Empowering Grids – the EGEE gLite middleware

Ludek Matyska

CESNET and Masaryk University
Czech Republic
Disclaimer

• This presentation is based on contribution from many gLite developers
• It uses pictures, numbers and sometime even whole slides from many other EGEE related presentations given at different fora
• Even if not explicitly referenced, all these information sources are highly appreciated

• Thanks to the whole JRA1 team
• **Pre-history**
  – DataGrid, focused on the initial middleware development (EDG)
  – 3 years, from 2001 to March 2004

• **EGEE**
  – Production oriented, based on middleware development in DataGrid, EDG, LCG and initial gLite middleware
  – 2 years, April 2004 to March 2006
  – 71 partners, 27 countries, operation federated (ROCs)

• **EGEE II**
  – Full scale deployment, the gLite middleware
  – 2 years, April 2006 to March 2008
  – 91 partners, 32 countries, 13 Federations
**EGEE Future**

- **EGEE III**
  - Just to be submitted (September 20\(^{th}\))
  - 94 partners, 34 countries, 12 federations
  - Real production (LHC deployment in 2008)
  - Strong support for other applications
    - Computational Chemistry
    - Astrophysics
    - Bioinformatics and medicine
    - Earth Sciences
    - *(Grid Observatory)*
  - Continued middleware development and support

- **EGI (European Grid Initiative)**
  - Post EGEE future
  - Design Study project (Started September 1\(^{st}\))
• **Large-scale production quality e-infrastructure**
  – HEP the main user
  – But other communities actively looked for and supported

• **High-throughput production environment**
  – Emphasis on large number of CPUs, sites, and independently submitted and run jobs
  – Goals: Tens to hundreds thousands jobs per day on the whole infrastructure

• **Data intensive (data Grid)**
  – Able to process PB of data
  – Data catalogues, access methods, …
  – Low, medium and high security requirements
Scale of EGEE Service

No. CPU

No. Sites

No. jobs / month - all

98k jobs/day
EGEE Middleware

- **Brand name**: gLite
- **Production quality**
  - Novelty less important
  - Must pass the real-use test
- **Testing and Integration**
  - Independent activity
  - Stay between development and operations
- **Foundation Services**
- **Higher Level Grid services**
• **Security infrastructure**
• **Information system, monitoring and accounting**
  – Information schema, simple resource discovery
  – Resource monitoring and notification interfaces
  – Accounting to provide appropriate aggregation and views
• **Compute Element (CE)**
  – Set of services to provide homogeneous secure access to heterogeneous computing resources
• **Storage Element (SE)**
  – Set of services to provide access to storage resources
  – SRM Interface
  – POSIX like I/O
Higher level Grid services

• **Job services**
  – Workload Management System (WMS)
    ▪ Resource brokerage
    ▪ Job Input and Output handling
    ▪ Automatic resubmission and persistence
    ▪ Job tracking – Logging and Bookkeeping service
    ▪ Permanent job information – Job Provenance service

• **Data management services**
  – Reliable asynchronous file transfer system
  – File and replica catalogues
  – Secure data management
  – Data encryption
• **EDG middleware**
  - DataGrid project
  - Maintained by the LHC Computing Grid – LCG middleware
  - LCG releases up to 2.7 (2005)

• **gLite middleware**
  - EGEE projects
  - Overlap with the LCG, but independent up to version 1.5 (2005)

• **gLite middleware 3.0**
  - Merge of gLite 1.5 and LCG 2.7 (2006)
  - Production release in EGEE project

• **gLite 3.1**
  - Increased stability and throughput, released
gLite services

- **Security**
  - Authentication
  - Authorization
  - Accounting

- **Computing Element**
- **Storage Element**
- **Information and Monitoring**
- **Workload Management**
  - Brokerage
  - Logging and Bookkeeping and Job Provenance

- **Data Management**
  - File transfers, Catalogues, Replicas
• **Authentication**
  - PKI with X.509 certificates providing single sign-on
  - Maintained list of trusted CA (EUGridPMA, IGTF)
  - Use of short term proxy credentials (lower risk)
    - Proxy delegation, MyProxy,

• **Authorization**
  - Virtual Organizations (VO)
    - User must be member of at least one VO
  - Resources are “assigned” to VOs (negotiation, includes priorities, access policies, etc.)
  - VOMS (VO Management Service)
    - Attribute certificates, capability based authorization
      - “Attached” to proxy certificate
    - Authorization information stored in VOMS servers
Long lived certificates may be replaced by short lived certificates provided by a Shibboleth identity Provider.

