European Master on

Theoretical Chemistry and Computational Modelling

*Grid platforms and Services*

# The CompChem VO

Université Paul Sabatier, Toulouse, France

Sept 4-5, 2008

Osvaldo Gervasi

Dept of Mathematics and Computer Science
University of Perugia
**osvaldo@unipg.it**

# Outline

- The CompChem Virtual Organization
- LCG-2 and Glite middleware
- Job submission
- The Resource Specification Language (RSL)

**Credits**: EGEE tutorials

# The Grid added value

- The Grid allows to carry out simulation of molecular systems increasing the quality and the quantity of properties investigated.
- The researcher should be able to perform computational campaign:
  - Massive submission of sequential jobs running on different input datasets
  - Submission of parallel jobs running on large pool of nodes
- The Grid offers efficient data management facilities

# The Grid added value

- Software integration into distributed workflows
  - to assemble applications out of various (different or complementary) distributed competences coordinated via the Grid: electronic structure, elementary dynamics, statistical averaging, interfacing the experiment.
- Collaborative Engineering of knowledge
  - to handle chemical information and knowledge including training and production of new knowledge
- Sharing competencies in a secure environment
  - The state-of-the-art tools to share computational resources and to share computational codes among institutions in a secure fashion.

# The Virtual Organization (VO)

- Grid systems and applications aim to integrate, virtualise, and **manage resources and services** within distributed, heterogeneous, dynamic **Virtual Organisations** across traditional administrative and organisational domains (*real organisations*)
- A Virtual Organisation (VO) comprises a set of individuals and/or institutions having direct access to computers, software, data, and other resources for collaborative problem-solving or other purposes
- The VO is a concept that supplies a context for operation of the Grid that can be used to **associate** users, their requests, and a set of resources.

# The Virtual Organization (VO) (i)

- The sharing of resources in a VO is necessarily highly controlled, with resource providers and consumers defining clearly and carefully just what is shared, who is allowed to share, and the conditions under which sharing occurs.
- This resource sharing is facilitated and controlled by a set of services that allow resources to be discovered, accessed, allocated, monitored and accounted for, regardless of their physical location. The VO access is granted on the basis of a **X.509 digital certificate**
- Users, resources and services have their own X.509 digital certificate

# The Molecular Science community and the EGEE project

- The EGEE Grid environment represents for this community the high valued infrastructure able to supply
  - the necessary computational power
  - A suitable middleware able to let people collaborate together and access the common resources in a secure way.
- The **CompChem VO** has been created to support such computational needs and pivoting the access to the EGEE Grid facilities.
- Several EGEE sites are supporting the VO
  - the **Italian EGEE** sites, **CESGA** (Spain), **CYFRONET** and **POZNAN** Supercomputing Center (Poland), **Hellas Grid** and GRNET (Greece), University of Cyprus (Cyprus).

# How to join CompChem VO

- Each partner may be involved at different levels:
  - User: implementation on the Grid of a suite of codes of exclusive interest for the implementing laboratory
  - Code offer: the laboratory confers to the VO a stable suite of codes
  - Service offer: the laboratory participates to the management of the Grid infrastructure (manpower, hardware, service brokering and monitoring, etc), the development of joint projects etc.
- Reference URL: **http://compchem.unipg.it**
- The user and the resources must have a valid X.509 digital certificate, released by a Certification Authority of the National Academic and Research Network facilities.
- With the X.509 certificate installed in the web browser, the user asks to join the VO through the URL: **https://voms.cnaf.infn.it:8443/voms/compchem/webui/request/user/create**

# How to join CompChem VO (i)

- In order to use the Grid, the user needs a User Interface (UI) host. She/He has to:
  - install her/his own copy of the Plag ‘n Play UI software (UIPnP) or access a UI host, like ui.grid.unipg.it, having an account defined there
  - Install the **user X.509 certificate** (**private** and **public keys**) in the **$HOME/.globus** directory
  - Set up the submission file (using the Job Decription Language, JDL)
  - Create a proxy certificate (if not exists)
  - Run the jobs and monitor the submissions

# How to use the grid

**Each  time a user belonging to a given VO needs to access the Grid, a valid proxy certificate must exist**

- voms-proxy-info
  - List the status of the proxy certificate
- voms-proxy-init –voms compchem
  - Activates the proxy certificate after having checked the X.509 certificate and the private key of the user
- voms-proxy-destroy
  - Destroys an existing proxy

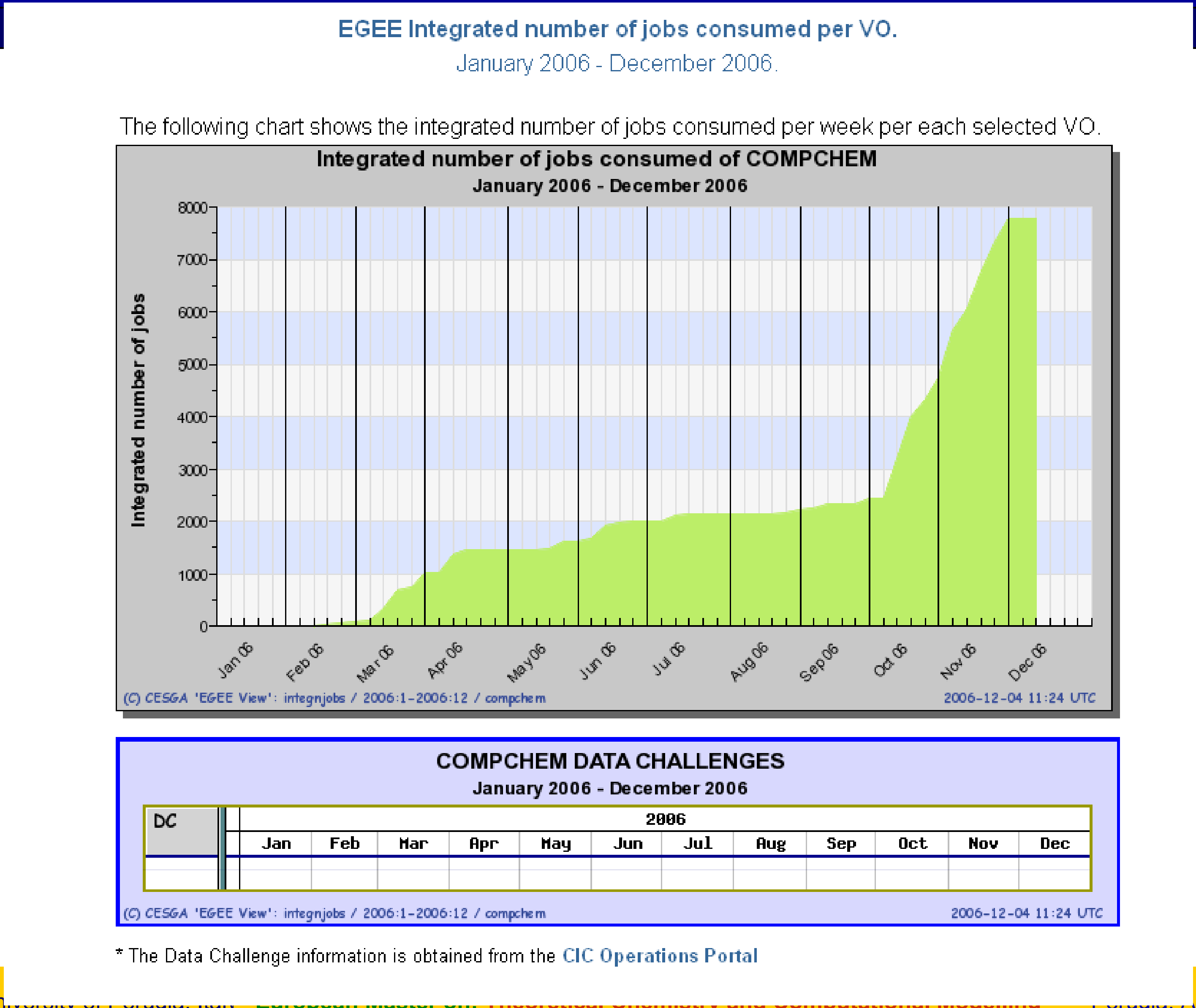
# lcg-infosites

```
#CPU Free  TotJobs Running Wait. ComputingElement
 180    34     0         0     0   fangorn.man.poznan.pl:2119/jobmanager-lcgpbs-compchem
 124   104     0         0     0   ce2.egee.cesga.es:2119/jobmanager-lcgpbs-compchem
  44    19     0         0     0   ares02.cyf-kr.edu.pl:2119/jobmanager-pbs-compchem
 186   186     0         0     0   ce.egee.man.poznan.pl:2119/jobmanager-lcgpbs-compchem
  14     0    16         5    11   egce.frascati.enea.it:2119/jobmanager-lsf-egee_long
  20     7     0         0     0   ce02.marie.hellasgrid.gr:2119/jobmanager-pbs-compchem
  64     1    38        25    13   ce01.isabella.grnet.gr:2119/jobmanager-pbs-compchem
 118    22     0         0     0   ce01.marie.hellasgrid.gr:2119/jobmanager-pbs-compchem
 118    22     0         0     0   glite-ce01.marie.hellasgrid.gr:2119/blah-pbs-compchem
 120     2     0         0     0   ce01.afroditi.hellasgrid.gr:2119/jobmanager-pbs-compchem
 118     0     0         0     0   ce01.kallisto.hellasgrid.gr:2119/jobmanager-pbs-compchem
 120    55     0         0     0   ce01.ariagni.hellasgrid.gr:2119/jobmanager-lcgpbs-compchem
 226   226     0         0     0   ce01.athena.hellasgrid.gr:2119/jobmanager-pbs-compchem
  80    21     5         0     5   gridba2.ba.infn.it:2119/jobmanager-lcgpbs-short
  22     0    25         0    25   grid002.ca.infn.it:2119/jobmanager-lcgpbs-grid
 322    24     6         4     2   grid012.ct.infn.it:2119/jobmanager-lcglsf-infinite
  48    29    35        14    21   grid0.fe.infn.it:2119/jobmanager-lcgpbs-grid
  28     3    25        25     0   griditce01.na.infn.it:2119/jobmanager-lcgpbs-grid
  62     0    33        28     5   prod-ce-01.pd.infn.it:2119/jobmanager-lcglsf-grid
 321    64    74        74     0   gridce.pi.infn.it:2119/jobmanager-lcglsf-gri
  37     0    29        29     0   grid003.roma2.infn.it:2119/jobmanager-lcgpbs-grid
   0     0     0         0  4444   grid001.ts.infn.it:2119/jobmanager-lcglsf-grid
  15     1    15        14     1   gridce.sns.it:2119/jobmanager-lcgpbs-grid
  14    14     0         0     0   ce2.egee.unile.it:2119/jobmanager-lcgpbs-grid
 118    10     0         0     0   spaci01.na.infn.it:2119/jobmanager-lcglsf-grid
   3     3     1         0     1   spacin-ce1.dma.unina.it:2119/jobmanager-lcgpbs-grid
  16     8     2         2     0   cex.grid.unipg.it:2119/blah-pbs-infinite
```

# Applications

- Quasiclassical (production)
  - ABCtraj
  - Venus
  - DL-POLY
- Quantum Time Dependent (production)
  - RWAVEP
- Quantum Time Independent (test)
  - APH3D
- Electronic structure (in test on a vanilla Globus environment)
  - MOLPRO
  - GAMESS, GAMESS-UK
  - Columbus (test phase will start soon)

# CompChem: some statistics



EGEE Integrated number of jobs consumed per VO.

January 2006 - December 2006.

The following chart shows the integrated number of jobs consumed per week per each selected VO.

Integrated number of jobs consumed of COMPCHEM

January 2006 - December 2006

(C) CESGA 'EGEE View': integnjobs / 2006:1-2006:12 / compchem 2006-12-04 11:24 UTC

COMPCHEM DATA CHALLENGES

January 2006 - December 2006

| DC | 2006 | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Jan | Feb | Mar | Apr | May | Jun | Jul | Aug | Sep | Oct | Nov | Dec |
| | | | | | | | | | | | | |

(C) CESGA 'EGEE View': integnjobs / 2006:1-2006:12 / compchem 2006-12-04 11:24 UTC

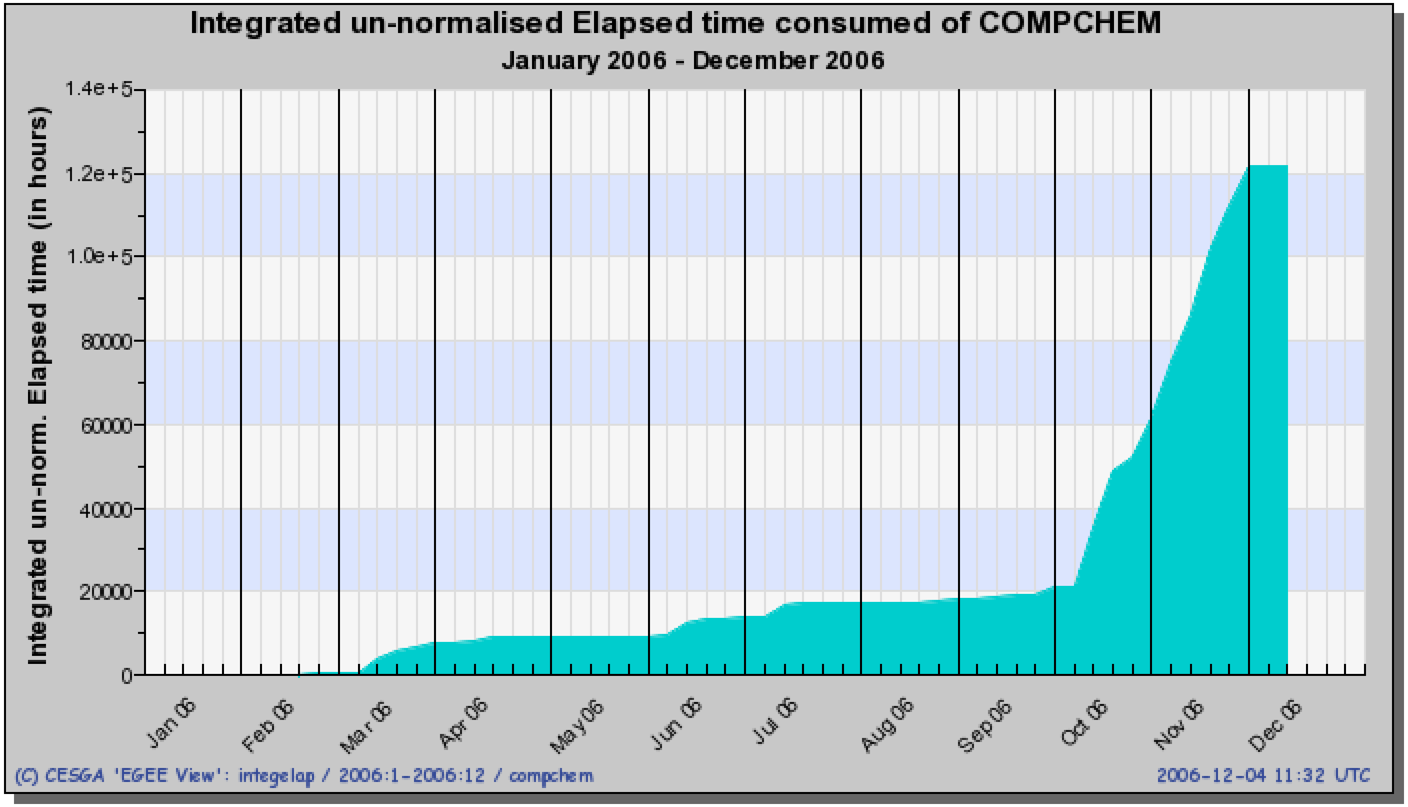
* The Data Challenge information is obtained from the CIC Operations Portal

# Hours of CPU per week

## EGEE Integrated un-normalised Elapsed time consumed per VO.

January 2006 - December 2006.

The following chart shows the integrated un-normalised Elapsed time consumed per week per each selected VO.



