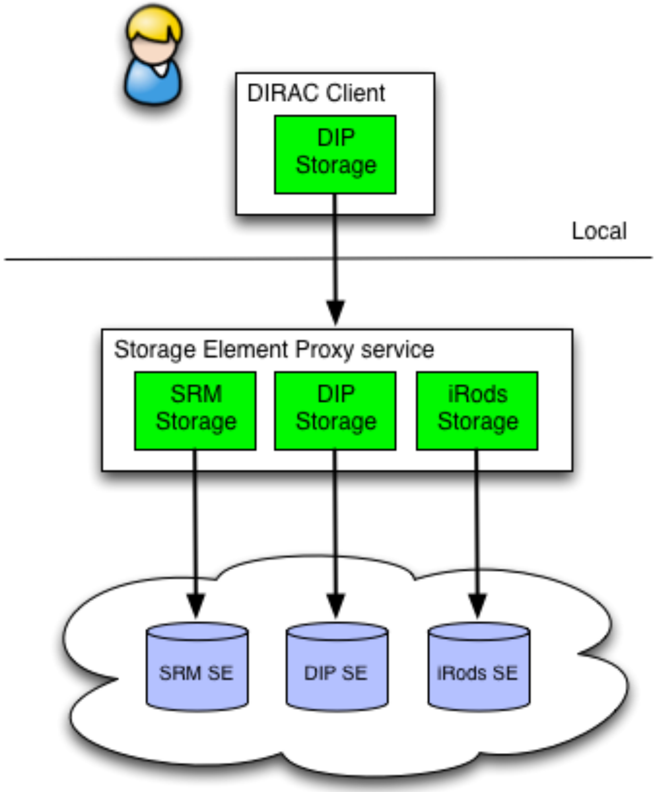
Storage plugins

- Storage element abstraction with a client implementation for each access protocol
  - DIPS, SRM, XROOTD, RFIO, etc
  - gfal2 based plugin gives access to all protocols supported by the library
    - DCAP, WebDAV, S3, …
- Each SE is seen by the clients as a logical entity
  - With some specific operational properties
  - SE's can be configured with multiple protocols



Storage Element Proxy

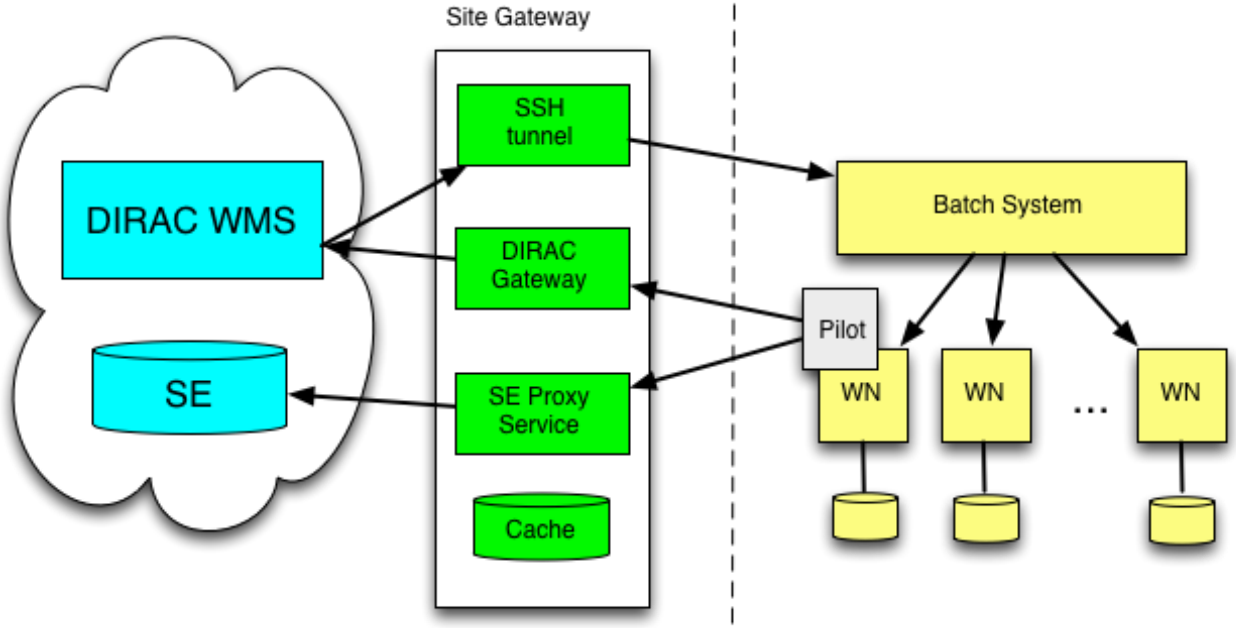
- SE Proxy Service translates the DIRAC data transfer protocol to a particular storage protocol
  - Using DIRAC authentication
  - Using credentials specific to the target storage system
  - Example: access to iRods storage
    - Using login/password authentication



