



cherenkov
telescope
array

CTA report

L. Arrabito¹, J. Bregeon², P. Maeght¹, M. Sanguillon¹

¹ LUPM CNRS-IN2P3, France

² LPSC CNRS-IN2P3, France

10th DIRAC User Workshop 10th – 11th May 2021

DIRAC functionalities in use 1/3

- We use DIRAC since ~10 years to manage CTA MC activities (production and analysis) and also to support users activities
- We plan to use DIRAC WMS also for real data processing during CTA operations (start around 2023)
- Currently we use almost all DIRAC functionalities
- For future CTA operations we may use DIRAC WMS coupled with Rucio (under discussion)

DIRAC functionalities in use 2/3

- Current installed version: v7r1p28
- WMS
 - Grid resources: HT-Condor, ARC, CREAM
 - Tests done with a standalone cluster (sshCE)
- Tests done with VMDIRAC to use Clouds (Openstack) (2017)
- DMS
 - All data operations (different protocols, XROOT, GSIFTP, SRM2, ...)
- DIRAC File Catalog (as Replica and Metadata Catalog)
 - More than 23 million replicas
 - About 20 meta-data defined to characterize CTA datasets
 - Using *datasets* to expose data selections to users and as input to transformations (currently 650 defined datasets)
- RMS
 - For Failover and data management transformations
- Accounting
- WebApp

DIRAC functionalities in use 3/3

- Transformation System
 - Mainly for processing (since 2015)
 - Recently also for data management (e.g. migrating about 1 PB from different SEs to tape SEs)
 - Coupled with FTS (**new in 2020**)
- Production System (**new in 2020**)
 - Developed and integrated in v7r0 (2019)
 - Used for the last large-scale MC productions (July 2020-now)
- Monitoring System with Elasticsearch backend (**new in 2020**)
 - Component Monitoring/WMS history
 - Installed/tested in 2020 (Patrick, Michèle and CC-IN2P3)

DIRAC functionalities we don't use

- Resource Status System
 - Lack of time for testing
 - At a first look, the configuration doesn't look trivial
- Central Logging
 - Lack of time for testing...
- Message Queuing System
 - Again lack of time...
- REST interface
 - No real need until now, maybe in future, not sure about the current status

CTADIRAC extension 1/2

- Extension of the Job API to easily configure CTA jobs
- Several scripts to launch transformations and productions for CTA workflows
 - Use CTADIRAC Job API
 - Create transformations for different kinds of CTA jobs taking as input a dataset name
 - Create productions for different kinds of CTA workflows
- Provenance Service to handle CTA provenance meta-data
 - Included in CTADIRAC DMS
 - Using a PostgreSQL DB as backend

-> Specific to CTA, cannot be ported to vanilla DIRAC

CTADIRAC extension 2/2

- Commands to manage transformations
 - Attach files to a transformation
 - Clean/delete a transformation
 - Create a 'Moving' transformation taking as input a dataset name, e.g. moving a dataset from Disk SEs to Tape SE
- Commands to manage datasets (show, dump)
- Command to get a summary view of jobs by Status and Site

-> All these can be generalized if interesting for others

What is your biggest frustration with DIRAC?

- We are generally happy with DIRAC
- However, it's not easy for newcomers to install a fully operational instance or to contribute to the development (not easy to prepare the development environment)
-> Next slide

You can magically add one feature to DIRAC, what is it?



