
AGATA Analysis on the GRID

R.M. Pérez-Vidal IFIC-CSIC

For the e682 collaboration

What is GRID?

Grid technologies allow that computers share through Internet or other telecommunication networks not only information, but also computing power (**Grid Computing**) and storage capacity (**Grid Data**).

In the early 21st century, the Large Hadron Collider (LHC) at CERN developed Grid technologies and infrastructures for High energy Physics experiments in order to provide a resource to store, distribute and analyse the 15 petabytes (15 million gigabytes) of data generated every year there. Such project, has been extended to other other disciplines and its infrastructures are becoming more stable and robust in the last years.

This grid-based infrastructure is the most effective solution to the data-analysis challenge for experiments with huge amount of data production, offering many advantages over a centralized system.

- Multiple copies of data can be kept at different sites.
- Uses high-capacity computing to model complex systems and to process experimental results.
- Ensures access for all scientists independent of geographical location
- There is no single point of failure
- Resources can be distributed across the world, for funding and sociological reasons.
- Secure access and trustworthy relations
- Users belong to a common Organizations (Virtual Organization)



The interaction between users and the resources is made possible by a software layer known as middleware.

What is GRID?

GRID is made up of four layers, or "tiers"

Tier-0

CERN Data Centre.

- Data recording
- Reconstruction
- Distribution.

Tier 1

Large computer centres.

- Permanent storage
- Reprocessing
- Analysis.

Tier 2

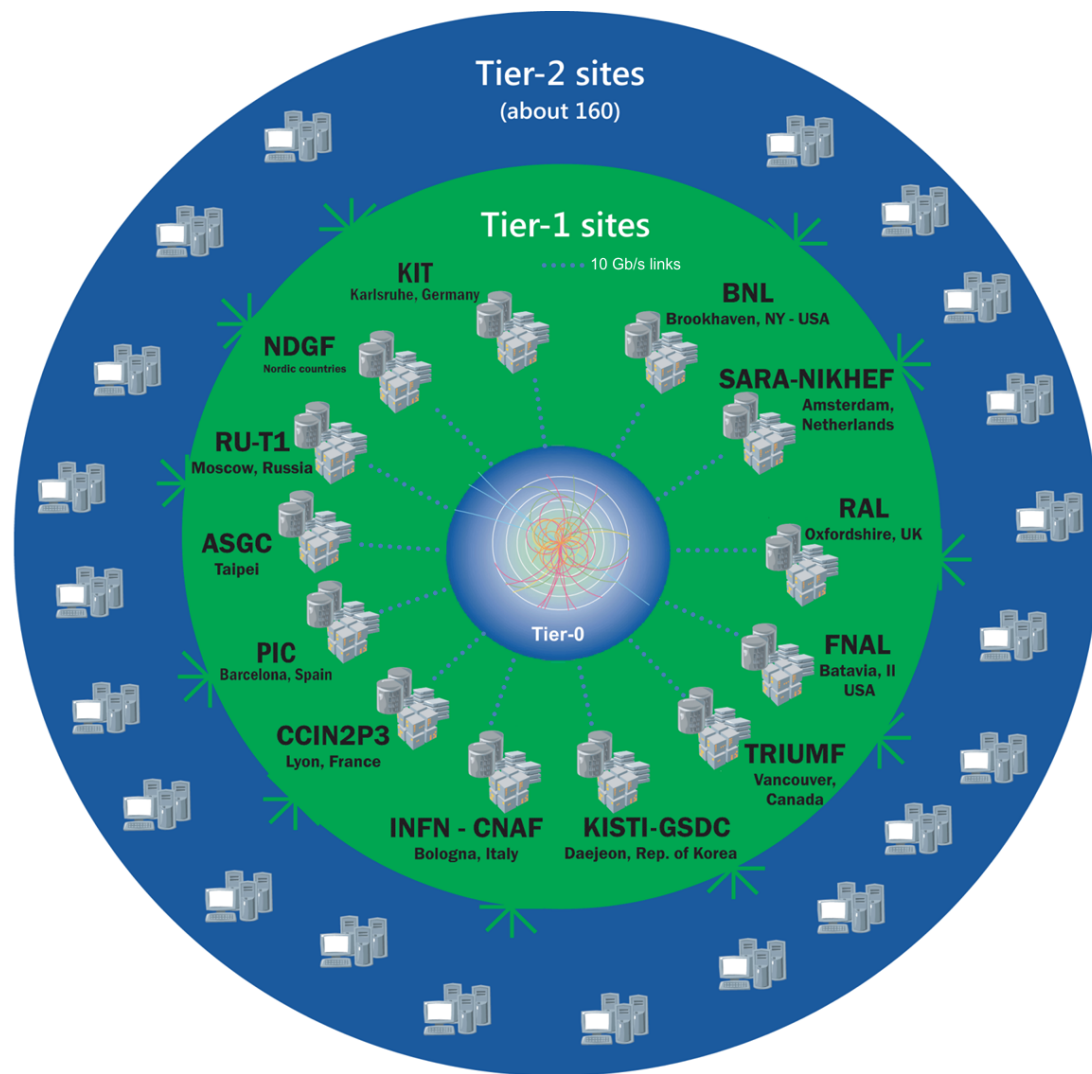
Universities and other scientific institutes.

- Simulations
- End-users analysis

Tier 3

Local computing resources (local clusters in a University Department, individual PC...)

- Access these facilities



What is GRID?

The GRID architecture is based on the **gLite** middleware which provides a **framework** for building applications tapping into distributed computing and storage resources across the internet and the following **set of services** to access remote sources:

SECURITY	Requirements: <ul style="list-style-type: none">- Authentication: Valid GRID certificate (CA)- Authorization: Be registered in a Virtual Organization (VO)
USER INTERFACE (UI)	Access point to the to the remote computing resources through the certificate and the proxy created with the VO. Each user has its personal account.
COMPUTING ELEMENT (CE)	Set of computing resources localized at a site (i.e. a cluster, a computing farm). It includes a collection of Worker Nodes (WNs), where the jobs are run.
STORAGE ELEMENT (SE)	Access to data storage resources. Controls simple disk servers, large disk arrays or tape-based Mass Storage Systems (MSS)
INFORMATION SERVICE (IS)	Information about the resources and their status.
WORKLOAD MANAGMENT SYSTEM (WMS)	Orchestrates the Job management on the Grid. It accepts user jobs, assigns them to the most appropriate CE, records their status and retrieves their output.

