

# Atlas sites Jamboree: Resource requirements on nodes

- ATLAS VO card
  - No (hardware) changes foreseen for Run-2
    - Max used physical non-swap i386 memory size (MB) 2048
    - Max used physical non-swap x86\_64 memory size (MB) 2048
    - Max size of scratch space used by jobs (MB) 20000
    - Max time of job execution (minutes) 3120
    - Job wall clock time limit (minutes) 5760
  - For details, see
    - <http://operations-portal.egi.eu/vo/view/voname/atlas>
  - But – configuration changes are needed
    - e.g. support for multicore, large-memory, long jobs
- Complex ATLAS job requirements
  - Run-2: things will change drastically
    - Mixture of single/multicore jobs
    - Significant fraction of high-memory jobs
    - Mixture of long (simulation) and short (analysis) jobs

# Atlas sites Jamboree: Resource requirements on nodes

- Benefits of ProdSys-2, Rucio
  - Scout job reports will be used to estimate the job resource limits for the rest of the jobs in a task
    - maxrss – maximum physical memory usage
    - maxcputime – maximum total cputime usage (multicore agnostic)
    - ... + any other metrics which might be useful (e.g. network, total I/O...)
  - Tasks will have job resource requirements much better defined
- Dynamic Scheduling and Resource Reservation
  - Passing the job resource requirements to the batch system
    - Part of WLCG Multicore Task Force effort
    - ARC-CE supports it out of the box
    - CREAM-CE supports it through add-on local “blah” extension scripts – Task Force will standardize and prepare default scripts for each batch system in use
    - OSG/Condor needs some more investigation
  - GOAL:
    - Optimize the resource usage
      - e.g. place 1x 6GB RSS reconstruction + 7x 1.4GB RSS simulation on a 16GB 8-core worker node
  - Limit the cputime/walltime for short jobs
    - Give higher priority
    - Use more opportunistic cores reserved other VOs (drains faster)
    - Backfilling the nodes which are draining for large (multicore) jobs
- Dynamic queue model:
  - AGIS: single, base queue with site properties
    - e.g. maxcores 64, maxmemory 256GB, maxwalltime 6 days
  - PanDA/AGIS creates task queues dynamically, as needed
    - e.g. ncores=8, memory=16GB, walltime=6 hours
  - Auto pilot-factory sends the pilots with corresponding resource limits to CE/batch
  - Number of different queues should be kept small

# Atlas sites Jamboree: Resource requirements on nodes

- Memory Requirements
  - Standard WN deployment: 2GB/core
  - Limits are needed
    - 64-bit code not auto-protected by 3.3GB RSS, 4GB VMEM limit
    - Too many high-memory jobs can kill a node
    - Shared sites usually impose strict limits
  - Recommendation:
    - Limit total RSS job usage, limit SWAP
    - RSS limit is only possible with cgroups enabled batch systems (SLURM, condor, UGE?)
      - Sites are encouraged to deploy it
    - VMEM only sites:
      - Jobs will request RSS, not VMEM → VMEM should be scaled by a factor of ~2
    - “ATLAS only” sites:
      - No limits on memory – the job-resource monitor will make sure the resources are not overused
    - Switching to RSS for jobs/tasks will happen early next year
- WN Disk space requirements
  - In Run-2, same as in Run-1
  - But the jobs will change:
    - All the output (and thus input) files will be optimized to fit in a 2-5GB range (possible with JEDI dynamic jobs), to reduce (or get rid of) the number of merge jobs
    - Most of the cpu-intensive jobs will be multi-core
      - Effective reduction of local disk space usage on the node
      - One 8-core job is I/O friendlier than 8 single-core jobs
  - Direct I/O:
    - Recommended for analysis (xrootd)
    - SE infrastructure must scale to a high number of connections and bandwidth
- Local network requirements
  - AtlasDerivation Framework:
    - Most of the common analysis processing will be driven centrally through the train production
    - Less need for CPU intensive user analysis
  - Most of the user analysis is expected to:
    - Process big amount of data
    - Use less cputime
    - A single analysis job on derived datasets can use 40MB/s
  - Local network:
    - 1Gb/s on fat nodes (32-core, 64-core) will be a bottleneck → consider 10Gb/s or 40Gb/s infrastructure
    - SE – WN bandwidth might need to be improved

