

# Table of Contents

<b>SitesSetupAndConfiguration.....</b>	<b>1</b>
<b>Help and Support.....</b>	<b>2</b>
<b>Recommendations and mandatory services.....</b>	<b>3</b>
Baseline middleware.....	3
Storage.....	3
Computing Element.....	3
Choice of Batch system.....	3
Batch system shares and limits.....	3
Forced Pilot termination.....	4
Worker Node hardware resources.....	4
Worker Node logical configuration.....	5
Squids.....	5
Network.....	5
Recommended CPU, Storage and Network capacity.....	5
Limit concurrent FTS transfers to a destination (Optional).....	5
<b>Configuration within ADC.....</b>	<b>7</b>
<b>Documentation of deployment steps.....</b>	<b>8</b>
<b>Documentation on operation.....</b>	<b>9</b>
Global overview WLCG.....	9
Global overview ATLAS.....	9
Site availability reliability.....	9
Site blacklisting.....	9
Typical ATLAS jobs.....	9
Grid Storage.....	9
<b>Grid site decommissioning.....</b>	<b>10</b>
<b>Grid component decommissioning or migration.....</b>	<b>11</b>
WLCG site : Replacing a SE with another SEs.....	11
WLCG site : migration to diskless sites.....	11
<b>Opportunistic Resources.....</b>	<b>13</b>
<b>Blacklisting of permanently broken site.....</b>	<b>14</b>
<b>FAQ.....</b>	<b>15</b>

# SitesSetupAndConfiguration

# Help and Support

- The first entry point for sites is their cloud squad support: atlas-adc-cloud-XX at cern.ch .
- In case of urgent matters please contact the ATLAS Computing Run Coordinator through atlas-adc-crc AT cern.ch .
- For information that you believe is worth being discussed within the whole ATLAS distributed computing community use (don't abuse) atlas-project-adc-operations AT cern.ch .
- The ICB should be informed of new resources using this procedure

# Recommendations and mandatory services

## Baseline middleware

- Sites should keep middleware and software up to date. Here is the link to the baseline versions for each product that requires a minimum version.
  - ◆ <https://wlcg-mw-readiness.cern.ch/baseline/current/>

## Storage

- Documentation on Grid storage deployment
- Since 2019, SRMless storage is being implemented especially in DPM sites. More informations here
- In order to keep the storages consistent with the Rucio catalogs, automatic consistency checks should run on regular basis. To do this, the sites are expected to provide storage dumps on a monthly or quarterly basis according the information here. Dumps are also expected in case a major incident affected the storage.

## Computing Element

- There are mainly 2 types of CEs, ARC-CE and HTCondorCE
- ARC-CE is now the one most commonly installed at new sites in Europe, HTCondorCE is typical in the US
- Sites with CREAM-CE should consider to migrate to another one

## Choice of Batch system

- ATLAS would strongly recommend that the site run a batch system which works well with the CE it uses, allows job requirement passing, e.g. memory, cores, walltime and is integrated with cgroups. HTCondor and SLURM are very well supported in our experience.
- ATLAS would prefer to have a fully dynamic configuration at site (possible with the above mentioned batch systems). The ATLAS payload scheduler, driven by the workflow management, will take care about optimizing the usage of the resources prioritizing the most mission critical jobs for ATLAS.
- You can find a batch system comparison table with information also about LSF, torque-maui and UGE/SoGE here

## Batch system shares and limits

- As of February 2020, the batch system configuration preferred by ATLAS is a single batch queue accepting both single-core and multi-core jobs, both analysis and production ("grand unified" queue). The shares between the different types of jobs are managed dynamically by ATLAS, with no hard partitions in the site batch system configuration, and (if possible) no soft limits either.
  - ◆ If needed, limits on jobs of a certain type (e.g. `resource_type_limits.SCORE`) can be set in the CRIC configuration of the PanDA queue.
  - ◆ For ATLAS, running on Grand Unified queues means that the workload can be adjusted dynamically based on current priorities. For sites, providing Grand Unified queues means that the queue can be assigned any kind of job at any time, so there should be fewer periods when the batch system cannot be filled due to lack of available jobs of a certain type.
  - ◆ NOTE: the migration to "Grand Unified" queues is in progress as of March 2020; most sites are still configured with separate production and analysis queues.