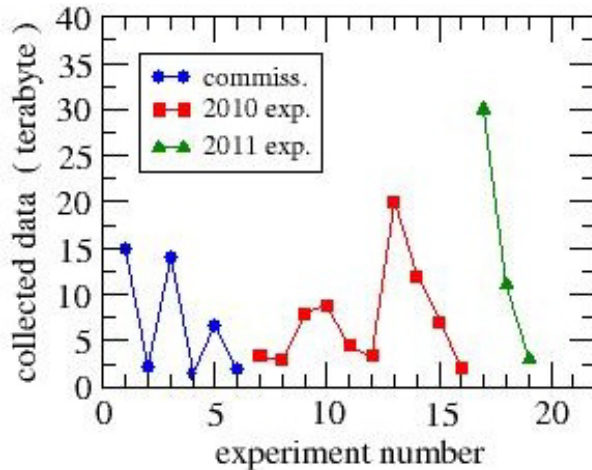
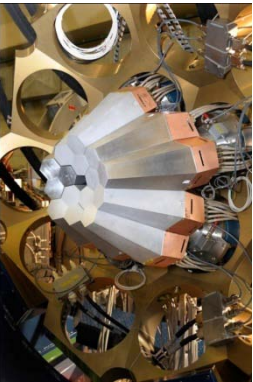
For more detailed info see the presentation of M. KACI (IFIC) shown at the 2nd EGAN School at GSI 2012

Migrations towards GRID

Previous Generation:
GAMMASPHERE, EUROBALL

Less than 300 GB data per experiment
Less than hundred sequential files to process
Data stored on Exabyte tapes and analyzed at home institutes

New Generation: **AGATA (PHASE-I)**

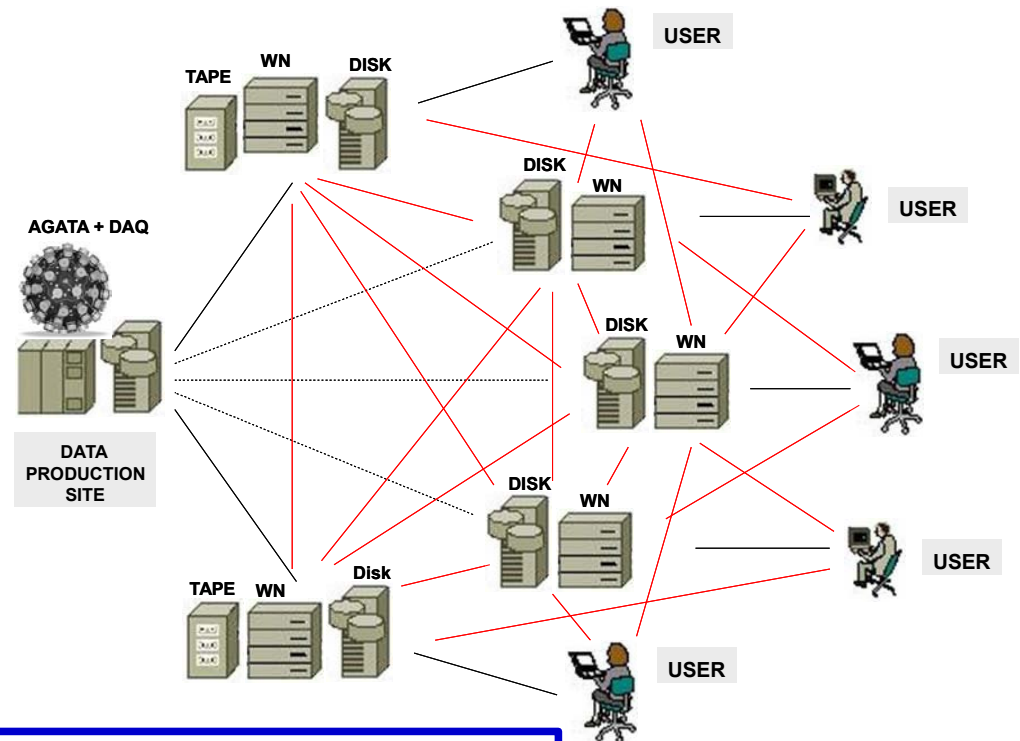


More than 157 TB of Raw Data produced
Around 8-10 TB per experiment
Thousands of files 3-5 GB each to process
Slow Data (re)Processing on a single CPU
Migration to Grid Storage

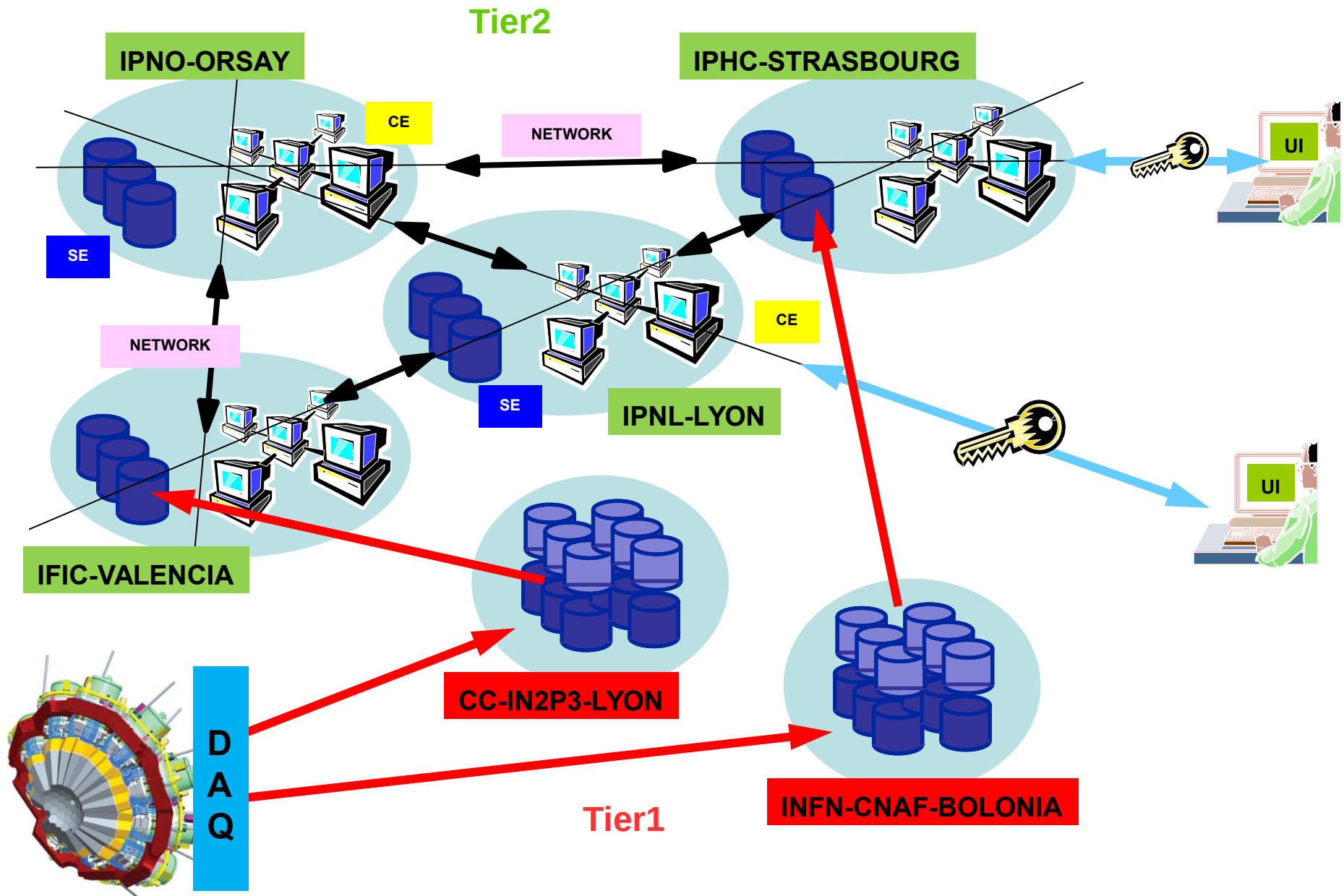
A new situation where the users share computing resources and data in a coordinated way (policy) within a Virtual Organization...

