An overview of the EGEE project and middleware

Gergely Sipos

MTA SZTAKI

Hungarian Academy of Sciences

www.eu-egee.org

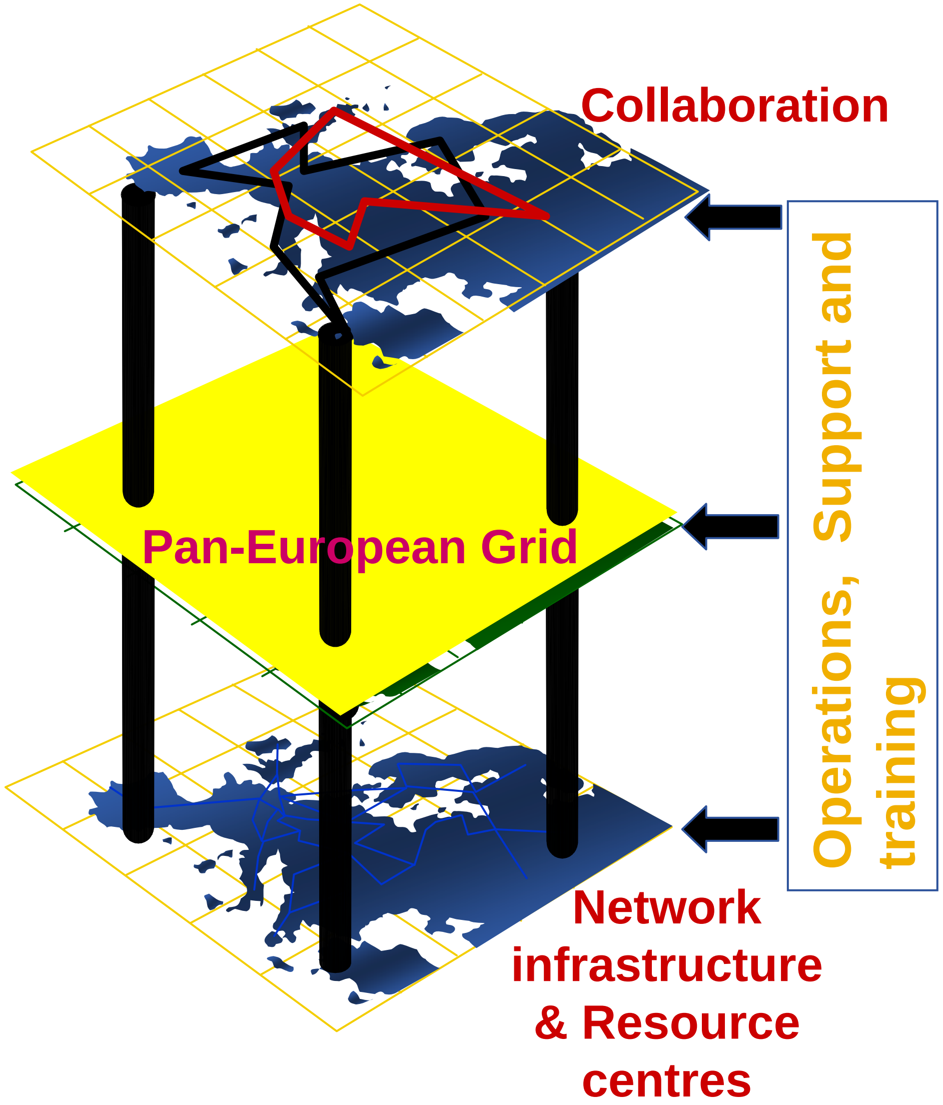
Contents

- What is EGEE?
- Overview of the main grid services

EGEE – international e-infrastructure

A four year programme:

- **Build, deploy and operate a consistent, robust a large scale production grid service that**
  - Links with and build on national, regional and international initiatives
- **Improve and maintain the middleware in order to deliver a reliable service to users**
- **Attract new users from research and industry and ensure training and support for them**

