DIRAC Data Production System

Andrew C. Smith

CERN

October 21, 2010

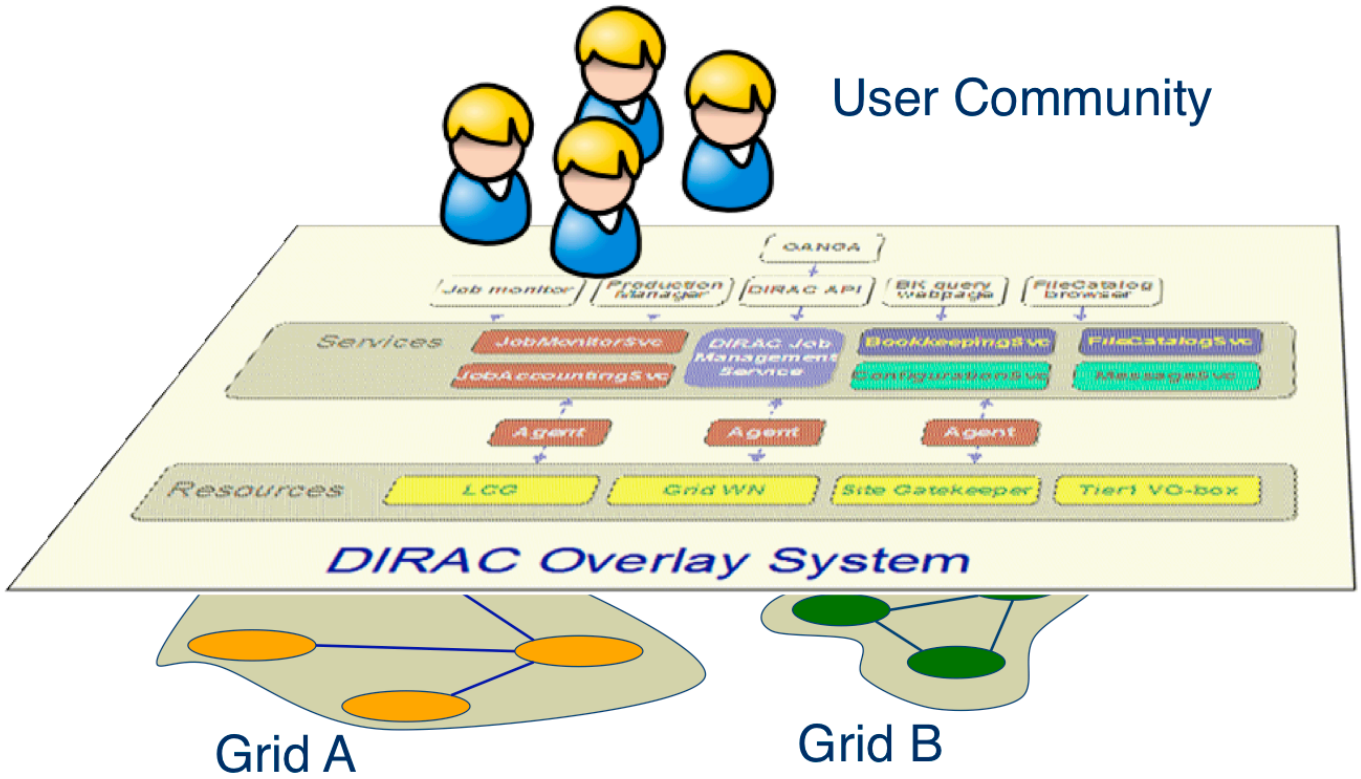
1. Introduction to DIRAC
2. DIRAC Data Production System
3. Using the Data Production System