MIGRATE TOWARDS...

NEED A GRID COMPUTING MODEL...



AGATA GRID computing resources



AGATA GRID computing resources

Security. Autentication

How to get a GRID certificate:

- You need a certificate (X.509) to authenticate (make your identity known) yourself to services, hosts and other users in the GRID environment
- Certificates are issued by a GRID Certification Authority (CA)
- Each country (or region) have their own CA
- Map with link to CA:s in Europe:
<https://www.eugridpma.org/members/worldmap/>
- Apply for a certificate in your home country
- Good instructions for this are available at the CA:s web sites
- Mention that you will be a member of the AGATA VO

AGATA GRID computing resources

Security. Authorization

Register as a user of the AGATA VO

- To access the AGATA GRID resources you need authorization for these specific resources
- This is obtained by becoming a member of the AGATA VO
- Visit
<https://cclcgvomsl101.in2p3.fr:8443/voms/vo.agata.org/>
and follow the instructions for the registration
- Note: You cannot access this site before you have imported your personal GRID certificate in your web browser

AGATA GRID computing resources

User Interface

For example, at IFIC (Valencia): **ui06.ific.uv.es**

Computing Elements (Tier2)

IPNL (Lyon): **lyogrid07.in2p3.fr:8443/cream-pbs-vo.agata.org**

IPNO (Orsay): **ipngrid04.in2p3.fr:8443/cream-pbs-agata**

IPHC (Strasbourg): **sbgse2.in2p3.fr:8443/cream-pbs-vo.agata.org**

IFIC (Valencia): **ce03.ific.uv.es:8443/cream-pbs-agataL**

Storage Elements

CC-IN2P3 (Lyon): **ccsrm02.in2p3.fr (Tier1)**

INFN-CNAF (Bologna): **storm-fe-archive.cr.cnaf.infn.it (Tier1)**

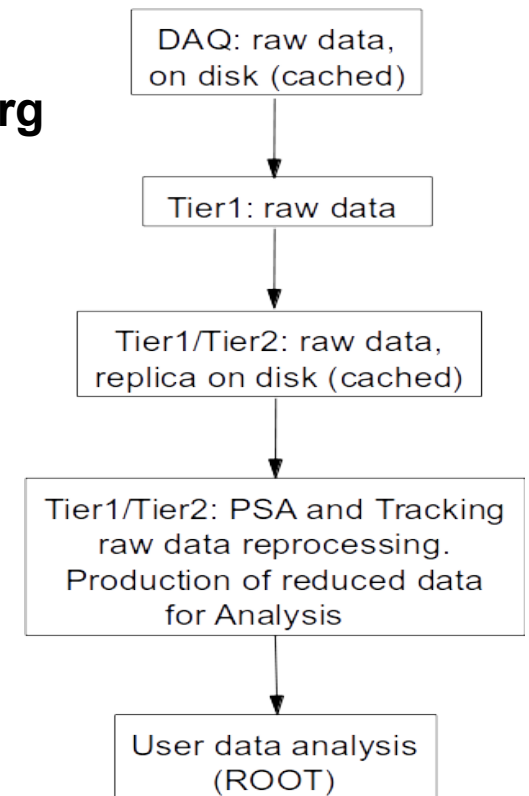
IPNL (Lyon): **lyogrid06.in2p3.fr (Tier2)**

IPNO (Orsay): **ipnsedpm.in2p3.fr (Tier2)**

IPHC (Strasbourg): **sbgse1.in2p3.fr (Tier2)**

IFIC (Valencia): **srmv2.ific.uv.es (Tier2)**

The AGATA Grid Computing uses the LHC Tiers structure. The Tier1s and Tier2s have similar roles in AGATA, but Tier1s provide Tape Storage



How to use GRID

1. Connect to a UI with the valid GRID certificate:

```
$ ssh -X username@ui06.ific.uv.es
```

2. Create a proxy:

```
$ voms-proxy-init-voms vo.agata.org
```

3. Use the following commands:

Computing resources info, CE and SE:

```
$ lcg-infosites --vo vo.agata.org ce  
$ lcg-infosites --vo vo.agata.org se
```

Available SEs:

```
for CNAF-Tier1 srm://storm-fe-archive.cr.cnaf.infn.it:8444/agata  
for LYON-Tier1 srm://ccsrm02.in2p3.fr:8443/pnfs/in2p3.fr/data/agata/  
for IPNL-Tier2 srm://lyogrid06.in2p3.fr/dpm/in2p3.fr/home/vo.agata.org/  
for IPHC-Tier2 srm://sbgse1.in2p3.fr/dpm/in2p3.fr/home/vo.agata.org/  
for IPNO-Tier2 srm://ipnsedpm.in2p3.fr/dpm/in2p3.fr/home/vo.agata.org/  
for IFIC-Tier2 srm://srmv2.ific.uv.es/lustre/ific.uv.es/grid/vo.agata.org/
```

Data management tools:

More detailed info in

<http://ns.ph.liv.ac.uk/EGAN/files/wednesday/GRID-HOWTO.pdf>

- List files info: `lcg-ls SE`

```
$ lcg-ls -l srm://storm-fe-archive.cr.cnaf.infn.it:8444/agata/
```

- Copy files: `lcg-cp fromSE_1 toSE_2`

```
$ lcg-cp -v srm://storm-fe-archive.cr.cnaf.infn.it:8444/agata/e682/run_0109.dat.05-07-15_19h35m58s/Data/00A/SRM_AGATA_event_mezzdata.cdat.0000  
srm://srmv2.ific.uv.es/lustre/ific.uv.es/grid/vo.agata.org/e682-test/run_0109.dat.05-07-15_19h35m58s/Data/00A/SRM_AGATA_event_mezzdata.cdat.0000
```

- Delete files (-l) or directories (-d): `lcg-del -? SE`

```
$ lcg-del -l srm://srmv2.ific.uv.es/lustre/ific.uv.es/grid/vo.agata.org/e682-test/run_0109.dat.05-07-15_19h35m58s/Data/00A/SRM_AGATA_event_mezzdata.cdat.0000  
$ lcg-del -d srm://srmv2.ific.uv.es/lustre/ific.uv.es/grid/vo.agata.org/e682-test/run_0109.dat.05-07-15_19h35m58s/Data/00A/
```

How to run a job on GRID

More detailed info in <http://ns.ph.liv.ac.uk/EGAN/files/wednesday/GRID-HOWTO.pdf>