# Atlas sites Jamboree: Resource requirements on nodes

- Batch System Recommendations
  - Move away from unsupported torque/maui
    - New features like cgroups support likely not to be available
  - Recommended, from ATLAS perspective, i.e. supporting RSS limits
    - HTCondor, SLURM, SGE
  - Move to generic batch queues (one)
    - Most batch systems support universal queues with dynamic job resource reservations
    - Hard partitioning between analysis/production, single/multicore is discouraged → sites will likely suffer inefficient use of resources (empty nodes)
    - Universal queue will be needed for dynamic PanDA queues
    - Deployment needs working parameter passing from CE to the batch

# Atlas sites Jamboree: Multicore

- Current problems
  - Bursty submission
    - Biggest problem makes any nullifies the effects of any optimization.
      - During Run2 will be a different story
  - Job lengths
    - Short jobs don't exploit the slots that took so long to drain
    - Job lengths should increase at least to 10-12h
  - Job inefficiencies
    - Partly due to a bug corrected in a release not used yet
    - Merge portion is SCORE and weighs more on shorter jobs
  - Memory requirements
    - Too large memory requirements means empty slots (next slides)
- Passing parameters to BS
  - Would simplify batch system job in allocating resources
  - Instead of relying on queue parameters which are usually really large
    - Would enable backfilling
    - Would enable jobs to request the memory they need making limits less important
  - Works at some sites but not all

# Atlas sites Jamboree: Multicore

- CEs&BS
  - type of CE's : ARC-CE, CREAM-CE, HTCondor CE (US)
  - 5 (main) batch systems: Torque/Maui, HTCondor, SGE, SLURM, LSF
- Starting from the BS
  - Reduced the number of params to 5
    - Check which parameters correspond to each batch system
    - Check what they do (do they behave in the same way)
    - Match them to whatever string the CE requires from the user after agreeing on a uniform meaning understood by sys admins and users

| Batch system | corecount    | rss           | rss+swap                 | vmem (address space) | cputime   | walltime |
|--------------|--------------|---------------|--------------------------|----------------------|-----------|----------|
| Torque/maui  | ppn          | mem           | -                        | vmem                 | cput      | walltime |
| *GE          | -pe          | s_rss         | -                        | s_vmem               | s_cpu     | s_rt     |
| UGE 8.2.0(*) | -pe          | m_mem_free    | h_vmem                   | s_vmem               | s_cpu     | s_rt     |
| HTCondor(**) | RequestCpus  | RequestMemory | VirtualMemory (in 8.3.1) | No default (Recipe)  | Recipe    | Recipe   |
| SLURM        | ntasks,nodes | mem-per-cpu   | -                        | No option            | No option | time     |
| LSF          | ?            | ?             | ?                        | ?                    | ?         | ?        |

- Proposal
    - GlueCEPolicyMaxCPUTime == maxtime
    - GlueCEPolicyMaxWallClockTime == ncores x maxtime
    - GlueHostMainMemoryRAMSize == maxrss
    - GlueHostMainMemoryVirtualSize == maxrss + maxswap
      - Using == is intentional
      - Currently wallclock > cputime at many (most?) sites
      - maxrss (new parameter), maxswap (new parameter)
        - Would replace current maxmemory which can be ambiguous
        - Brokering can be done using RSS+swap (as now) sites may choose not to set swap and the brokering will be done using RSS
      - Use of Glue1 maybe changed to Glue2
- (\*) ARC-CE has a HTCondor backend with \*Limit parameters which make it simpler

# Atlas sites Jamboree: Multicore

- **Virtual Memory**
  - Many sites limit vmem because they want to limit RSS+swap
    - Kernels have changed years ago and vmem doesn't mean RSS+swap anymore it's the size of the address space
      - SCORE 32bit vmem-RSS+swap was still negligible in first approximation
      - 64bit address space much larger difference will increase
  - Standard tools do not report the memory correctly anymore nor are able to limit RSS+swap
    - Processes may look like they are using 40GB of vmem but if one looks at RSS+swap with other tools the same processes don't go above 20GB
      - Swap for multicore jobs is negligible
    - ulimit used to be able to distinguish for example it could limit RLIMIT\_RSS now it limits only RLIMIT\_AS which affects all memory allocation and mapping functions
- **Memory multicore case**
  - To the previous slide we need to add that multicore (v)memory is wrong by default because the shared memory is accounted multiple times.
    - Even without counting the experiments asking for more to cover the 5 minutes peaks
  - Some sites limiting the (v)memory had to increase the limit
    - Problem when limit = allocation of resources
  - Some sites are oversubscribing the memory by a factor
    - Useful particularly for multicore when most of the time the memory is not used.
    - Recipes for maui and HTcondor exist
- **Memory and cgroups**
- Some sites are enabling cgroups.
  - Allows more accurate monitoring (see Gang presentation)
  - Allows smart soft limit without allocating memory
    - If jobs exceed this the kernel tries to recover unused memory from the cgroup before killing jobs
  - Allows hard limit jobs just get killed