- SE Proxy Service allows access to storages not having access libraries on a given client machine
  - DIRAC or HTTP protocol

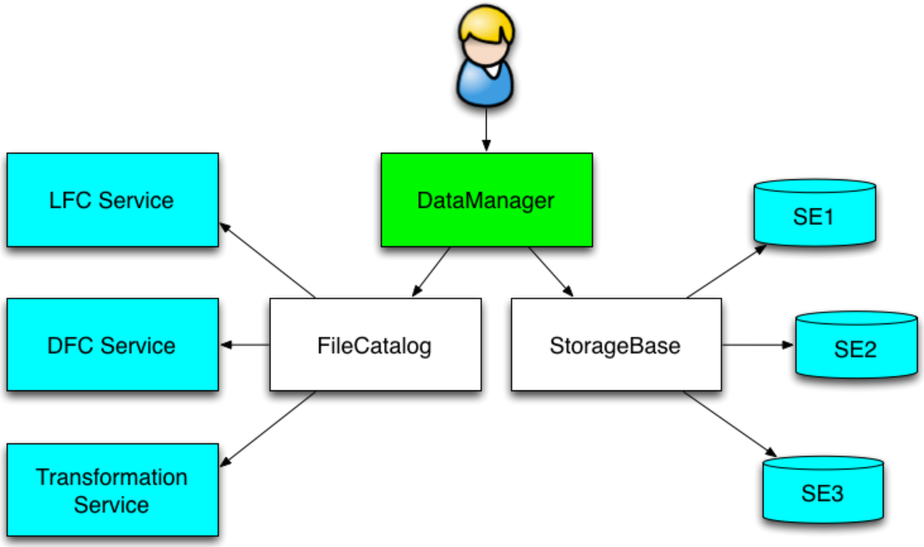
HPC example

- Pilot submitted to the batch system through the SSH tunnel
- Pilot communicates with the DIRAC service through the Gateway proxy service
- Output upload to the target SE through the SE proxy



File Catalog

- Central File Catalog (DFC, LFC, …) is maintaining a single global logical name space
- Several catalogs can be used together
  - The mechanism is used to send messages to "pseudocatalog" services, e.g.
    - Transformation service (see later)
    - Bookkeeping service of LHCb
  - A user sees it as a single catalog with additional features
- DataManager is a single client interface for logical data operations



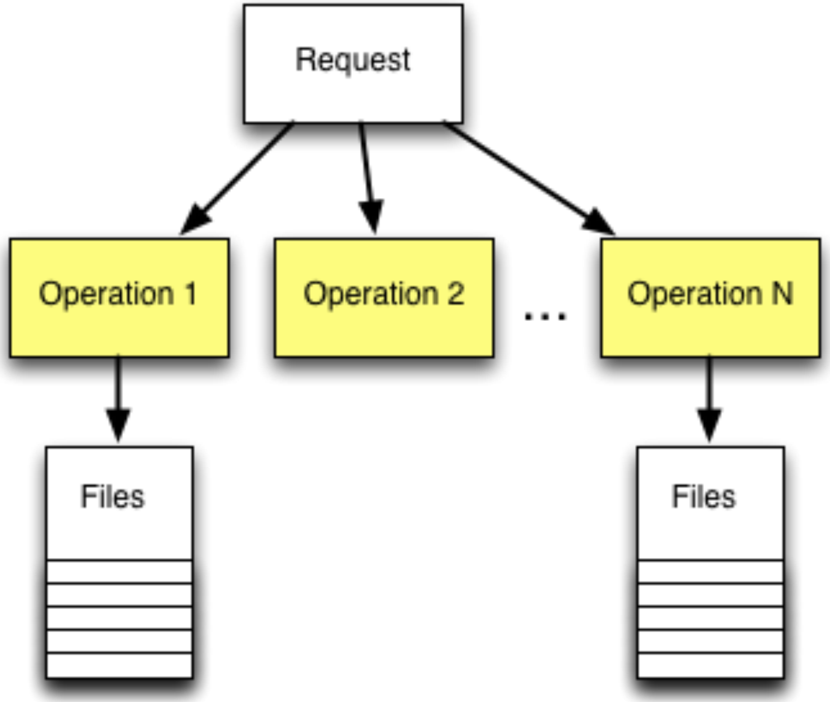
Asynchronous operations

- Request Management System (RMS)
  - Keeps the database of Requests

- Request is a sequence of Operations executed in a certain order
  - Operations can have associated Files