(C) CESGA 'EGEE View': integelap / 2006:1-2006:12 / compchem

2006-12-04 11:32 UTC

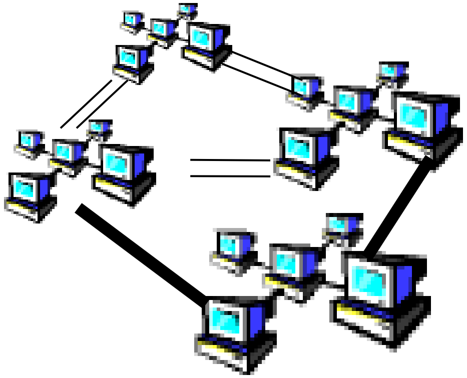
# What are the characteristics of a Grid system?

## Numerous Resources

Ownership by Mutually Distrustful Organizations & Individuals

Connected by Heterogeneous, Multi-Level Networks

Different Security Requirements & Policies Required



Different Resource Management Policies

Potentially Faulty Resources

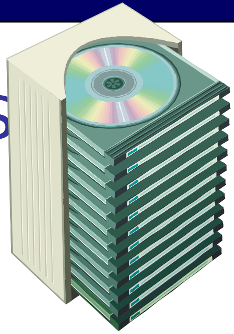
Geographically Separated

Resources are Heterogeneous

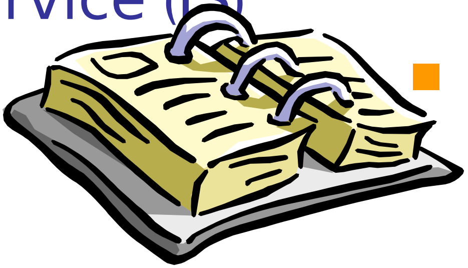
# Main Logical Machine Types (Services) in LCG-2/Glite

- User Interface (UI)

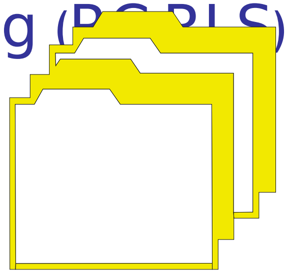
- Storage Element (S



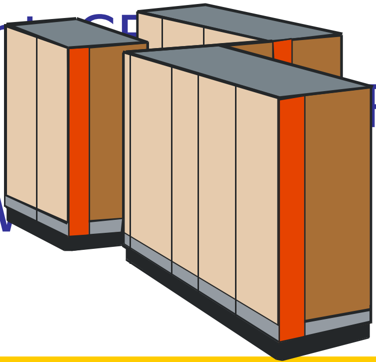
- Information Service (IS)



- Replica Catalog (RC,RLS)



- Computing Element (CE
  - Frontend Node
  - Worker Nodes (W



- Resource Brok (RB)

# User Interface

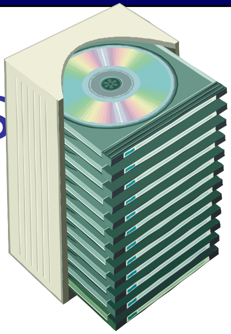
- The initial point of access to the LCG2/Glite Grid is the User Interface
- This is a machine where
  - Glite users have a personal account
  - The user's certificate is installed
- The UI is the gateway to Grid services
- It provides a Command Line Interface to perform the following basic Grid operations:
  - submit a job for execution on a Computing Element;
  - list all the resources suitable to execute a given jo
  - replicate and copy files;
  - cancel one or more jobs;
  - retrieve the output of one or more finished jobs;
  - show the status of one or more submitted jobs.

- One or more UIs are available at each site part of the LCG-2/Glite Grid

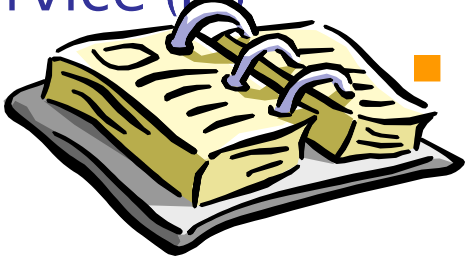
# Main Logical Machine Types (Services) in LCG-2/Glite

- User Interface (UI)

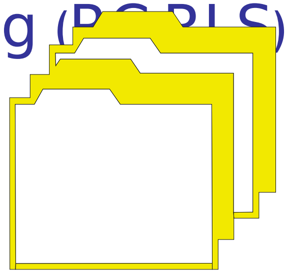
- Storage Element (S



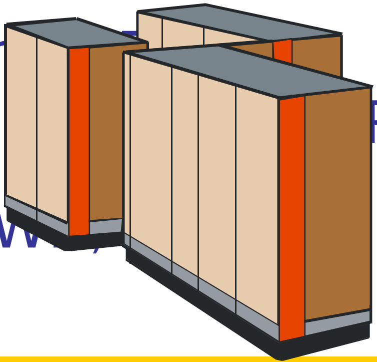
- Information Service (IS)



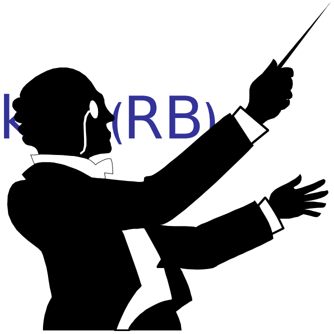
- Replica Catalog (RC,RLS)



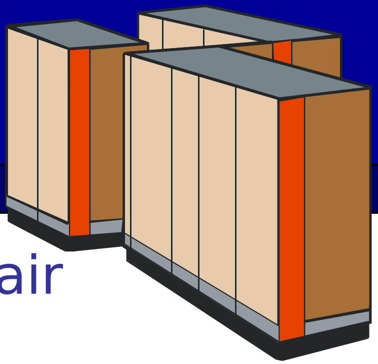
- Computing Elemen
  - Frontend Node
  - Worker Nodes (W )



- Resource Brok (RB)



# Computing Element (CE)

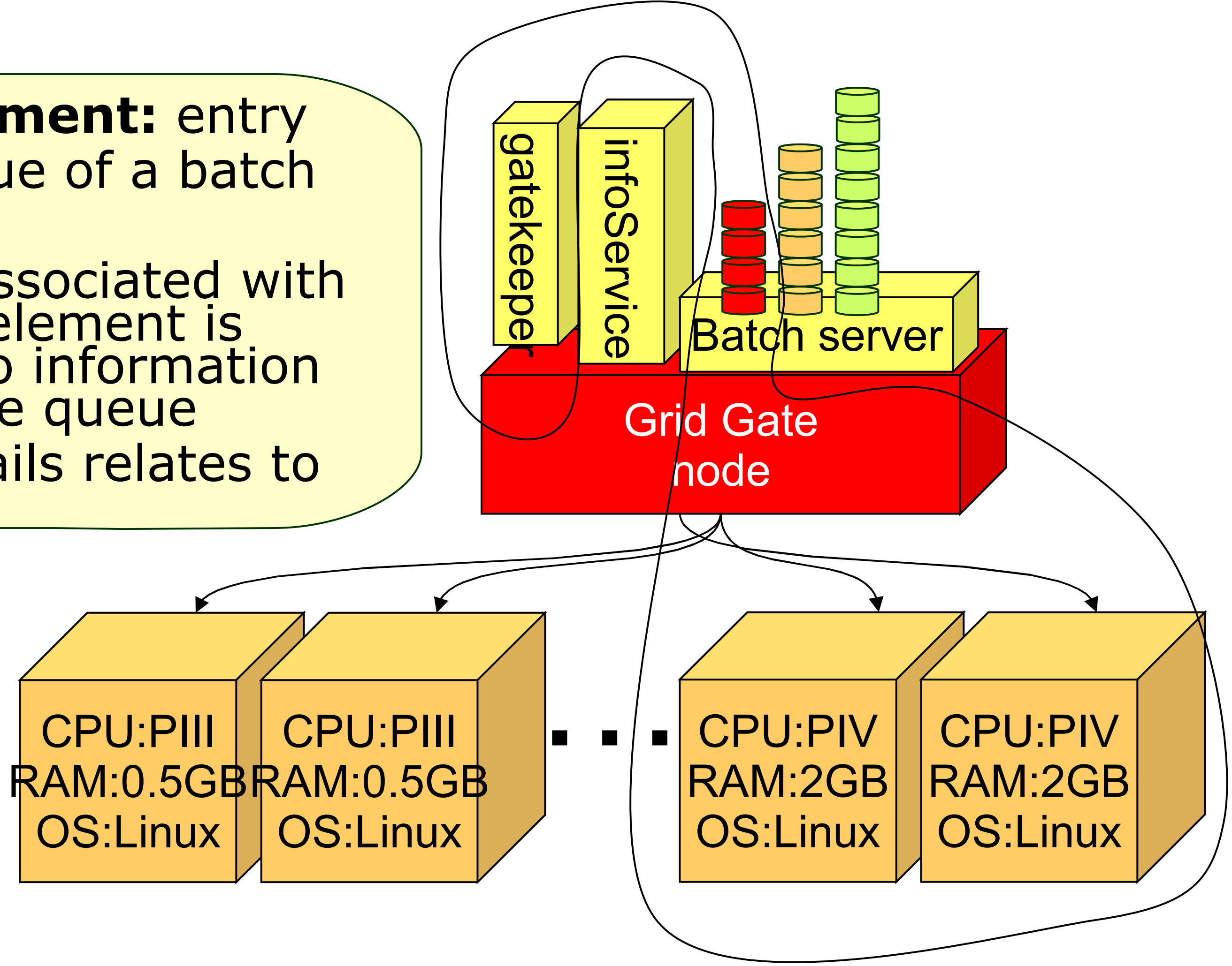


- Defined as a Grid batch Queue and identified by a pair
  <hostname>:<port>/<batch queue name>
- Several queues defined for the same hostname are considered different CEs. For example:
  ce.grid.unipg.it:2119/jobmanager-lcgpbs-long
  adc0015.cern.ch:2119/jobmanager-lcgpbs-short
- A Computing Element is built on a homogeneous farm of computing nodes (called Worker Nodes)
- One node acts as a ***Grid Gate (GG)*** or front-end to the Grid and runs:
  - a Globus gatekeeper
  - the Globus GRAM (Globus Resource Allocation Manager)
  - the master server of a Local Resource Management System that can be:
    - PBS, LSF or **Condor**
  - a local Logging and Bookkeeping server
- Each LCG-2 site runs at least one CE and a farm of WNs behind it.

# Computing Element

**Computing Element:** entry point into a queue of a batch system

- information associated with a computing element is limited only to information relevant to the queue
- Resource details relates to the system

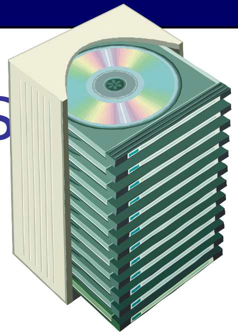


in the example
the red queue
is assigned
for two hosts

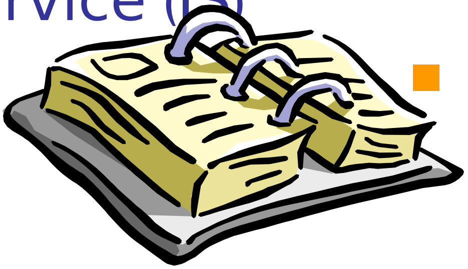
# Main Logical Machine Types (Services) in LCG-2/Glite

- User Interface (UI)

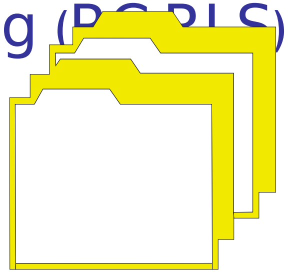
- Storage Element (S



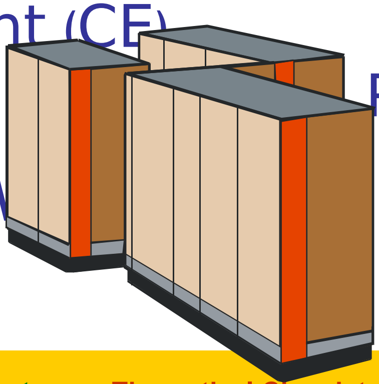
- Information Service (IS)



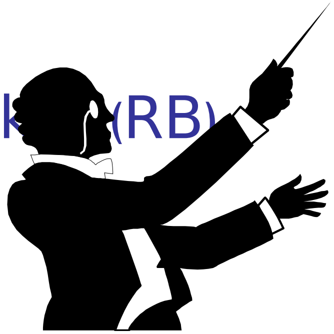
- Replica Catalog (RC RLS)



- Computing Element (CE)
  - Frontend Node
  - Worker Nodes (W

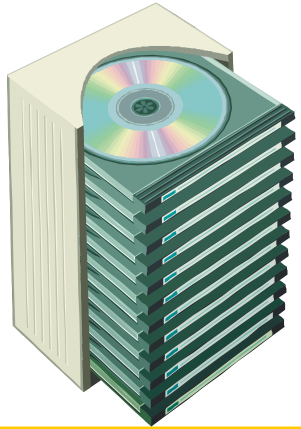


- Resource Brok (RB)



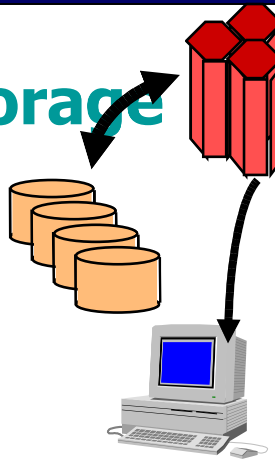
# Storage Element (SE)

- A ***Storage Element (SE)*** provides uniform access and services to large storage spaces.
- Each site includes at least one SE
- They use two protocols:
  - ***GSIFTP*** for file transfer
  - ***Remote File Input/Output (RFIO)*** for file access



# Storage Resource Management (SRM)

Data are stored on **disk pool servers** or **Mass Storage Systems**



- storage resource management needs to take into account
  - Transparent access to files (migration to/from disk pool)
  - Space reservation
  - File status notification
  - Life time management
- SRM (Storage Resource Manager) takes care of all these details
  - SRM is a Grid Service that takes care of local storage interaction and provides a Grid interface to outside world

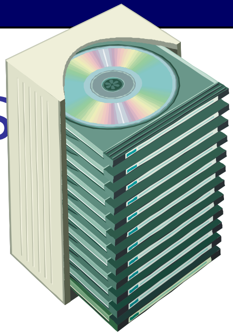
# Storage Resource Management

- Support for **local policy**
  - Each storage resource can be managed independently
  - Internal priorities are not sacrificed by data movement between Grid agents
- Disk and tape resources are presented as a **single element**
- **Reservation** on demand and advance reservation
  - Space can be reserved for registering a new file
  - Plan the storage system usage
- File **status** and **estimates** for planning
  - Provides info on file status
  - Provide estimates on space availability/usage

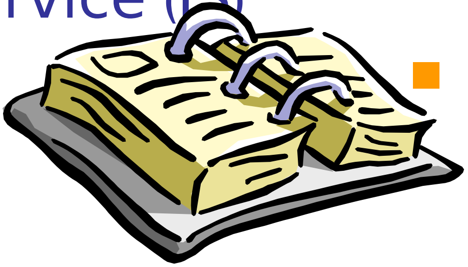
# Main Logical Machine Types (Services) in LCG-2/Glite

- User Interface (UI)

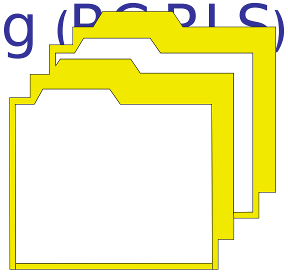
- Storage Element (S



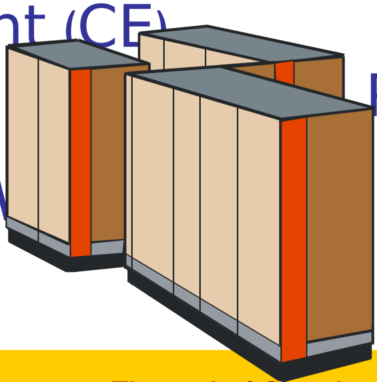
- Information Service (IS)