Outline

1. Introduction to DIRAC
2. DIRAC Data Production System
3. Using the Data Production System

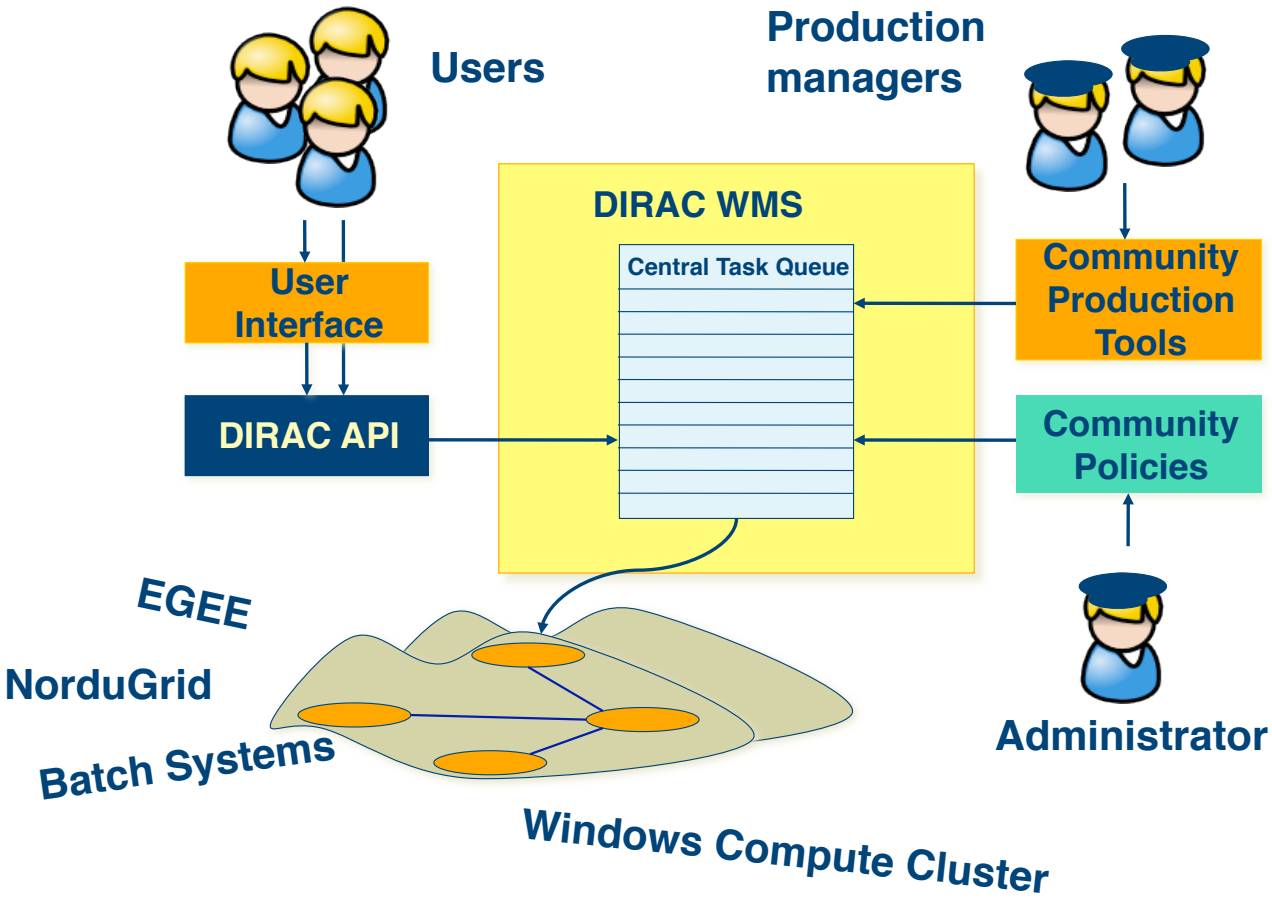
Community Grid Solution

The DIRAC project is a complete Grid solution.
Designed to be used by a community of users.
Services and agents of DIRAC overlay resources.
Transparent use of different grids e.g. gLite, NorduGrid etc.
Integration of non-grid resources e.g. local, clouds, batch systems etc.
Grid compliant security framework (OpenSSL with X509 certificates).



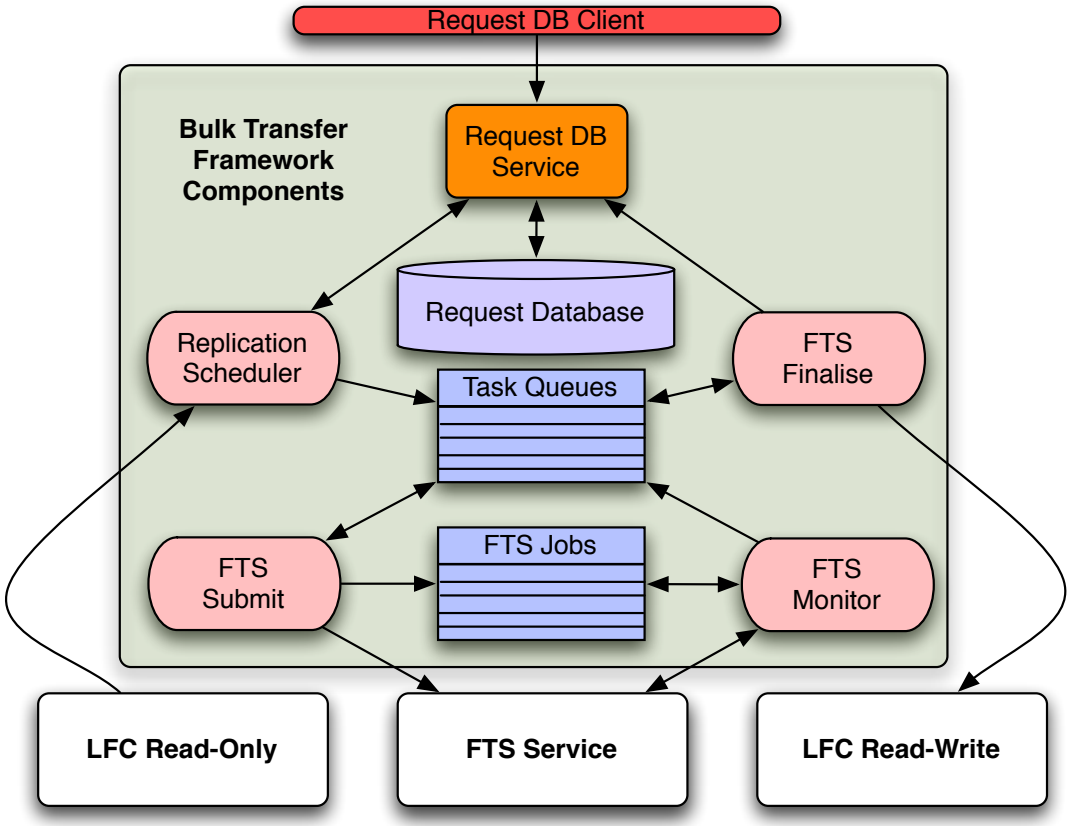
Workload Management

The DIRAC WMS pioneered the use of Pilot Agents.
The Workload is optimised in a central Task Queue.
Allows community priorities and shares to be applied.