- ◆ NOTE: a few sites still do not support running single-core and multi-core jobs on the same batch queue; in this case, single-core production and single-core analysis will be grand-unified, with multi-core production remaining on a separate queue. The share should be 80% multi-core (8 core) and 20% single core jobs with a dynamic setup.
- ◆ NOTE: depending on site needs, other queue configurations are still supported. E.g. a site not supporting multi-core could run a grand-unified single-core analysis+production queue. A site not supporting analysis could run a unified single-core+multi-core production queue.
- The type of jobs can be identified
  - ◆ Analysis jobs come with /atlas/Role=pilot while Production jobs with /atlas/Role=production
  - ◆ /atlas/Role=lcgadmin is used only for SAM tests which are very few jobs per hour (usually one) and they should have the highest priority and small fair share
- As an example, on March 3rd 2020, the target global shares<sup>↗</sup> set by ATLAS are 83% production, 11% analysis, 6% others - these will change dynamically based on ATLAS needs, and should not be reflected in hard settings in batch system configurations.
  - ◆ Analysis jobs come in burst so it can be that only few analysis jobs are assigned to the site
- See the Job Monitoring<sup>↗</sup> dashboard in grafana to monitor analysis vs production jobs - specifically, plots by 'prod source' and by 'production type'
- cgroups: to control job resource usage as CPU/memory on kernel level, cgroups are a desirable feature. SLURM, HTCondor, LSF (>9.1), UGE (>8.2) all support cgroups at least to limit the memory.
- ATLAS sites should really avoid killing ATLAS jobs based on VMEM info, VMEM does not represent physical memory used anymore, it indicates just the memory that can be mapped at some point, and in the 64bit era it can become a huge number compared to the memory actually used.
  - ◆ More about memory and how it is mapped to the different batch systems and CEs parameters can be found in the WLCG Multicore TF pages.
  - ◆ Batch systems like torque and SoGE<sup>↗</sup> are not integrated with cgroups and cannot handle limiting the memory correctly anymore. To protect sites from memory leaks even at these sites ATLAS jobs can now monitor the memory they use with a tool that extracts memory information from smaps and is working to make sure jobs are not exceeding the site specification using this information.

## Forced Pilot termination

- ATLAS sites might be forced to kill misbehaving ATLAS jobs for different reasons. The Pilot can trap the signals listed here<sup>↗</sup>. When such a signal is received, the Pilot aborts the job, data transfer or whatever it is doing at the moment, and informs the server with the corresponding error code.
- **Notice** that the pilot typically needs 3-4 minutes to wrap up the job so we recommend to wait at least that time before an eventual SIGKILL signal is sent.

## Worker Node hardware resources

A node should typically provide the following amount of hardware resources *per single-core job slot*

- 20 GB of disk scratch space, although 10-15 GB is workable.
- At least 2 GB of (physical) RAM, but having 3-4 GB would be beneficial
- Enough swap space such that RAM + swap  $\geq$  4 GB
- As a rule of thumb, about 0.25 Gbit/s of network bandwidth (might want higher for more powerful CPUs).
- CPU performance increases of up to ~40% (according to HEP-SPEC06) can be gained by using hyperthreading; in this case each node would require additional disk and RAM (and to a lesser extent, network bandwidth) to support the additional virtual cores.

## Worker Node logical configuration

See AtlasWorkerNode : OBSOLETE

### Squids

- Sites are requested to have a squid (ideally two for resilience) to allow WN to access conditions data (Frontier) and CVMFS data (SW releases) in an efficient manner which won't put load on the ATLAS central services.
- A Frontier Squid RPM, which works for both CVMFS and Frontier access, has been created which will setup a squid with suitable default settings, requiring only minimal configuration. See v2 or v3 instructions.
- A standard Squid (3.x) [can](#) be configured to allow access to CVMFS, and may work for Frontier access (but there are potential issues for both v2 and v3 v2 and v3)
  - ◆ [here](#) are instructions for configuring a standard squid for CVMFS access
- In either case follow the ATLAS-specific deployment instructions at [AtlasComputing/T2SquidDeployment](#)

### Network

- perfSONAR [is](#) also mandatory to understand and monitor Networks.
- Each site should have two sets of perfSONAR services running: latency and bandwidth
- It is possible since perfSONAR v3.4 to run both services on a single node which has at least two suitable NICs (network interface cards). See the link for details about deploying and configuring perfSONAR.

