



Deployment of Job Priority mechanisms in the Italian Cloud of the ATLAS experiment

Alessandra Doria¹, Alex Barchiesi², Simone Campana³, Gianpaolo Carlino¹,
Claudia Ciocca⁴, Alessandro De Salvo¹, Alessandro Italiano⁴, Elisa Musto¹,
Laura Perini⁵, Massimo Pistolese⁵, Lorenzo Rinaldi⁴, Davide Salomoni⁴,
Luca Vaccarossa⁵, Elisabetta Vilucchi⁶

*1 – INFN Sez. di Napoli, 2 – INFN Sez. di Roma1, 3 – CERN, 4 – INFN CNAF
5 – INFN Sez. di Milano, 6 – INFN Lab. Naz. di Frascati*



Why Job Prioritization and Fairshare?



ATLAS needs a mechanism to:

- grant different shares of computing resources to the various activities in the Grid.
- Maximize system utilization
- Incorporate historical resource usage and political issues

Fairshare implemented at the batch system level, assigning different resource shares to users with different VOs and VOMS attributes.



Current usage of resources by VOMS groups and roles must be published to the information system, to be used for matchmaking by WMS.



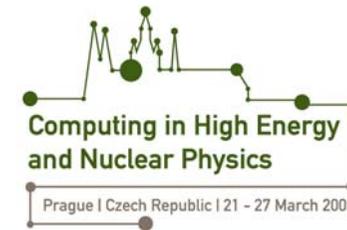
Outline



- Basic features of the ATLAS Computing model
- ATLAS job submission
 - Montecarlo Production System– PANDA
 - Distributed Analysis tools – GANGA & Pathena
- Deployment and test of VOViews
 - VOViews publication
 - Use of VOViews by the WMS
- Deployment and test of Fair Share in batch systems.
 - PBS/Torque + MAUI test @ Tier2
 - LSF test @ Tier1
- Conclusions



ATLAS computing model



Tier0 at CERN. Immediate data processing. Stores on tape all ATLAS data

Tier1. Data storage and reprocessing of data with better calibration or alignment constants. Physics Group Analysis

**Tier-2 . Complete replica of analysis data (AOD and DPD).
MC Simulation and User Analysis.**

Three Grid middleware infrastructures are used by the ATLAS distributed computing project:





Atlas Computing Model: the Grid interfaces



The Atlas Grid tools interface to all middleware types and provide uniform access to the Grid environment