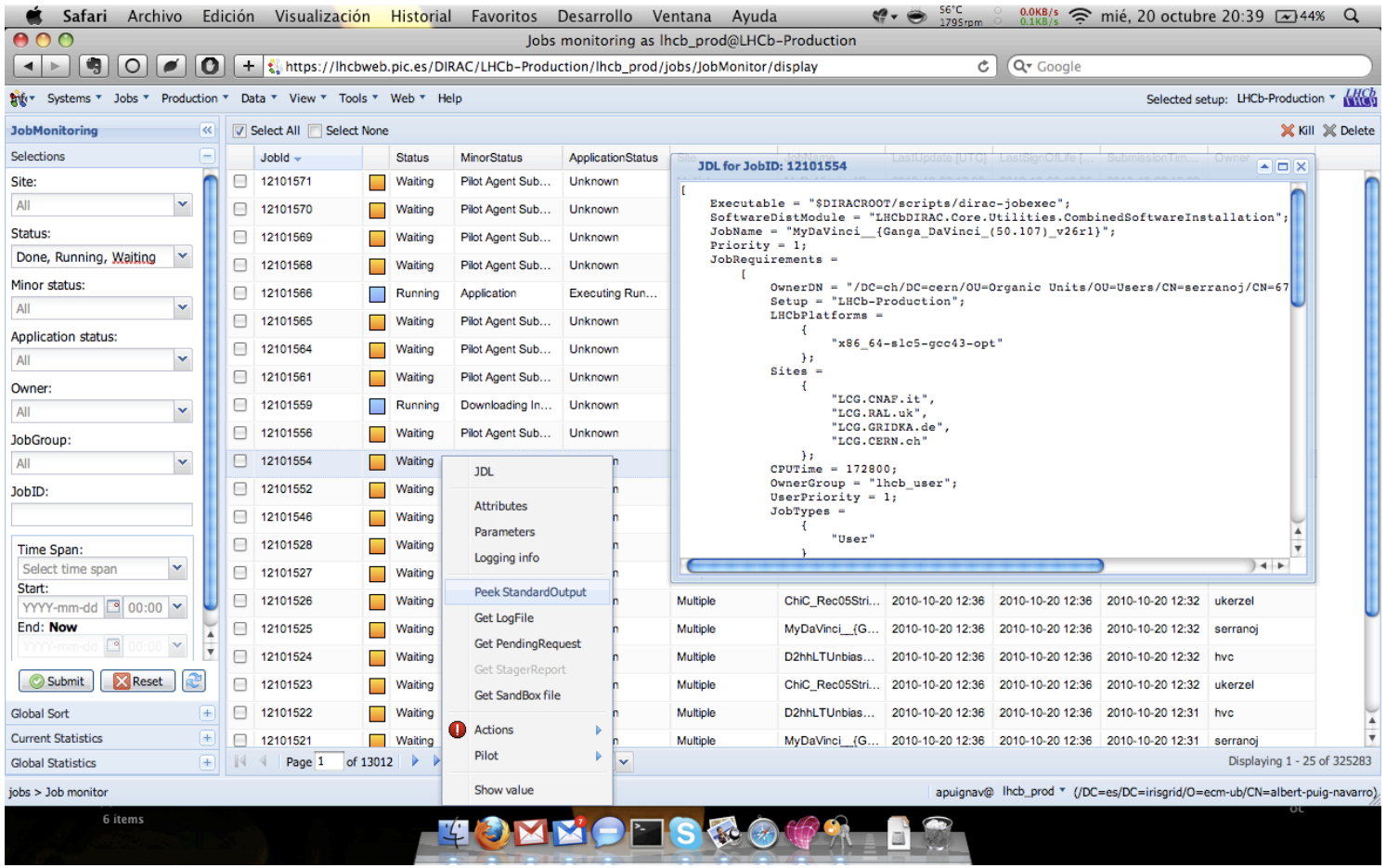
DIRAC Data Management System

DIRAC DMS provides tools for interacting with catalogs and storage.
Upload, download and replication of data to/from/between storages.
Includes bulk transfer framework based on FTS.



Interfaces

DIRAC API is a Python wrapper used by external projects (for example Ganga).
DIRAC CLI is based on DIRAC API in Python.
DIRAC GUI is secure web based interface build in JS.
Mimics desktop application for short learning curve.



Outline

1 Introduction to DIRAC

2 DIRAC Data Production System

3 Using the Data Production System

Motivation

Each user community has specific workflows of varying complexity:

- Single application jobs
- Multi-application jobs
- Jobs that require input from previous jobs
- Replication of output data
- Removal of transient data
- etc. etc. etc.

