
Introduction to gLite

Mehdi Sheikhalishahi
Grid Computing Group
IPM
4 Dec. 2008



Outline

- Grid Middleware Toolkits
- Why gLite?
- gLite Middleware Services

Grid Middleware Toolkits

Name	Held By	Sponsors
Globus Toolkit (1998)	University of Chicago	DoE, EPSRC, NSF
ProActive (2002)	Inria Sophia Antipolis	INRIA
UNICORE (1997)	Juelich Supercomputing Centre	German ministry for education and research
gLite (2004)	Europe	WLCG, EGEE
Alchemi (2004)	Gridbus	eWater, The University of Melbourne, ARC, Microsoft

Why gLite?

- Globus Toolkit is the de facto standard in the Grid
 - But it is for developers
 - It takes a lot of time for building a Grid Infrastructure
 - Just offers the main services of Grid (not LRM and Broker)
 - Should be integrated with other Technologies like LRMs
 - Needs a lot of knowledge of its components and other technologies

Why gLite?

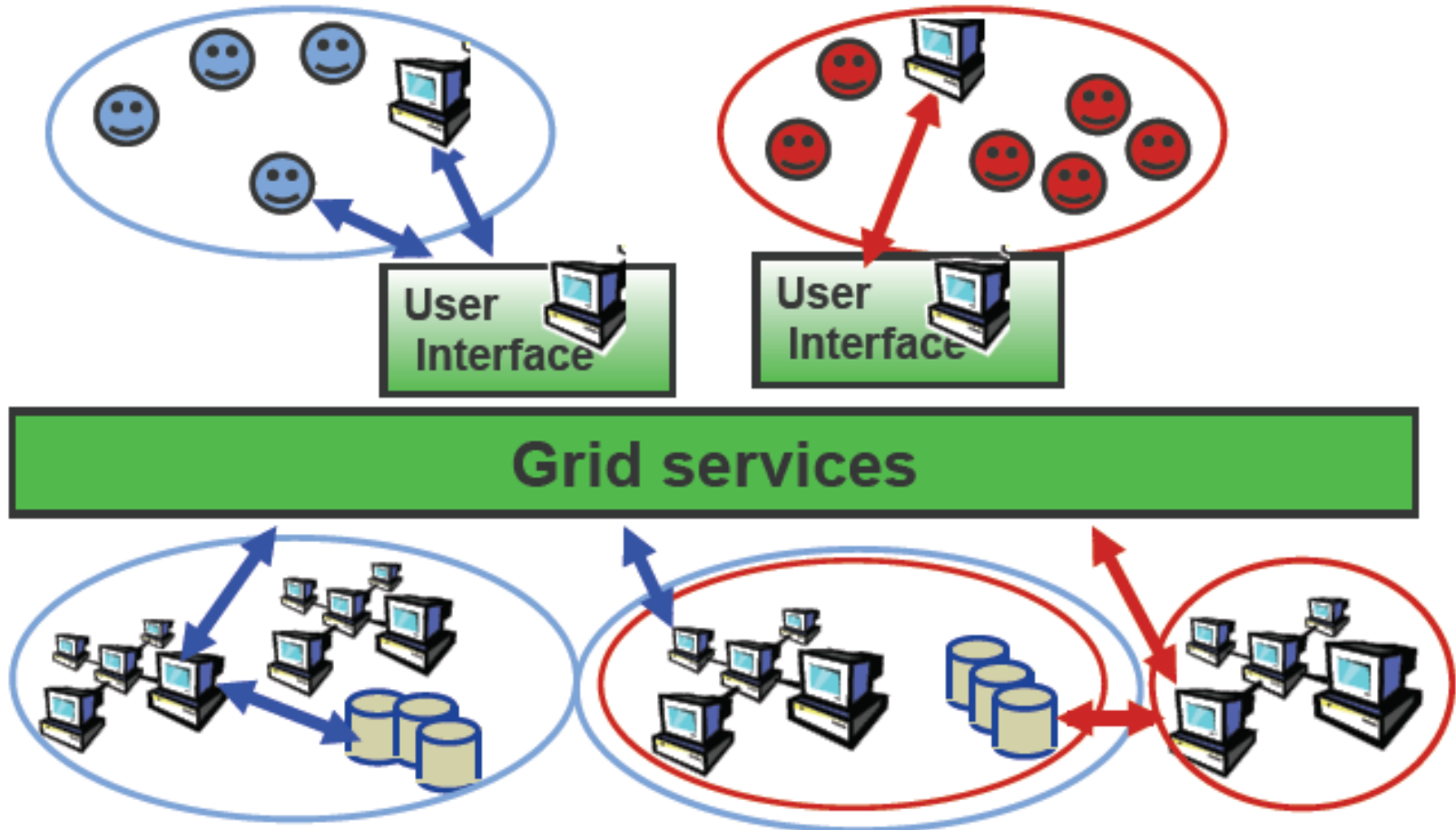
- It contains the most important technologies related to the Grid like Globus Toolkit
- A complete Grid Middleware
- Data/Computation Grid