# Atlas sites Jamboree: Multicore

- **cgroups and BS**

- Can it work everywhere?
  - Really easy to enable in Htcondor
  - Supported in SLURM
  - UGE has been patched
  - SoGE/OGE no support
  - torque/maui no support
    - At last count still 100 sites
- Sites moving away from torque should look into it
  - HTCondor recipe really easy
  - SLURM probably easy too

| Batch system | rss | rss+swap | vmem      | needs cgroups |
|--------------|-----|----------|-----------|---------------|
| Torque/maui  | -   | -        | RLIMIT_AS | N/A           |
| *GE          | -   | -        | RLIMIT_AS | N/A           |
| UGE 8.2.0    | yes | yes      | RLIMIT_AS | yes           |
| HTCondor     | yes | in 8.3.1 | -         | yes           |
| SLURM        | yes | -        | -         | yes           |
| LSF          | -   | -        | -         | -             |

- **Smaps**

- smaps reports things correctly
  - Not used by standard tools
  - Not used by batch systems either
- CentOS has a ps\_mem tool
- Atlas is working on code to put in the pilot to monitor and limit jobs using smaps



# Atlas sites Jamboree: ARC CE

- Fully functioned Grid Compute Element part of UMD
  - BDII, APEL, GOCDB, SAM
- Unique functionality
  - data input and output done outside batch job
- Installation straightforward
  - SL, Debian, SUSE(builds ok)
- single config file
- major batch systems supported
  - PBS, SGE, Slurm, HTCondor, Loadleveler, LSF, Boinc
  - bash scripts for submit,stat,cancel
  - perl for infosys
- Operation
  - no scale limit observed yet:
  - debug is easier, e.g. log file per job
- Use cases for ARC CE
  - Traditional Computer centers/HPC
- No local SE
  - data handling by CE, outside batch job
  - maybe suitable for Clouds – asynchronous data movement
- ssh backend to ARC CE
  - no service on site, just passwordless ssh
    - sshfs, batch commands via ssh – works well.
- Replace CREAM CE for pilot submission
  - Use to submit normal ATLAS pilot jobs
  - no data staging required(goes via SRM)
- Why do some need an alternative?
  - Rod Walker : “Cream CE is unreliable, difficult to configure/debug/upgrade, not scaling well, high RAM usage”
  - not alone: read lcg-rollout list
- Flexibility of choosing custom job limit
  - recall surprise RAM usage of pile MCore – no manual action required

# Atlas sites Jamboree: ARC Control Tower

- Developed so that Nordugrid sites could integrate to Panda
  - pilot jobs required WN network – not allowed
    - cvmfs, Frontier work! Selected IPs is allowed.
    - want known job with custom limits
  - WN pulls data from SE – no local SE on NG
    - must know inputs and pre-fetch
  - aCT spoofs WN pilots
    - pulls job from Panda, sends heartbeats, final update
- The payload
  - In between preloaded pilot is submitted to ARC CE
    - input data known
    - resource limits, RSS+wall, known and set
  - Lose late binding
    - instead must set BS priority for eg. merge job
    - modifying priorities possible(with dev) but maybe not necessary
- Dynamic queues vs aCT
  - Range of limits(mem,wall,cores) in Panda queues(static or chosen by JEDI)
    - send pilots to each queue with activated jobs
  - Late binding messed up here too
    - cannot choose which pilot starts next
    - priorities between jobs in different queues NOT respected
  - aCT sets priority per job
    - Panda chooses next job to run – better
- aCT and ARC CE crucial for certain use-cases
  - HPC, strict CC, Boinc, no local SE, Cloud(?)
- Useful for ATLAS production
  - fine-grained & flexible job limits via aCT
    - aCT currently only supports ARC CE
    - no strong case/manpower to support preloaded pilots to Cream
    - no more empty pilots!
  - resource limit setting for dynamic queues