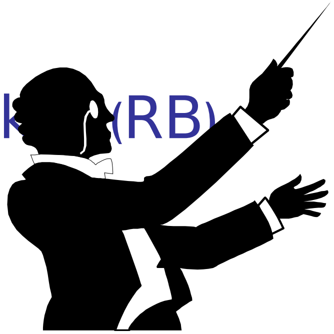
- Replica Catalog (RC RLS)



- Computing Element (CE)
  - Frontend Node
  - Worker Nodes (W

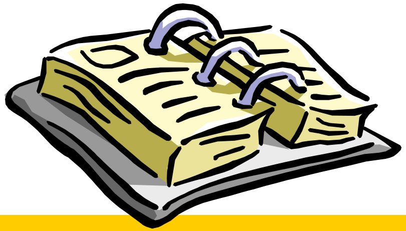


- Resource Brok (RB)



# Information System (IS)

- The Information System (IS) provides information about the LCG-2/Glite **Grid resources and their status**
- The IS of LCG2 is based on **LDAP**: a directory service infrastructure which is a specialized database optimized for reading, browsing and searching information.
- The IS of Glite is based on **UDDI** (Universal Description, Discovery and Integration, the Web Services Index)
- the IS schema implements the ***GLUE (Grid Laboratory for a Uniform Environment) Schema***



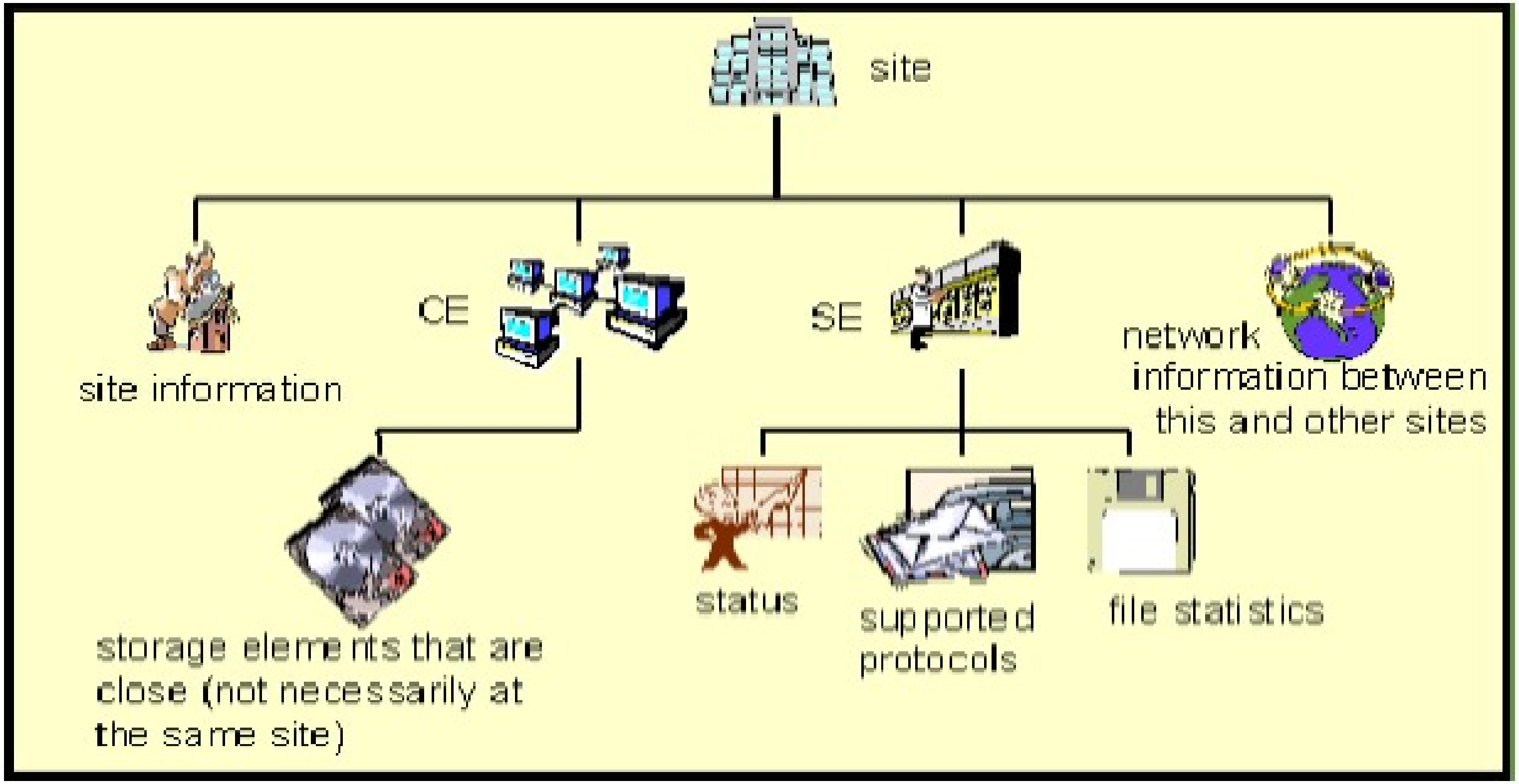
# How to store Information?

- The LDAP information model is based on entries.
- An ***entry*** usually describes an object such as a
  - person,
  - a computer,
  - a server, and so on.
- Each entry contains one or more ***attributes*** that describe the entry.
- Each attribute has a type and one or more ***values***.
- Each entry has a name called a ***Distinguished Name (DN)*** that uniquely identifies it.
- A DN is formed by a sequence of attributes and values.
- Example: The DN of a particular CE entry would be:
  - an attribute identifying the site (**site_ID**=cern) and
  - an attribute identifying the CE (**CE_ID**=lxn1102.cern.ch),
  - so the complete DN would be:

    **CE_ID=lxn1102.cern.ch,site_ID=cern.**

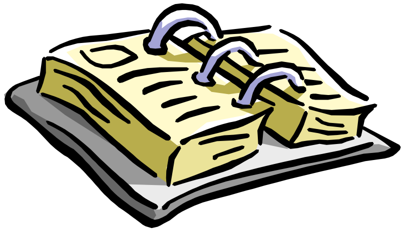
# The Directory Information Tree

- Based on their DNs, the entries can be arranged into a hierarchical tree-like structure.
- This tree of directory entries is called the ***Directory Information Tree (DIT)***.


site
site information
CE
SE
network
information between this and other sites
storage elements that are close (not necessarily at the same site)
status
supported protocols
file statistics

# Information System (IS)

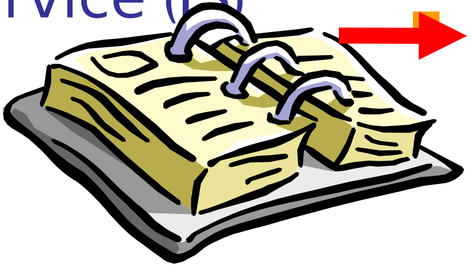
- The IS is a hierarchical system with 3 levels from bottom up:
  - **GRIS (*Grid Resource Information Servers)*** level (CE and SE level)
  - ***Grid Index Information Server (GIIS)*** level (site level)
  - Top, centralized level (Grid level)
- the Globus ***Monitoring and Discovery Service (MDS) mechanism*** has been adopted at the **GRIS** level
- The other two levels use the ***Berkeley DB Information Index (BDII) mechanism***



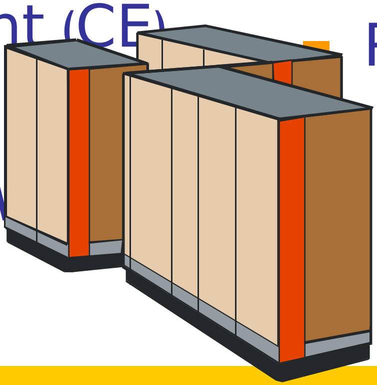
# Main Logical Machine Types (Services) in LCG-2/Glite

- User Interface (UI)

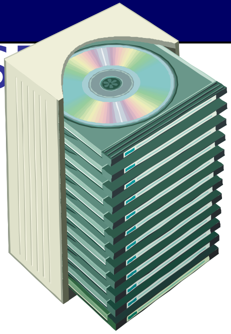
- Information Service (IS)



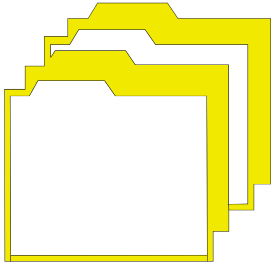
- Computing Element (CE)
  - Frontend Node
  - Worker Nodes (W



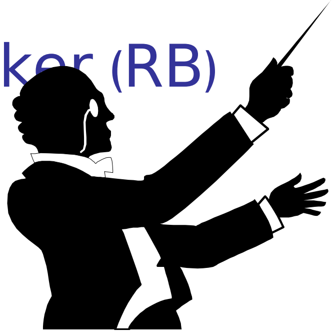
- Storage Element (S



- Replica Catalog (RC,RLS)

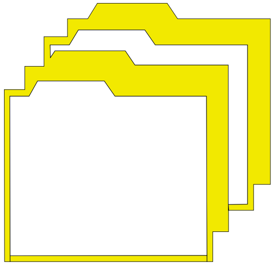


- Resource Broker (RB)



# Data Management

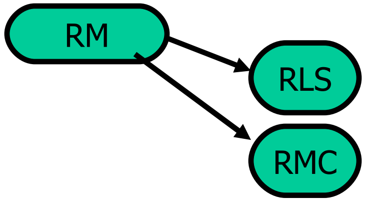
- The Data Management services are provided by
  - the ***Replica Management System (RMS)*** of EDG
  - and the ***LCG Data Management*** client tools
- In LCG, the data files are replicated:
  - on a temporary basis,
  - to many different sites depending on
  - where the data is needed.
- The users or applications do not need to know where the data is located, they use logical files names
- the Data Management services are responsible for locating and accessing the data.



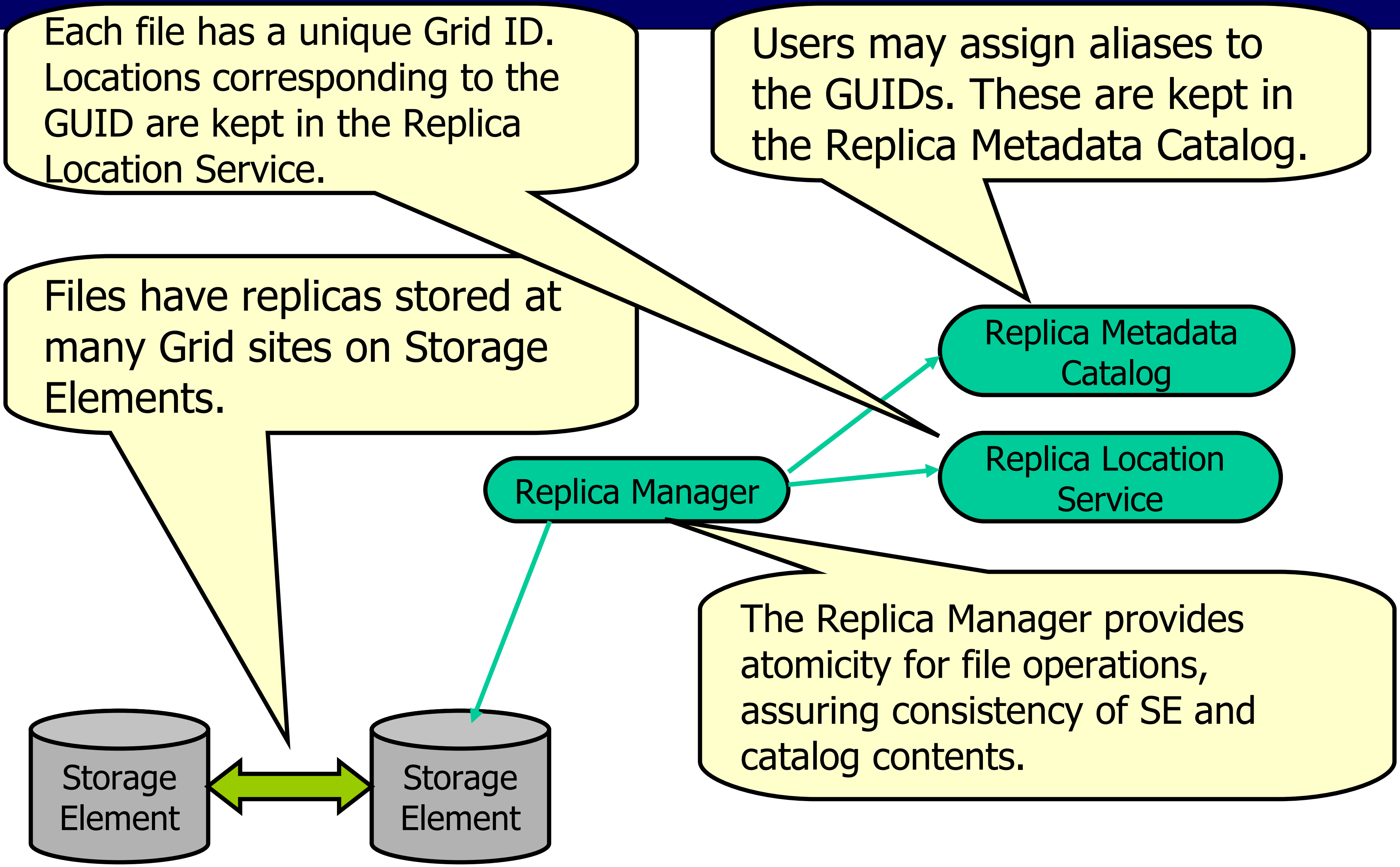
# Data Management Tools

- Tools for
  - Locating data
  - Copying data
  - Managing and replicating data
  - Meta Data management

- You have
  - Replica Manager (RM)
  - Replica Location Service (RLS)
  - Replica Metadata Catalog (RMC)



# Replication Services: Basic Functionality



Each file has a unique Grid ID. Locations corresponding to the GUID are kept in the Replica Location Service.

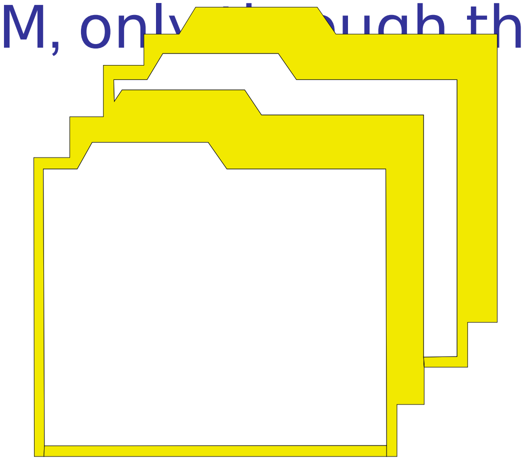
Users may assign aliases to the GUIDs. These are kept in the Replica Metadata Catalog.

Files have replicas stored at many Grid sites on Storage Elements.

The Replica Manager provides atomicity for file operations, assuring consistency of SE and catalog contents.