The LHCb computing model outlines several activities:

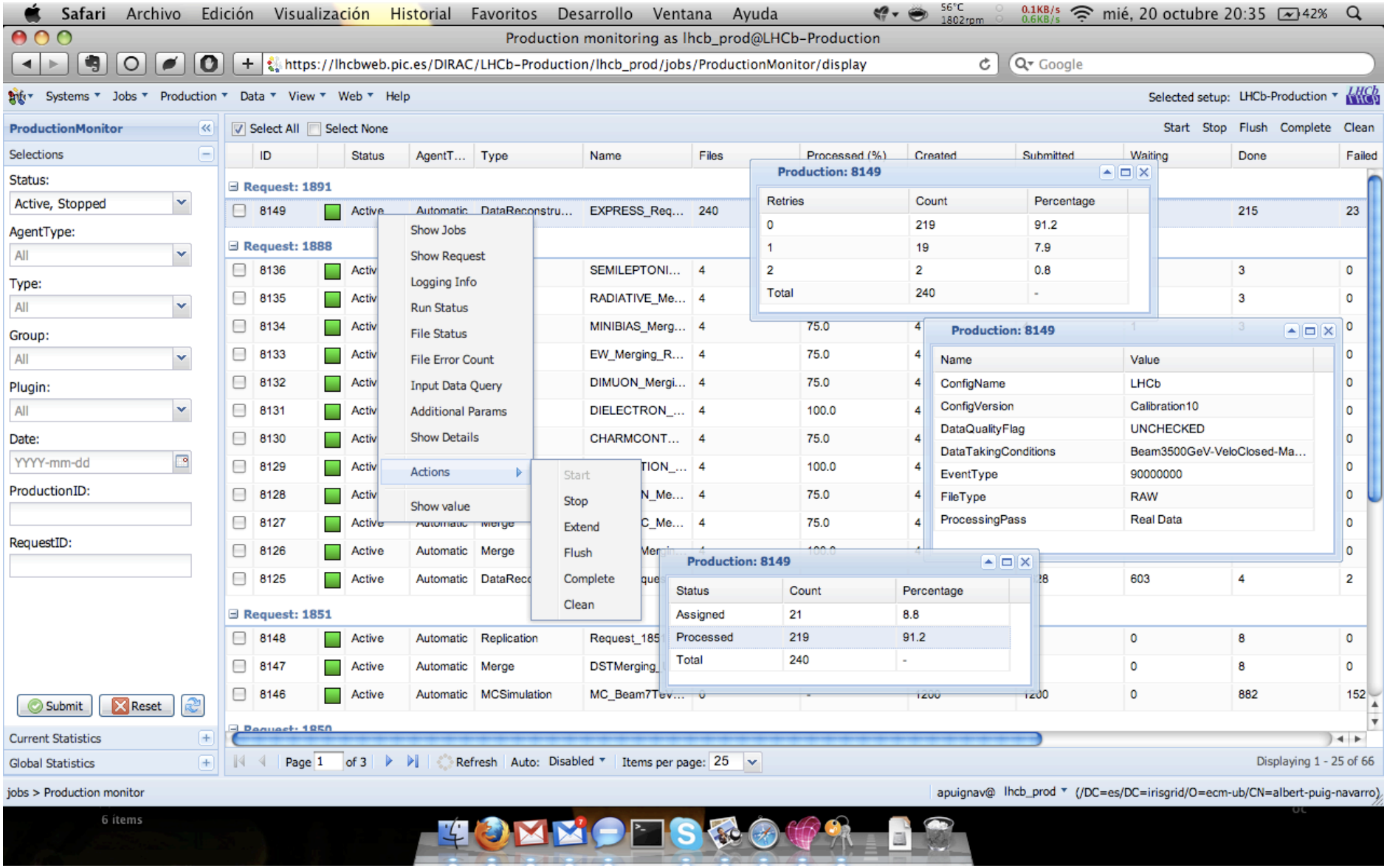
- Real-time RAW data reconstruction and stripping.
- Regular re-reconstruction and re-stripping exercises.
- Massive Monte Carlo campaigns.
- Replication of data for redundancy and availability.