# Atlas sites Jamboree: Job-Property-Based Scheduling at Sites

- Sites maintained correspondence between PanDA queues, Globus RSL queue and internal Site Batch queues
  - With increasing number of functionally specialized PanDA queues (e.g. MCORE, LMEM, SHORT) this may no longer be desirable
    - Directly effects how pilot submission is handled
      - APF is in the middle between the grid view of the site, and the state of the WMS queues
- could be improved by eliminating grid-level queues and always submit Jobs with their requirements expressed as standard attributes (e.g. cpucount, maxwalltime, maxmemory)
  - E.g. this would allow us to get rid of “PROD” vs. “ANALY”
  - If the job properties are ~the same we could let PanDA broker both types to the same PanDA Pilot
- Work can be submitted to the Site’s Default Queue
- APF/PanDA Interactions
  - APF must be able to query PanDA for ready/activated Jobs within an equivalence class
    - Each Pilot guaranteed to have a Job for dispatch to it
    - Any system where equivalence classes can change, are inconsistent, or overlapping, is unworkable
- PROD vs. ANALY
  - Goal is to dispense PROD vs. ANALY concept
  - ANALY vs. PROD used as a proxy for I/O intensity
    - Better would be to let Scout calculate walltime and cputime and calculate I/O intensity factor
- Site Processing
  - Proposed concept works with popular Batch Systems
  - E.g. HTCondorCE job router provides hooks for scripts to transform incoming Class Ad to local Batch ClassAd

# Atlas sites Jamboree: Storage Protocol overview

- SRM
  - In use currently for storage management (e.g. data deletion, free/used space)
  - Possibility to use WebDAV for deletion
  - Possibility to use some ad hoc tool/whatever for free space
  - In use for Tape Management (staging)
    - No reasonable alternative available.
- gridFTP
  - Is needed for 3rd party transfers
    - Currently : SRM+gridFTP (or directly gridFTP)
  - Xrootd/webDav are under testing/commissioning xrootd
- Xrootd
  - Needed for direct I/O. Possibly download and upload.
- WebDav
  - can be used for all the rest (download/upload/deletions)
  - but till matures to production we need other protocols.
- Actions
  1. Short Term : Eliminate the need of space tokens (rely on paths) for accounting.
  2. Short Term: Move to gridFTP only for 3rd party transfer (requires 1)
  3. Short Term: Move to xrootd/http for uploads and downloads (requires 1)
  4. Short/Medium Term: commission http/WebDAV to production quality for deletions
  5. Medium Term: Decommission SRM from non tape sites (**requires all the above**)
  6. Short Term: Move to xrootd (or file) all the directIO
  7. Short Term: Decommission other directIO protocols (**requires 6**)
  8. Medium/Long Term: Commission xrootd and WebDAV for 3rd party transfers
  9. Medium/Long Term: Decommission gridFTP (**requires 8**)
  10. Long Term: Consolidate Davix and evaluate Davix and xrootd for directIO
  11. Long Term: Keep both webDAV and xrootd or decommission one
- The new framework (Rucio and ProdSys2) will (can) allow Atlas to differentiate the protocol per activity per site:
  - sites can configure themselves and help us in testing, integrating and commissioning "new" protocols.
- Atlas needs help from the community in taking onboard initiative of testing other protocols and to implement the missing functionalities.

# Atlas sites Jamboree: FTS deployment model

- FTS == FTS3.
- 3 FTS instances:
  - CERN, RAL and BNL
- Rucio, in case of submissions problems, automatically failover on the others servers. Present config:
  - US,CA,DE,IT (312 DDMEndpoints):
    - FTS3: 1) BNL, 2) CERN, 3) RAL
  - CERN,ES,NL,ND,RU,TW (178 DDMEndpoints):
    - FTS3: 1) CERN, 2) RAL, 3) BNL
  - UK,FR (163 DDMEndpoints):
    - FTS3: 1) RAL, 2) BNL, 3) CERN
- Main DDM ATLAS monitor
  - <http://dashb-atlas-ddm.cern.ch/ddm2>
  - failed transfers have links to the FTS3 monitoring with log files to debug errors
- FTS dashboard
  - <http://dashb-Ftstransfers.cern.ch/ui/>
  - All the experiments
- FTS3 Server monitor:
  - More details about each instance:
    - <https://fts3.cern.ch:8449>
    - <https://fts.usatlas.bnl.gov:8449>
    - <https://lcfsts3.gridpp.rl.ac.uk:8449>
- FTS3 users guide:
  - <http://fts3-service.web.cern.ch/>