# Replica Manager (RM)

- High level data management on the Grid, takes care of:
  - Location of data
  - Replication of data
  - Efficient access to data
- Hides the SRM (Storage Resource Manager):
  - User cannot access directly the SRM, only through the RM
- Coordinates the use of
  - Replica Location Service
  - Replica Metadata Catalog

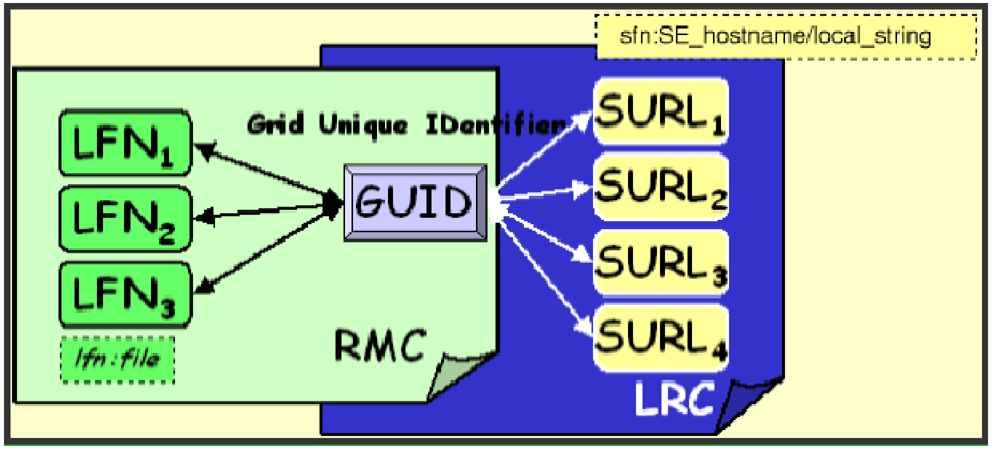


# File References and Replica Catalogs

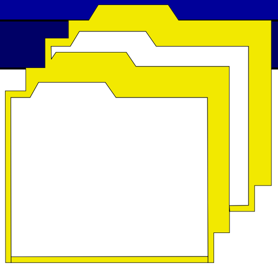
- The files in the Grid are referenced by different names:
  - ***Grid Unique IDentifier (GUID)***
  - ***Logical File Name (LFN)***
  - ***Storage URL (SURL)***
  - ***Transport URL (TURL)***.
- the GUID or LFN refer to files and not replicas, and say nothing about locations
- the SURLs and TURLs give information about where a physical replica is located.

**RMC : Replica Metadata Catalog**

**LRC : Local Replica Catalog**



# Abstract file names



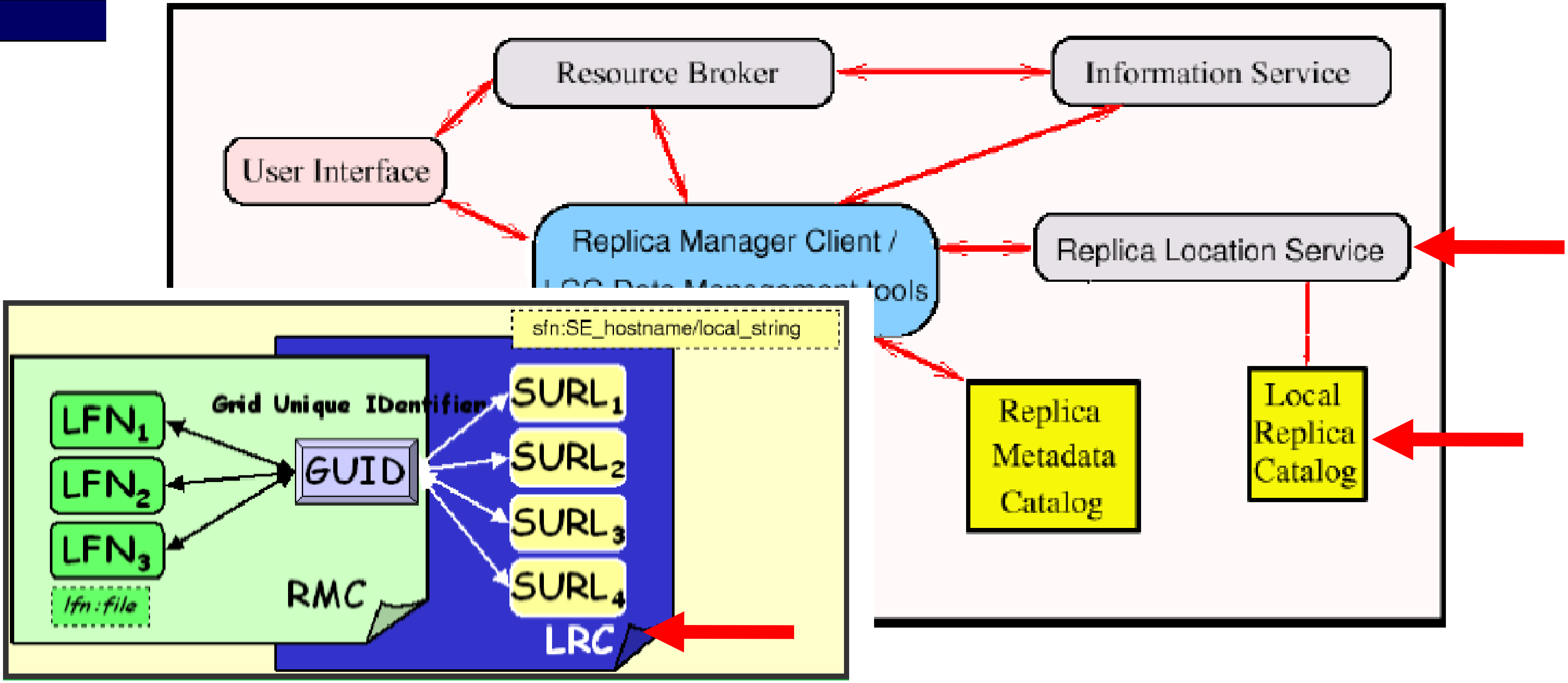
- **GUID**
  - A file can always be identified by its GUID
  - GUID is assigned at data registration time
  - GUID is based on the UUID standard to guarantee unique IDs
  - A GUID is of the form: guid:<unique string>
    (“guid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6”)
  - All the replicas of a file will share the same GUID
- **LFN**
  - In order to locate a Grid accessible file, the human user will normally use a LFN
  - LFNs are human-readable strings, they are allocated by the user as GUID aliases
  - LFN’s form is: lfn:<any alias> (“lfn:cms/20030203/run2/track1”)

# Physical file names

- **SURL**
  - used by the RMS to find where a replica is physically stored and by the SE to locate the file
  - SURLs are of the form: sfn:<SE hostname>/<local string>
  - where <local string> is used internally by the SE to locate the file.
  - The location of an actual piece of data on a storage system, e.g.
    "srm://pcrd24.cern.ch/flatfiles/cms/output10_1" (SRM)
    "sfn://lxshare0209.cern.ch/data/alice/ntuples.dat" (Classic SE))
- **TURL**
  - TURL gives the necessary information to retrieve a physical replica, including
    - Hostname, path, protocol, port (as any conventional URL);

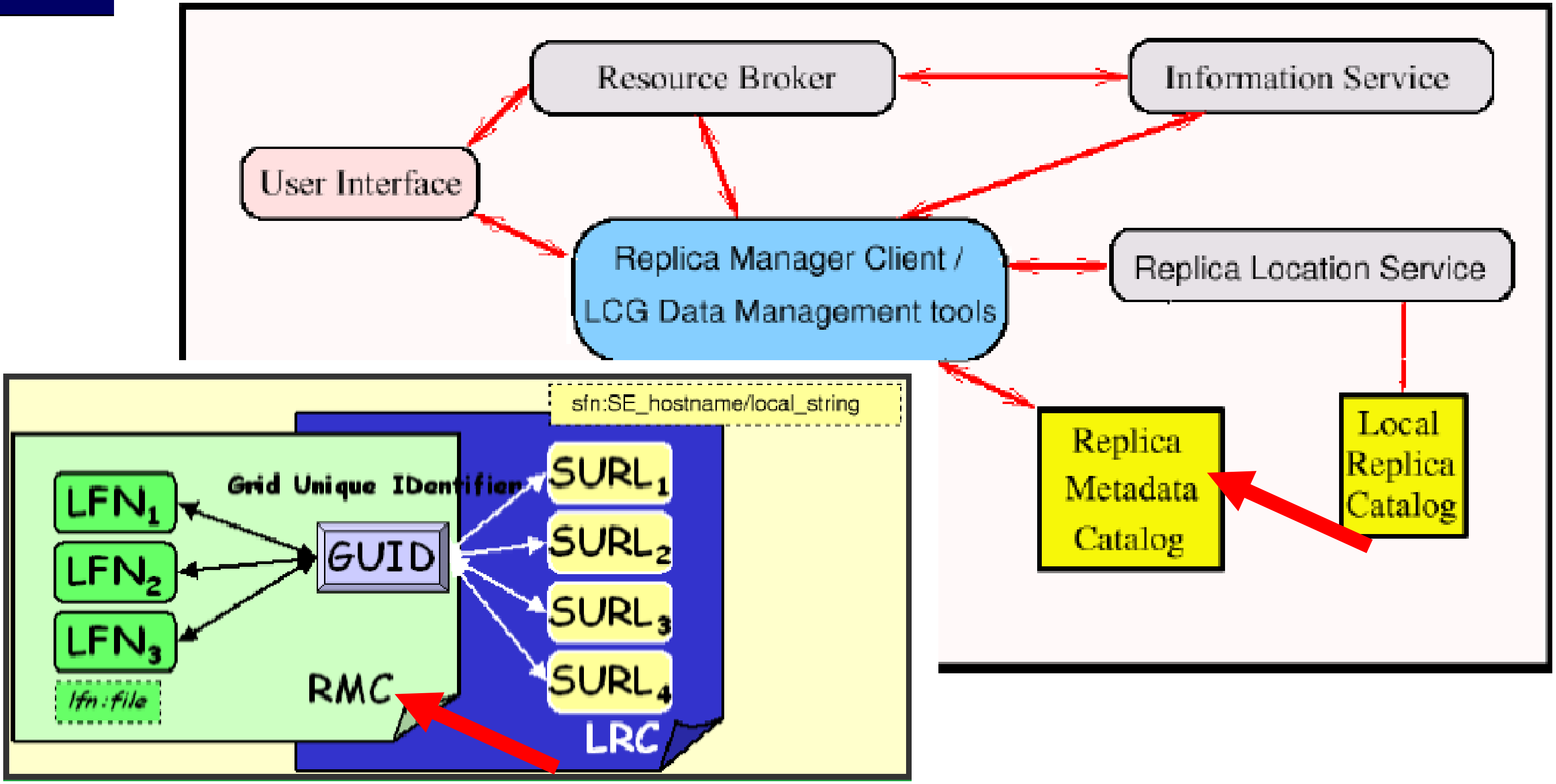
- Temporary locator of a replica + access protocol: understooc SE, e.g. "rfio://lxshare0209.cern.ch//data/alice/ntuples.dat"

# Replica Location Service (RLS)



- **RLS** maintains information about the physical location of the replicas (mapping with the GUIDs).
- It is composed of several **Local Replica Catalogs (LRCs)** which hold the information of replicas for a single VO.

# Replica Metadata Catalog (RMC)

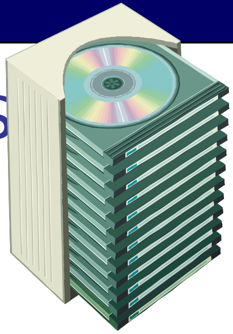


- The **RMC** stores the mapping between GUIDs and the respective aliases (LFNs)
- Maintains other metada information (sizes, dates, ownerships. . . )

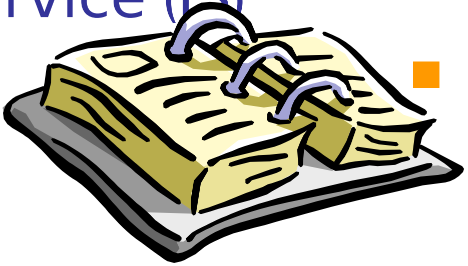
# Main Logical Machine Types (Services) in LCG-2/Glite

- User Interface (UI)

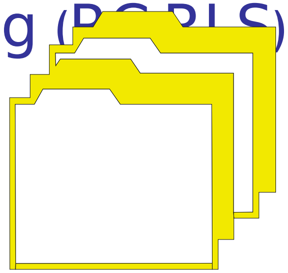
- Storage Element (S



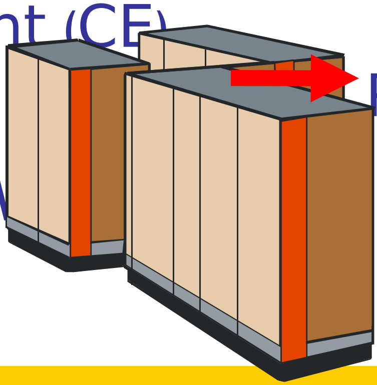
- Information Service (IS)



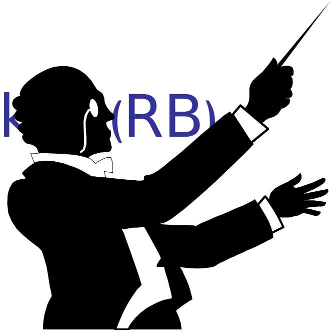
- Replica Catalog (RC,RLS)



- Computing Element (CE)
  - Frontend Node
  - Worker Nodes (W



- Resource Brok (RB)



# Job Management

- The user interacts with Grid via a **Workload Management System (WMS)**
- The Goal of WMS is the **distributed scheduling** **and** **resource management** **in a Grid environment**.
- What does it allow Grid users to do?
  - To submit their jobs
  - To execute them on the “best resources”
    - **The WMS tries to optimize the usage of resour**
  - To get information about their status
  - To retrieve their output

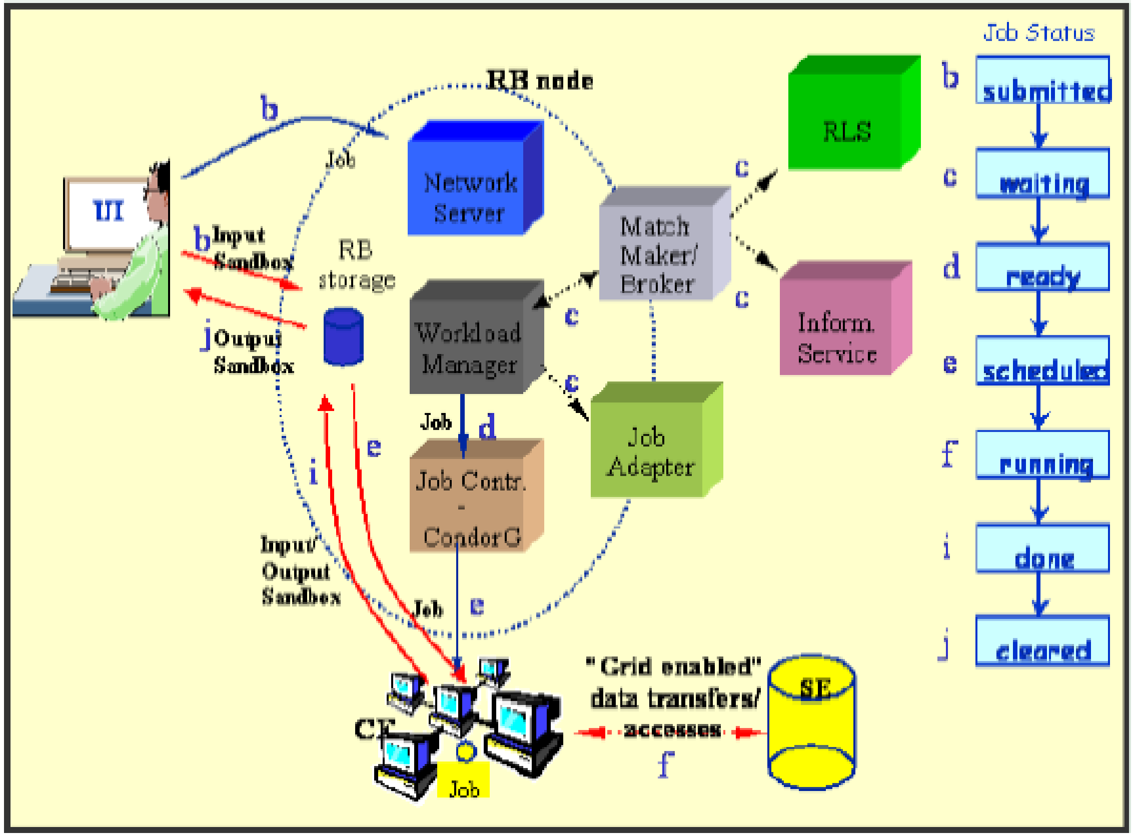
# WMS Components

- WMS is currently composed of the following parts:
  - **Workload Manager**, which is the core component of the system
  - **Match-Maker** (also called Resource Broker), whose duty is finding the best resource matching the requirements of a job (match-making process).
  - **Job Adapter**, which prepares the environment for the job and its final description, before passing it to the Job Control Service.
  - **Job Control Service (JCS)**, which finally performs the actual job management operations (job submission, removal. . .)
  - **Logging and Bookkeeping** services (LB) : store Job Info available for users to query