gLite Middleware Services

- Security
- Execution Management
- Information Service
- Data Management

Users view of the Grid



User Interface

- The access point to the Grid
- Keeps user accounts and user's certificates
- User can be authenticated and authorized to use the Grid resources
- It is the responsibility of the user to describe his jobs and their requirements, and to retrieve the output when the jobs are finished

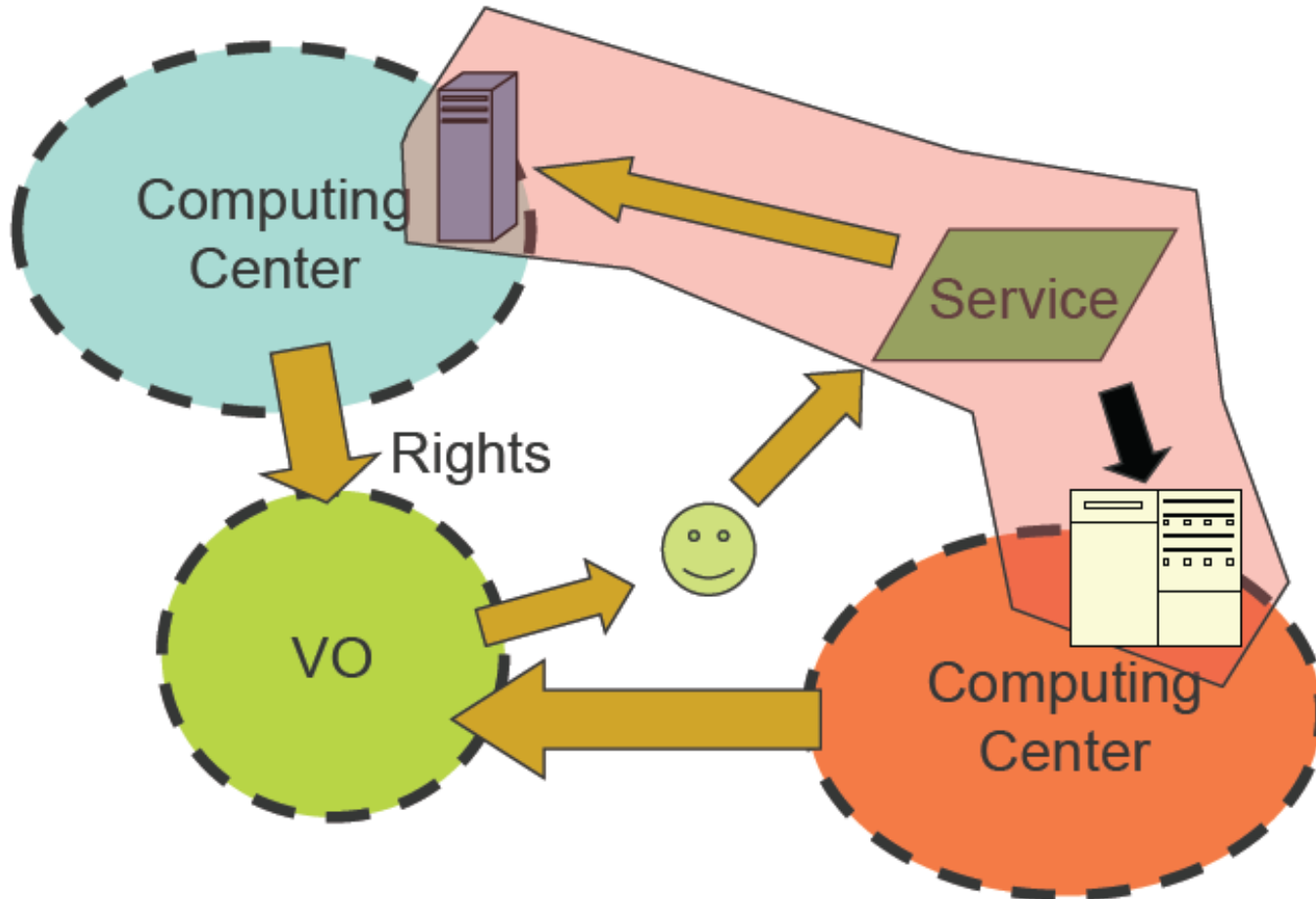
UI

- Job Management
 - Submit jobs for execution
 - Cancel jobs
 - Retrieve the output of finished jobs
 - Show the status of submitted jobs
 - Retrieve the logging and bookkeeping information of jobs
- Information
 - List all the resources suitable to execute a given job
 - Retrieve the status of different resources from the Information System
- Data Management
 - Copy, replicate and delete files from the Grid