Supported features

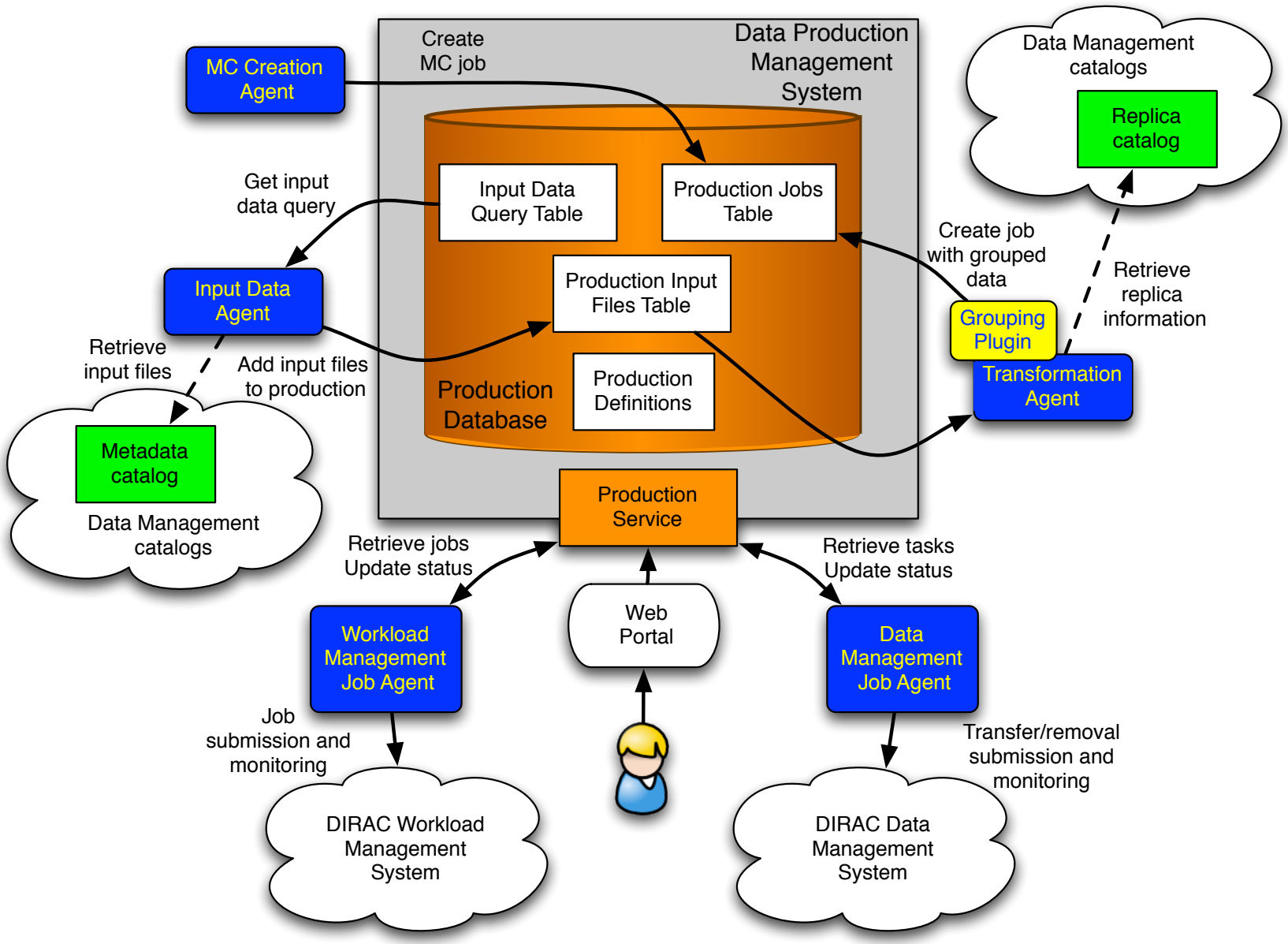
To support this the Data Production System provides:

- Job description library for chaining applications in a single job.
- Job creation based on input data availability.
- Chaining jobs based on output/input data.
- Submission and monitoring of workload and data management jobs.
- Automatic creation of jobs to reach requested amount.
- Post workflow data integrity checking suite.
- Web monitoring and control interface.

What does this system look like



What does this system look like (really)



Key components - Production database

The central component is the **Production Database**; a simple relational database that persists a full description of a production, its input data and jobs created. It is polled and updated (through the service) by the system agents and clients (including the web GUI).

The key tables are:

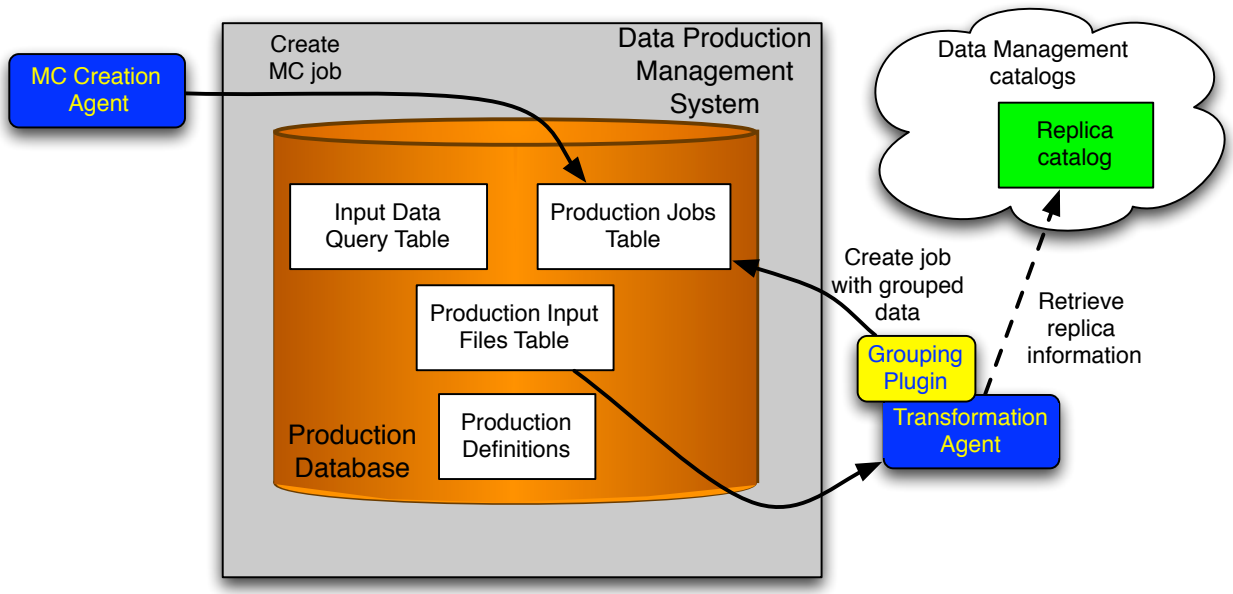
- **Production** - production parameters and job description template.
- **Production Input Query** - metadata query describing the input data.
- **Production Input** - input files and their statuses.
- **Production Jobs** - created jobs, their parameters and status.
- **Production Job Inputs** - mapping production inputs to jobs.