- Each Operation type has a dedicated Executor
  - Execution is done with the credentials of the Owner of the Request
    - E.g. user defined operations
  - Examples: ForwardDISET, ReplicateFile, RemoveFile

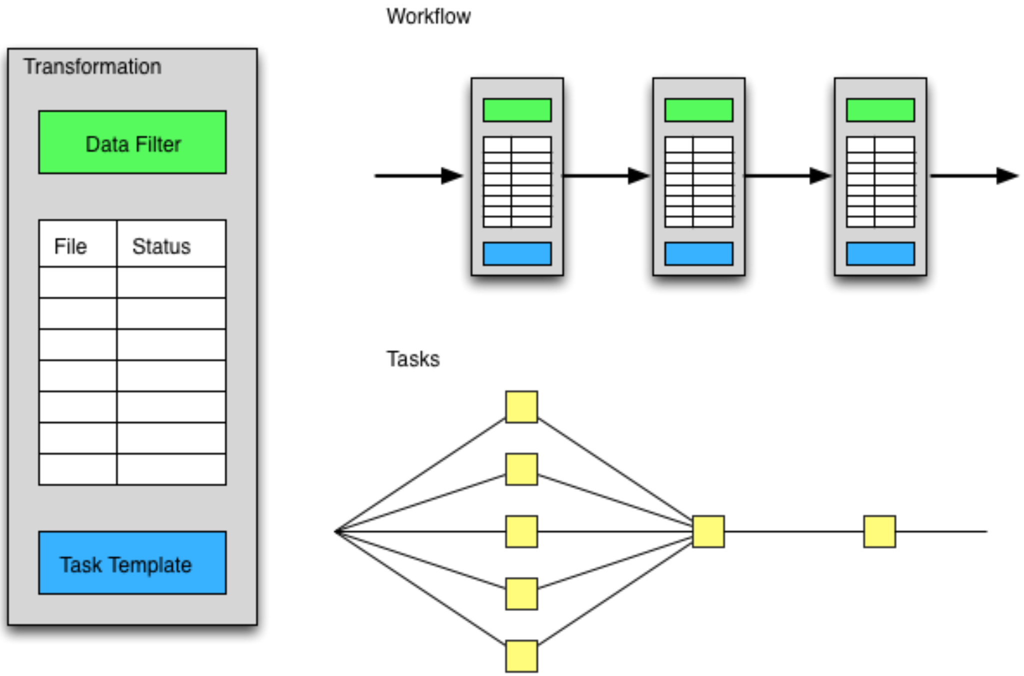
- Executors are invoked by an agent running in a background
  - Retry logic in case of failures



Transformation System

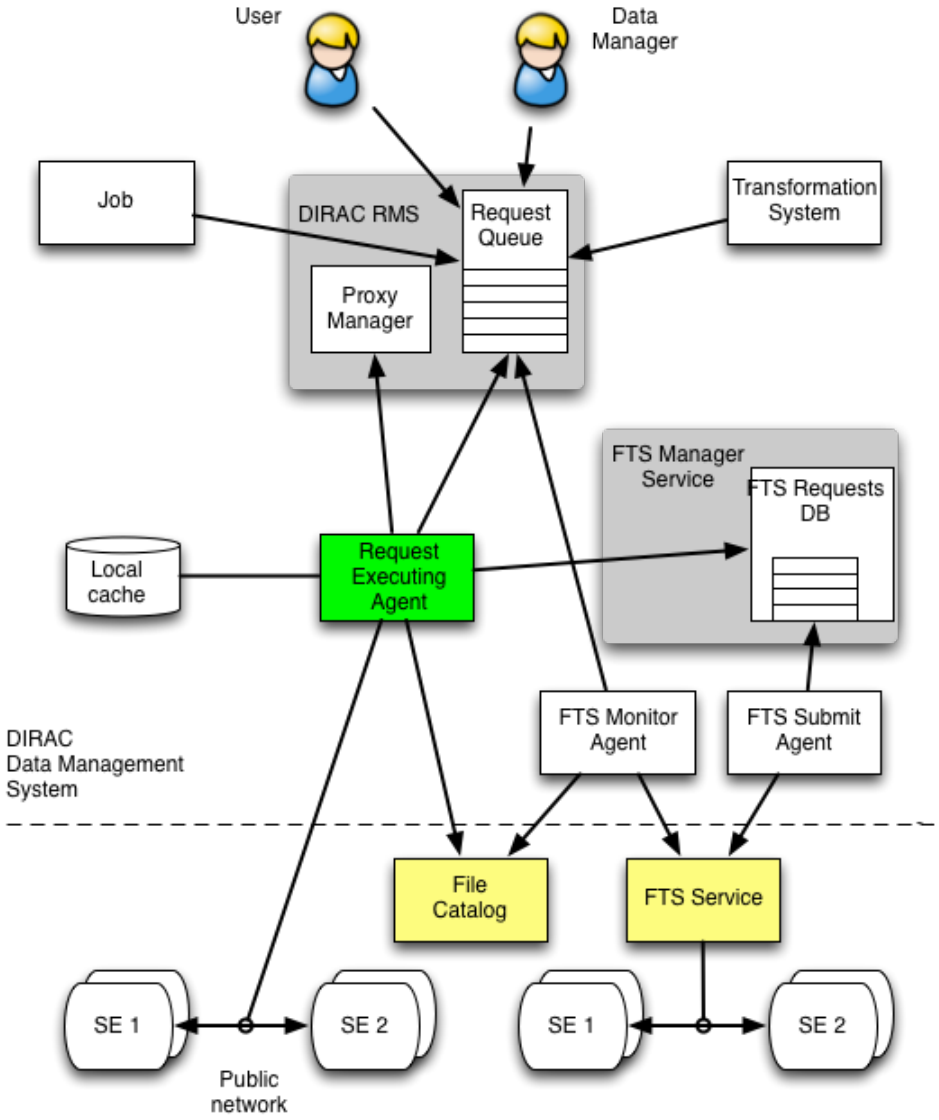
- Data driven workflows as chains of data transformations
  - Transformation: input data filter + recipe to create tasks
  - Tasks are created as soon as data with required properties is registered into the system
  - Tasks: jobs, data operations, etc