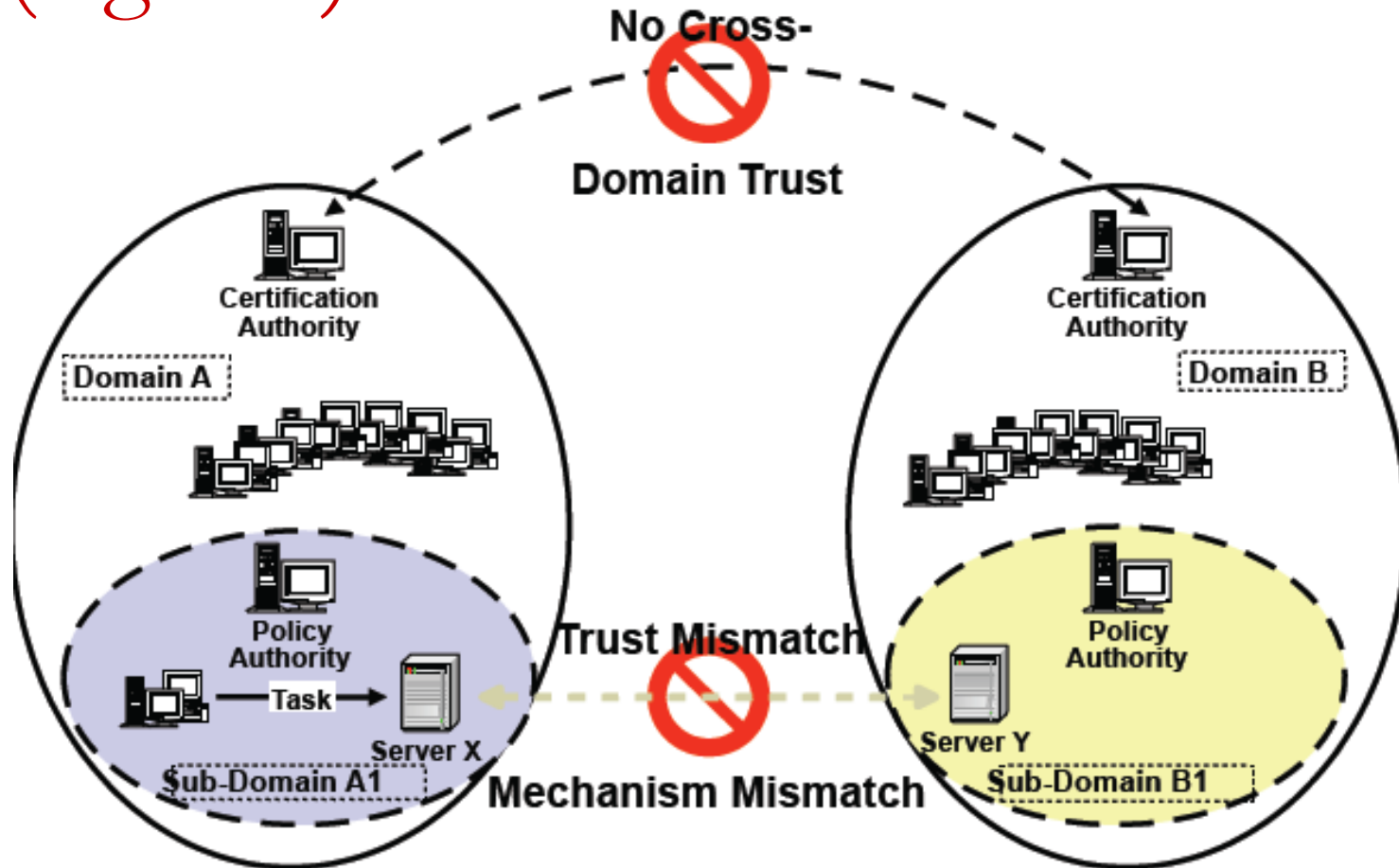
Internet vs. Grid

- Internet: Client/Server
 - DNS, Router
 - Access to Google.com
- Grid: Delegation, Single Sign-on
 - Are not just client/server
 - But service-to-service on behalf of the user
 - Requires delegation of rights by user to service
 - Resource Broker
 - Services may be dynamically instantiated

Delegation for dynamic distributed system



Security: Multi-institution issues (IcG-CA)



Grid Security

- Resources being used may be valuable & the problems being solved sensitive
 - Both users and resources need to be careful
- Dynamic formation and management of user groups
 - Large, dynamic, unpredictable...
- Resources and users are often located in distinct administrative domains
 - Can't assume cross-organizational trust agreements
 - Different mechanisms & credentials

Virtual Organization

- Virtual organizations (VOs) are groups of Grid users (authenticated through digital certificates)
- VO Management Service (VOMS)
 - serves as a central repository for user authorization information
 - providing support for sorting users into a general group hierarchy
 - keeping track of their roles, etc.
- VO Manager, according to VO policies and rules, authorizes authenticated users to become VO members

Grids and VOs

- Resource centers (RCs) may support one or more VOs, and this is how users are authorized to use computing, storage and other Grid resources
- VOMS allows flexible approach to A&A on the Grid

Logging to the Grid

- To run programs, authenticate to Grid:
 - `voms-proxy-init -voms VONAME`
 - Enter PEM pass phrase: `*****`
- Creates a temporary, local, short-lived proxy credential for use by our computations
- Delegation = remote creation of a (second level) proxy credential, which allows remote process to authenticate on behalf of the user



gLite Middleware Services

- Security
- Execution Management
- Information Service
- Data Management

Workload Management Service

- Makes running jobs easier for the user
- WMS manages jobs on users' behalf (Delegation)
- Accept job submissions
- Matchmaking: Dispatch jobs to appropriate Compute Element (CE)
- Balances workload
- Distributed resource management
 - With the help of Information System