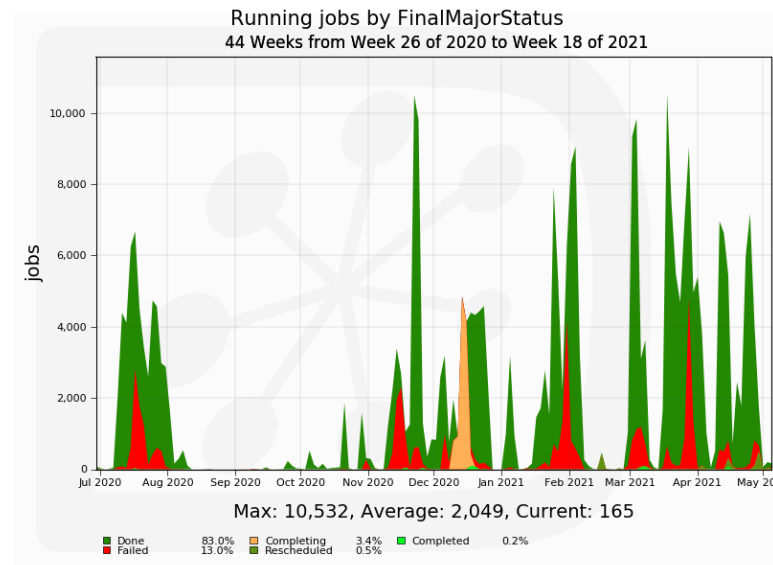
- Tools helping to install almost automatically a fully operational server instance (e.g. using pre-defined containers for server and client)
- It would allow newcomers to quickly have a running installation to play with
- Coupled with a test suite to check that everything works fine
 - Launch jobs, transformations, productions, do data management operations, etc.

Additional desired features

- Tools to easily monitor storage space occupancy
- Monitoring System
 - Job Parameters stored in ES are available only through Kibana
 - It would be nice to have them in WebApp
 - We are interested to collect other Job Parameters, such as used CPU, RAM, etc. and make plots out of them (distributions, eventually 2D correlation plots, ...)
- Accounting
 - It would be nice to have a **'Pie plot of Normalized used CPU'**, while today one can have either a 'Pie plot of used CPU' either a 'Cumulative plot of Normalized used CPU'...
- Better specification of job RAM requirements
 - We regularly have failures due to jobs exceeding RAM queue limits
 - Not found an easy way to handle this...

Operations incident in the last year

- Mainly DB incidents (DB not responding) happened several times during last year
 - In principle not related to DIRAC
- We currently rely on a MariaDB cluster at CC-IN2P3 shared with other experiments
 - It seems that the overload was due to other experiments activities
 - We will move to a dedicated server soon



Other systems in use than DIRAC

- CVMFS for sw distribution
 - CTA repository (Stratum-0 at CC-IN2P3 and 2 Stratum-1 at CC-IN2P3 and DESY Zeuthen)
- FTS for bulk data transfer
 - CERN pilot instance
- In future we consider using IAM for A&A
 - Need to make it compatible with DIRAC

How would you rate the communication?