- Transformations can be used for automatic data driven bulk data operations
  - Scheduling RMS tasks
  - Often as part of a more general workflow



Bulk transfers

- Replication/Removal Requests with multiple files are stored in the RMS
  - By users, data managers, Transformation System
- The Replication Operation executor
  - Performs the replication itself or
  - Delegates replication to an external service
    - E.g. FTS
  - A dedicated FTSManager service keeps track of the submitted FTS requests
  - FTSMonitor Agent monitors the request progress, updates the FileCatalog with the new replicas



File Catalog

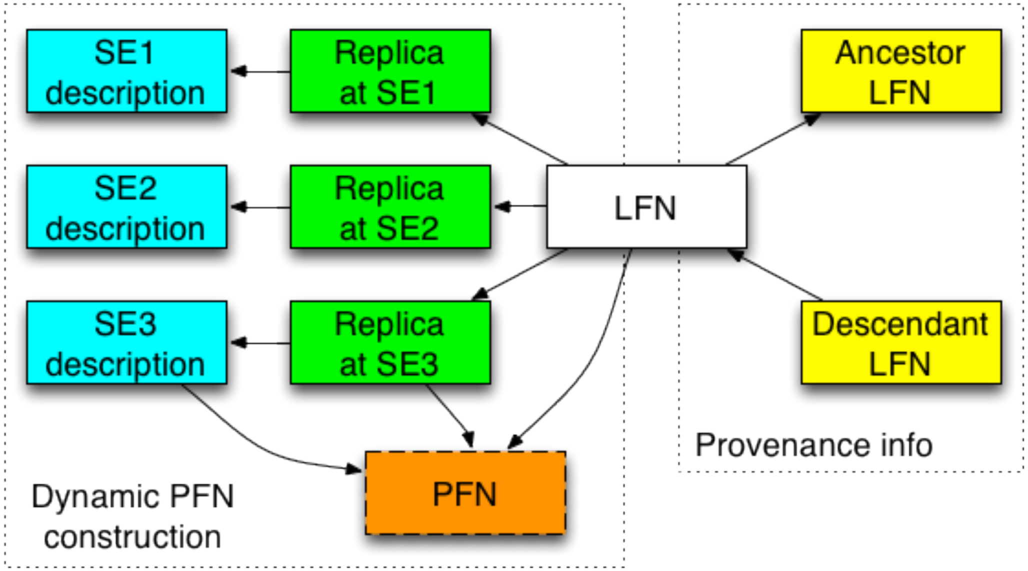
- DFC is the central component of the DIRAC Data Management system

- Defines the single logical name space for all the data managed by DIRAC

- Together with the data access components DFC allows to present data to users as single global file system