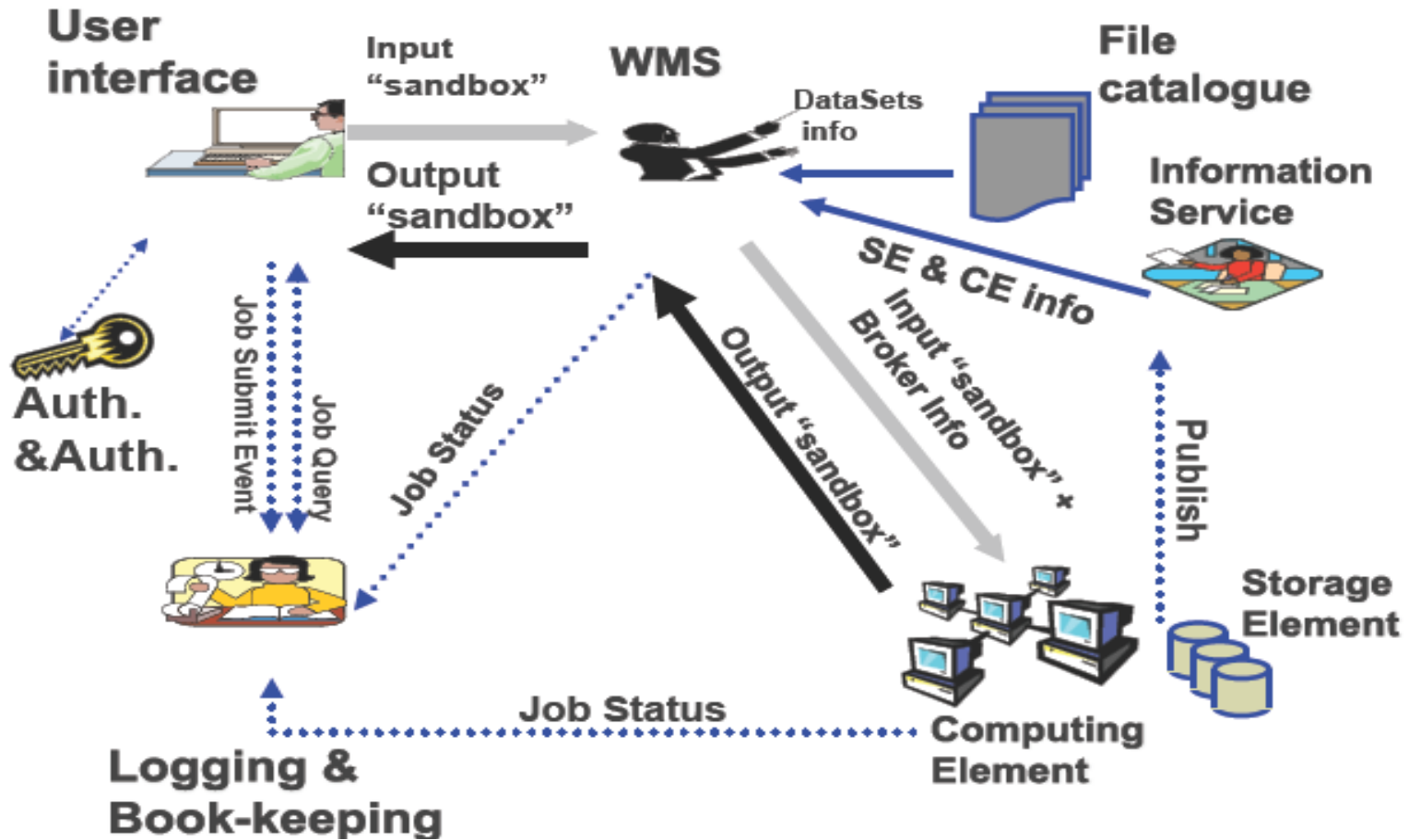
Workload Management Service

- Allow users
 - Manages jobs and their files
 - To get information about their status
 - To retrieve their output
- When a user submits a job, JDL options are to:
 - Specify CE
 - Allow WMS to choose CE (using optional tags to define requirements)
 - Specify SE (then RB finds “nearest” appropriate CE, after interrogating File catalogue service)