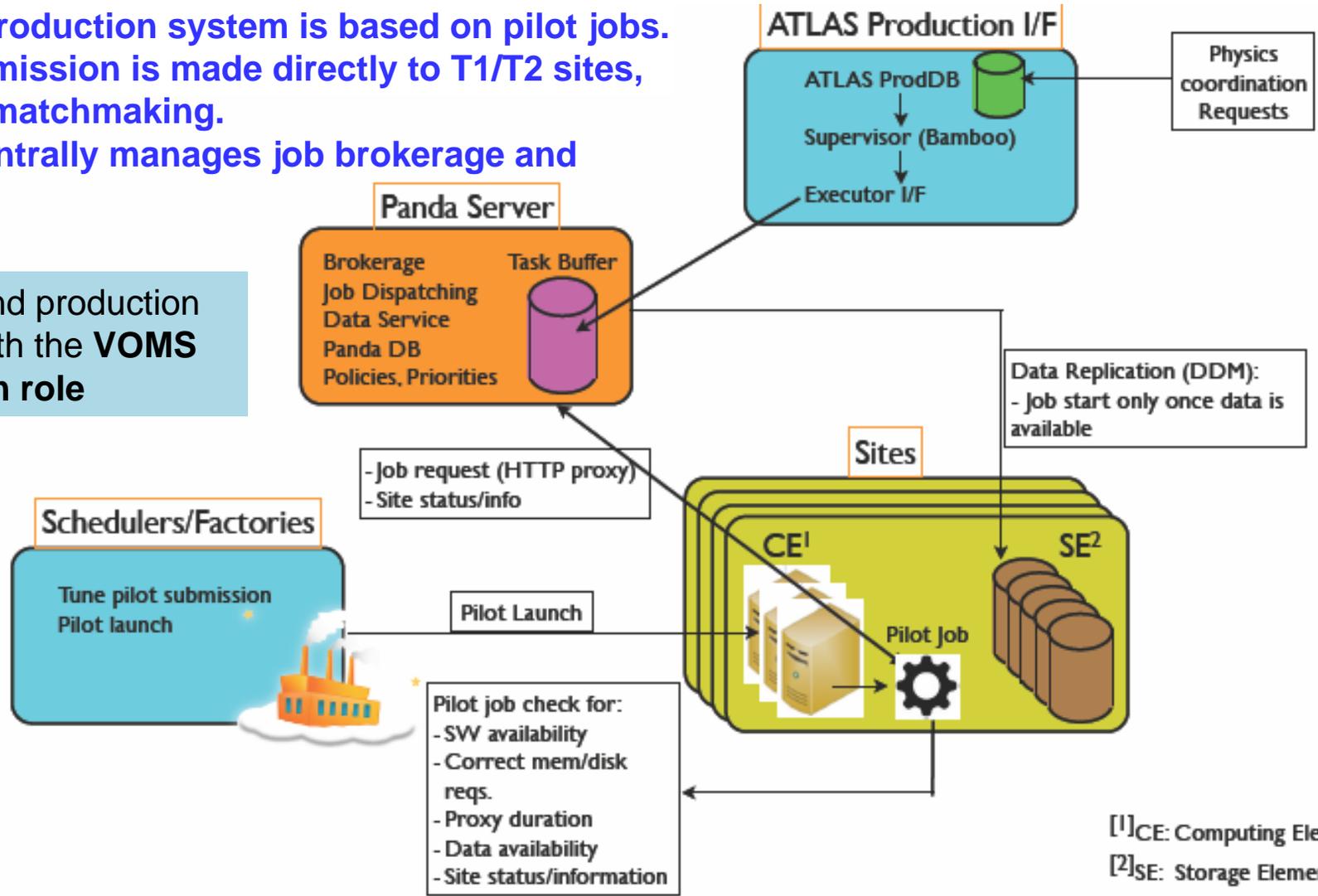
- The **VOMS** database contains the privileges of all ATLAS members; it is used to allow ATLAS jobs to run on ATLAS resources and store their output files on ATLAS disks
- the **DDM** (Distributed Data Management) system catalogues all ATLAS data and manages the data transfers
- The **ProdSys/Panda** production system schedules all organized data processing and simulation activities
- The **Ganga** and **Pathena** interfaces allow the analysis job submission: jobs go to the sites holding input data and output data can be stored locally or sent back to the submitting site.



ATLAS Production System

PANDA production system is based on pilot jobs.
Pilot submission is made directly to T1/T2 sites,
no WMS matchmaking.
Panda centrally manages job brokerage and priority.