1. Connect to a UI with the valid GRID certificate:

```
$ ssh -X username@ui06.ific.uv.es
```

2. Create a proxy:

```
$ voms-proxy-init-voms vo.agata.org
```

3. Create a JDL file and a Script file to be executed on the Grid: myJob.jdl, myJob.sh

4. Use the following commands:

myJob.sh: contains all the actions to be executed on WN

myJob.jdl: contains the Job Description Language

Executable=myJob.sh

StdOutput=terminal output

StdError=terminal output errors

InputSandBox= files to be moved to the WN

OutputSandBox=terminal output files

Available CEs:

for IPNL-Tier2 lyogrid07.in2p3.fr:8443/cream-pbs-vo.agata.org

for IPHC-Tier2 sbgce2.in2p3.fr:8443/cream-pbs-vo.agata.org

for IPNO-Tier2 ipngrid04.in2p3.fr:8443/cream-pbs-agata

for IFIC-Tier2 ce03.ific.uv.es:8443/cream-pbs-short

for IFIC-Tier2 ce03.ific.uv.es:8443/cream-pbs-agataL

Job management tools: `$ glite-wms-job-xxx --help`

- List the available CEs:

```
$ glite-wms-job-list-match -a myJob.jdl
```

- Submit the job (-o stores the job ID in a file, -r selects de CE)

```
$ glite-wms-job-submit -a -o jobID -r CE myjob.jdl
```

- Status of the submitted job (-i takes the file with the job ID)

```
$ glite-wms-job-status -a -i jobID
```

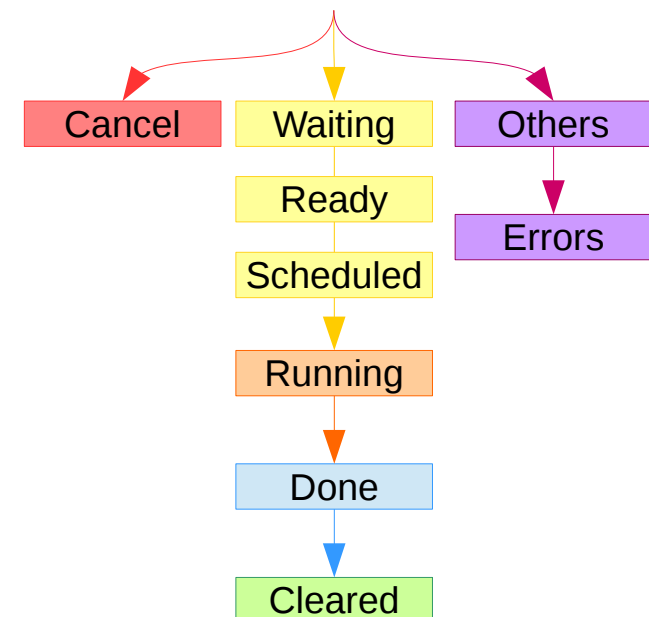
- Retrieve of the results when the job status is done:

```
$ glite-wms-job-output -a -i jobID
```