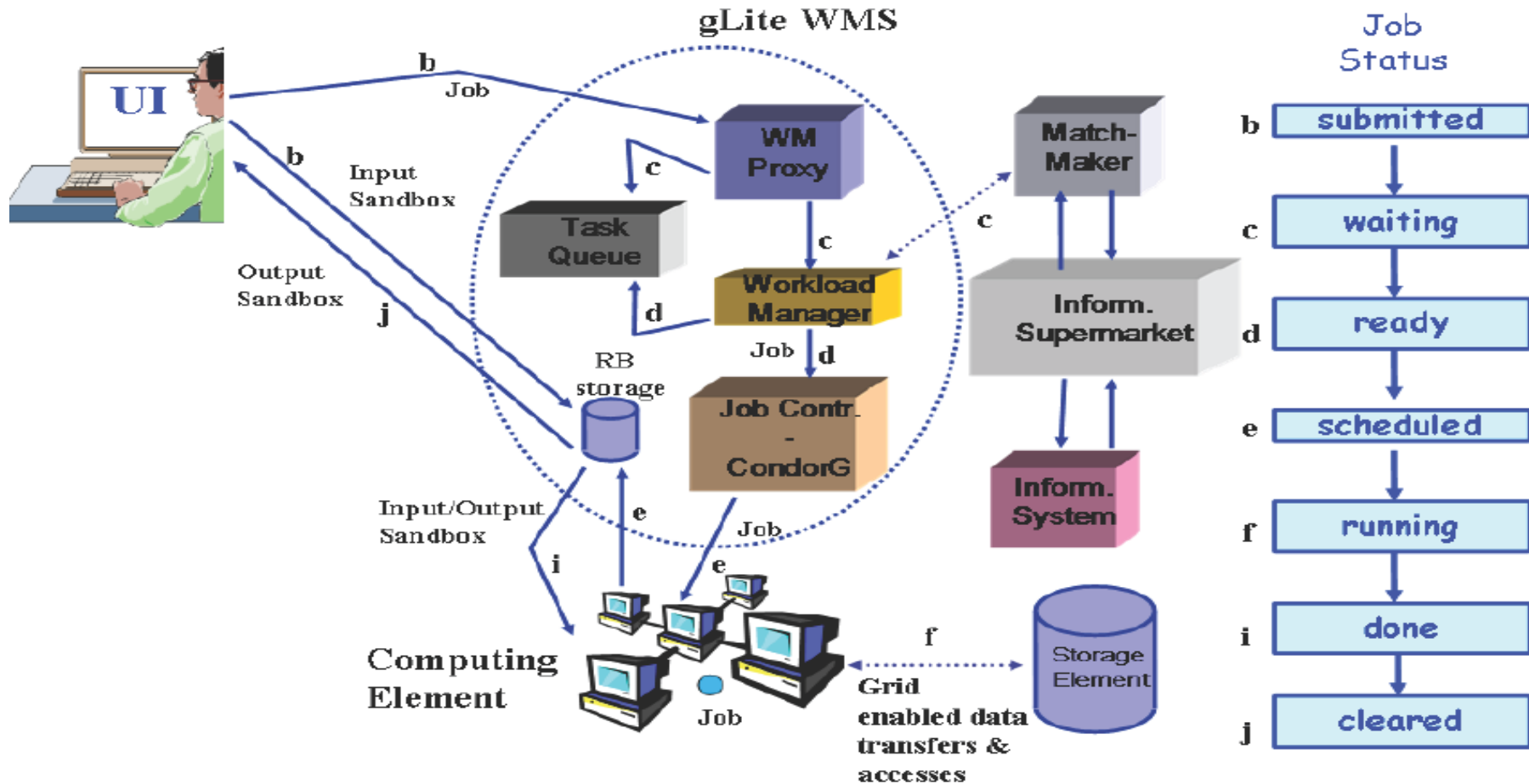
Logging and Bookkeeping

- Who did what and when?
- What is happening to my job?
- Usually runs on the WMS node

What really happens?