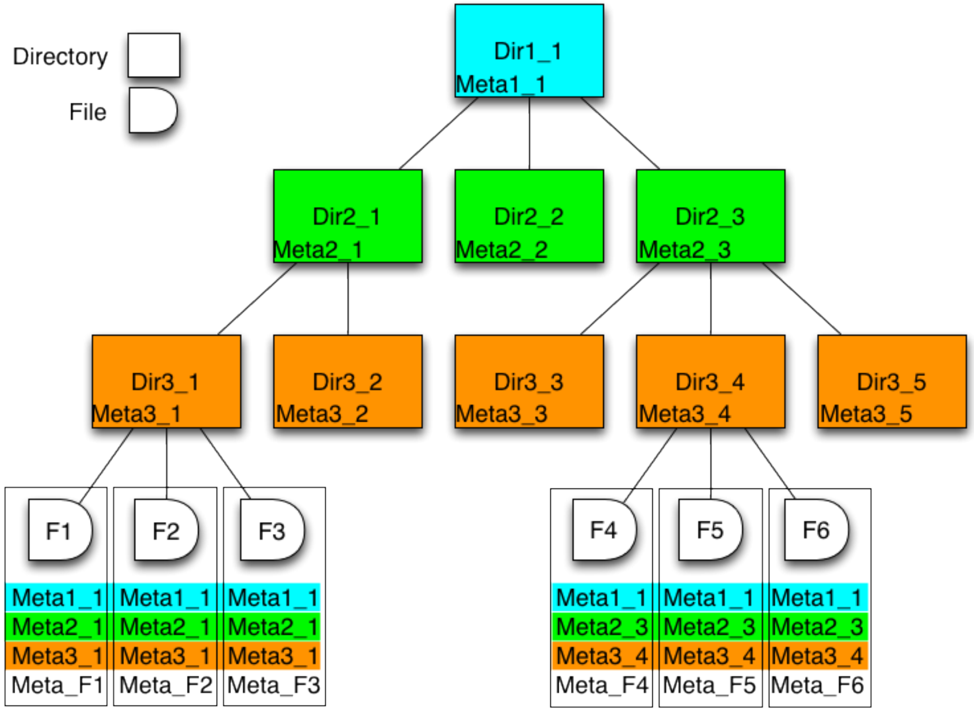
File Catalog: Replicas

- File standard metadata
  - Size, ownership, time stamps, ACL, checksum
- Standard Replica Catalog functionality
  - Optimized for bulk queries
- On the fly PFN construction
  - Small database footprint
  - Full PFN can be stored if necessary
- Ancestor-descendent relations
- Efficient storage usage reports



File Catalog: Metadata

- DFC is Replica and Metadata Catalog
  - User defined metadata
  - The same hierarchy for metadata as for the logical name space
    - Metadata associated with files and directories
    - Allow for efficient searches
  - Efficient Storage Usage reports
    - Suitable for user quotas



- Example query:
  - `find /lhcb/mcdata LastAccess < 01-01-2012 GaussVersion=v1,v2 SE=IN2P3,CERN Name=*.raw`

Support for datasets

- Datasets defined as a resulting list of files from a given metaquery
  - Particular case: all the files under a given directory
- Dataset objects are stored in the same directory hierarchy as files
  - ACLs, ownership, show up in the output of ***ls*** command as for files
- Datasets can be frozen in order not to change from one query to another
  - Can be refreshed by an explicit command, a quick check if changed since the last update
- Datasets can be annotated
- Operations on datasets
  - Replica lookup for all the files in a dataset
  - Total size, number of files report
  - Replication, removal, etc

DFC usage examples

- LHCb accomplished migration from LFC to DFC
  - ~17M files, 7M directories
  - Needed to develop a specific ACL plugin where several DIRAC groups have same ACLs for a given data
  - Not using the Metadata features of the DFC except for the Storage Usage reports
  - Using Transformation System of DIRAC for bulk data driven operations (e.g. replication, processing tasks submission, etc)
- ILC, BES III, CTA use intensively DFC as both Replica and Metadata Catalog
  - BES III performed a detailed performance comparison with the AMGA metadata service
- Pierre Auger Observatory
  - ~30M files
  - Working on complex metadata queries and dataset algebra (dataset relations, intersections, unions, etc)
- Eiscat-3D prototype
  - ~8M files (complete sample of ~100M files)
  - Scalability and usability tests with metadata operations
- FG-DIRAC multi-community service
  - ~1.5M files
  - Several VOs using the same catalog service

DM interfaces

- Command line tools
  - Multiple dirac-dms-… commands
- COMDIRAC
  - Representing the logical DIRAC file namespace as a parallel shell
  - **dls, dcd, dpwd, dfind, ddu** etc commands
  - **dput, dget, drepl** for file upload/download/replication
- Web Interface
  - Using a standard file browser paradigm
    - Possibility to define metadata queries
  - Under development
    - Better connection to other systems (WMS)
    - Better support of the DIRAC "computer" paradigm