Phase 1: Shibboleth enabled SLCS

Phase 2: Attribute transfer into VOMS

SLCS

Shibboleth IdP

VOMS

MyProxy

CE, SE

debelates

pX.509

pX.509 w/ VOMS attributes

Voms_proxy_init (DN)

authZ

authN

Config Files gridmapfiles blacklist
File System ACL LCAS LCMAPS

PDP #n
PDP #2
PDP #1

submits jobs
• **Abstraction of a computational resource**
  – Common set of interfaces/services for heterogeneous resources

• **Cluster a typical CE**
  – Head node
  – Several worker nodes (WN)
  – Single (local) batch system to dispatch jobs among WNs

• **Different realizations (interfaces)**
  – LCG-CE
  – gLite-CE
  – CREAM
• **LCG-CE**
  - Globus Toolkit version 2 GRAM service
  - Never ported to GT4
  - Deprecated

• **gLite-CE**
  - GSI-enabled Condor-C
  - Still needs some tuning
  - Phased out

• **CREAM**
  - WS-I interface (OGF-BES)
  - BLAH (Batch Local Ascii Helper) connector
    - Job management operations
    - Job status changes
Workload management system

• **Resource brokering**
  – Matchmaking: user requirements vs. grid state
  – CE selection

• **Workflow management**
  – Compound jobs

• **I/O management**
  – Takes into consideration also data resources

• **Additional features**
  – Persistency
    ▪ Deep and shallow resubmission
    ▪ Recovery from WMS crashes
  – Security
    ▪ Proxy renewal
Supported job types

- “Normal” (batch like)
- DAG workflow
- Collection
- Parametric
- MPI
- Interactive

- Deprecated
  - Checkpointable
  - Partitionable
Real time job tracking

- **Logging and Bookkeeping Service**
  - Keep track of Grid jobs across components
    - Reliable and secure collection of events (non-blocking)
    - Multiple event sources (redundancy)
  - Capture job control flow
  - Provide job state information
    - Job state updated on new event arrival
  - Support user generated events
  - Secure
    - Mutual authentication of all components
    - Encrypted data transmission
    - VOMS based authorization
  - All data collected on LB server
    - Multiple instances (one job – one LB server)
Job Provenance

• **Long term preservation of information about Grid jobs**
  – Information on job control flow and execution environment complements actual job results
  – Based on data from LB, extended by input and sandbox, small output files, additional user annotations

• **Architecture optimized for storage AND retrieval**
  – JP Primary Server
    ▪ One for several VO
    ▪ Huge amount of raw data
    ▪ Fast write
  – JP Index Servers
    ▪ Many instances per JP PS (registration, support for >1 PS)
    ▪ Provide “views” on data
  – Support for data-mining

• **Assist job re-submission**
Accounting

- **Collection of data on resource usage**
  - By VO, group or a single user
- **Metering sensors on all resources**
- **Pricing – cost of use of resources**
  - If enabled, market-based resource brokering
- **High privacy**
  - Access to data granted to authorized personnel
  - Information collected in GOC (Grid Operation Centre)
- **Functionality provided by APEL**
  - Uses R-GMA to propagate job accounting information for infrastructure monitoring
- **Full support via DGAS**
  - Complex architecture (site and central databases)
  - Used by INFN, gLite certification pending
Abstraction of file storage

**Interface: SRM (Storage Resource Management)**
- Current version 2.2

**Handles authorization**

**Various implementations**
- Disk based: DPM, dCache
- Tape based: Castor, dCache

**POSIX like I/O (rfio)**
- GFAL (Grid File Access Layer)
Disk Pool Manager (DPM)

• Manages storage on disk servers
• SRM support
  – 1.1
  – 2.1 (for backward compatibility)
  – 2.2 (released in DPM version 1.6.3)
• GSI security
• ACLs
• VOMS support
• Targets small to medium sites
  – Single disks or several disk servers
• **LCG File catalogue**
• **Stores mapping between**
  - Users’ file names
  - File locations on the Grid
• **Provides**
  - Hierarchical Namespace
  - GSI security
  - Permissions and ownership
  - ACLs (based on VOMS)
  - Virtual ids
    ▪ Each user is mapped to (uid, gid)
  - VOMS support
    ▪ To each VOMS group/role corresponds a virtual gid
File Transfer Service (FTS)