WMS and job states

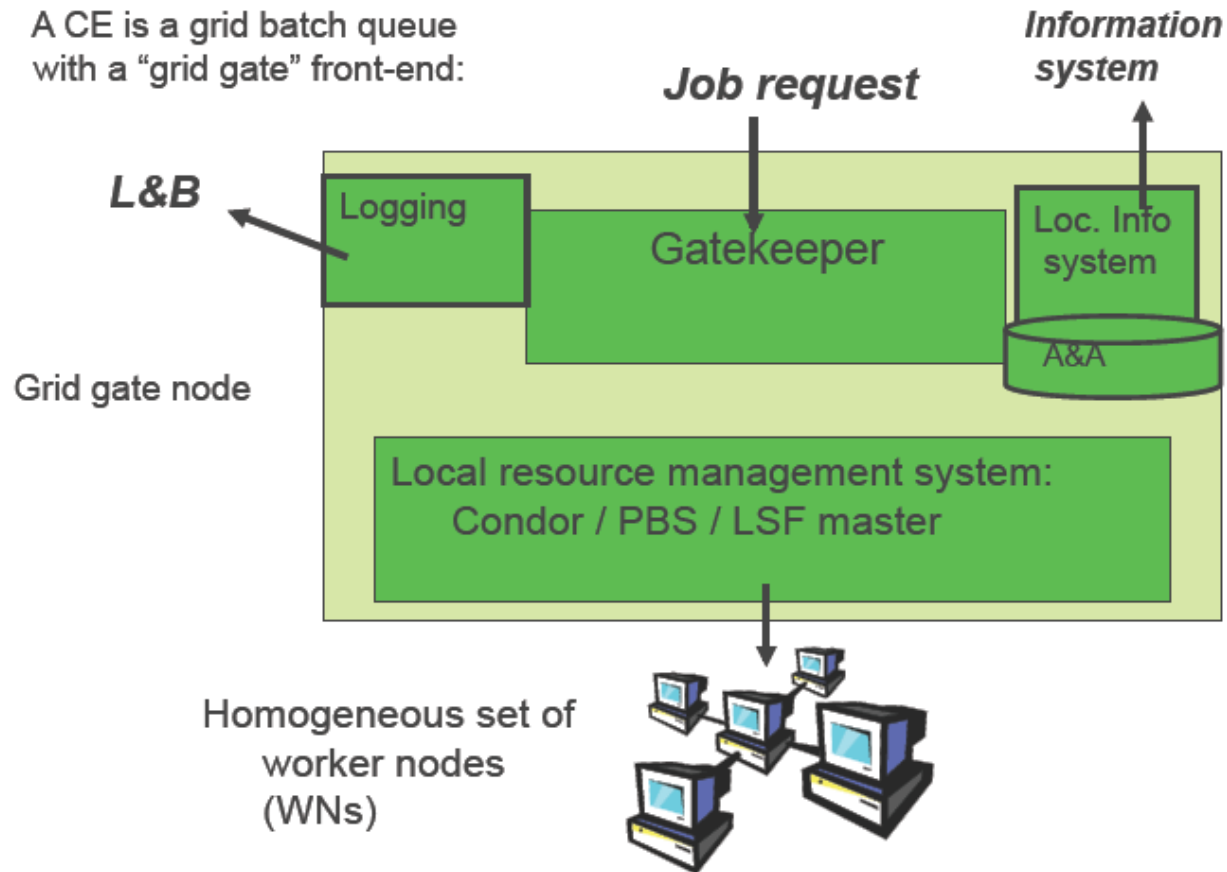


Computing Element

- CE: In Grid terminology, is some set of computing resources localized at a site (i.e. a cluster, a computing farm).
 - Authorization
 - Grid Gate: which acts as a interface to the cluster; a Local Resource Management System (LRMS) (sometimes called batch system), and the cluster itself

Compute Element







A CE is a grid batch queue with a "grid gate" front-end:



Worker Node

- The GG is responsible for accepting jobs and dispatching them for execution on the WNs via the LRMS
- These nodes are the constituents computing nodes of a cluster behind a CE
- This is where
 - The actual computations are performed
 - The user's software will be installed
- The user doesn't work directly with the worker nodes
 - since all the job requests pass through the gatekeeper, which hides all the specific details of the worker nodes (their management, etc.) from the user.

Local Resource Manager

Local Resource Manager	High Throughput Computing	High Performance Computing
Fork (Unix)		
Condor (1986)		
Platform LSF (1992)		
OpenPBS (1994)		
PBSPPro (1995)		
Sun Grid Engine (2000)		
- Torque (2005)		

Local Resource Manager Features

- Queue
- Various scheduling algorithms
- Different scheduling and management policy



gLite Middleware Services

- Security
- Execution Management
- **Information Service**
- **Data Management**

Information System

- Hierarchical Database
 - Lightweight Database Access Protocol: LDAP
- Local Information System
- Grid Information System
- Receives periodic (~5 min) updates from CE, SE, etc.
- Used by WMS (RB) node to determine resources to be used by a job