# Let’s think the way the Grid thinks!

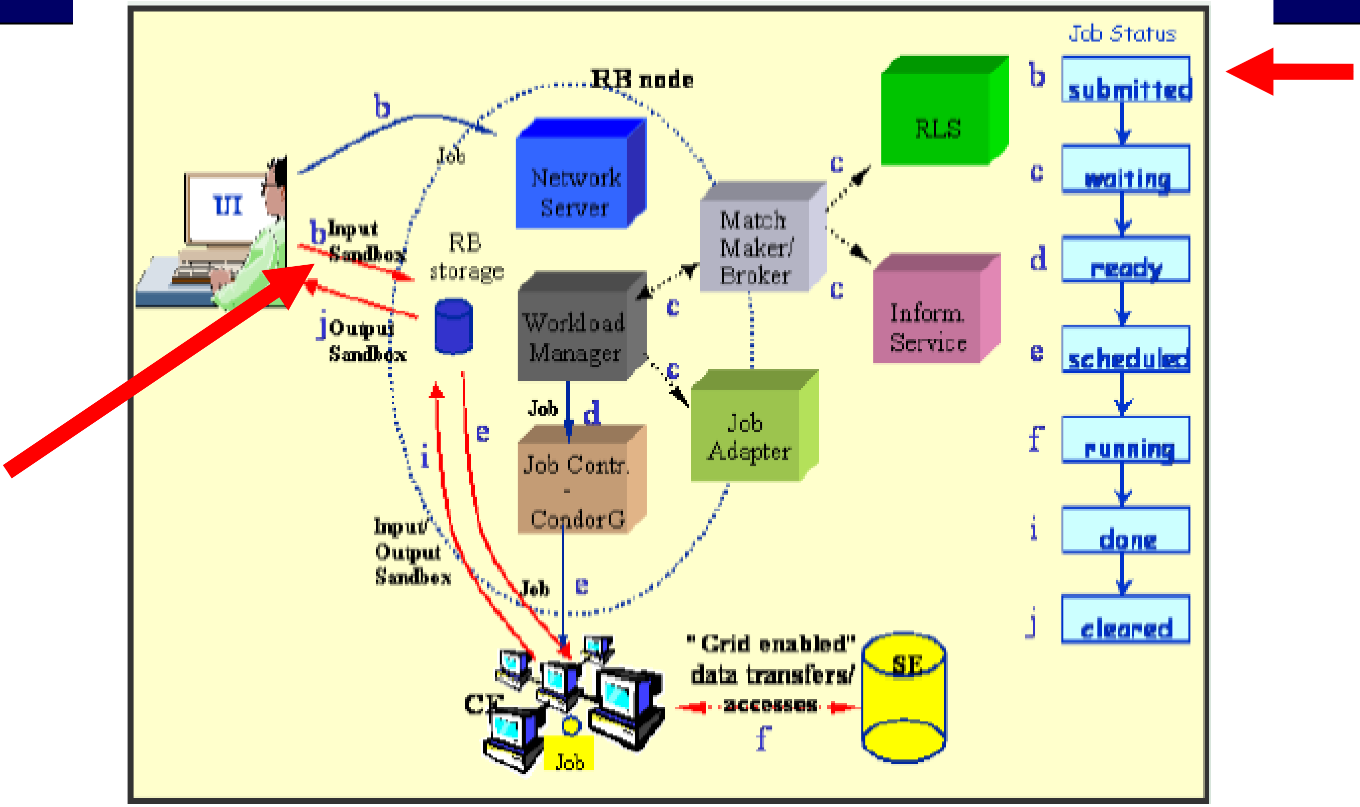
- Information to be specified
  - Job characteristics
  - Requirements and Preferences of the computing system
  - Software dependencies
  - Job Data requirements
- Specified using a Job Description Language (JDL)

# Job Flow



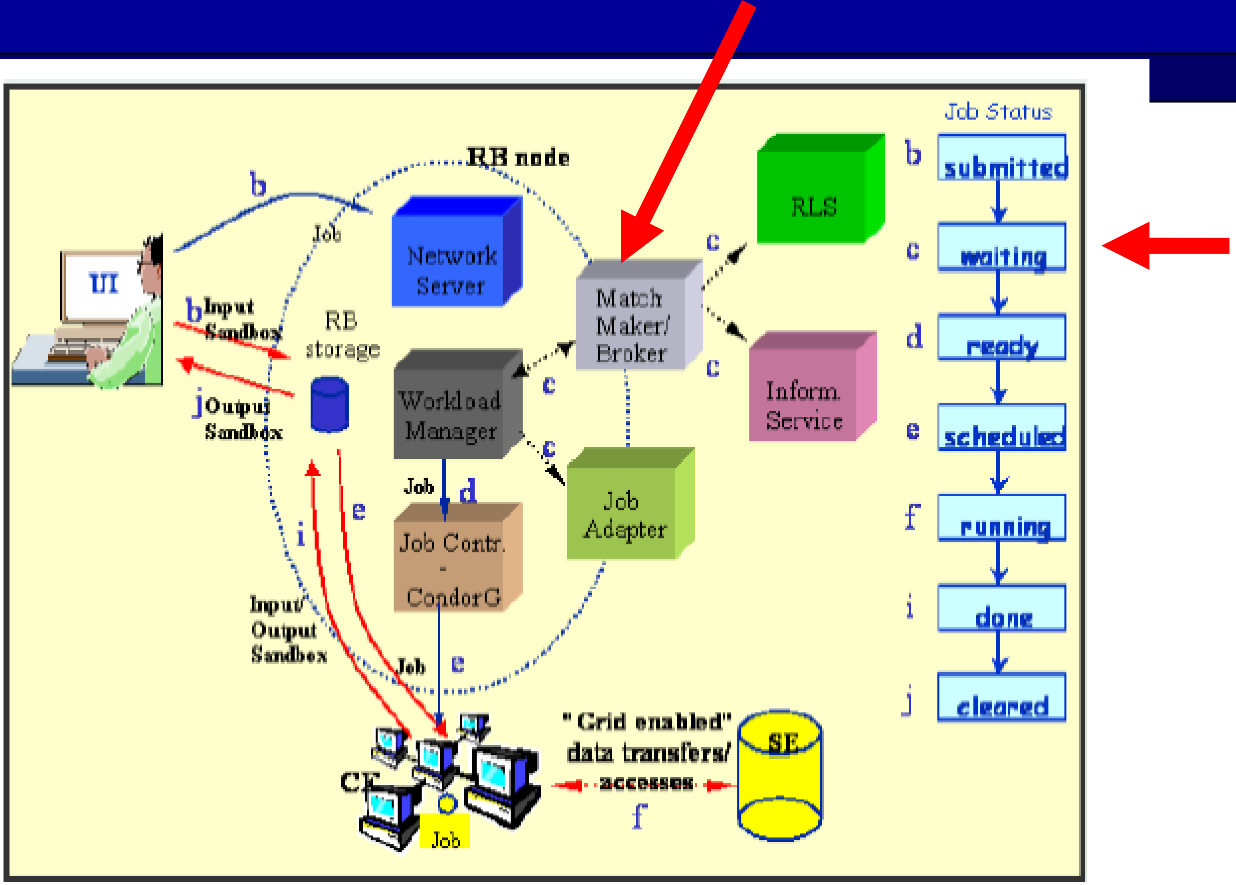
- a. the **user logs** to the **UI machine** and creates a **proxy certificate** that authenticates her in every secure interaction, and has a limited lifetime.

# Job Flow



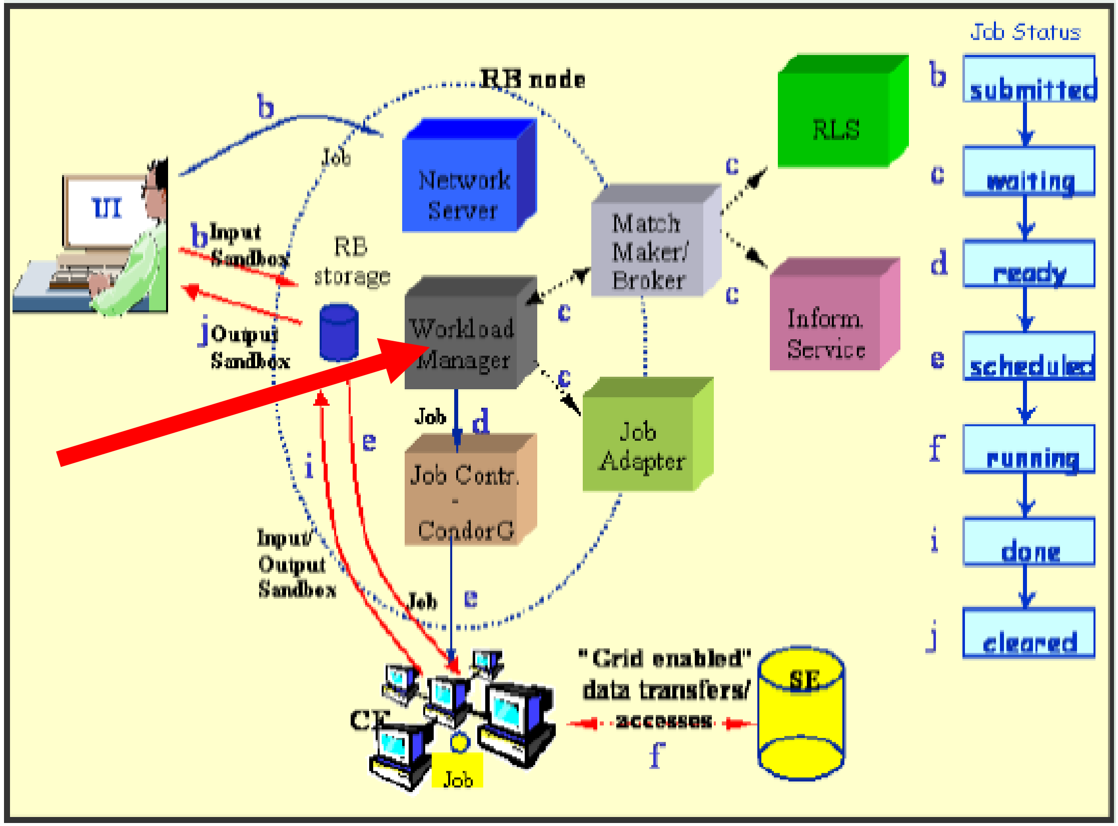
- b. The user **submits the job** from the UI to the WMS.
- The user can specify in the JDL one or more files to be copied from the UI to the RB node; this set of files is called ***Input Sandbox***. The event is logged in the LB and the status of the job is **SUBMITTED**.

# Job Flow



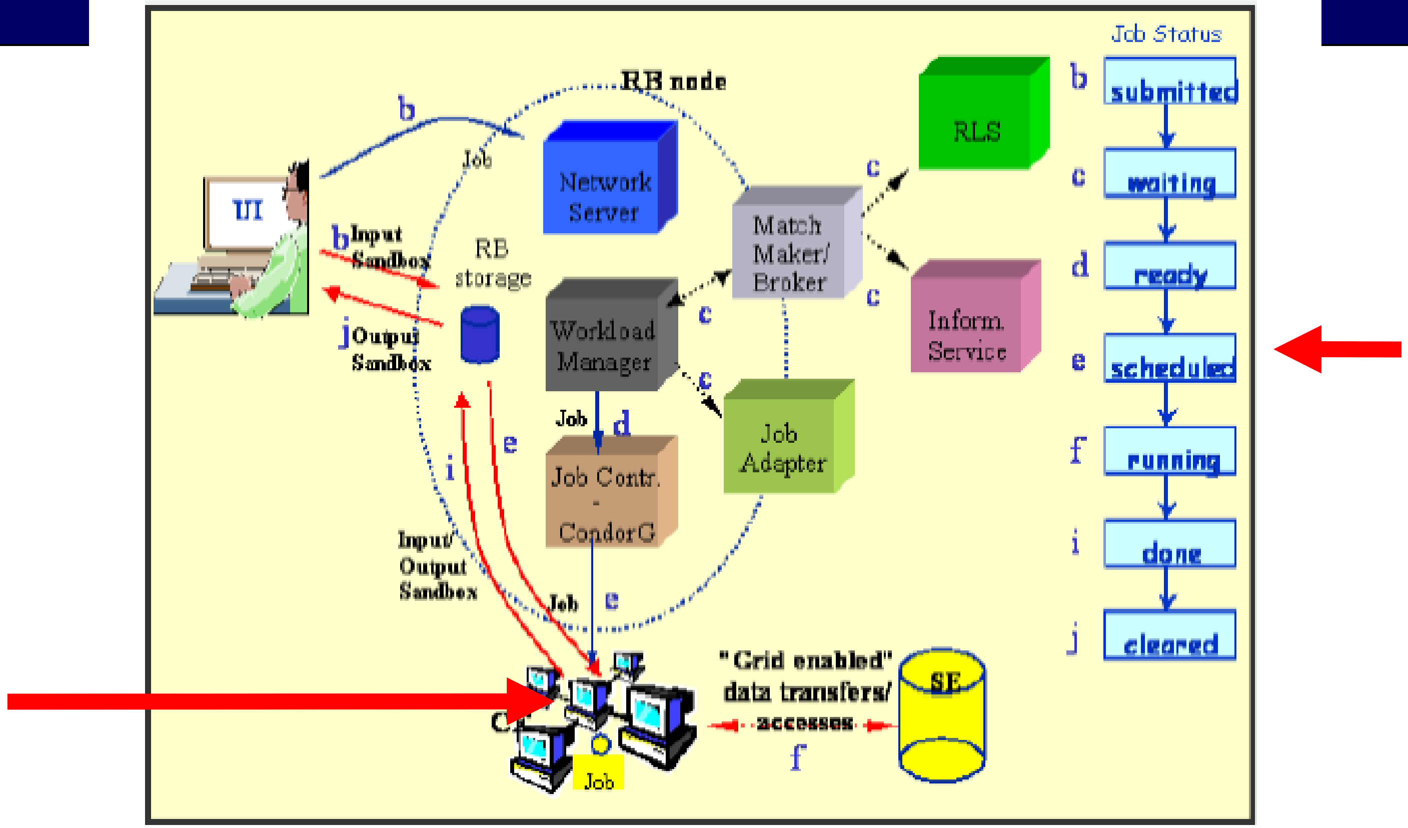
- c. The WMS, and in particular the Match-Maker component, looks for the best available CE to execute the job. The Match-Maker interrogates the BDII to query the status of CEs and SEs, and the RLS to find location of data. The event is logged in the LB and the status of the job is **WAITING**.