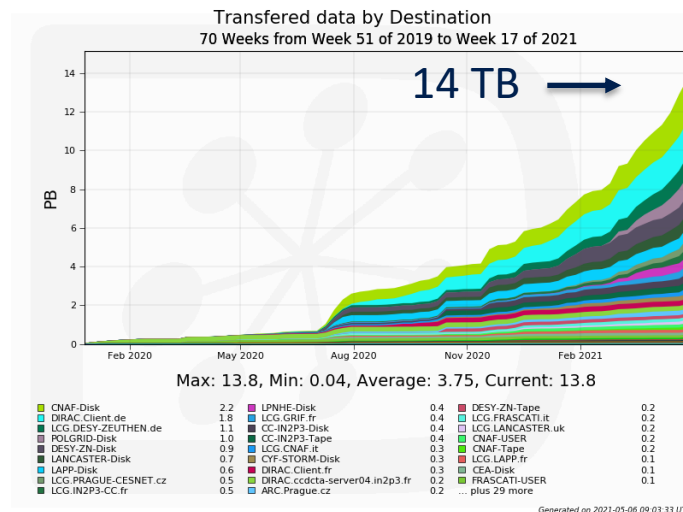
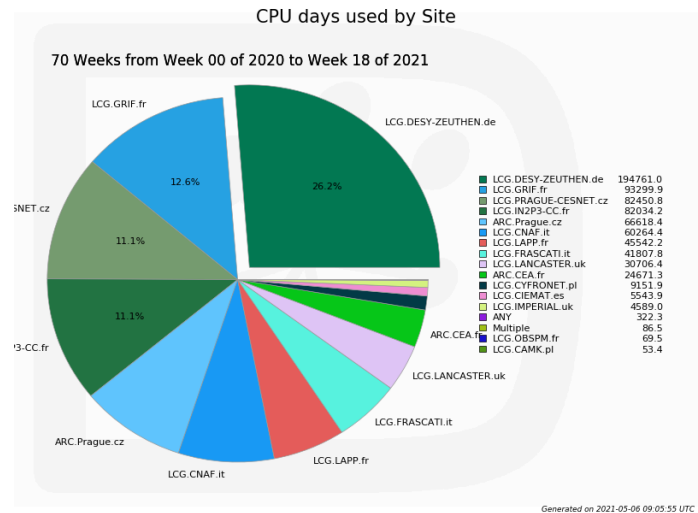
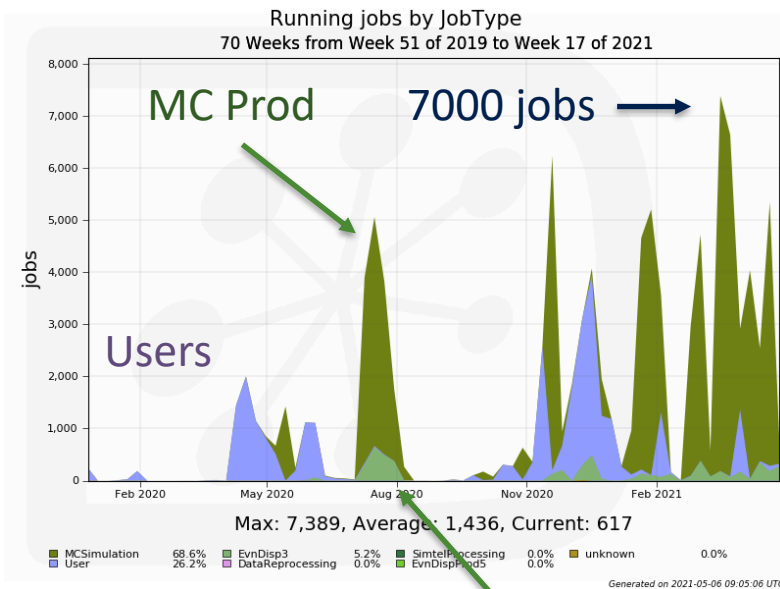


-
- The communication is good, but documentation could be better organized
 - dirac-grid forum is very useful
 - Ask help and check if anybody had a similar issue/question
 - Release notes very helpful, e.g. to update the server instance
 - Not so much time to attend BiLD meetings but they are also a good place for discussion
 - We also have 'DIRAC French meetings' (DIRAC@IN2P3 project) with Andrei, CC-IN2P3 and different users communities
 - News on DIRAC, sorting issues, share experience ...

Resource usage with DIRAC during last 2 years (plots from 1st Jan. 2020)