- **Reliable data movement fabric service**
  - Performs bulk file transfers between multiple sites
  - Transfers are made between any SRM-compliant storage elements (both SRM 1.1 and 2.2 supported)

- **It is a multi-VO service**
  - Balance usage of site resources according to the SLAs agreed between a site and the VOs it supports

- **VOMS aware**

- **Secure**
  - All data is transferred securely using delegated credentials with SRM / gridFTP
  - Service audits all user / admin operations

- **Deployment**
  - Tier 0 at CERN (target 1GB/s 24/7 service)
  - Among ~10 Tier 1 centers and also Tier 1 – Tier 2 transfers
• Request from medical community
• Strong security requirements
  – anonymity (patient data is separate)
  – fine grained access control (only selected individuals)
  – privacy (even storage administrator cannot read data)
• Solution based on many components:
  – image ID is located by AMGA (metadata management)
  – key is retrieved from the Hydra key servers
  – file is accessed by SRM (access control in DPM)
  – data is read and decrypted block-by-block in memory only (GFAL and hydra-cli)
Some statistics

- Stress tests performed by the HEP experiments
  - ATLAS and CMS
- **gLite 3 with “standard” testing and certification procedure**
  - Results not satisfactory for end users
- **gLite 3.1**
  - Closed loop between developers and users
  - Intensive work on started in 2007
  - Visible improvements
## Requirements for the gLite WMS

### CMS | ATLAS
---|---
**Performance**
2007 | 50K jobs/day | 20K production jobs/day + analysis load
2008 | 200K jobs/day (120K to EGEE, 80K to OSG) Using <10 WMS entry points | 100K jobs/day through the WMS; Using <10 WMS entry points
**Stability**

<1 restart of WMS or LB every month under load
Based on the experiment requirements, some criteria have been defined to decide if the gLite WMS satisfies the requirements:

- At least 10000 jobs/day submitted for at least five days
- No service restart required for any WMS component
- The WMS performance should not show any degradation during this period
- The number of zombie jobs should be less than 0.5% of the total
Results of the acceptance test

- **115000 jobs submitted in 7 days**
  - ~16000 jobs/day well exceeding acceptance criteria
  - The "limiter" prevented submission when load was very high (>12)
- **All jobs were processed normally but for 320**
  - ~0.3% of jobs with problems, well below the required threshold
  - Recoverable using a proper command by the user

- The WMS dispatched jobs to computing elements with no noticeable delay
- Acceptance tests were passed

No stale jobs
Number of Jobs Error Breakdown: January to August 2007

- gLite WMS: ~22%
- Data Management: 36%
- ATLAS SW: 8%
Number of Jobs Error Breakdown: July and August 2007

- gLite WMS: ~13%
- Data Management: 47%
- ATLAS SW: 11%

gLite WMS category includes also site specific issues and problematic job distribution (with subsequent proxy expiration).
WallClockTime Error Breakdown: January to August 2007

ATLAS SW: 28%

Data Management: ~60%

gLite WMS: negligible
CMS supports submission of analysis jobs via WMS

- Using two WMS instances at CERN with the latest certified release
- For CSA07 the goal is to submit at least 50000 jobs/day via WMS
- The Job Robot (a load generator simulating analysis jobs) is successfully submitting more than 20000 jobs/day to two WMS
Summary

- **gLite middleware reached production quality**
  - Large scale deployment on an EGEE Grid
  - Hundreds of sites, tens thousands jobs every day
    - Scalability limits much higher
    - Multiple deployment of key services possible
  - File transfers at PB level already achieved (over half a year)
- **On-going performance tuning**
  - Closer collaboration between users and developers beneficial to fast development of high performing components
    - Experimental services approach
- **On-going reliability improvements**
- **Ready for use – new users welcome**