Further auxiliary tables also exist to store any additional parameters and log the production history.

Key components - Job creation agents

The **MC Creation Agent** creates jobs for productions without input data.

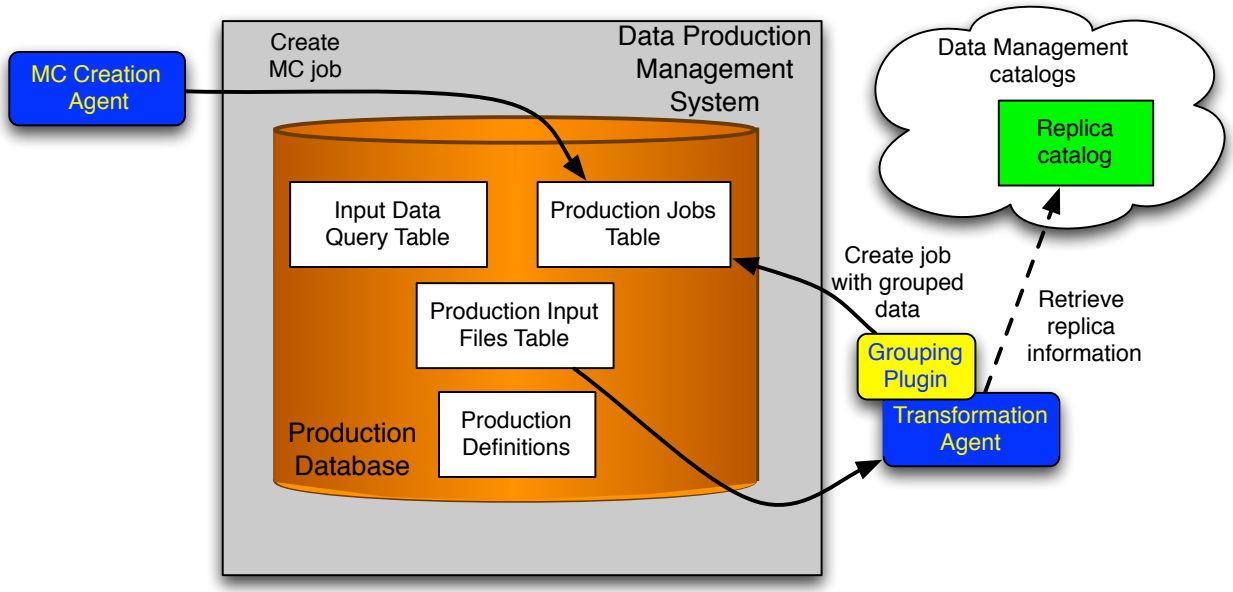
- Retrieves required number of successful jobs from definition.
- Determines the number of jobs to create based on current status.
- Maximum number of jobs created per iteration to avoid flooding.



Key components - Job creation agents

The **Transformation Agent** creates jobs for productions with input data.

- Retrieves unused input data for productions.
- Queries replica catalog for input data location.
- Applies *transformation plugin* to group input data.
- e.g. by size, by run, by file type, according to shares.
- Jobs are created for each of the resulting groups.

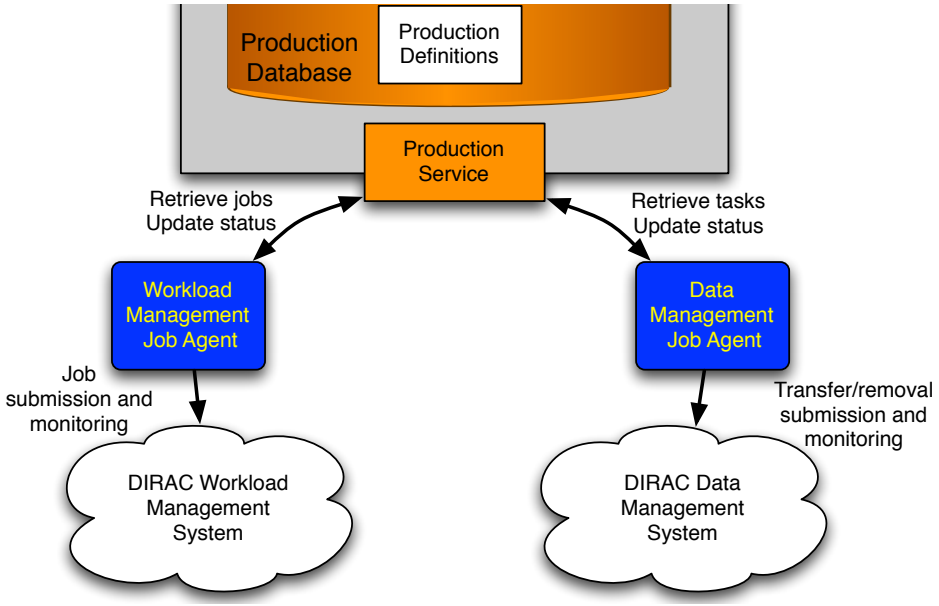


Key components - Job submission and monitoring agents

The **Workload Management Job Agent** and **Data Management Job Agent** submit and monitor jobs destined for the WMS and DMS respectively.

- Created jobs are retrieved for their respective productions.
- Job specific parameters are applied to the job description template.
- Submitted job identifiers are inserted into the database.
- Identifiers used for subsequent updates of statuses.