All pilots and production jobs run with the **VOMS production role**



[1]CE: Computing Element
[2]SE: Storage Element



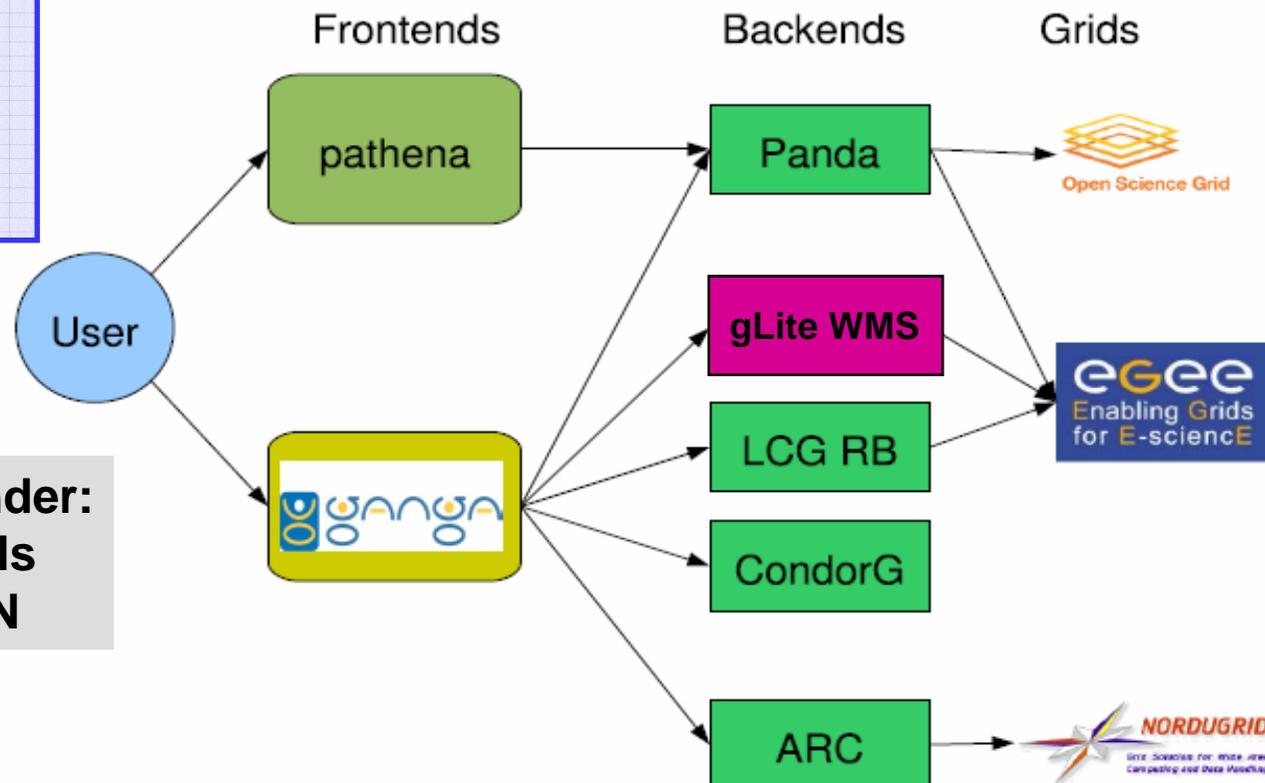
ATLAS distributed analysis framework



PATHENA: a client tool for PANDA to submit the user defined jobs on GRID
Same submission mechanism for production and analysis jobs.

GANGA: a single tool to submit/manage/monitor the user jobs on GRID
A first level of job brokering is based on data location.

GANGA submission via glite WMS takes advantage of Job Priorities



User analysis runs under:

- plain atlas credentials
- regional group FQAN





VOViews



- VOView concept is implemented in the information system: publishing the number of running, waiting jobs, free slots, etc. as a function of the VOs and also of the VOMS **FQANs***.
- LCMAPS in site Computing Elements maps different VOMS FQANs to different groups of users, associated to different resource shares or priorities in the underlying LRMS.
- CE configuration has been completely made by YAIM.
- The **lcg-info-dynamic-scheduler** produces the correct information to be published in the VOViews.
- **VOViews must be mutually exclusive**: achieved via appropriate use of **DENY tags**.

Based on the solution proposed
by the **Job Priorities WG**

*Fully Qualified Attribute Name consists of a VO,
a group, a role and a capability



Information publishing tested



```
dn: GlueVOViewLocalID=atlas, GlueCEUniqueID=atlasce01.na.infn.it:2119/jobmanager-  
lcgpbs-atlas, Mds-Vo-name=INFN-NAPOLI-ATLAS, o=grid  
GlueCEAccessControlBaseRule: VO:atlas  
GlueCEAccessControlBaseRule: DENY:/atlas/it  
GlueCEAccessControlBaseRule: DENY:/atlas/Role=production  
GlueCEStateRunningJobs: 0  
GlueCEStateWaitingJobs: 45
```

FQAN VOViews enabled in CE

```
dn: GlueVOViewLocalID=/atlas/Role_production, GlueCEUniqueID=atlasce01.na.infn.it  
:2119/jobmanager-lcgpbs-atlas, Mds-Vo-name=INFN-NAPOLI-ATLAS, o=grid  
GlueCEAccessControlBaseRule: VOMS:/atlas/Role=production  
GlueCEStateRunningJobs: 1  
GlueCEStateWaitingJobs: 0
```

```
dn: GlueVOViewLocalID=/atlas/it, GlueCEUniqueID=atlasce01.na.infn.it:2119/jobma  
nager-lcgpbs-atlas, Mds-Vo-name=INFN-NAPOLI-ATLAS, o=grid  
GlueCEAccessControlBaseRule: VOMS:/atlas/it  
GlueCEStateRunningJobs: 189  
GlueCEStateWaitingJobs: 473
```