# Job Flow



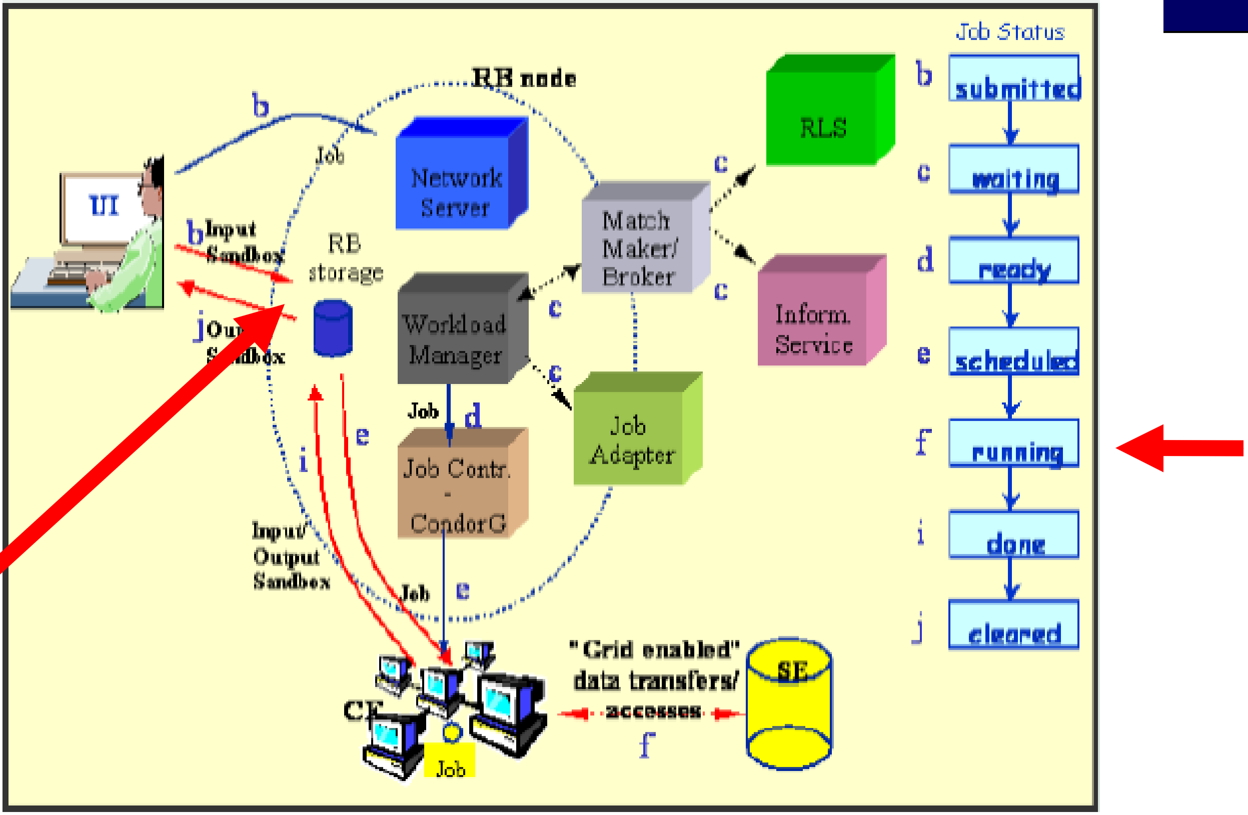
- d. The WMS Job Adapter prepares the job for submission creating a wrapper script that is passed, together with other parameters, to the JCS for submission to the selected CE. The event is logged in the LB and the status of the job is **READY**.

# Job Flow



- e. The **Globus Gatekeeper** on the CE receives the request and sends the Job for execution to the LRMS (e.g. PBS, LSF or Condor). The event is logged in the LB and the status of the job is **SCHEDULED**.

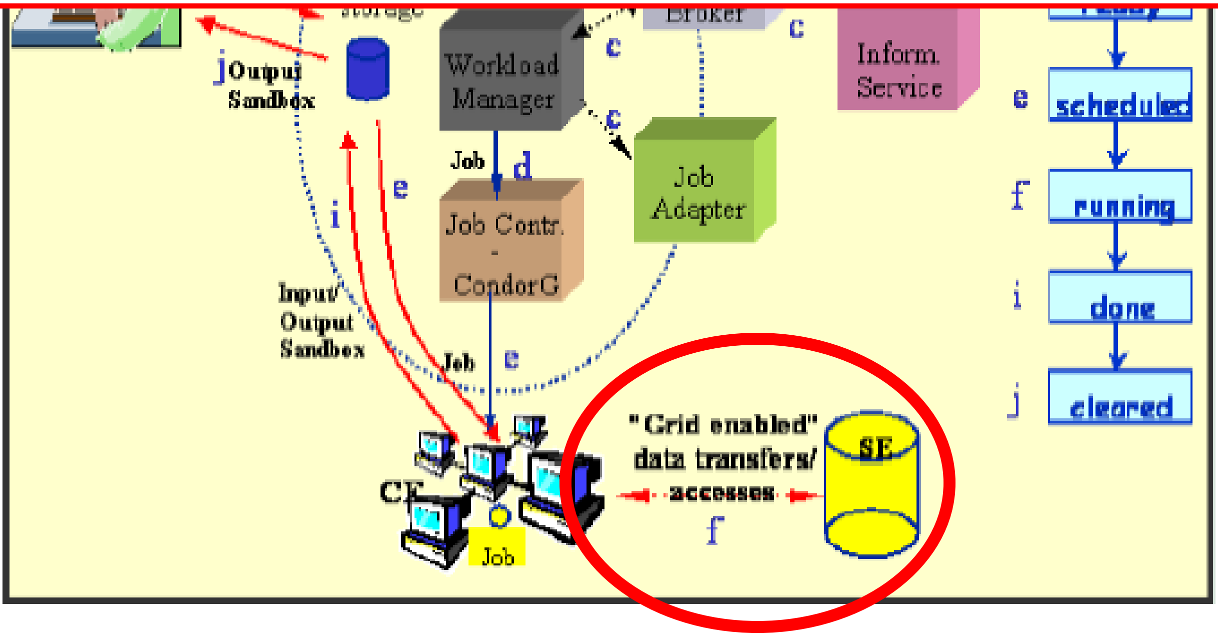
# Job Flow



- f. The LRMS handles the job execution on the available local farm worker nodes. User's files are copied from the RB to the WN where the job is executed. The event is logged in the LB and the status of the job is **RUNNING**.

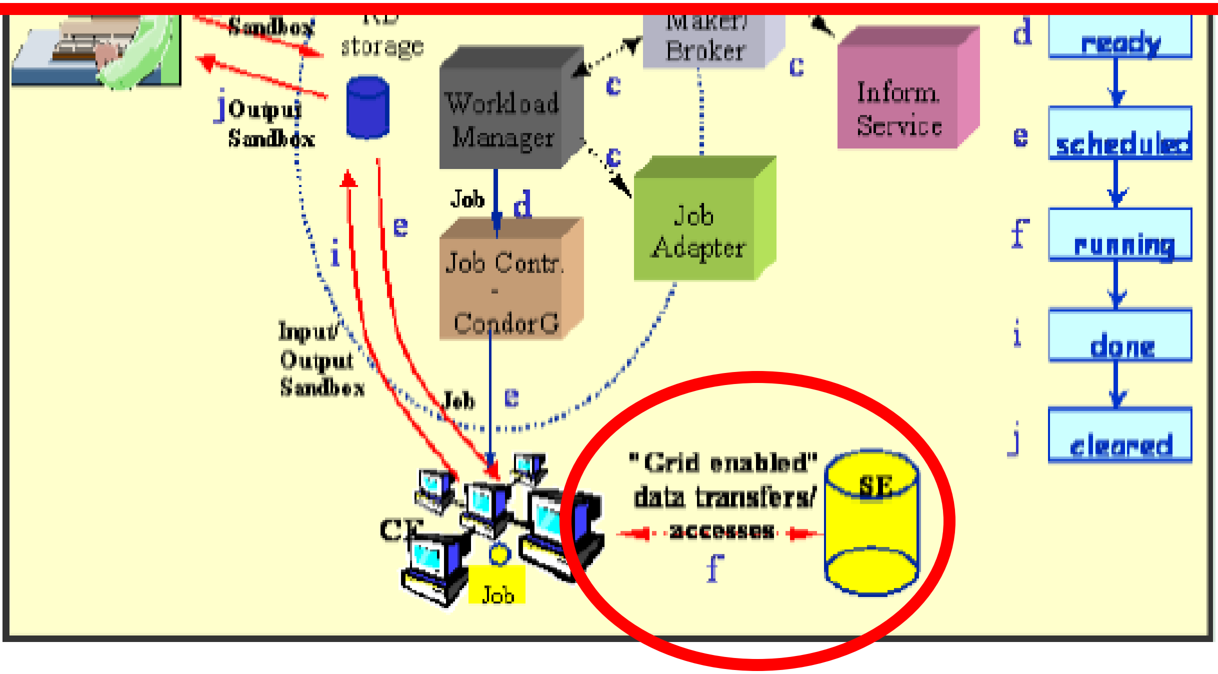
# Job Flow

- f. While the job runs, **Grid files can be accessed on a (close) SE** using either the RFIO protocol or local access if the files are copied to the WN local filesystem. In order for the job **to find out which is the close SE**, or what is the result of the **Match-Maker** process, **a file with this information is produced by the WMS** and shipped together with the job to the **WN**. This is known as the **.BrokerInfo file.** Information can be retrieved from this file using the BrokerInfo CLI or the API library.

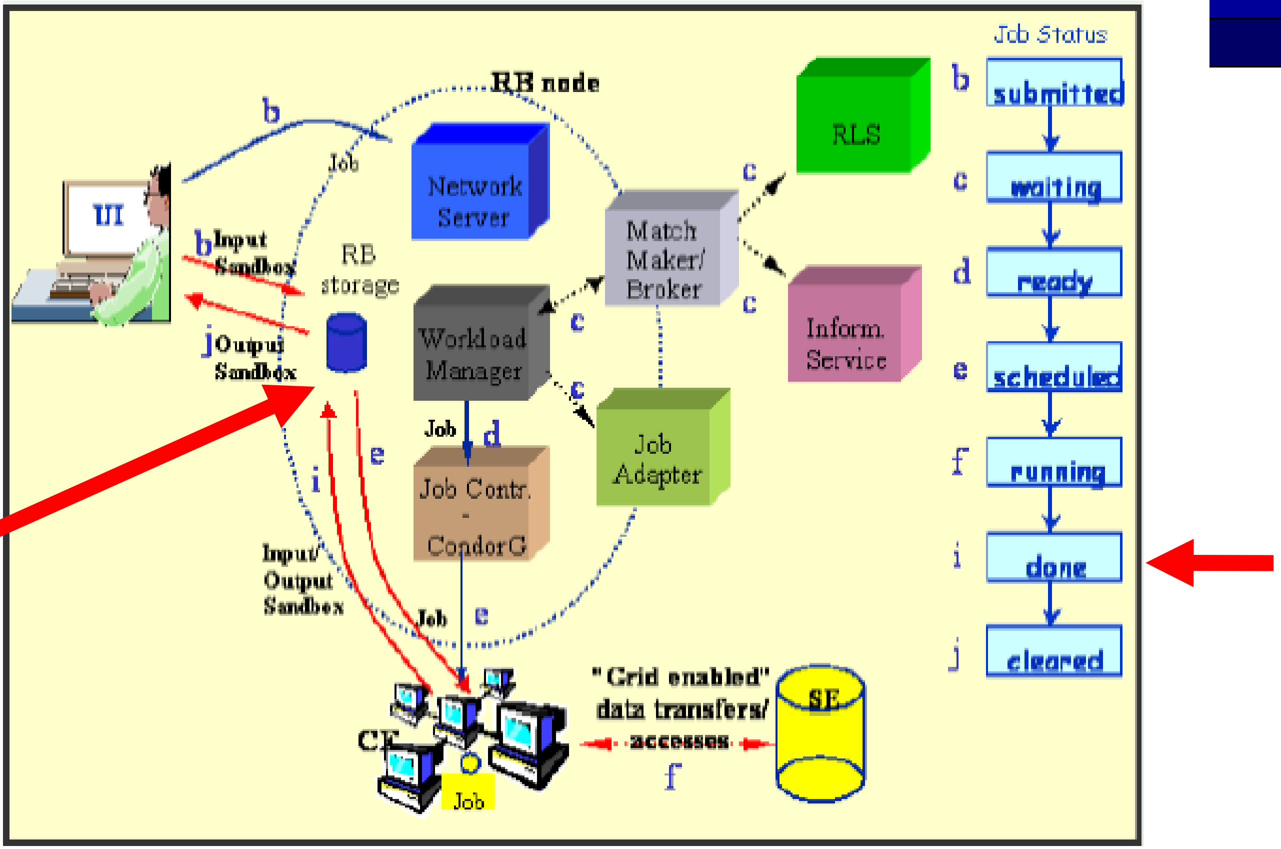


# Job Flow

- f. The **job can produce new output** data that can be uploaded to the Grid and made available for other Grid users to use. This can be achieved using the Data Management tools. **Uploading a file** to the Grid means
  - copying it on a Storage Element and
  - registering its location, metadata and attribute to the RMS.
- During job execution, data **files can be replicated** between two SEs using again the Data Management tools.

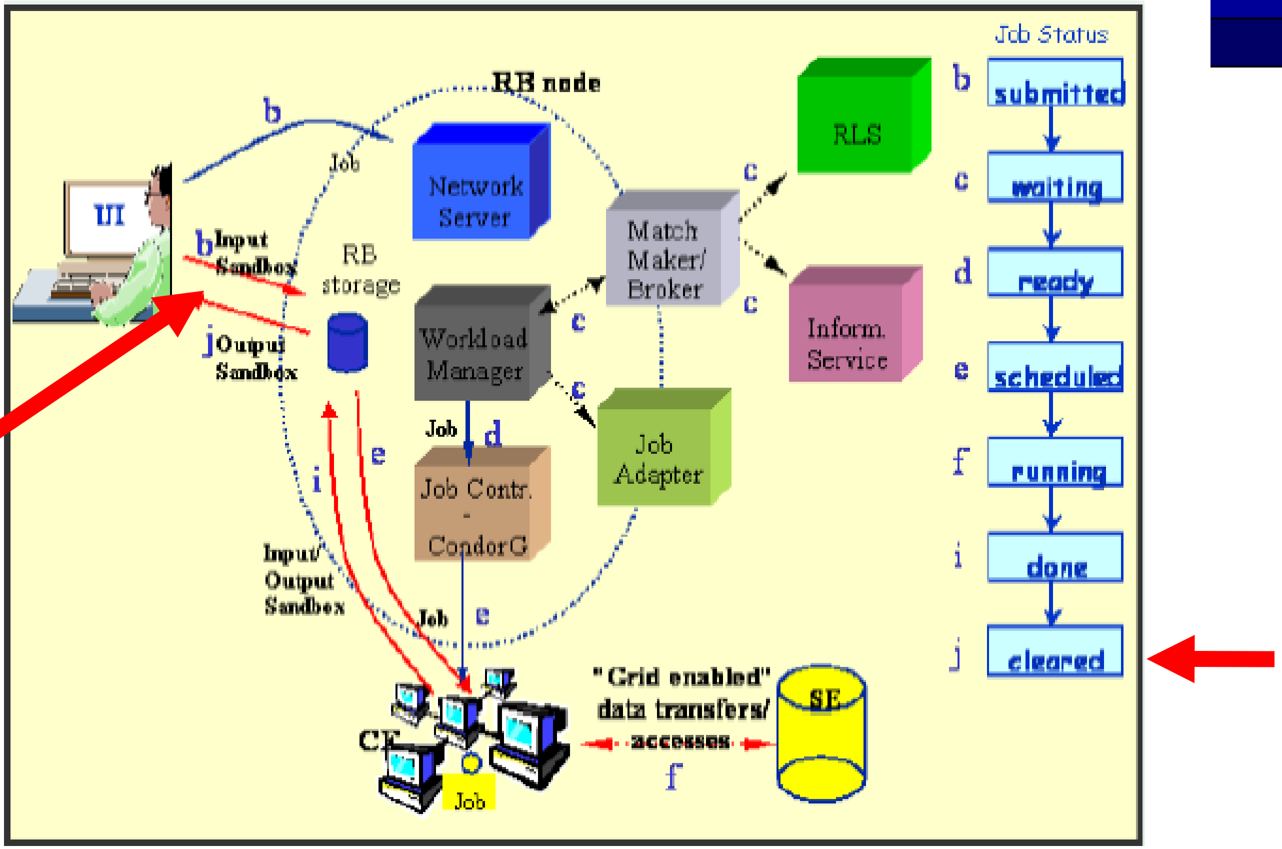


# Job Flow



- i. If the job reaches the end without errors, the output (not large data files, but **just small output files** specified by the user in the so called ***Output Sandbox***) is transferred back to the RB node. The event is logged in the LB and the status of the job is **DONE**.

# Job Flow



- j. At this point, the user can retrieve the output of his/her job from the UI using the WMS CLI or API. The event is logged in the LB and the status of the job is **CLEARED**.