- Cancel a submitted job :

```
$ glite-wms-job-cancel -a -i jobID
```

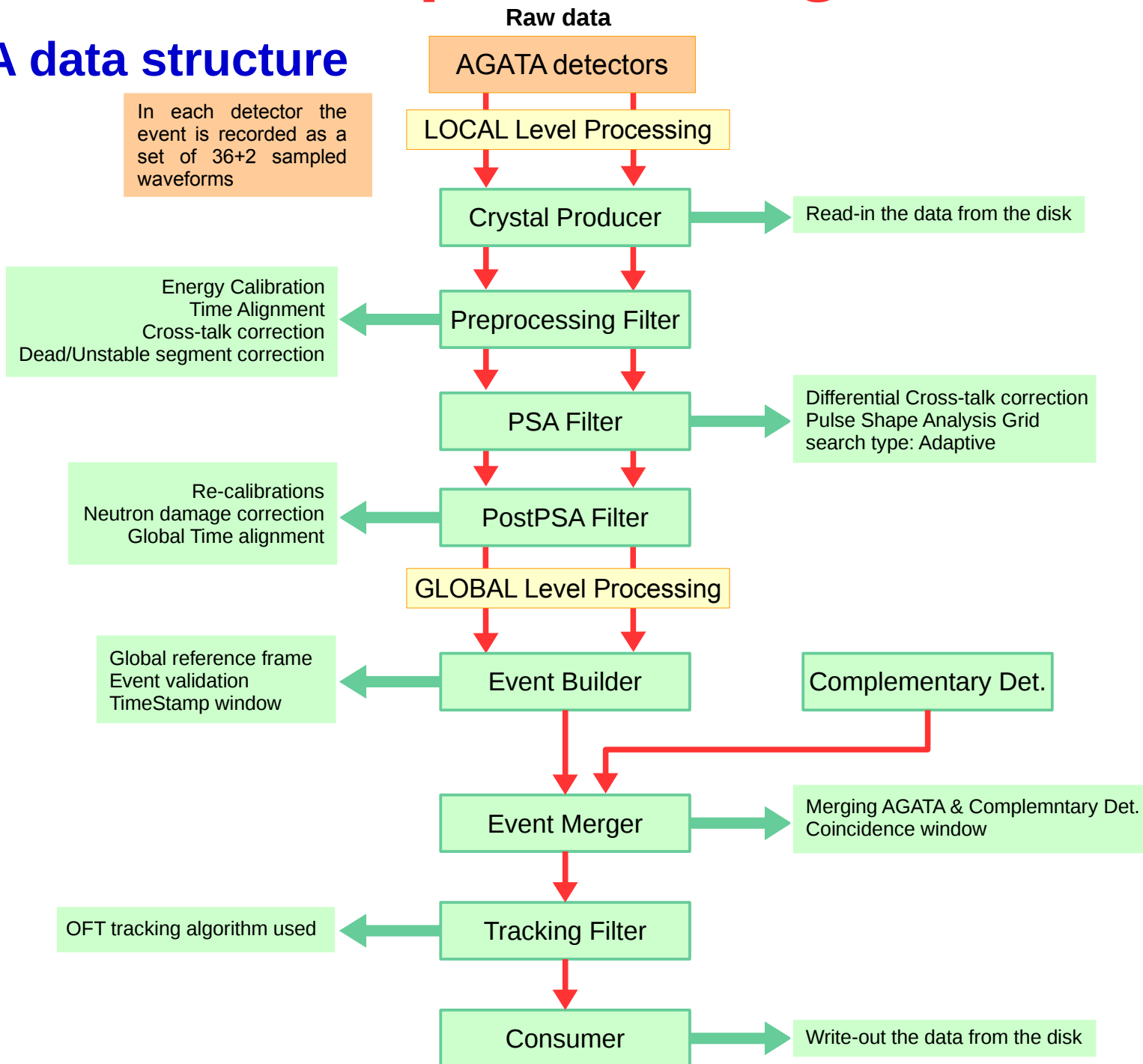
Job Status after submission



AGATA data processing on GRID

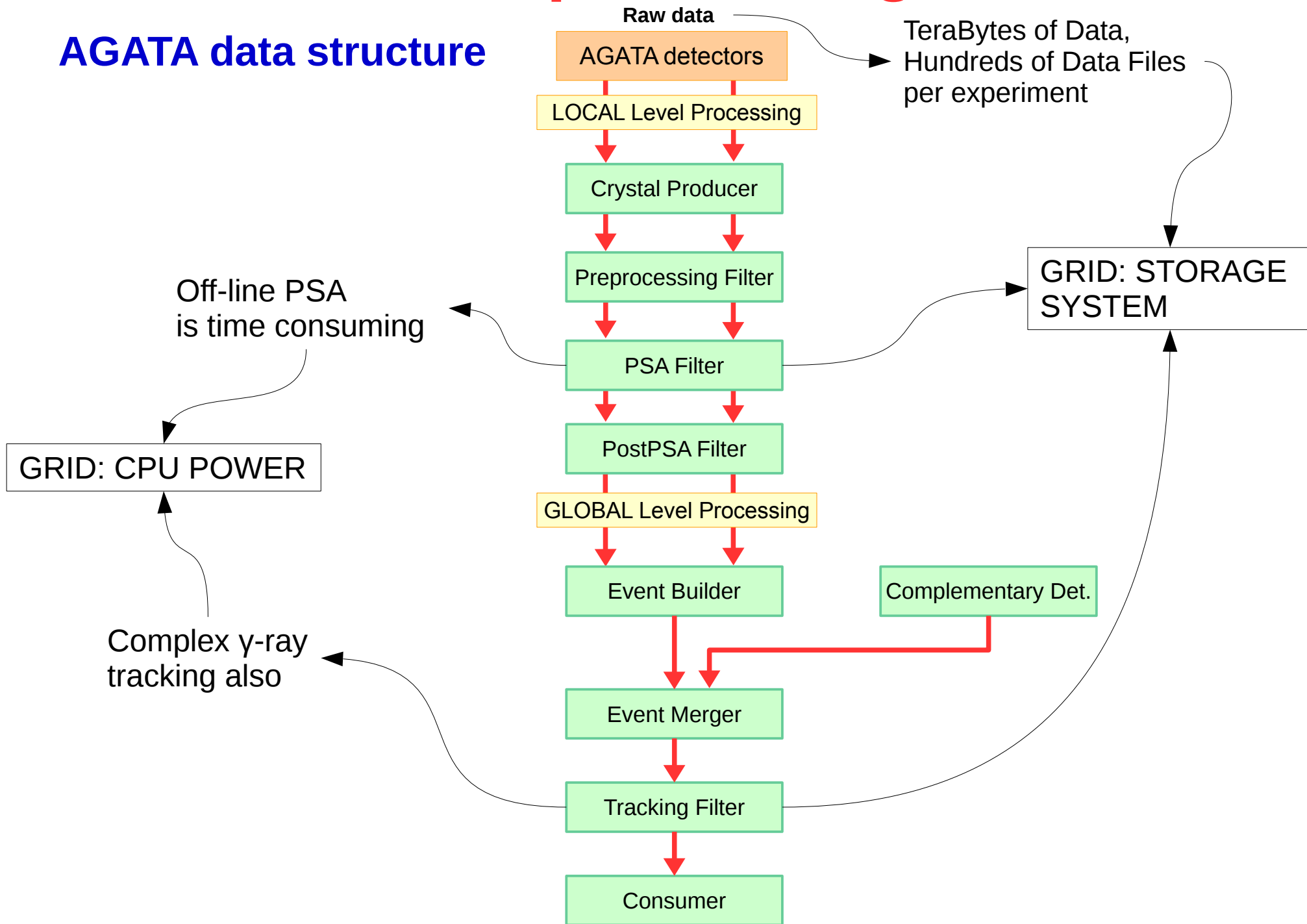
AGATA data structure

In each detector the event is recorded as a set of 36+2 sampled waveforms



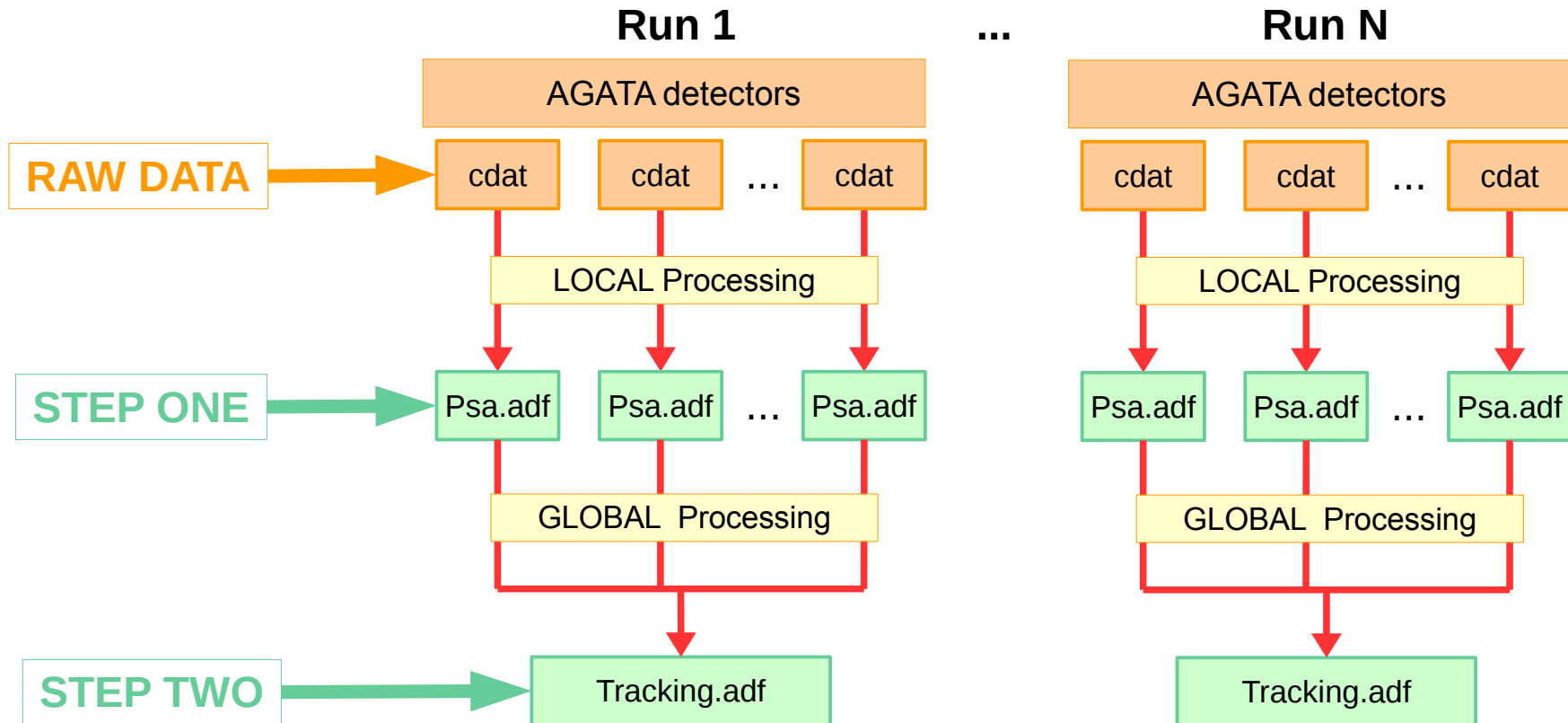
AGATA data processing on GRID

AGATA data structure



AGATA data processing on GRID

Methodology



1. Local processing file by file to decompose recorded waves: **PSA Algorithm**

2. Global processing run by run to reconstruct the gamma-ray path: **Tracking Algorithm**

AGATA data processing on GRID

What do we need for the replay?