We verified that the IS publishes the correct number of Waiting and Running jobs for each GlueVOViewLocalID



ATLAS requirements on gLiteWMS



- Shares corresponding to different activities should be exposed to the gLite WMS via the Information System, for two reasons:
 - to let the WMS estimate the rank of a CE based on the information available for the share the user will be mapped to
 - to let the user know how many jobs are running, waiting, etc. separately for each share available to the VO.
- the gLite WMS should not match more than one VOView for the same CE.

Requirements have been verified by our tests



Deployment of Fair Share in batch systems



- In the Italian Cloud, lcg-CE is used with two different batch systems:
 - **PBS/Torque+Maui** at Napoli, Milano and LNF Tier2s
 - **LSF** at Roma1 Tier2 and CNAF Tier1.
- Both systems were configured to deploy a Fairshare policy between :
 - Atlas, without any role or group
 - Atlas with production role
 - Atlas Italian analysis group
 - Any other supported VO

```
vo:/atlas
```

```
VOMS:/atlas/Role=production
```

```
VOMS:/atlas/it
```

For our purposes, we use **group based Fairshare** , where each group corresponds to a different VOView, for example to a different physical activity or regional community.



PBS-MAUI @ Napoli Tier2



- In the MAUI scheduler (version 3.2.6p20) , a large number of parameters can contribute to the calculation of job priorities, with different weights.
- Fairshare based priority is regulated by the following parameters:
 - **FSTARGET** - percentage of resources to be used by a credential
 - **FSINTERVAL** - duration of each fairshare window
 - **FSDEPTH** - number of fairshare windows factored into current fairshare utilization
 - **FSDECAY** - decay factor applied to weighting the contribution of each fairshare window – set to 1 for all tests
 - **FSPOLICY** - metric to use when tracking fairshare usage, set to job wallclocktime for all tests.
- MAUI recalculates the job priorities at the end of each window. The fairshare contribution increases the priority of a job if the usage of resources by the job credential is lower than the FSTARGET.



PBS-MAUI @ Napoli Tier2



Test with two groups: atlas user and production

The relative usage of resources is represented by the average wall clock time calculated from the test beginning .

The trend of the first 10 windows depends on the past resource usage, when only production group was running

The asynthetic share corresponds to the expected Fairshare targets

Test parameters:

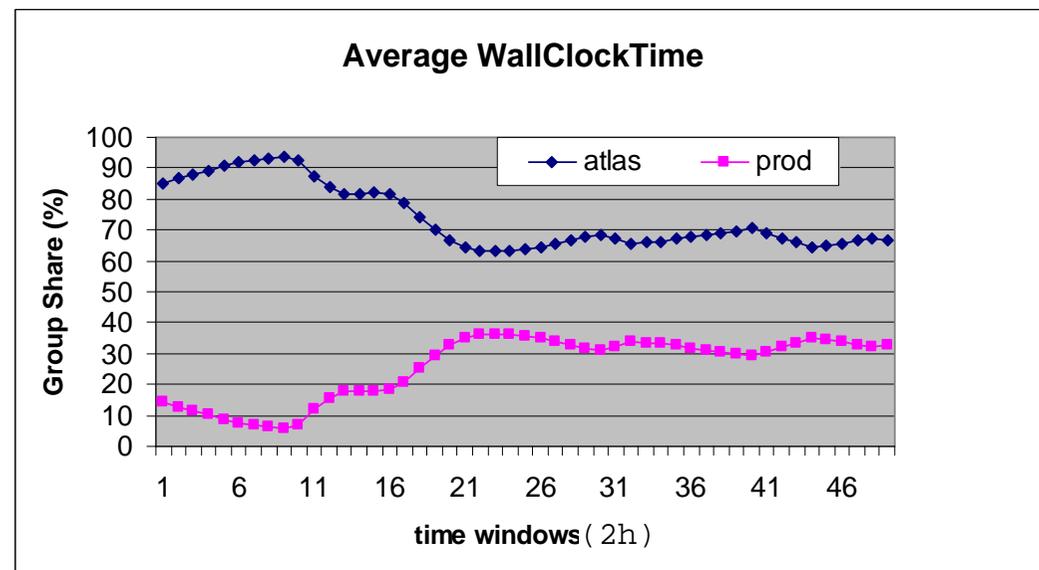
Fairshare Targets:

ATLAS User 70%

ATLAS Production 30%

10 Time Windows 2h long

Job length: 4 hours





PBS-MAUI test @ Napoli Tier2



Test with three groups

This test confirms that the expected shares are reached, also with more groups.

Test parameters:

Fairshare Targets:

ATLAS 40%

ATLAS PROD 30%

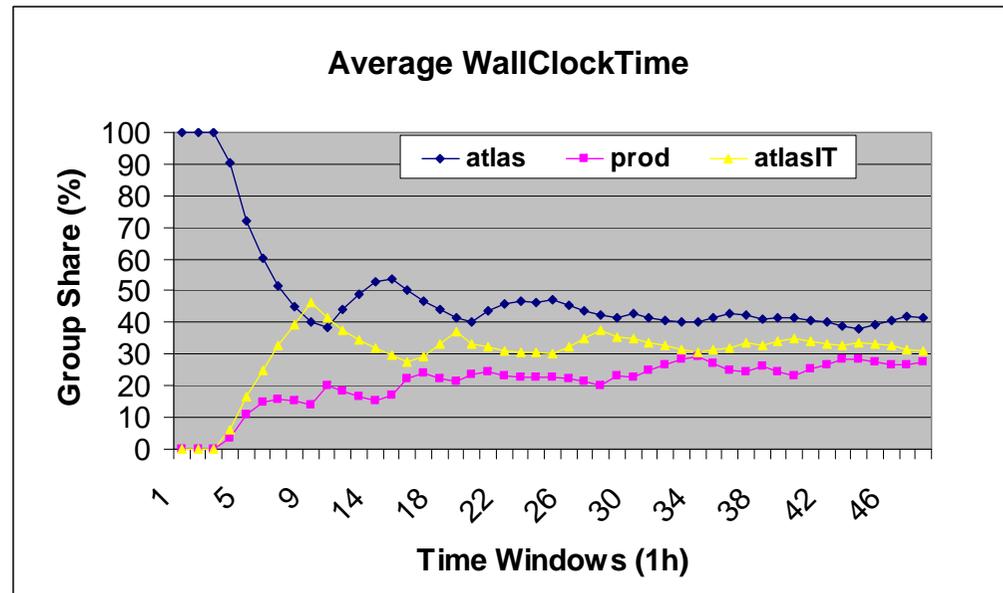
ATLAS IT 30%

10 Time Windows 1h long

Job length: 2 hours

NB: Values used as Fairshare targets are just for test.

In the final configuration we allocate 50% of resources to production jobs; the remaining 50% to the different analysis groups, with a share conforming to ATLAS policies.



It's good practice to make the window length comparable to the typical job duration. Having shorter jobs and windows allows to reach the fairshare target faster. More parameters, like QUEUETIME can be added to determine the job priority.



LSF Test @ CNAF Tier1



- **Batch system configuration**
 - Batch system: Paltform LSF, version 7.0.2
 - Scheduling algorithm: Hierarchical FairShare
 - FairShare based user's priority calculation:
 - Formula: Shares/Resources used
 - History time window: 5 hours
 - Time period used by the scheduler to compute the total resources used by every user in the time window.
- **Two different tests:**
 - **intra-VO FairShare for a single VO**
 - Only two users subgroups, belonging to the same VO, submitted jobs so they could use all the available resources.
 - **multiple intra-VO FairShare.**
 - Two VOs have been involved in the test; for each VO two users subgroups have submitted jobs.