Specifying CE and data

```
Type = "Job";  
Executable = "/bin/hostname";  
Arguments = "";  
StdError = "stderr.txt";  
StdOutput = "stdout.txt";  
InputSandbox = "";  
OutputSandbox = {"stderr.txt", "stdout.txt"};  
  
Requirements = other.Architecture=="INTEL" &&  
    other.GlueCEInfoTotalCPUs > 480;  
Rank = other.GlueCEStateTotalJobs;  
  
InputData = "lfn:/grid/gridbox/antun/file.txt";
```

**WMS uses
Information System
to find CE**

**WMS uses File
Catalog to find file**



gLite Middleware Services

- Security
- Execution Management
- Information Service
- **Data Management**

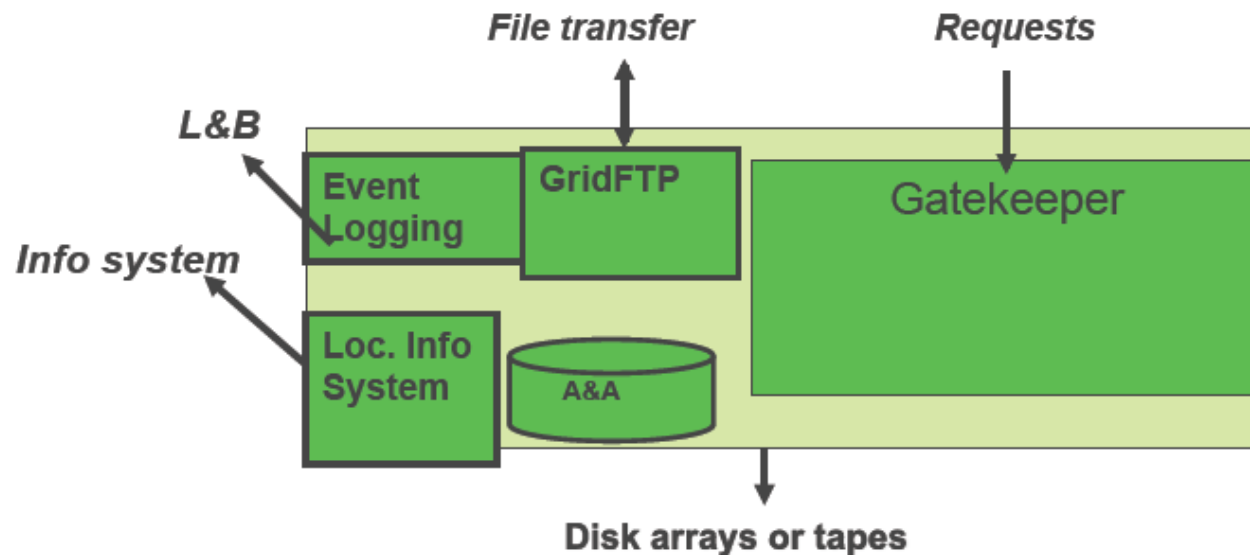
Storage Element

- Uniform access to data storage
- Storage elements hold files: write once, read many
- Replica files can be held on different SE:
 - “close” to CE
 - share load on SE
- File Catalogue - what replicas exist for a file and where are they?

Storage Element

- These nodes provide uniform access to data storage.
- The storage element may control mass storage systems such as large disk arrays but they are presented to the user as a unique storage space.

SE



Job types

- **Job Collections**
 - Type = "Collection";
- **DAG jobs (Direct Acyclic Graphs)**
 - Type = "Dag";
- **Parametric jobs**
 - JobType = "Parametric";
- **Interactive Jobs**
 - JobType = "Interactive";
- **MPI Jobs (Message Passing Interface)**
 - JobType = "MPICH";

JDL-file attributes

- **Type:** “Job” for sequential jobs
- **Executable:** sets the name of the executable file
- **Arguments:** command line arguments of the program
- **StdOutput, StdError:** files for storing the standard output and error messages output
- **InputSandbox:** set of input files needed by the program, including the executable
- **OutputSandbox:** set of output files which will be written during the execution, including standard output and standard error; these are sent from the CE to the WMS for you to retrieve
- **ShallowRetryCount:** in case of grid error, retry job this many times (“Shallow”: before job is running)

In sum up

- Grid structure is complicated but hidden from end-users, enabling all the comfort they need
- Users just need to obtain certificates and join the VO

Thanks

?