DIRAC for CTA: DIRAC File Catalog

- In use since 2012 in parallel with LFC. Full migration to DFC in summer 2015
- More than 21 M of replicas registered
- About 10 meta-data defined to characterize MC datasets

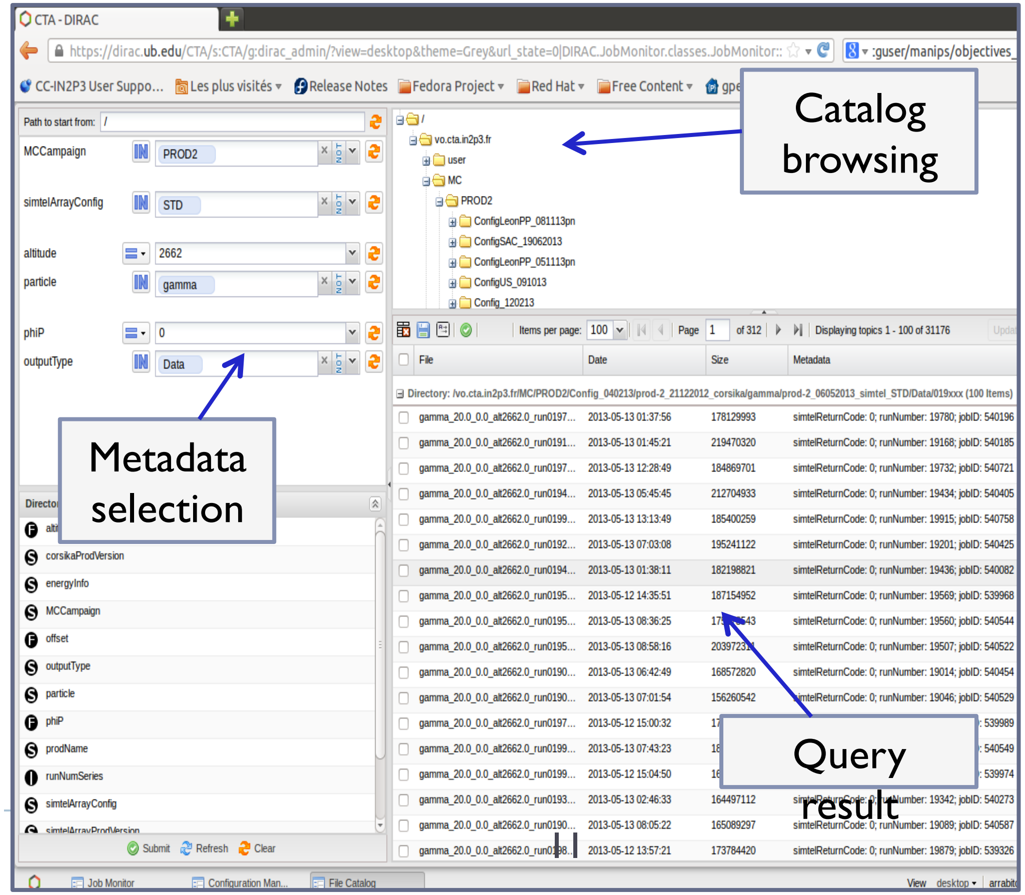
Query example:

cta-prod3-query --site=Paranal --particle=gamma --tel_sim_prog=simtel --array_layout=hex --phiP=180 --thetaP=20 --outputType=Data

Typical queries return several hundreds of thousands of files

L.Arrabito, LUPM

DFC web interface



CTA - DIRAC

https://dirac.ub.edu/CTA/s:CTA/g:dirac_admin/?view=desktop&theme=Grey&url_state=0|DIRAC.JobMonitor.classes.JobMonitor::

Path to start from: /

MCCampaign	IN	PROD2
simtelArrayConfig	IN	STD
altitude	=	2662
particle	IN	gamma
phiP	=	0
outputType	IN	Data

Metadata selection

Directory: alti[illegible] · corsikaProdVersion · energyInfo · MCCampaign · offset · outputType · particle · phiP · prodName · runNumSeries · simtelArrayConfig · simtelArrayProdVersion

Submit · Refresh · Clear

/ › vo.cta.in2p3.fr › user · MC › PROD2 › ConfigLeonPP_081113pn · ConfigSAC_19062013 · ConfigLeonPP_051113pn · ConfigUS_091013 · Config_120213

Catalog browsing

Items per page: 100 · Page 1 of 312 · Displaying topics 1 - 100 of 31176

Directory: /vo.cta.in2p3.fr/MC/PROD2/Config_040213/prod-2_21122012_corsika/gamma/prod-2_06052013_simtel_STD/Data/019xxx (100 Items)