# Atlas sites Jamboree: Network Monitoring via Sonar Tests

- Important links
  - Site Status Board (SSB):
    - <http://dashb-atlas-ssb.cern.ch/dashboard/request.py/siteviewhome>
  - Sonar twiki:
    - <http://twiki.cern.ch/twiki/bin/viewauth/AtlasComputing/DDMSonarTests>
  - Brief description of DDM Sonar:
    - <http://iopscience.iop.org/1742-6596/396/3/032019>
  - Network monitoring directly from File Transfer Service (FTS):
    - <http://egg.bu.edu/atlas/studies%7Btype>
- T2D concept
  - T2D was a useful concept when we had cloud boundaries. Atlas is breaking the cloud boundaries now, since almost all of T2s are T2D.
  - Why did atlas need T2D up to now:
    - direct transfers between T2Ds
    - high analysis availability (A), i.e they can get more data
    - funding, motivation for T2 to be better
  - Motivation for classifying T2s in the future:
    - funding
    - warn T2s with critical connectivity
    - prepare a network mesh for Rucio to allow direct transfers

# Atlas sites Jamboree: Perfsonar

- perfSONAR has been deployed to monitor the network
  - Info on the working group is at:  
<https://twiki.cern.ch/twiki/bin/view/LCG/NetworkTransferMetrics>
  - perfSONAR install details at:  
<https://twiki.opensciencegrid.org/bin/view/Documentation/DeployperfSONAR>
  - Primary challenges: newest version 3.4.1 installed, properly configured, firewalls not blocking operation
  - <https://twiki.opensciencegrid.org/bin/view/Documentation/DeployperfSONAR>
  - [http://grid-monitoring.cern.ch/perfsonar\\_coverage.txt](http://grid-monitoring.cern.ch/perfsonar_coverage.txt)
  - **deadline 8<sup>th</sup> January (we start ticketing sites after...)**
- Via perfSONAR we gather a number of metrics:
  - Topology/path-information via traceroute
  - One-way delay via OWAMP
  - Packet-loss via OWAMP
  - Usable bandwidth via BWCTL
- ESnet has some nice pages on using perfSONAR to identify problems
  - <http://fasterdata.es.net/performance-testing/evaluating-network-performance/>
- Some specific examples: <https://indico.cern.ch/event/354593/>

# Atlas sites Jamboree: FAX and Panda WAN usage

- FAX infrastructure
  - Sites should pay more attention to status of their FAX endpoints (automatic or personal mails): multiple days without a service are hurting the system
- Monitoring situation
  - <http://dashb-xrootd-comp.cern.ch/cosmic/ATLASmigrationMonitoring/>
  - [http://dashb-ai-621.cern.ch/cosmic/DB\\_ML\\_Comparator/](http://dashb-ai-621.cern.ch/cosmic/DB_ML_Comparator/)
  - <http://waniotest.appspot.com/>
- Panda WAN modes
  - Failover
    - In the case stage-in fails due to a temporary SE related problem, the pilot will re-attempt the stage-in a second time after a few minutes. If that fails as well, the pilot has the option to attempt stage-in from a remote SE using FAX.
  - Overflow
    - When deciding where to broker a task, JEDI can estimate that it is better to send it to a site that does not have the input data and let it read from FAX, rather than let it sit in the queue of the site that has the data. Limited to analysis queues.
  - Explicit overflow
    - If a user explicitly requires CE that does not have the input data, the task will be brokered to that CE and FAX used to get the data
- Site controls
  - Failover
    - Setup per Panda queue using two AGIS fields:
    - allowfax=True will enable FAX retries.
    - Faxredirector sets the FAX access point to be used. For optimal performance it should be set to the site's closest redirector
  - Overflow
    - Other queue settings:
  - wansinklimit\*\*- limits the bandwidth that jobs overflowed to the site can use.
  - wansourcelimit\*\*- limits the bandwidth that site's FAX endpoint can deliver to jobs overflowed elsewhere.
- Overflow - technical details
  - JEDI brokering
    - Chooses a possible alternative queue based on weights defined by the "busyness" of the queue, possible input dataset replicas, and the network cost (as determined by the "cost matrix" discussed yesterday)
  - FAX endpoint used
    - JEDI sets a variable "source site" for each job, but for a technical reason the actual FAX access endpoint used comes from the Fax redirector value set for the destination queue (uses FAX redirection)
- Enabling Overflow
  - There is a need for it - according to JEDI brokering
  - Expect to see issues as new queues are added; also queue-level tuning is needed (initially)
  - At the scale of 500-2000 finished jobs per hour not a serious load on the infrastructure.
  - Brokering decisions could be analyzed to determine optimal tuning (analytics).
  - CPU efficiency will improve with the adoption of the new ROOT release and switch to xAOD