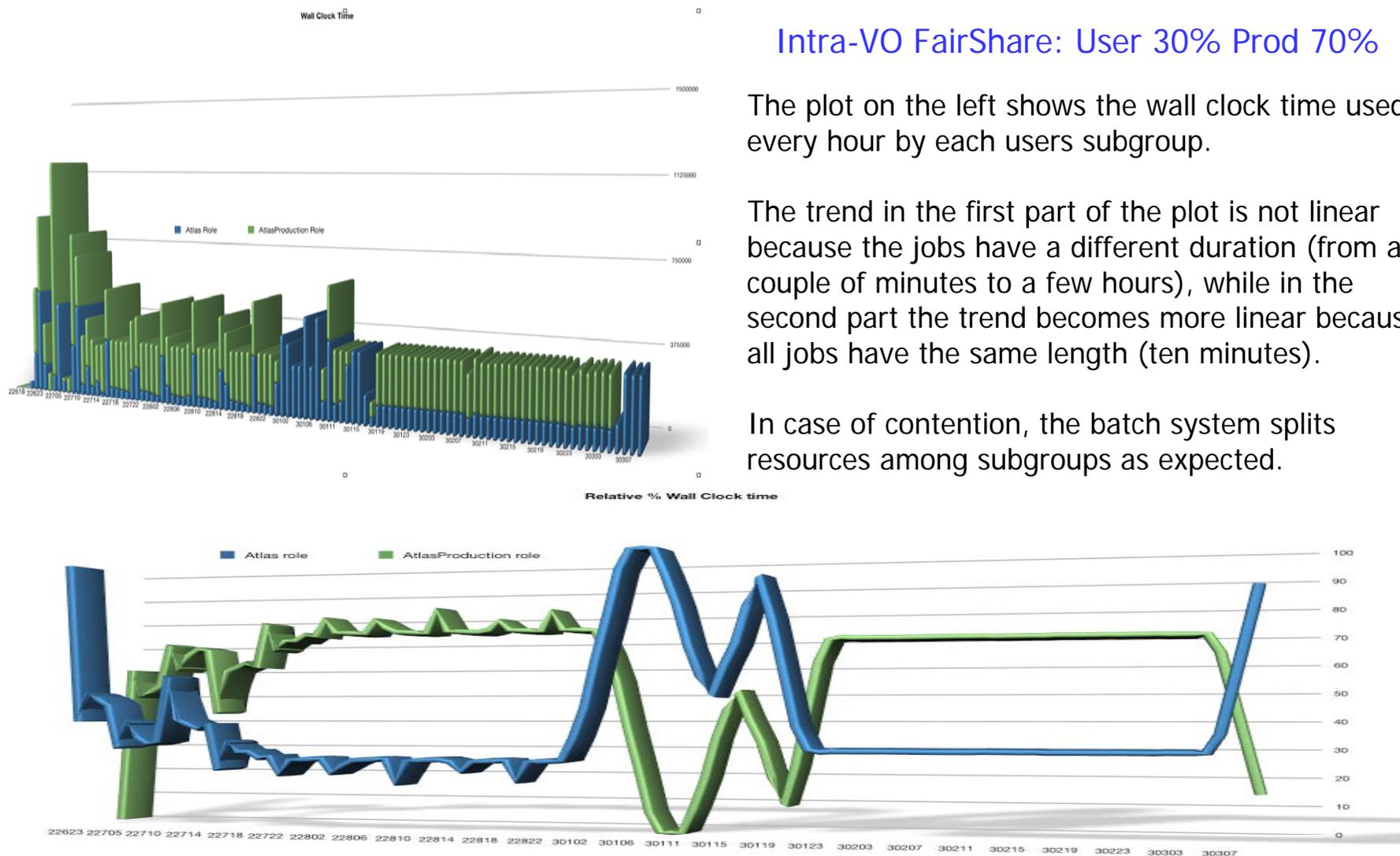
LSF Test @ CNAF: Results(1)

Intra-VO FairShare: User 30% Prod 70%

The plot on the left shows the wall clock time used every hour by each users subgroup.

The trend in the first part of the plot is not linear because the jobs have a different duration (from a couple of minutes to a few hours), while in the second part the trend becomes more linear because all jobs have the same length (ten minutes).

In case of contention, the batch system splits resources among subgroups as expected.