File	Date	Size	Metadata
gamma_20.0_0.0_alt2662.0_run0197...	2013-05-13 01:37:56	178129993	simtelReturnCode: 0; runNumber: 19780; jobID: 540196
gamma_20.0_0.0_alt2662.0_run0191...	2013-05-13 01:45:21	219470320	simtelReturnCode: 0; runNumber: 19168; jobID: 540185
gamma_20.0_0.0_alt2662.0_run0197...	2013-05-13 12:28:49	184869701	simtelReturnCode: 0; runNumber: 19732; jobID: 540721
gamma_20.0_0.0_alt2662.0_run0194...	2013-05-13 05:45:45	212704933	simtelReturnCode: 0; runNumber: 19434; jobID: 540405
gamma_20.0_0.0_alt2662.0_run0199...	2013-05-13 13:13:49	185400259	simtelReturnCode: 0; runNumber: 19915; jobID: 540758
gamma_20.0_0.0_alt2662.0_run0192...	2013-05-13 07:03:08	195241122	simtelReturnCode: 0; runNumber: 19201; jobID: 540425
gamma_20.0_0.0_alt2662.0_run0194...	2013-05-13 01:38:11	182198821	simtelReturnCode: 0; runNumber: 19436; jobID: 540082
gamma_20.0_0.0_alt2662.0_run0195...	2013-05-12 14:35:51	187154952	simtelReturnCode: 0; runNumber: 19569; jobID: 539968
gamma_20.0_0.0_alt2662.0_run0195...	2013-05-13 08:36:25	17[illegible]543	simtelReturnCode: 0; runNumber: 19560; jobID: 540544
gamma_20.0_0.0_alt2662.0_run0195...	2013-05-13 08:58:16	2039723[illegible]	simtelReturnCode: 0; runNumber: 19507; jobID: 540522
gamma_20.0_0.0_alt2662.0_run0190...	2013-05-13 06:42:49	168572820	simtelReturnCode: 0; runNumber: 19014; jobID: 540454
gamma_20.0_0.0_alt2662.0_run0190...	2013-05-13 07:01:54	156260542	simtelReturnCode: 0; runNumber: 19046; jobID: 540529
gamma_20.0_0.0_alt2662.0_run0197...	2013-05-12 15:00:32	17[illegible]	[illegible]: 539989
gamma_20.0_0.0_alt2662.0_run0199...	2013-05-13 07:43:23	18[illegible]	[illegible]: 540549
gamma_20.0_0.0_alt2662.0_run0199...	2013-05-12 15:04:50	16[illegible]	[illegible]: 539974
gamma_20.0_0.0_alt2662.0_run0193...	2013-05-13 02:46:33	164497112	simtelReturnCode: 0; runNumber: 19342; jobID: 540273
gamma_20.0_0.0_alt2662.0_run0190...	2013-05-13 08:05:22	165089297	simtelReturnCode: 0; runNumber: 19089; jobID: 540587
gamma_20.0_0.0_alt2662.0_run0198...	2013-05-12 13:57:21	173784420	simtelReturnCode: 0; runNumber: 19879; jobID: 539326