## Recommended CPU, Storage and Network capacity

The latest basic recommendation for CPU and Storage and Network were presented in 2014 at the International Computing Board ( [link](#); the file is attached to this twiki for people without ICB indico access 20140227\_ADCOpsSiteClassification\_rev01.pdf

- The numbers are just indications, they strongly depend on the sites and their configuration, obviously the WAN and LAN should be dimensioned depending on the number of cores.
  - DirectI/O (from WN to local storage) and remote data access also impact the networking infrastructure.
  - As order of magnitude: A minimal Tier2 has 2k HEPSPEC06 and 1000TB of disk space and the international connectivity should be ~ 10 GB/s (updated in march 2020)
  - For LAN, we do have I/O hungry jobs which can require 40 GB of inputs for an 8 core jobs and last 3/6 hours, 10-20MB/s from/to WN - storage would be reasonable (check also above in the WN part the "rule of thumb" for the network bandwidth)
- 
- [Recommendations for sites](#)
  - [Justification on network activity](#)

## Limit concurrent FTS transfers to a destination (Optional)

The following lines can be used to limit the total number of concurrent transfers to a SE. Normally, input files are located in few places so only few concurrent FTS channels are used. But in case of dat rebalancing or PU distribution, the input files can be in > 50 sites which can be used by all FTS transfers. To protect storages with few disk servers, the number of parallel transfers can be limited with the following code. This should not be usefull by default.

## SitesSetupAndConfiguration < AtlasComputing < TWiki

```
from rucio.transfertool.fts3 import FTS3Transfertool

limits = {'srm://tech-se.hep.technion.ac.il': (150, 150)}

fts_hosts = ['https://lcgfts3.gridpp.rl.ac.uk:8446', 'https://fts3-pilot.cern.ch:8446', 'https://']

for fts_host in fts_hosts:
    fts = FTS3Transfertool(fts_host)
    for se in limits:
        fts.set_se_config(se, inbound_max_active=limits[se][0], outbound_max_active=limits[se][1])
        #fts.set_se_config(se, staging=limits[se][0])
```

# Configuration within ADC

- The ATLAS Grid Information System (CRIC)[↗](#) is the service where the ATLAS site services and the ATLAS topology are described.
- Once your site has been setup you need to get in contact with your cloud squad (atlas-adc-cloud-XX at cern.ch ) to make sure that your site, the DDM endpoint (AKA RucioStorageElement RSE), and the PandaResources(AKA PandaQueue) are properly defined in CRIC.
  - ◆ The instructions for this are available at Tier2ProductionSiteConfiguration
- As noted in the perfSONAR instructions you also need to register your perfSONAR installation in either GOCDB or OIM (for OSG sites).



# Documentation of deployment steps

Follow this procedure

# Documentation on operation

## Global overview WLCG

- Vocard: it contains very simplified requirements for sites.  
<http://operations-portal.ege.eu/vo/view/voname/atlas>
- new WLCG Ops portal for sys admins with a lot of reference for sys admins regarding services at sites (CE, SE, Batch system, CVMFS)

## Global overview ATLAS

### Site availability reliability

- SAM tests : Build metric of WLCG site availability/reliability for ATLAS (rely on services) : Global monitoring
- Etf tests : (Another view of SAM tests ?) :Storage example
- Atlas SAM Performace metric : Build metric of ATLAS site reliability/availability (rely on ATLAS jobs and transfers)
  - ◆ Production monitoring
  - ◆ Analysis monitoring

### Site blacklisting

Site status can be found in ATLAS Sam monitoring or CRIC

- Based on site downtime : Switcher
  - ◆ Storage downtime : Site admins are requested to declare downtime for ALL published access protocols at the site. If not all protocols are declared 'stopped', data access will be attempted through allowed protocols.
- Based on site validation with HammerCloud jobs :
  - ◆ Monitoring
  - ◆ HammerCloudTutorialATLASsiteAdmins

## Typical ATLAS jobs

In this twiki ATLASJobs we are trying to summarize some of the typical ATLAS jobs in terms of memory requirements, I/O, etc

## Grid Storage

- All DDM Operations including site admin or squad actions.

# Grid site decommissioning

*In order to minimise the impact on ATLAS users and production, site decommissioning should be organised well in advance and coordinated with ADC. \*Do NOT disable panda queues or DDM endpoints in CRIC without first consulting ADC\**

If the site provides pledged resources, the ICB must be informed, even if the pledge is moving to another site in the same federation.

The decommissioning follow up should be done through a JIRA ticket in ADCINFR project [☞](#) (squad/site responsible).

The CPU could be used until the last moment by accessing data still on SE or remote SE. In contrary, the decommissioning of the storage part should be organised well in advance (up to 3 months) to replicate data somewhere else (if necessary) and clean properly Rucio catalog : it avoids to still host data when the deadline is reached. The main bottleneck for this procedure is the discovery and proper cleaning of lost files.