Topology. conf	List of Narval actors to run and chain together all the configuration files
ADL/	Calculated signal basis for each ATC, needed for the PSA
ADF.conf	Definition of adf frames used for this analysis
gen_conf.py*	Generator of the configuration files for each actor and output directory.
Conf/	Configurations of actors and calibrations directory, read during initialization
Data/	Link to the experimental data and spectra directory (ln -s Path_to_Data/Data/Data)
Out/	Output data and spectra directory produced during data replay. It is created by gen_conf.py. Same structure as Data directory.
FEMUL*	Narval-emulator program used when the data needs to be replayed after an experiment has been performed. This program is essentially the same as the version used during experiment except the actors are written in C++.

*Modifications:

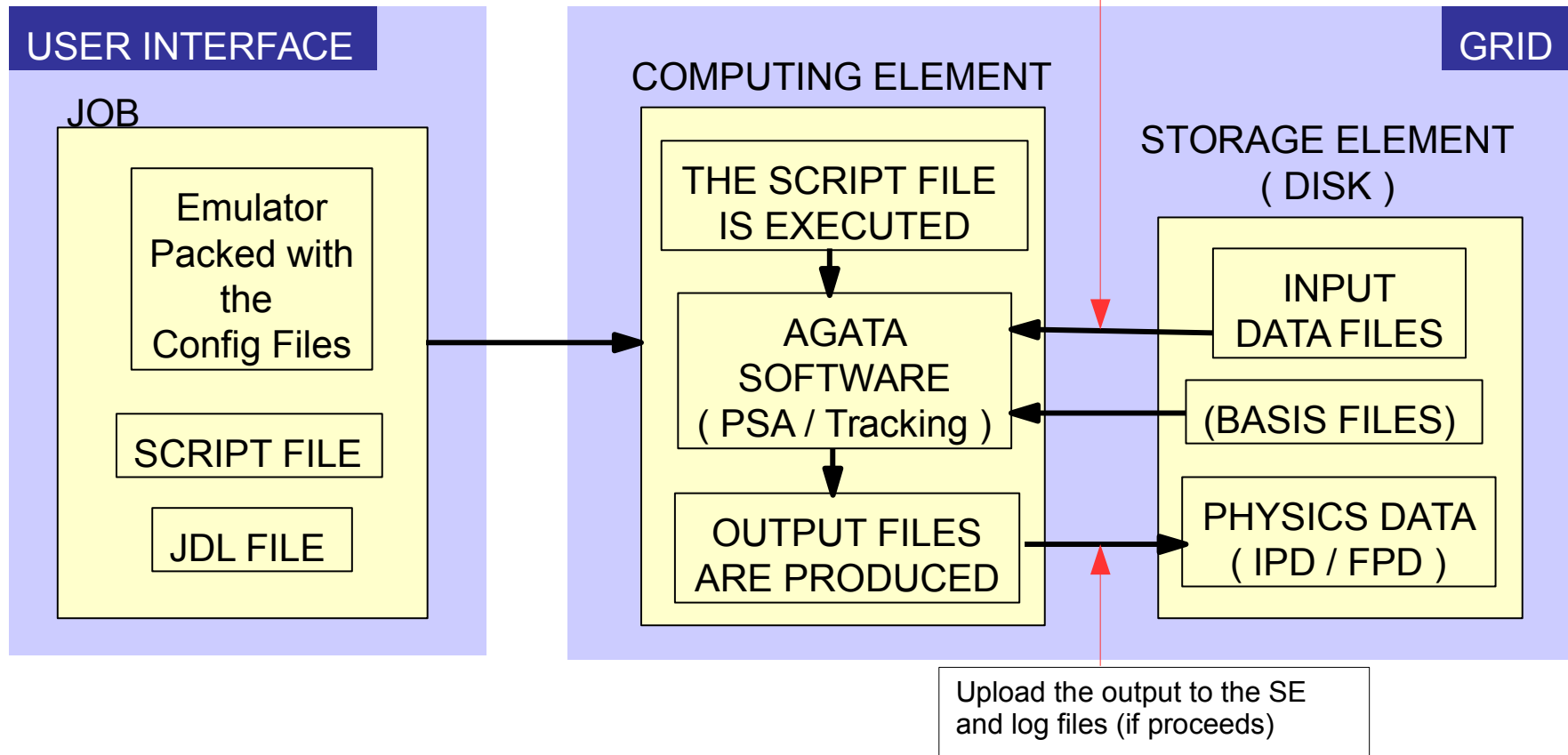
- add the new path for the psa basis on GRID
- comment "AllInputFiles" in the CrystalProducer actor for reading the cdat files individually (optional, depends on how you write the script)
- any analysis correction you could need (optional)

*Depending on the version of Scientific Linux in the UI you may need to recompile the FEMUL software on GRID.

AGATA data processing on GRID

How to run a job

- Define the topology you want to use
- Prepare the gen_conf.py for the Job and pack with the Configuration directory
- Pack the emulator (here femul)
- Create the script file to be executed in the WN
- Create the JDL file describing your Job
- Select the CE to use
- Launch the job



AGATA data processing on GRID

How to run a job. Example of files

myjobAgata.sh

myjobAgata.jdl

#Job description for myJobAgata.jdl

```
Executable = "myJobAgata.sh";
StdOutput = "std.out";
StdError = "std.err";
InputSandbox =
{"myJobAgata.sh", "AgataSoftware.tar.bz2",
Conf.tar.bz2", "Topology.conf", "ADF.conf"};
OutputSandbox = {"std.out", "std.err"};
```

Topology. conf	Local or Global topology sent to GRID → myjobAGATA.jdl
ADL/*	Basis for PSA stored in the SE of IFIC
ADF.conf	Sent to GRID → myjobAGATA.jdl
gen_conf.py	Packed within Conf.tar.bz2
Conf/	Packed and sent to GRID → myjobAGATA.jdl
Data/*	Raw data stored in the SE of IFIC
Out/	Created in GRID → myjobAGATA.sh
FEMUL	Software packed and sent to GRID → myjobAGATA.jdl

* To avoid downloading the files to do the replay they have been previously copied from to the Tier1 SE to IFIC SE which can be seen from the CE of IFIC.