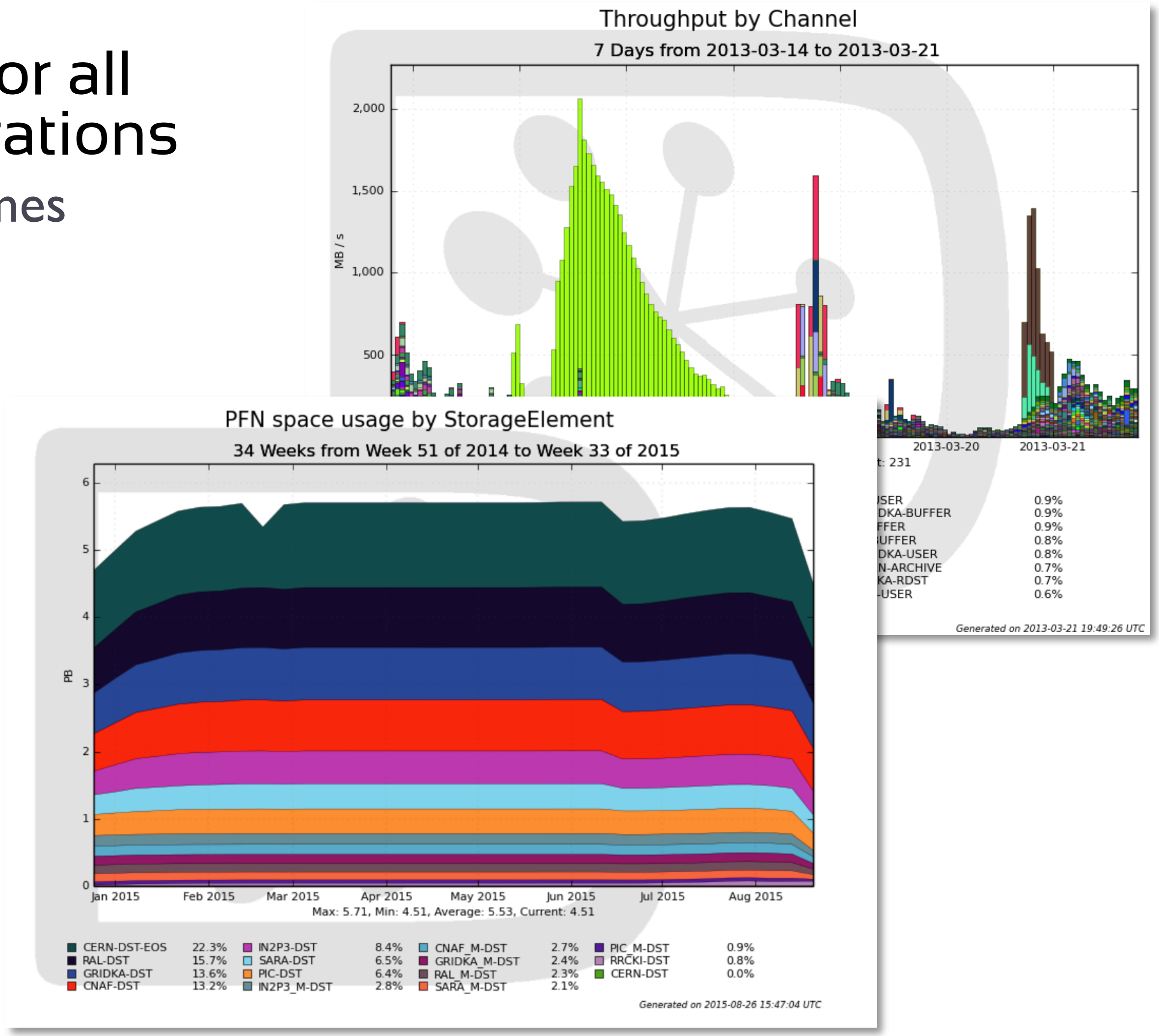
Query result

FSDIRAC

- DIRAC extension to mount the DIRAC File System as a local one
- Using FUSE, fuse-python
- Needs X509 credentials to browse data
- Logical File Namespace can be looked up in a standard file browser on Mac, Linux
- Mostly for quick read-only access
- Write access is complicated especially in the case of multiple file replicas

Accounting

- DIRAC includes a general purpose Accounting System
  - For all the activities
- Accounting reports for all the data related operations
  - Transfer rates and volumes
  - Storage usage
  - Success/failure rates
  - Etc
- Plots selectable by
  - Storage Elements
  - Transfer channels
  - Owner of the data
  - Dates
  - Etc, etc



Throughput by Channel

7 Days from 2013-03-14 to 2013-03-21

Generated on 2013-03-21 19:49:26 UTC

PFN space usage by StorageElement

34 Weeks from Week 51 of 2014 to Week 33 of 2015

Generated on 2015-08-26 15:47:04 UTC

Other services

- Data Logging service
  - Each operation on a chosen subset of name space changing the status of the file is recorded
    - Storage, identity of the operation initiator, status, etc
  - Useful in debugging problems with the data flows
- Data Integrity service
  - Each file access problem can be reported and accumulated in the Data Integrity database
  - Problem resolution either automatically or manually
- FTS Manager service
  - Interacts with the FTS3 service
  - Keeps track og ongoing FTS operations
- Staging service
  - Bringing data online before job submission
    - Asynchronous staging requests with polling for progress status
    - Pin time management

DIRAC
THE INTERWARE

DIRAC Framework

- DIRAC has a well defined architecture and development framework
  - Standard rules to create DIRAC extension
    - LHCbDIRAC, BESDIRAC, ILCDIRAC, …
- Large part of the functionality is implemented as plugins
  - Almost the whole DFC service is implemented as a collection of plugins
- Examples
  - Support for datasets first added to the BESDIRAC
  - LHCb has a custom Directory Tree module in the DIRAC File Catalog
- Allows to customize the DIRAC functionality for a particular application with minimal effort

Conclusions

- DIRAC combines various distributed computing and storage resources in a coherent system seen by the user as a single large computer
- The Data Management Model of DIRAC is organizing storage resources in a large distributed logical File System optimized for massive operations with data
- Recurrent bulk data operations can be automated an their integrity is ensured by the Request Management System
- DIRAC DMS is extensible due to its modular architecture and can be easily adapted to the needs of particular applications