The code required for interacting with the WMS/DMS is contained in specific plugins. Allows easy extension for use with other systems.



Additional Components - Data Integrity Checking

The mutual consistency of data management catalogs is imperative.

To reduce data integrity problems a full consistency check is performed:

- The output directories for the production are retrieved
- For each directory the catalog contents are verified against the storage.
- The storage contents are verified against the catalog.
- Problematic files are stored and invisible to user queries.
- Resolution methods are applied to problematic files.

Checks can be triggered at any time by changing production status.

Outline

1 Introduction to DIRAC

2 DIRAC Data Production System

3 Using the Data Production System

Current users of DIRAC

DIRAC developed by LHCb to support their computing model.

Evaluated by other projects including ILC/LCD, Belle, EELA/GISELA.

New communities bring new use-cases:

- Amazon EC2 resource integration for Belle
- MPI job support for EELA

Adoption by new communities simple because of modular design.

Become part of the production infrastructure in some grid segments:

- Latin American - GISELA
- NGI France
- NGI Spain

LHCb workflows - Monte-Carlo generation

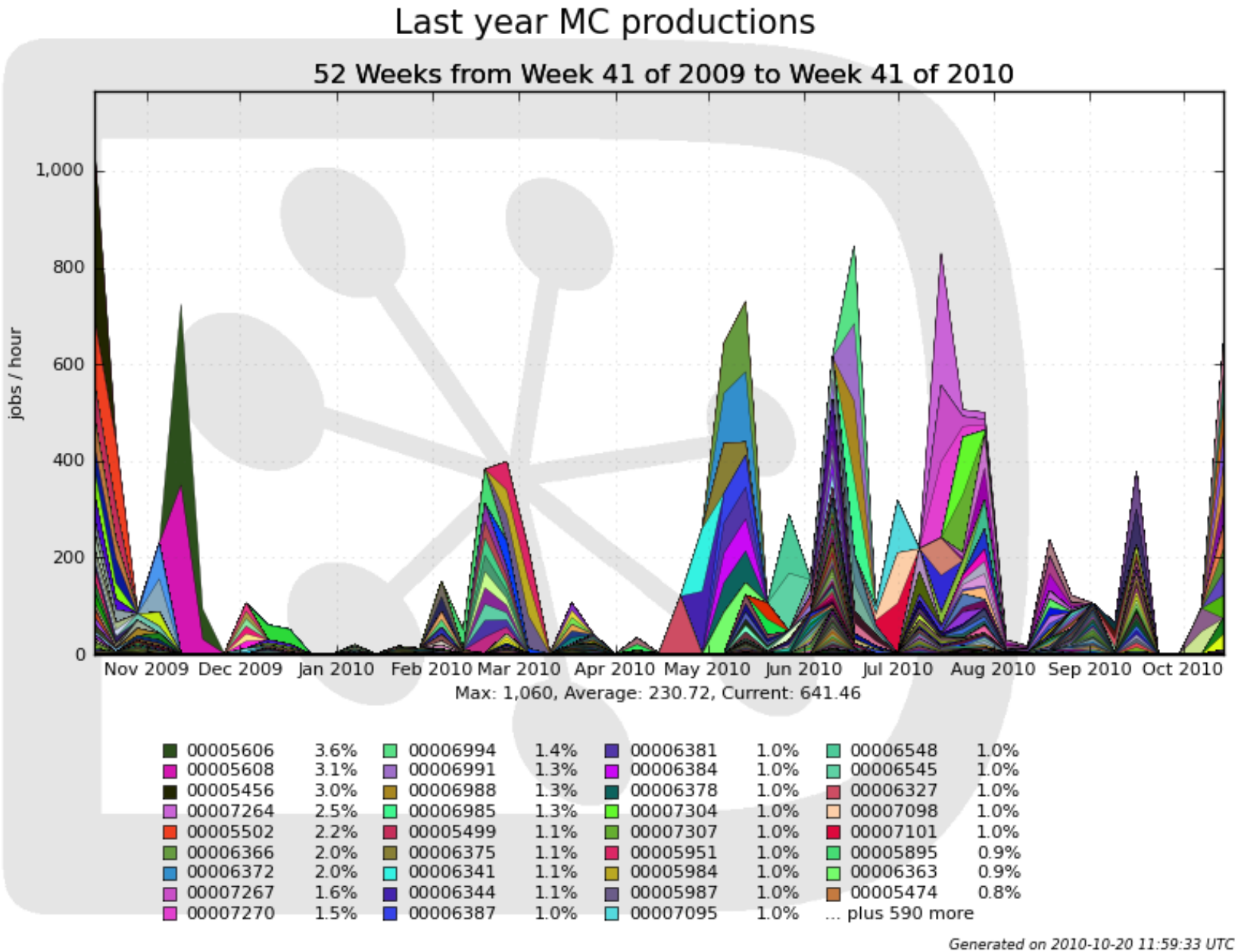
Monte-Carlo generation

Requires three productions (bullets 1-3).

1. Three application-step jobs to generate, digitise and reconstruct events.
2. Single application-step jobs submitted to merge files.
3. Replication of merged files to two Tier1s.
4. Data integrity check of all merged files.
5. Removal of all un-merged files.

Productions 2 and 3 take their input data from the previous production.