- Total executed jobs: 3.3 million
- Total CPU: 245 millions HS06 hours

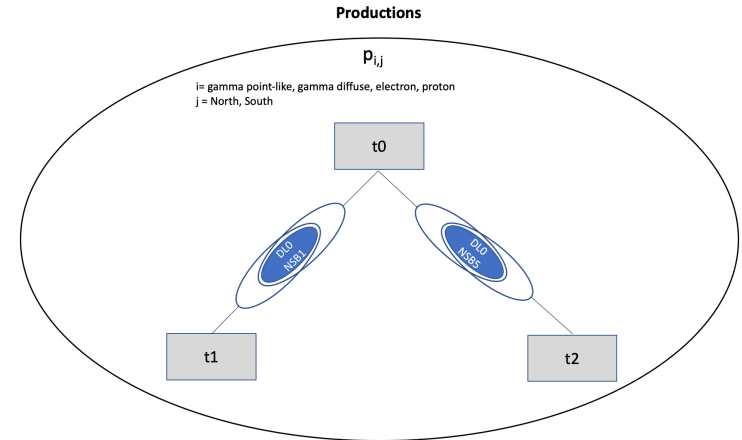


Other news since last workshop

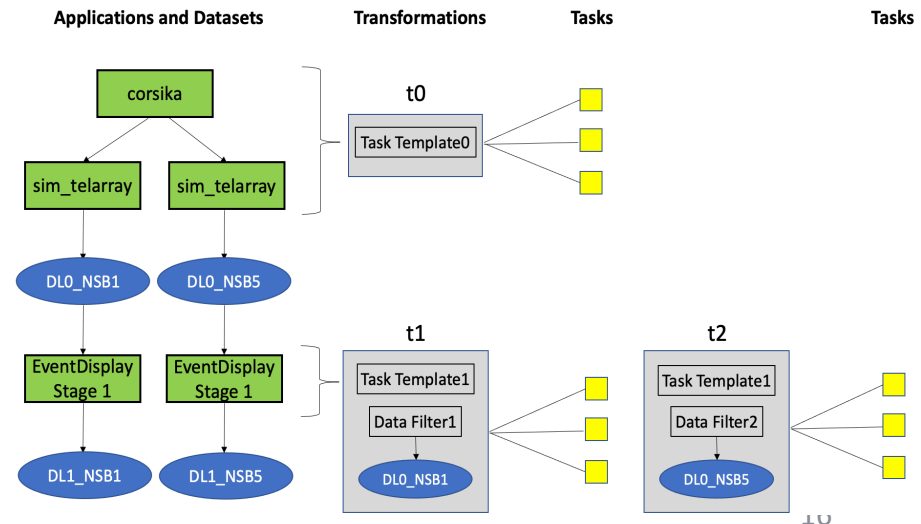
- Installed a DIRAC single-server instance for testing purpose
 - Very useful to test new features or new DIRAC releases before applying them to the production instance
 - e.g. Used to install/test Monitoring with ElasticSearch
 - Preliminary tests done with Rucio Catalog plugin
 - Currently testing v7r2
- First large-scale usage of the Production System (Johan)
- First usage of Transformation System and FTS for bulk data transfers
- The team at LUPM grewed 😊 (Michèle, Patrick and 1 more person should start in July for a 2-years contract)

First large-scale usage of the DIRAC Production System

- Prepare the production description
 - Define the connected transformations
 - Input/Output datasets as metadata queries



- Run the production
 - Automatic transformation creation
 - Failed jobs automatically retried
- In this example: each production composed of 3 transformations and 8 productions for each MC campaign
 - > Much easier to manage than tens of transformations



Future plans

- Work on testing/CI infrastructure for CTADIRAC
- Improve failure recovery or finalization of transformations to obtain fully automatized data-processing
- Improve the Production System interface (eventually developing a web interface) and **prepare a tutorial**
- Thanks to all DIRAC community for the help and support

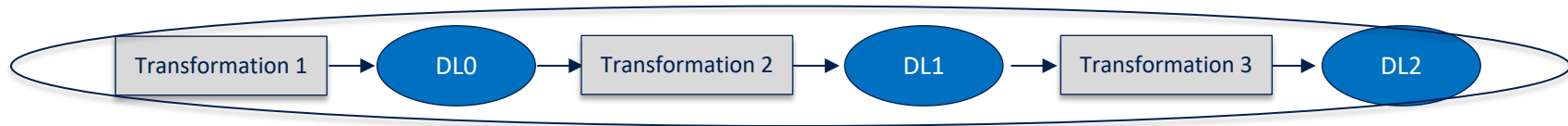
Backup



First large-scale usage of the DIRAC Production System



- Workflow in DIRAC as a series of *data transformations*



- Transformation System

- Tasks (jobs) creation and submission to WMS

- Production System (since v7r0)

- Designed to handle workflows at high level

- **Production (or workflow)** : a series of **connected transformations**

- Connected transformations

- Intersection of Input and Output datasets (defined as metadata queries)