Example for PSA

```
#!/bin/sh
```

#uncompress AgataSoftware

```
tar -xjf AgataSoftware.tar.bz2
```

#uncompress Configuration

```
tar -xjf Conf.tar.bz2
```

#set environment (and compile the femul if needed)

```
cd AgataSoftware; source scripts/add2path.sh $PWD /lib /bin; cd -
```

#generate the configuration and the output directory

```
mv Conf/gen_conf.py .
```

```
./gen_conf.py
```

#link input RAW data and PSA basis files

```
ln -s /lustre/ific.uv.es/grid/vo.agata.org/e682/run_0027.dat.25-06-15_18h07m37s/Data/13C/SRM_AGATA_event_mezzdata.cdat.0011 Data/13C/event_mezzdata.cdat
```

```
ln -s /lustre/ific.uv.es/grid/vo.agata.org/BaseFilesGANIL
```

#run femul

```
AgataSoftware/agapro/WinCtest/femul ./Topology.conf | tee Out/13C/13C.log
```

#upload the output files (adf and spectra) to the SE(IFIC)

#adf copy

```
mv -v ./Out/13C/psa_0000.adf ./Out/13C/psa_0011.adf
```

```
lcg-cp -v --vo vo.agata.org file://`pwd`/Out/13C/psa_0011.adf
```

```
srm://srmv2.ific.uv.es/lustre/ific.uv.es/grid/vo.agata.org/e682/run_0027.dat.25-06-15_18h07m37s/PSAData_test/13C/psa_0011.adf
```

```
rm -f ./Out/13C/psa_0011.adf
```

#spectra copy

```
tar -cjf Outputs_13C_cdat.0011.tar.bz2 Out/
```

```
lcg-cp -v --vo vo.agata.org file://`pwd`/Outputs_13C_cdat.0011.tar.bz2
```

```
srm://srmv2.ific.uv.es/lustre/ific.uv.es/grid/vo.agata.org/e682/run_0027.dat.25-06-15_18h07m37s/PSAOut_test/13C/Outputs_13C_cdat.0011.tar.bz2
```

AGATA data processing on GRID

How to run a job. Summary

I. Login into a User Interface machine and create a valid proxy

```
$ ssh -X username@ui06.ific.uv.ues  
$ voms-proxy-init-voms vo.agata.org
```

II. Write a JDL file and a Script file to be executed on the Grid

Content of a Script file:

- Uncompress the femul software with the right configuration
- Compile the software (if proceeds)
- Download the Data files to be processed (if proceeds)
- Run femul
- Upload the output and log files (if proceeds)

III. Grid commands for Job management

Submit the Job to the Grid :

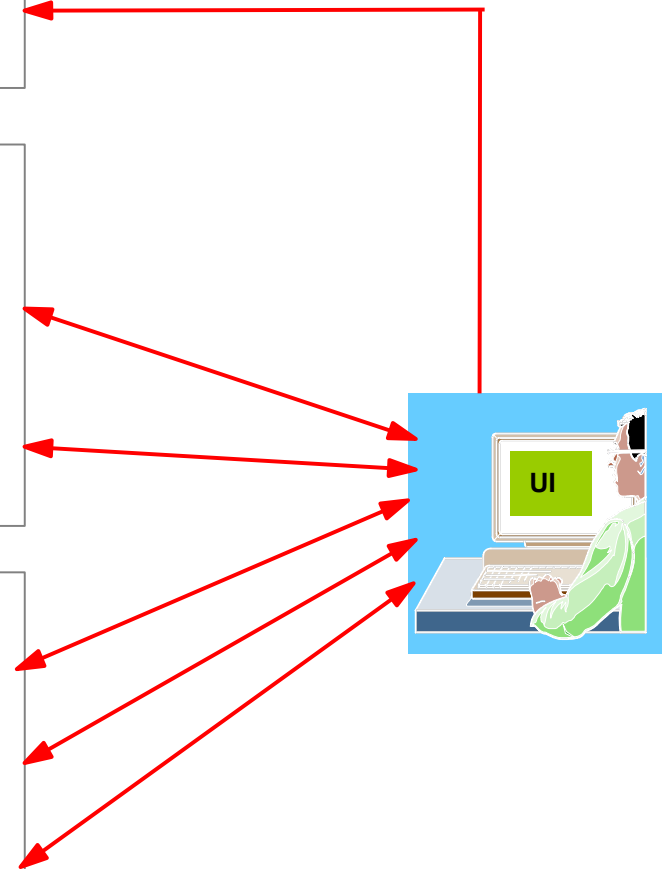
```
$ glite-wms-job-submit -a -o jobsID myJob.jdl
```

Follow up the status of the Job:

```
$ glite-wms-job-status -i jobsID
```

Retrieve the outputs:

```
$ glite-wms-job-output -i jobsID
```



WRITE A SCRIPT TO REPEAT AUTOMATICALLY ALL THE ACTIONS FOR EACH JOB

Running jobs on GRID

Example: 246 cdat 842Gb of a calibration run

- **Number of files processed:** 246 (max 3,7 Gb per file)
- **ProcessType:** PSA → AGATA reprocessing from cdat
- **Data access:** Lustre → All data is stored on the SE of IFIC
- **Site:** IFIC → CE used: ce03.ific.uv.es:8443/cream-pbs-agataL
- **Number of cores available:** ~100 → Check the cores available before launching the job with:
`lcg-infosites --vo vo.agata.org ce`
- **Jobs failed:** 0

Computing	Time of execution
PC 1 cpu local disk	~ 2,5 d
PC 20 cpu local disk	~ 5,5 h
GRID 246 cpu	~ 1-2h*