LHCb workflows - Monte-Carlo generation



LHCb workflows - RAW data processing

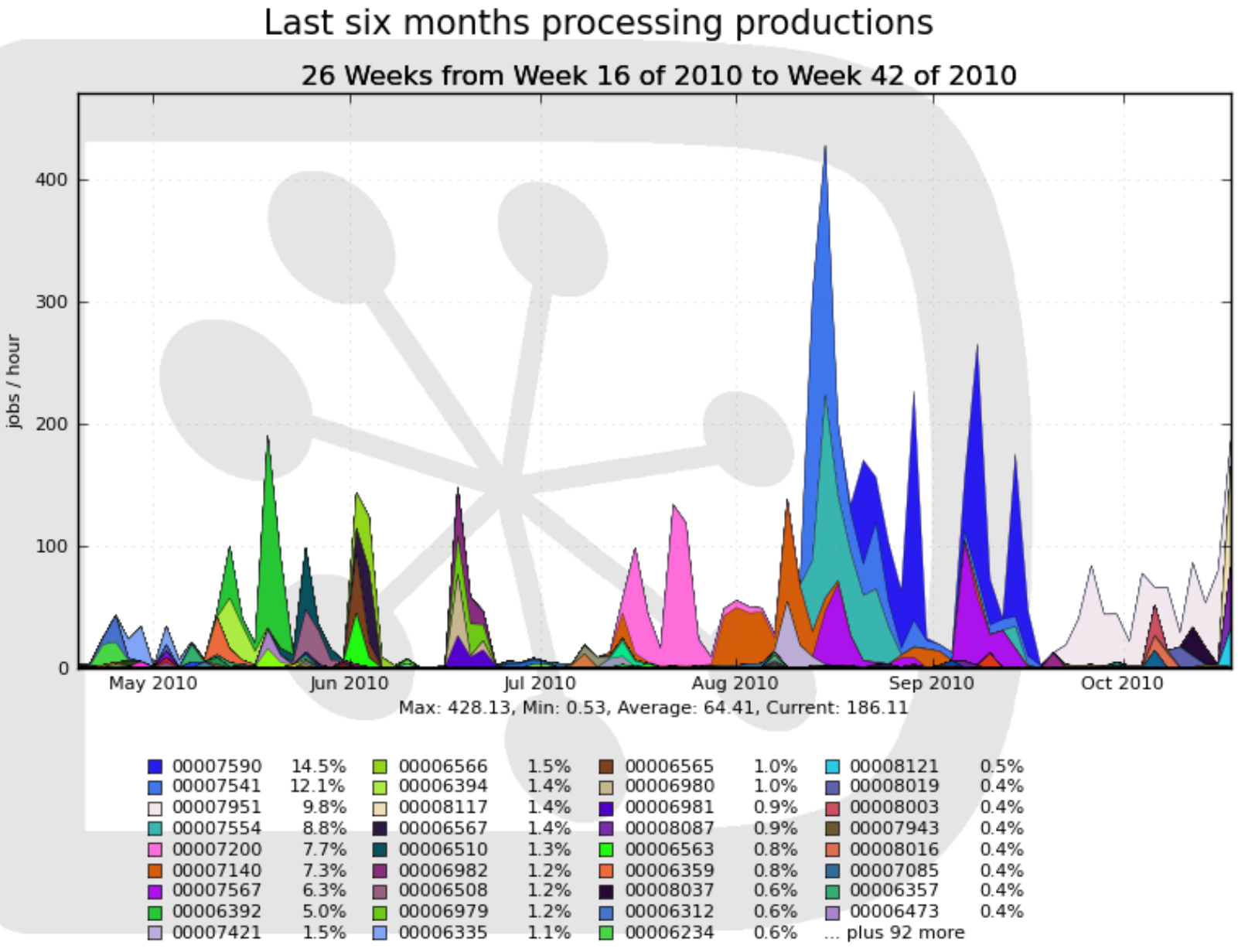
Real-time RAW data taking

Requires four productions (bullets 1,3,5,6).

1. Replication of RAW data from CERN to Tier1s.
2. Target sites chosen by shares (controlled by dedicated plugin).
3. Two application-step jobs submitted on RAW to reconstruct and strip.
4. Jobs sent to sites according pledged CPU.
5. Two application-step jobs submitted on stripped files to merge and tag.
6. Replication of merged and tag files to all LHCb Tier1s.
7. Data integrity check of all merged and tag files.
8. Removal of all un-merged files.

Productions 3, 5 and 6 take their input data from the previous production.

LHCb workflows - RAW data processing



ILC/LCD adoption and experience

The ILC/LCD group at CERN required massive production of MC events.

Adoption of DIRAC for grid production and analysis.

- Configuration of workflow was done in a couple of weeks.
- Initially, small scale productions submitted as "user" jobs.
- Allowed rapid development and testing of job descriptions.
- After six months production workflows moved to Data Production System.

Flexibility of the system allows to add new functionality very quickly.

- Users from outside CERN now using it.
- Prospects to make it the general grid solution for the ILC community.

More information

For more information see here:

- DIRAC project page
- DIRAC Overview at CHEP09
- DIRAC Pilot Framework
- DISET Secure Framework
- Secure Web Interface
- DIRAC MPI in EELA2
- Usage of DIRAC by other communities
- Belle experience with Amazon EC2 - PS50-2-483/PO-WED-054

Email DIRAC project leader, Andrei Tsaregorodtsev, atsareg@in2p3.fr