- Updating Panda queues
  - ◆ if the CPU usage should be stopped : **site admin** declares downtime in GOCB/OIM the services associated to the PQ (CE, squid). A quicker procedure is to ask **cloud squad or ADC Central** to set OFFLINE in CRIC
  - ◆ if the CPU usage should continue **cloud squad or ADC Central** should update the PQ :
    - ◇ to write output in new SE (Update the read\_wan)
    - ◇ to add, when necessary, a read\_lan to access remote SE
    - ◇ to remove, when necessary, the read\_lan to local SE
- For the Grid storage part, follow this procedure
- When previous steps done, stop all other services in CRIC (\* squad responsible or ADC central\*)

# Grid component decommissioning or migration

To stop the usage of a Grid component (CE, SE , Panda queue, DDM endpoint)

- a JIRA ticket in ADCINFR project [☞](#) should be issued to follow up
- a downtime of the service should be declared in GOCDB/OIM
- CRIC [☞](#) has to be updated accordingly (squad or site responsibility) This is done by the local squad or ADC central (first this last case, post a JIRA ticket in ADCINFR project [☞](#)).

## WLCG site : Replacing a SE with another SEs

- Follow Tier2ProductionSiteConfiguration to deploy and populate your new SE with datasets/files for Functionnal Tests
- Procedure to decommission old SE. Please note that current procedure will migrate datasets randomly to available SEs in ATLAS Grid.

## WLCG site : migration to diskless sites

Small WLCG sites are recommended to focus investments in CPUs instead of storage to optimise the ADC support and site manpower. This does not prevent keeping a LOCALGROUPDISK endpoint. To contribute to production, such sites should be setup to run local CPUs and access input files from a remote site and write output to the remote site ('diskless sites'). The technical requirements to pair a diskless site to a remote SE :

- A network connectivity big enough
- A remote SE able to sustain the additional load.

This setup is similar to cloud sites. Another option is to setup an ARC-CE to benefit from its caching mechanism.

To transform small sites with existing SE to diskless ones, the following procedure should be followed :

- The decision to migrate a WLCG site to diskless is taken by the cloud coordination and the country ICB representative. They should also define the remote SE (usually the biggest one in the country). These informations are transferred to ADC coordination to trigger the migration. *To initiate this migration in 2017, the ICB-ADC responsible has contacted ICB representatives to recommend this migration.*
- ADC coordination initiates a JIRA ticket in ADCINFRA, with site representative and associated squad in CC to :
  - ◆ Request somebody at the diskless site to check on a WN that it can read/write a file on the remote Storage Element using protocole at destination (should fail only in case of port not opened outside)
    - ◇ Testing read : `lsetup rucio ; Setup proxy for ATLAS ; rucio download --rse ENDPOINT --nrandom 1 DATASET (DATASET=hc_test.pft or hc_test.aft )`
    - ◇ Testing write : \* Can only be done with Production Role to write on DATADISK\* ;  
`rucio upload ...`
  - ◆ Transform Panda queues to use remote SE (more details) . ATLAS SAM tests on local SE will stopped automatically. This new configuration is tested over few weeks to validate that the remote SE can sustain this additional load. Untill this point it is easy to switch to the local SE again
  - ◆ After the validation period, **DDM Ops and cloud squad** will make the necessary steps to decommission the SE : Procedure
    - ◇ Cleaning LOCALGROUPDISK endpoints : Organised by **cloud squad** (deletion by replica owner or with `atlas//Role=production`)

◇ Remove the SE description from CRIC (**cloud squad**)

When the migration on the ATLAS side is finished, the site admin proceeds with the decommissioning in WLCG/OIM. The recommended steps are :

- It should update the GOCDB to declare the SE no more in production. Then Ops SAM tests will be stopped automatically
- The SE can be physically decommissioned.

# Opportunistic Resources

In this twiki OpportunisticResources

# Blacklisting of permanently broken site

In the ICB meeting on 12th December 2017, the Funding Agency reps have agreed to endorse the following policy for permanently broken sites :

- Broken site over > 1 month with no concrete action although informed through GGUS -> site permanently blacklisted in CRIC (site informed through the pending GGUS ticket which can be closed)
- Each year, permanently blacklisted sites are reviewed and most probably completely decommissioned in CRIC and Rucio (ICB rep + CRIC site contact informed)

Added after ICB :

- If the issue concerns SE, the site PQ can be changed to point to another SE but it requires the agreement of the destination site

# FAQ

Frequently asked questions by sites

---

## Major updates:

-- AleDiGGi - 2015-09-16

Responsible: AleDiGGi

Last reviewed by: **Never reviewed**

- 20140227\_ADCOpsSiteClassification\_rev01.pdf: 20140227\_ADCOpsSiteClassification\_rev01.pdf
  - DISKLESS\_MEMO\_EV\_.pdf: Procedure set Panda queue to access remote SE
- 

This topic: AtlasComputing > SitesSetupAndConfiguration

Topic revision: r61 - 2021-10-20 - DavidCameron



Copyright &© 2008-2022 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.  
or Ideas, requests, problems regarding TWiki? use Discourse or Send feedback