# Job Flow Status and Errors

- Status queries from the UI machine:
  - **job status queries** are addressed to the LB database.
  - **Resource status queries** are addressed to the BDII
- **If the site** where the job is being run **falls down**, the job will be **automatically resent** to another CE that is analogue to the previous one, w.r.t. requirements the user asked for.
- In the case that this new submission is disabled, the job will be marked as **aborted**.
- Users can get information about what happened by simply questioning the LB service.

# How do I login on the Grid ?

- Distribution of resources: secure access is a basic requirement
  - secure communication
  - security across organisational boundaries
  - single "sign-on" for users of the Grid
- Two basic concepts:
  - Authentication: *Who am I?*

    - "Equivalent" to a pass port, ID card etc.

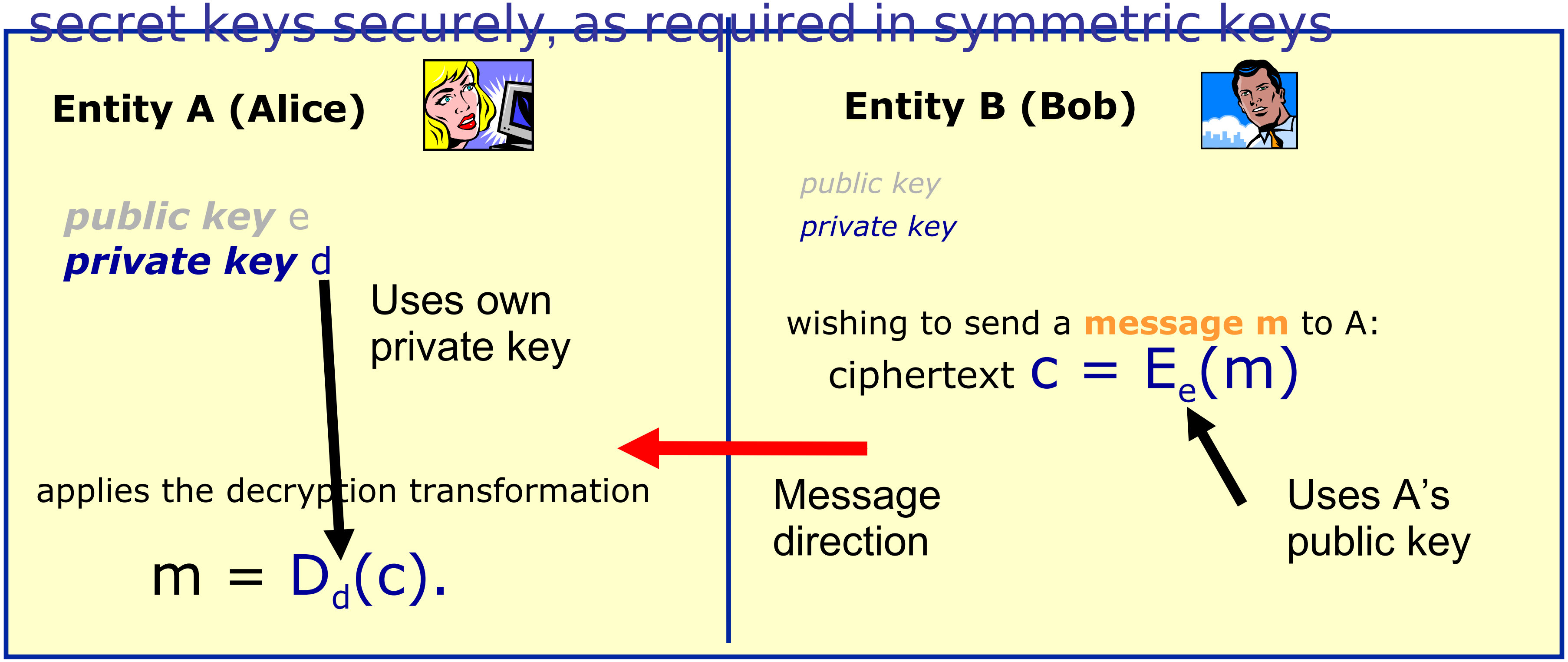
  - Authorisation: *What can I d*
    - Certain permissions, duties etc.

# Security in the Grid

- In industry, several security standards exist:
  - Public Key Infrastructure (PKI)
    - PKI keys
    - SPKI keys (focus on authorisation rather than certificates)
    - RSA
  - Secure Socket Layer (SSL)
    - SSH keys
  - Kerberos
- Need for a common security standard for Grid services
  - Above standards do not meet all Grid requirements (e.g. delegation, single sign-on etc.)
- Grid community mainly uses X.509 PKI for the Internet
  - Well established and widely used (also for www, e-mail, etc.)

# PKI – Basic overview

- Public Key Infrastructure (also called asymmetric cryptography)
- One primary advantage: it is generally easier than distributing secret keys securely, as required in symmetric keys



**Entity A (Alice)**

***public key*** e

***private key*** d

Uses own private key

applies the decryption transformation

$m = D_d(c)$.

**Entity B (Bob)**

*public key*

*private key*

wishing to send a **message m** to A:

ciphertext $c = E_e(m)$

Message direction

Uses A's public key

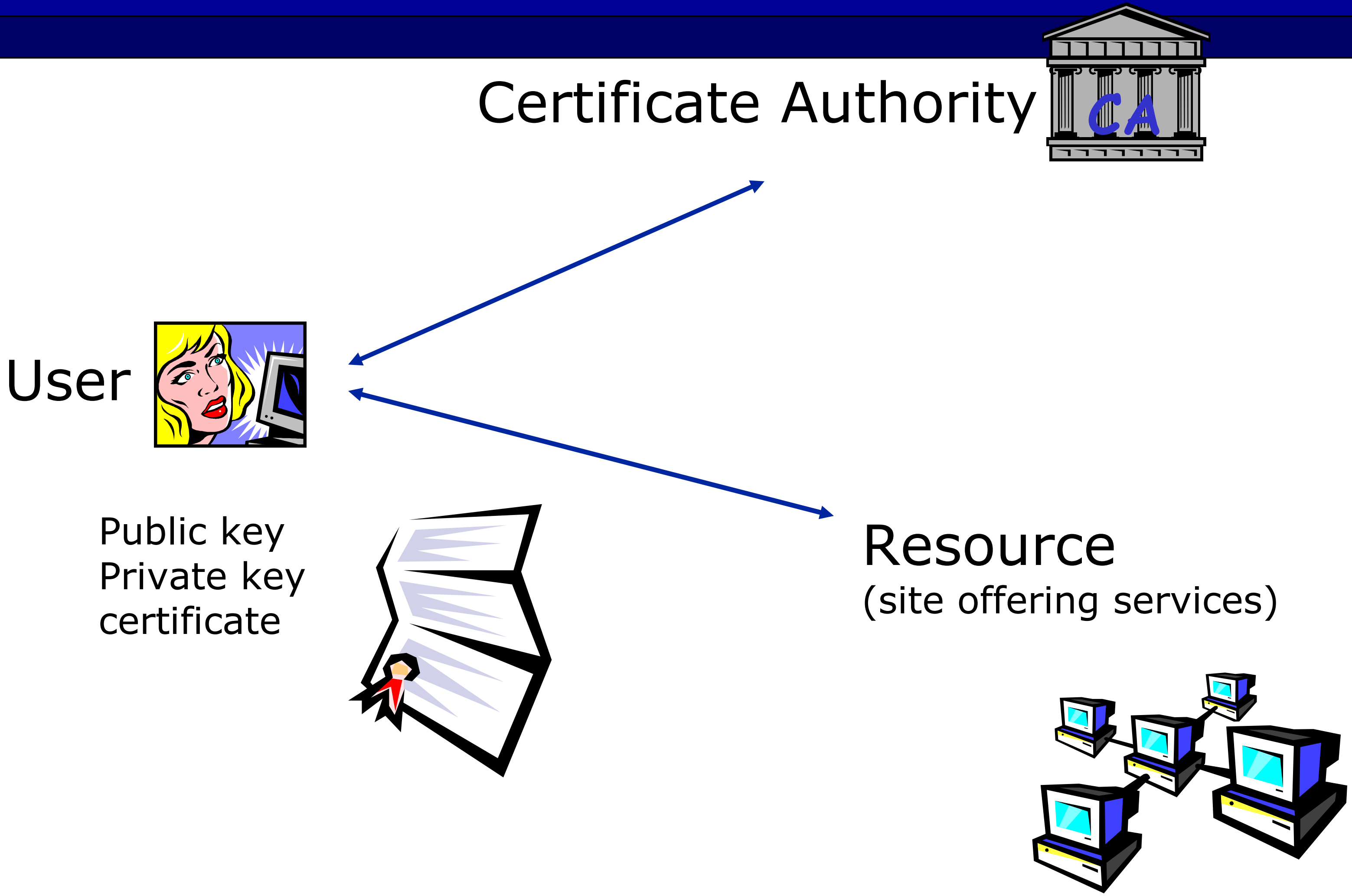
*encryption transformation* $E_e$

*decryption transformation* $D_d$

# Digital Certificates

- How can B be sure that A’s public key is really A’s public key and not someone else’s?
  - A ***third party*** guarantees the correspondence between public key and owner’s identity, by signing a document which contains the owner’s identity and his public key (**Digital Certificate**)
  - Both A and B must trust this third party
- Two models:
  - **X.509: hierarchical organization**;
  - PGP: “web of trust”.

# Involved entities



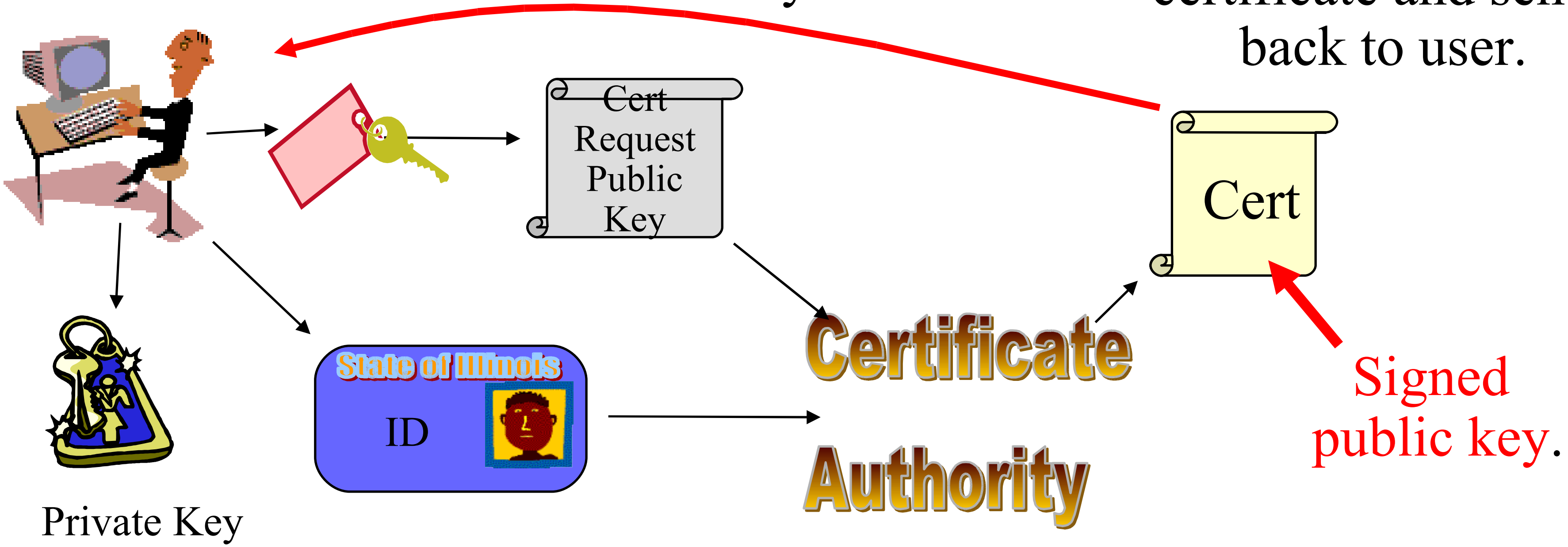
Certificate Authority

User

Public key
Private key
certificate

Resource
(site offering services)

# Certificate Request



User generates public/private key pair.

User send public key to CA along with proof of identity.

CA confirms identity, signs certificate and sends back to user.

Signed public key.

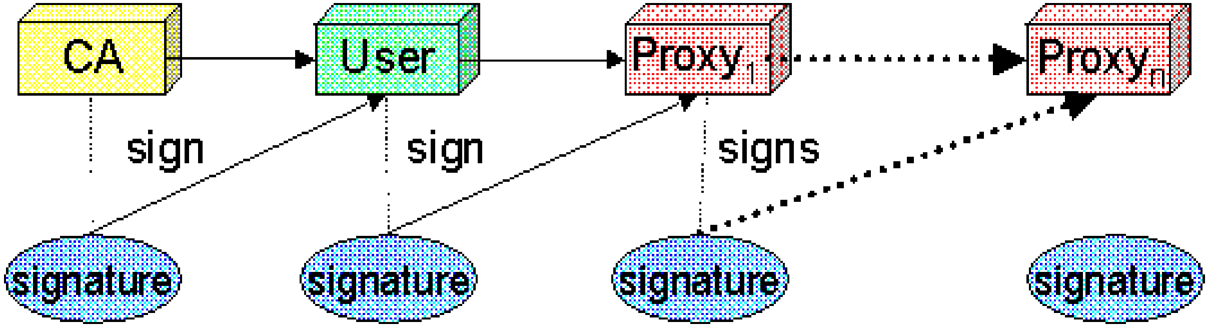
Private Key encrypted on local disk

# Grid Security Infrastructure (GSI)

- **Globus Toolkit™** proposed and implements the Grid Security Infrastructure (**GSI**)
  - Protocols and APIs to address Grid security needs
- GSI protocols extend standard public key protocols
  - Standards: X.509 & SSL/TLS
  - Extensions: **X.509 Proxy Certificates (single sign-on) & Delegation**
- **Proxy Certificate:**
  - Short term, restricted certificate that is derived form a long-term X.509 certificate
  - Signed by the normal end entity cert, or by another proxy
  - Allows a process to act on behalf of a user
  - Not encrypted and thus needs to be securely managed by file system

# Delegation

- Proxy creation can be recursive
  - each time a new private key and new X.509 proxy certificate, signed by the original key
- Allows remote process to act on behalf of the user
- Avoids sending passwords or private keys across the network
- The proxy may be a "Restricted Proxy": a proxy with a *reduced* set of privileges (e.g. cannot submit jobs).



# Resource Management Review

- Resource Specification Language (RSL) is used to communicate requirements
- The Grid Resource Allocation and Management (GRAM) API allows programs to be started on remote resources, despite local heterogeneity
- A layered architecture allows application-specific resource brokers and co-allocators (e.g. DUROC) to be defined in terms of GRAM services

# Resource Specification Language

- Much of the power of GRAM is in the RSL
- Common language for specifying job requests
  - GRAM service translates this common language into scheduler specific language
- GRAM service constrains RSL to a conjunction of (attribute=value) pairs
  - E.g. &(executable="/bin/ls")(arguments="-l")
- GRAM service understands a well defined set of attributes

# RSL Attributes For GRAM

- Type=Value
  - Value is one of “Job” or “DAG”
    - Job: a simple job
    - DAG: a Direct Acyclic Graph of dependent jobs. Although DAGs represent sets of jobs, they are described through a single JDL and can therefore be submitted all at once to the WMS. Upon submission of such kind of requests, the WMS, in addition to the ids of the single nodes, will provide the user with a collective identifier that will allow monitor and control of the whole set of jobs through a single handle (the DAG Id).