LSF Test @CNAF: Results(2)

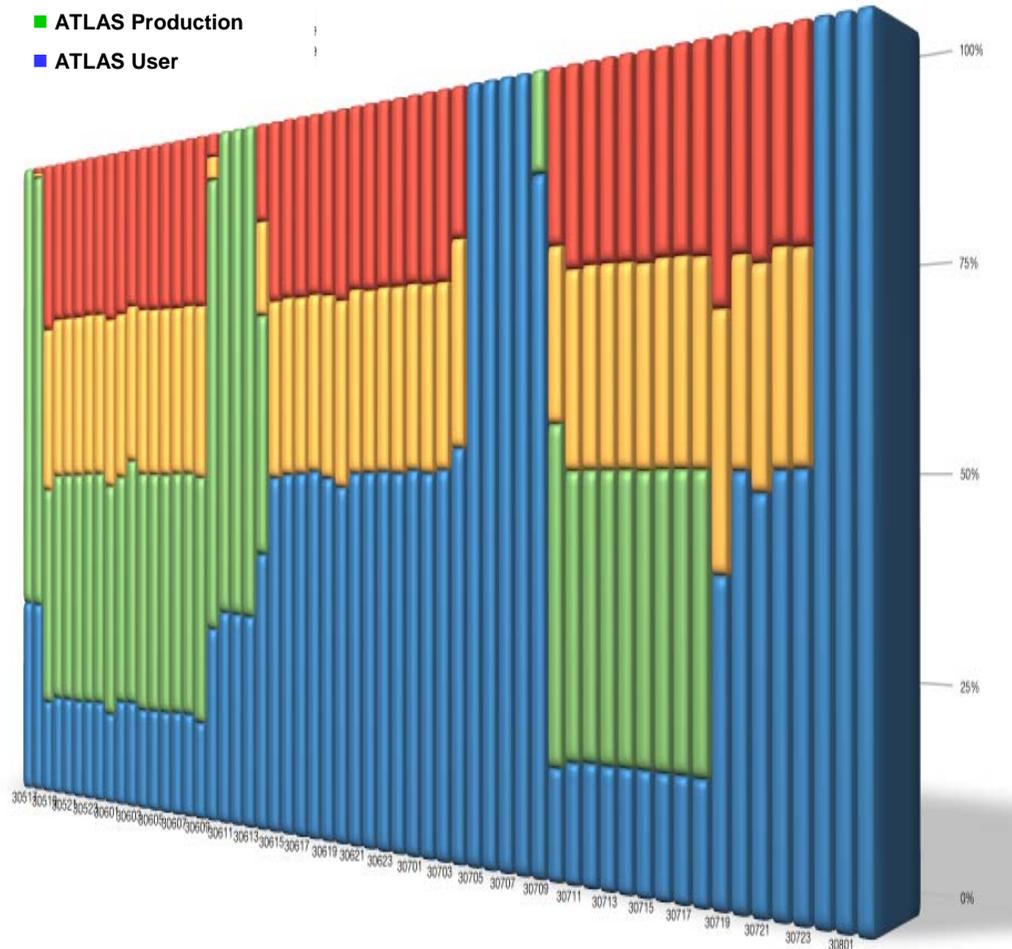
- LHCb production
- LHCb User
- ATLAS Production
- ATLAS User

Share %

Multiple intra-VO FairShare: 50 % for each VO

ATLAS: 30% User, 70% Production

LHCb: 50% User, 50% Production



The plot shows the percentage of wall clock time used every hour by each users subgroup.

In case of contention, the batch system grants the same amount of wall clock time to each VO . At the same time the batch system splits the VO's available resources among its users subgroups according to the configured share.

When there isn't contention, whoever is running is free to use all the available resources.



Conclusions



- Contention of computing resources between experiment central activities and user analysis requires mechanisms to determine resource sharing according to site, cloud and/or VO level policies.
- We deployed and tested Job Priorities and Fairshare in local LRMS.
- The correct handling of VOViews by the Information System and the gLite WMS has been successfully verified in the ATLAS Italian Cloud.
- In order to tune an optimal configuration for ATLAS, we will make more realistic tests, using real production jobs and HammerCloud analysis tests .