# Atlas sites Jamboree: Site consistency tools

- CCC - complete consistency checker
  - Developed at MWT2

`ccc_pnfs_rucio.py` will download the list of datasets listed in Rucio for a site, compute Rucio paths for each file in the datasets, and check that against what is in PNFS. It will additionally check PNFS against disk. A report is generated in the output directory with a summary of discrepancies found.

- Available in Github:
- <https://github.com/DHTC-Tools/ATLAS/tree/master/Complete%20Consistency%20Check>

## Report - flags:

---

**GHOST:** A file that is present in a higher level storage catalog, but not in a lower one or on disk. Ghost files often cause job failures when a job runs at the site and fails to fetch the missing input file.

**ORPHAN:** A file that is missing in the higher level storage catalog, but is present in the lower one or on disk. Orphans do not cause job failures. They are typically 'dark data', and may need to be removed manually.

**DAMAGED DATASET:** A dataset that is listed as having a complete replica at the site, but lower level catalogs or disk show missing files. Damaged datasets may cause job failures, for the same reason as ghost files. **MISSING DATASET:** A dataset that is listed as having a complete replica at the site, but lower level catalogs or disk show all files are missing. Missing datasets may cause job failures, for the same reason as ghost files.

**INCOMPLETE DATASET:** A dataset is listed as having an incomplete replica at the site, and the lower level catalog and file system confirm that. These are not a problem.

**EMPTY DATASET:** A dataset listed as having 0 files in the catalog. These are not a problem.

**UNKNOWN DATASET:** The catalog lists the dataset as present at the site, but when the catalog was queried for a list of files it returned an 'unknown dataset' error. If a dataset appears in this category repeatedly it may be broken in the catalog and need attention by the catalog admins.

**OK DATASET:** A dataset that is listed as having a complete replica at the site, and the lower level catalogs and file system show all files present. Most datasets should be in this category.

# Atlas sites Jamboree: LOCALGROUPDISK management tools

- US cloud:
- Policy 1: Management tool by “disk usage of User”
  - A warning message will be sent automatically if usage >3TB per site per user.
  - Warned owners have to reduce their disk space (< 3TB).
  - If not possible < 3TB, fill the web form providing information (data, group, size, period) to be needed.
  - Usage >3TB , keep the information about the data(lifetime).
  - Usage >20TB per user per site requires an approval from management
  - In case of no answer to 3 warning messages (1 week interval), will be discussed with management
- Policy 2: Management tool by “dataset”
  - Popularity :
    - Contact to owner with dataset not accessed at least one year.
    - If the data is needed, owner is required to fill out the web form.
  - Multiple dataset :
    - If dataset has >7 replicas over the grid, contact to owner verifying if all those replicas are essential, or number of copies can be reduced.
- Monitoring tool
  - written in “python”.
  - The “web monitor” are built in the web application frame work “Django”.
  - The “web monitor” accesses the database (ddm05.usatlas.bnl.gov) which is updated with “python” files.
  - To update the database, compare dataset by dataset and file by file between central catalog from RUCIO and the database.
  - e.g. If missing dataset/file exists in the monitor (central catalog), put into (delete from) the database.
- To update for all LOCALGROUPDISK in US, it takes ~ 2days.

# Atlas sites Jamboree: LOCALGROUPDISK monitoring

- DE cloud:
  - Project to be found at <https://pilot.pleiades.uni-wuppertal.de>
  - based on Django (<https://www.djangoproject.com/>)
  - Updated every day
  - dq2 command line requests too slow for all DE sites → atlas provides daily database dumps
  - Privacy protection:
  - authentication by grid certificate, access to monitoring page only for ATLAS DE users