# RSL Attributes For GRAM

- jobType=value
  - Value is one of “MPICH”, “normal”, “Interactive”, “Checkpointable”, “”or “Partitionable”
    - Normal: A simple batch job, default option.
    - MPICH: Run the program using “mpirun -np <count>” (parallel jobs)
    - Interactive: a job whose standard streams are forwarded to the submitting client
    - Partitionable: a job which is composed by a set of independent steps/iterations, i.e. a set of independent sub-jobs, each one taking care of a step or of a sub-set of steps, and which can be executed in parallel
    - Checkpointable: a job able to save its state, so that the job execution can be suspended and resumed later, starting from the same point where it was first stopped

# RSL Attributes For GRAM

- Executable=string
  - Program to run
  - A file path (absolute or relative) or URL
- Directory=string
  - Directory in which to run (default is $HOME)
- Arguments="arg1 arg2 arg3…"
  - List of string arguments to program

# RSL Attributes For GRAM

- StdInput=string
  - stdin for program
  - A file path (absolute or relative) or URL
- StdOutput=string
  - stdout for program
  - A file path (absolute or relative) or URL
- StdError=string
  - stderr for program
  - A file path (absolute or relative) or URL

# RSL Attributes For GRAM

- InputSandBox=string
  - Every file to be sent to the RB during the job submission using edg-job-submit or glite-job-submit

  InputSandbox = {“test.sh",“input"};

- OutputSandBox=string
  - Output files to be retrieved with the command edg –job-get-output or glite-job-output

  OutputSandbox = {“std.out",“std.err“,”results”};

# RSL Attributes For GRAM

- Environment=string
  - Sets the environment. The argument is a list of values of type: “VAR_NAME=VAR_VALUE “

    Environment = {“JOB_LOG_FILE=/tmp/myjob.log”,

    “ORACLE_SID=edg_rdbms_1”,

    “JAVABIN=/usr/local/java”};
- VirtualOrganization = vo
  - Define the VO to which the job and the user belong
- Retrycount=n
  - Number of resubmissions in case of error

# RSL Attributes For GRAM

- NodeNumber = n
  - Number of WN to be user for parallel jobs
- ListenerPort = p
  - Port for communicating with the client for Interactive jobs

# RSL Attributes For GRAM

- Requirements= list
  - List of resource required to the CE (Glue schema for the CE) using ClassAd builtin-functions:
    - anyMatch()
    - whichMatch()
    - allMatch()
  - Service types:

http://forge.cnaf.infn.it/plugins/scmsvn/viewcvs.php/v_1_2/mapping/ldap/schema/openldap-2-1/Glue-CE.schema?rev=10&root=glueschema&view=markup

  - Glue.CE.Info.LRMS.Type: (OpenPBS, LSF, Condor, BQS, CondorG, Torque, PBSPro, SGE, NQE, fork, other)
  - Glue.SE.ControlProtocol.Type (SRM, org.edg.SE, classic, other)
  - Glue.SE.AccessProtocol.Type (gsiftp, nfs, afs, rfio, gsirfio, dcap, gsidcap, root, https, other)

# Requirements

```
Requirements = other.GlueCEInfoLRMSType == "PBS" &&
    other.GlueCEInfoTotalCPUs > 2 &&
      Member("IDL1.7",other.GlueHostApplicationSoftwareRunTimeEnv
      ironment);

Requirements = other.GlueCEPolicyMaxCPUTime >= 1500 &&
  other.GlueCEPolicyMaxWallClockTime >= 6000;

Requirements =
  Member(other.GlueHostApplicationSoftwareRunTimeEnvironm
  ent ,"ALICE-3.07.01");
```

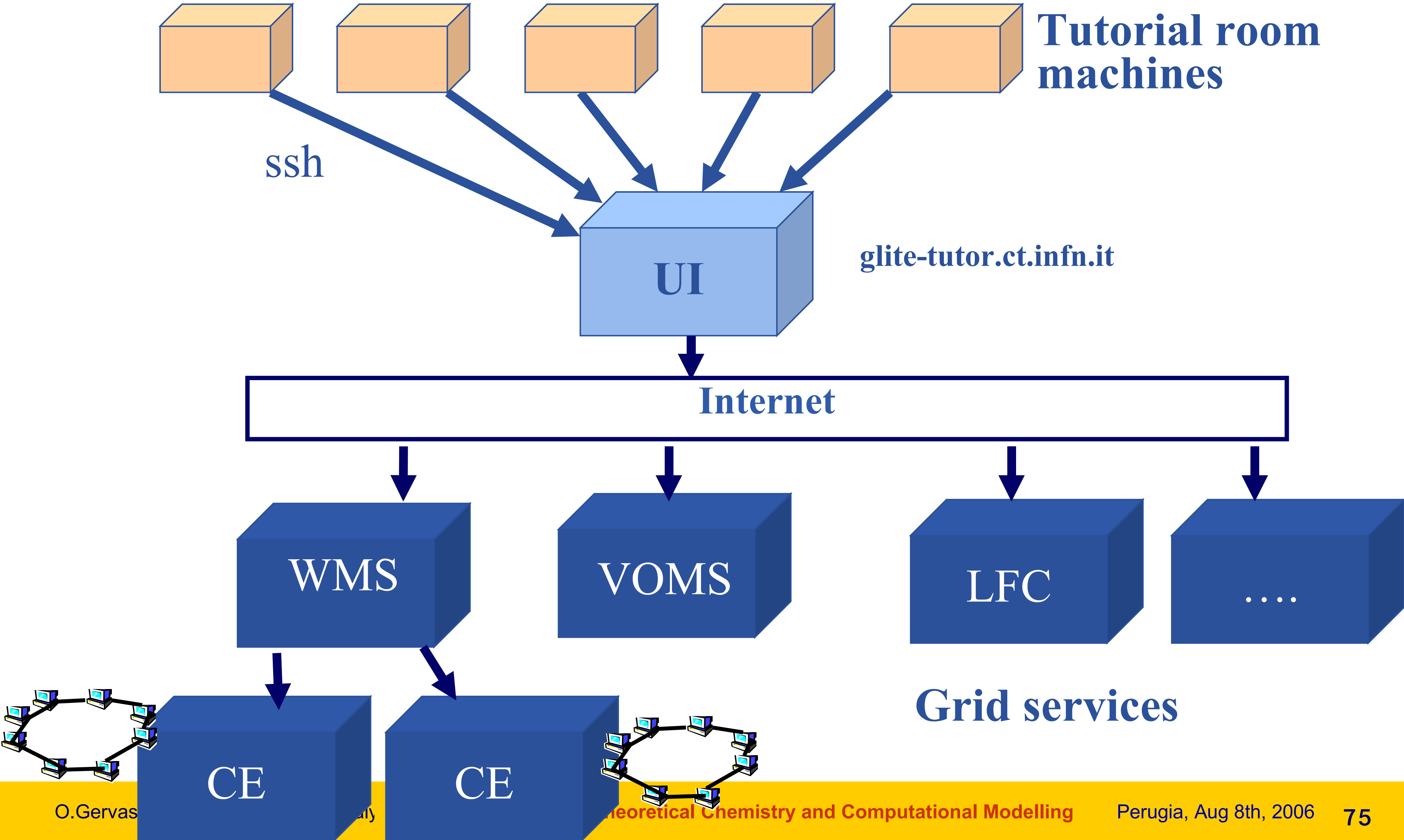
# Info for the tutorial session

- SSH to: glite-tutor.ct.infn.it or

  glite-tutor2.ct.infn.it
- User: toulouse01 → toulouse16
- Password: GridTOU01 → GridTOU16
- The pass phrase to access the X.509 digital certificate is TOULOUSE for all

```
ssh toulouse01@glite-tutor.ct.infn.it
```

# Our setup


Tutorial room machines
ssh
UI
glite-tutor.ct.infn.it
Internet
WMS
VOMS
LFC
….
CE
CE
Grid services

# proxy creation

```
voms-proxy-init --voms gilda
```

```
Your identity: /C=IT/O=GILDA/OU=Personal
  Certificate/L=TARTU/CN=TARTU14/Email=emidio.giorgio@ct
  .infn.it
Enter GRID pass phrase:
Creating temporary
  proxy .................................................
  Done
Contacting  voms.ct.infn.it:15001
  [/C=IT/O=GILDA/OU=Host/L=INFN
  Catania/CN=voms.ct.infn.it/Email=emidio.giorgio@ct.inf
  n.it] "gilda" Done
Creating
  proxy .................................................
  .. Done
Your proxy is valid until Thu Jun 29 00:08:12 2006
```

TOULOUSE

# Job example

```
Type = "job";
JobType = "normal";
VirtualOrganisation = "gilda";
Executable = "test.sh";
StdOutput = "test.out";
StdError = "test.err";
OutputSandbox = {
"test.err", "test.out"
};
RetryCount = 3;
InputSandbox = {
"/home/toulose01/test.sh"
};
```

Test.sh:

```
#!/bin/sh
/bin/hostname
```

# Session example

```
[simbex@ui simbex]$ glite-voms-proxy-info
Couldn't find a valid proxy.
[simbex@ui simbex]$ glite-voms-proxy-init -voms compchem
Your identity: /C=IT/O=INFN/OU=Personal Certificate/L=University of Perugia Dept Maths
    and CompSci/CN=Osvaldo Gervasi
Creating temporary proxy ............................... Done
Contacting  voms.cnaf.infn.it:15003 [/C=IT/O=INFN/OU=Host/L=CNAF/CN=voms.cnaf.infn.it]
    "compchem" Done
Creating proxy .................................. Done
Your proxy is valid until Tue Aug  8 00:05:43 2006
[simbex@ui simbex]$ glite-wms-job-submit test.jdl

Selected Virtual Organisation name (from proxy certificate extension): compchem
Connecting to host egee-rb-01.cnaf.infn.it, port 7772
Logging to host egee-rb-01.cnaf.infn.it, port 9002

****************************************************************************************

                       JOB SUBMIT OUTCOME
 The job has been successfully submitted to the Network Server.
 Use edg-job-status command to check job current status. Your job identifier (edg_jobId) is:

 - https://egee-rb-01.cnaf.infn.it:9000/JMyibHA5rqK9wsIQxH_TBQ
****************************************************************************************
```

# Session example (i)

```
[simbex@ui simbex]$ glite-wms-job-status
                    https://egee-rb-01.cnaf.infn.it:9000/YYWccOJwTmhTbn4vyjXgww
***************************************************************
BOOKKEEPING INFORMATION:
Status info for the Job : https://egee-rb-01.cnaf.infn.it:9000/YYWccOJwTmhTbn4vyjXgww
Current Status:     Ready
Status Reason:      unavailable
Destination:        ce.grid.unipg.it:2119/jobmanager-lcgpbs-short
reached on:         Mon Aug  7 13:04:02 2006
***************************************************************
[simbex@ui simbex]$ glite-wms-job-status
                    https://egee-rb-01.cnaf.infn.it:9000/YYWccOJwTmhTbn4vyjXgww
***************************************************************
BOOKKEEPING INFORMATION:
Status info for the Job : https://egee-rb-01.cnaf.infn.it:9000/YYWccOJwTmhTbn4vyjXgww
Current Status:     Done (Success)
Exit code:          0
Status Reason:      Job terminated successfully
Destination:        ce.grid.unipg.it:2119/jobmanager-lcgpbs-short
reached on:         Mon Aug  7 13:09:46 2006
***************************************************************
```

# Session example (ii)

```
[simbex@ui simbex]$ glite-wms-job-output
                    https://egee-rb-01.cnaf.infn.it:9000/YYWccOJwTmhTbn4vyjXgww
[simbex@ui simbex]$ Retrieving files from host: egee-rb-01.cnaf.infn.it ( for https://egee-
   rb-01.cnaf.infn.it:9000/YYWccOJwTmhTbn4vyjXgww )

***************************************************************************************
               JOB GET OUTPUT OUTCOME

 Output sandbox files for the job:
 - https://egee-rb-01.cnaf.infn.it:9000/YYWccOJwTmhTbn4vyjXgww
 have been successfully retrieved and stored in the directory:
 /tmp/jobOutput/simbex_YYWccOJwTmhTbn4vyjXgww

***************************************************************************************
```

# Session example (iii)

```
[simbex@ui simbex]$ edg-job-list-match  test.jdl
Selected Virtual Organisation name (from proxy certificate extension): compchem
Connecting to host egee-rb-01.cnaf.infn.it, port 7772
***********************************************************************

                   COMPUTING ELEMENT IDs LIST
 The following CE(s) matching your job requirements have been found:
              *CEId*
 ares02.cyf-kr.edu.pl:2119/jobmanager-lcgpbs-compchem
 ce.egee.man.poznan.pl:2119/jobmanager-lcgpbs-compchem
 ce.grid.unipg.it:2119/jobmanager-lcgpbs-infinite
 ce01.athena.hellasgrid.gr:2119/jobmanager-pbs-compchem
 ce01.isabella.grnet.gr:2119/jobmanager-pbs-compchem
 ce01.kallisto.hellasgrid.gr:2119/jobmanager-pbs-compchem
 ce01.marie.hellasgrid.gr:2119/jobmanager-pbs-compchem
 ce101.grid.ucy.ac.cy:2119/jobmanager-lcgpbs-compchem
 ce2.egee.cesga.es:2119/jobmanager-lcgpbs-compchem
 ce01.isabella.grnet.gr:2119/jobmanager-pbs-short
 gridba2.ba.infn.it:2119/jobmanager-lcgpbs-short
 grid0.fe.infn.it:2119/jobmanager-lcgpbs-grid
 grid002.ca.infn.it:2119/jobmanager-lcgpbs-grid
 gridce.sns.it:2119/jobmanager-lcgpbs-grid
 gridba2.ba.infn.it:2119/jobmanager-lcgpbs-long
 atlasce01.na.infn.it:2119/jobmanager-lcgpbs-grid
 gridit-ce-001.cnaf.infn.it:2119/jobmanager-lcgpbs-gridit
 griditce01.na.infn.it:2119/jobmanager-lcgpbs-grid
 gridba2.ba.infn.it:2119/jobmanager-lcgpbs-infinite
 spaci01.na.infn.it:2119/jobmanager-lcglsf-grid
```

# Useful commands

- LCG syntax (preferred, acts on LCG+Glite sites):
  - edg-job-list-match test.jdl
  - edg-job-submit -o list_of_jobs test.jdl
  - edg-job-status -i list_of_jobs (or the URL of the job)
  - edg-job-get-output -i list_of_jobs (or the URL of the job)
- Glite commands (acts only on Glite sites):
  - glite-job-list-match test.jdl
  - glite-job-submit -o list_of_jobs test.jdl
  - glite-job-status -i list_of_jobs (or the URL of the job)
  - glite-job-output -i list_of_jobs (or the URL of the job)

# AA : References

- VOMS on EGEE: User Guide available at http://glite.web.cern.ch/glite/documentation/default.asp
- VOMS
  - Available at http://infnforge.cnaf.infn.it/voms/
  - Alfieri, Cecchini, Ciaschini, Spataro, dell'Agnello, Fronher, Lorentey, From gridmap-file to VOMS: managing Authorization in a Grid environment
  - Vincenzo Ciaschini, A VOMS Attribute Certificate Profile for Authorization
- GSI
  - Available at www.globus.org
  - A Security Architecture for Computational Grids. I. Foster, C. Kesselman, G. Tsudik, S. Tuecke. *Proc. 5th ACM Conference on Computer and Communications Security Conference*, pp. 83-92, 1998.
  - A National-Scale Authentication Infrastructure. R. Butler, D. Engert, I. Foster, C. Kesselman, S. Tuecke, J. Volmer, V. Welch. *IEEE Computer*, 33(12):60-66, 2000.
- RFC
  - S.Farrell, R.Housley, An internet Attribute Certificate Profile for Authorization, RFC 3281

# WMS : References

WMS User's Guide

- https://edms.cern.ch/file/572489/1/EGEE-JRA1-TEC-572489-WMS-guide-v0-2.pdf

JDL Attributes Specification

- Fabrizio Pacini
- https://edms.cern.ch/file/555796/1/EGEE-JRA1-TEC-555796-JDL-Attributes-v0-8.pdf

# Summary

- LCG-2 is based on Glous Toolkit 2 and do not supports Web Services
- Glite is the new EGEE middleware, that enhance the security and the facilities of LCG-2 middleware.
- Glite is based on Web Services approach
- Glite Workload Management System is based on the Web Services Interface
- The main elements of the EGEE Grid architecture are: the Resource Broker (RB), the BDII server, the Computing Element (CE), the Storage Element (SE), the Worker Nodes (WN), the User Interface (UI).
- Glite uses the Relational Grid Monitoring Architecture (R-GMA) to monitor EGEE Grid