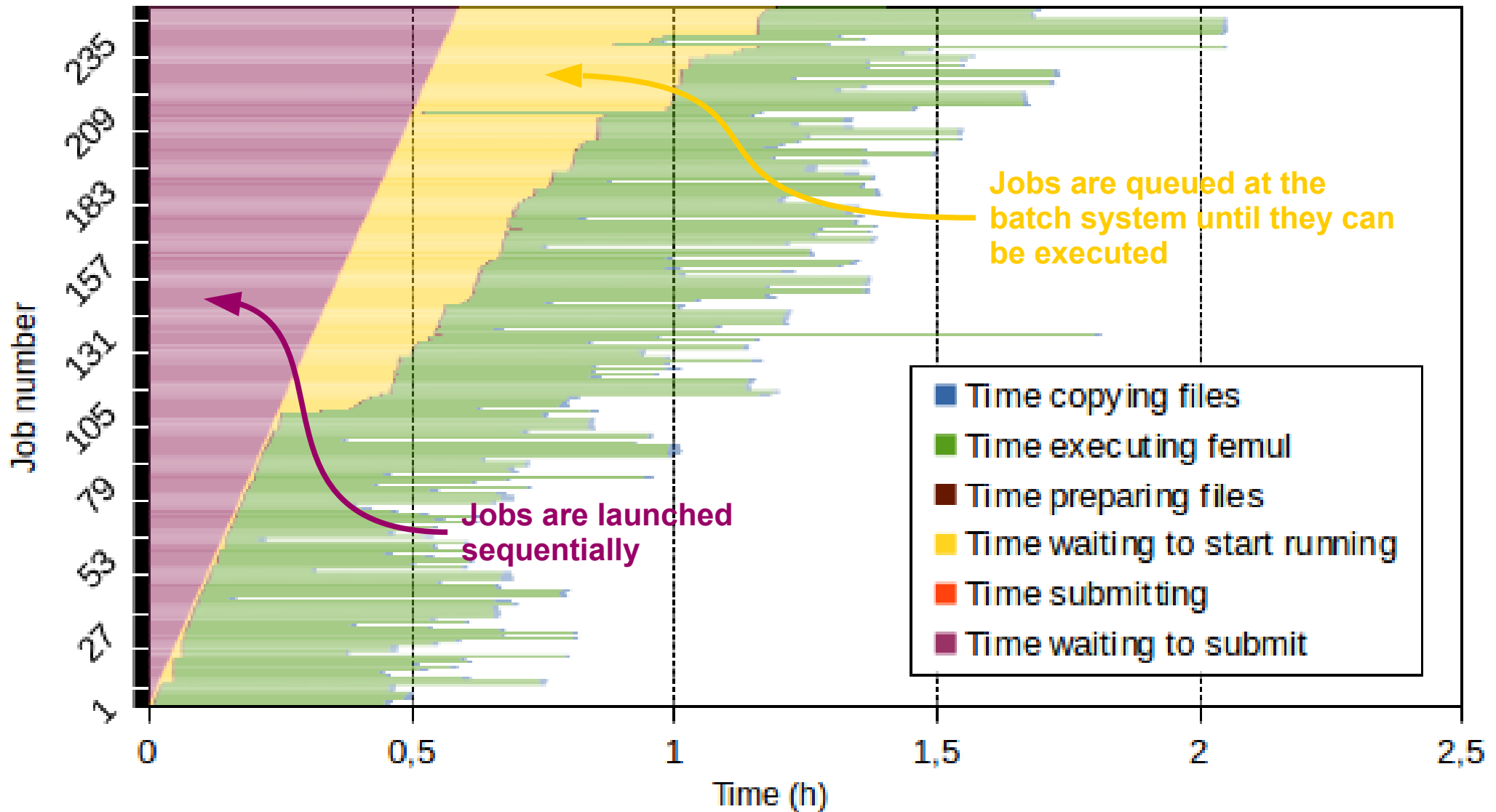
*Includes the copy of the data

Running jobs on GRID

Example: 246 cdat8 842Gb of a calibration run

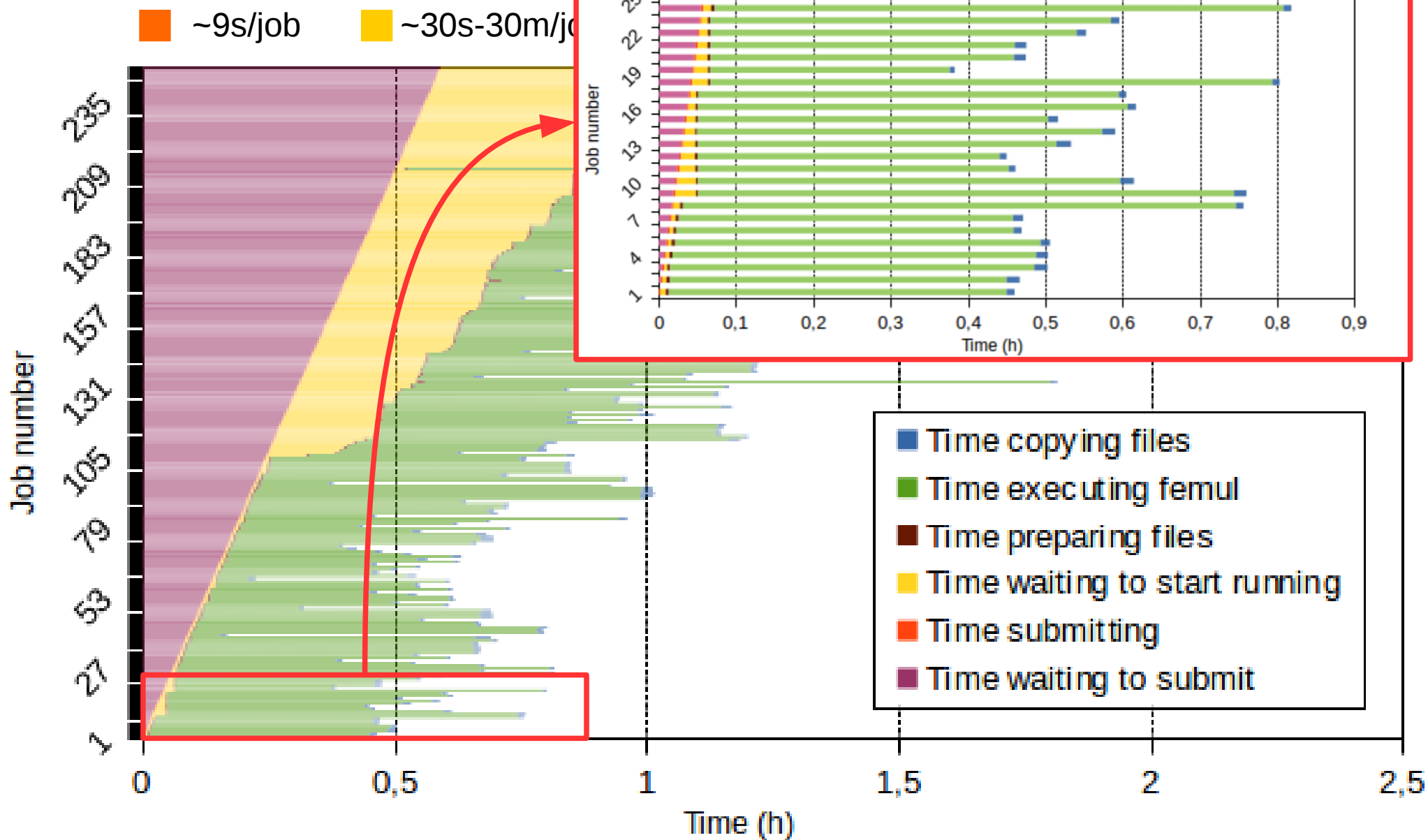
Time of execution

■ ~9s/job ■ ~30s-30m/job ■ ~30s/job ■ ~30m/job ■ ~30s-1m/job



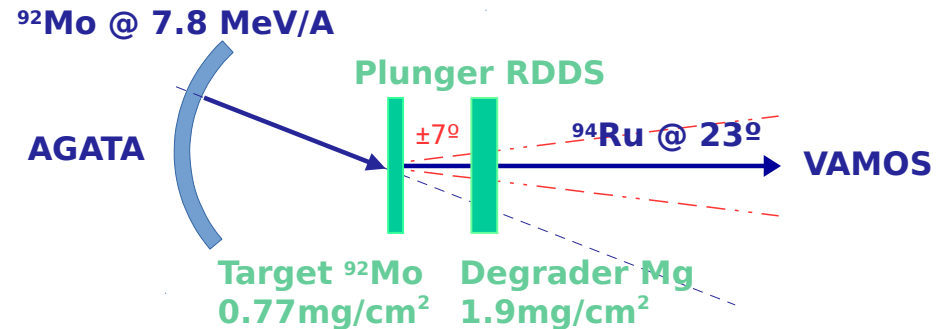
Running jobs on GRID

Example: 246 cdats 842Gb of a calibration run



Experiment e682 on GRID

- 23 AGATA crystals
- Nominal position
- Counting rate per crystal 50 kHz
- Total number of runs: 44
- Total number of cdat files: 1651 (3.9T)
- Data access: Lustre → All data is stored on the SE of IFIC
- Site: IFIC → CE used: `ce03.ific.uv.es:8443/cream-pbs-agataL`



PSA off-line

- Number jobs submitted: 1651
- ProcessType: PSA
- Number of cores available: ~300
- Jobs failed: 78
- Time of execution: ~ 10 h

Tracking off-line

- Number jobs submitted: 44
- ProcessType: Tracking
- Number of cores available: ~200
- Jobs failed: 0
- Time of execution: 1h 33m