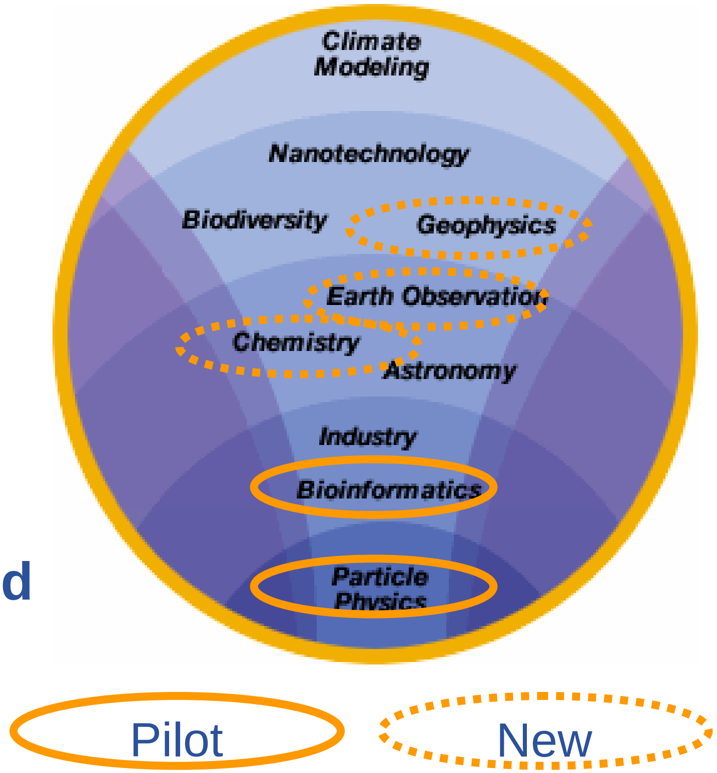

Collaboration
Pan-European Grid
Network infrastructure & Resource centres
Operations, Support and training

In the first 2 years EGEE

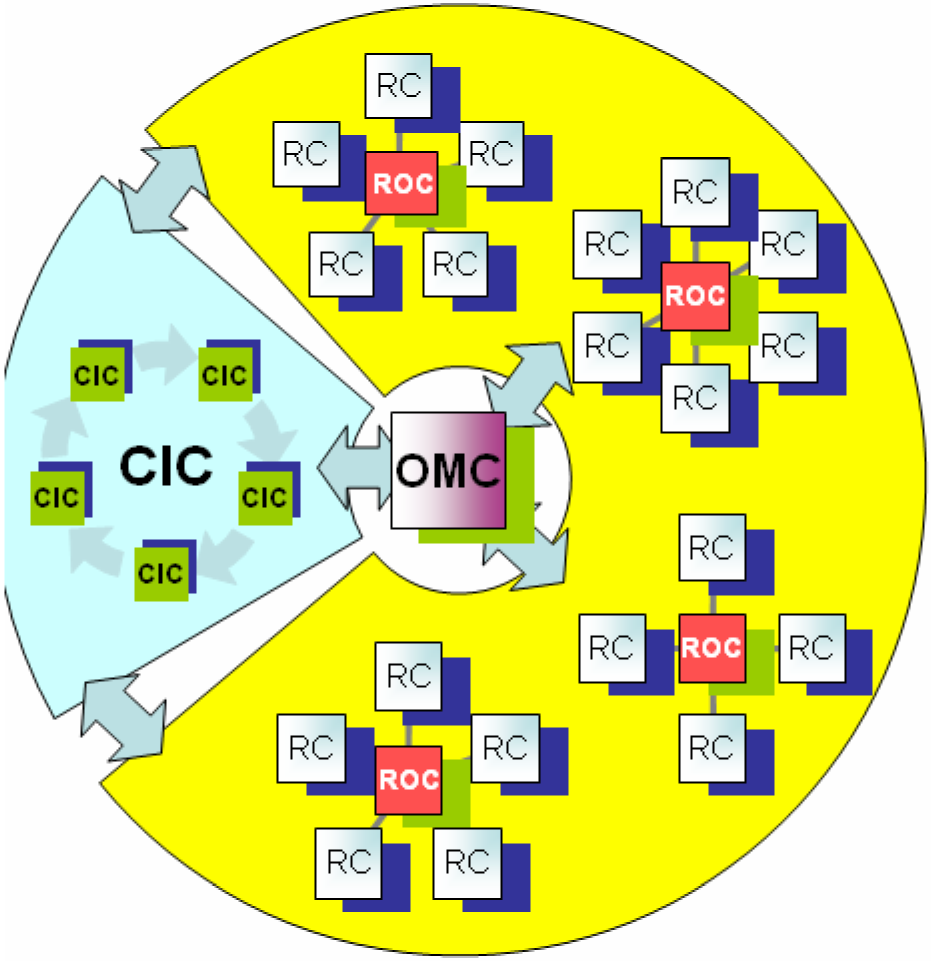
- **Established production quality sustained Grid services**
  - 3000 users from at least 5 disciplines
  - Goal was to integrate 50 sites into a common infrastructure → currently 180
  - offer 5 Petabytes (10^{15}) storage

- **Demonstrated a viable general process to bring other scientific communities on board**

- **Secured a second phase from April 2006**



Grid Operations



RC = Resource Centre
ROC = Regional Operations Centre
CIC = Core Infrastructure Centre
OMC = Operations Management Centre

- **CICs act as a single Operations Centre**
  - Operational oversight *(grid operator)* responsibility
  - rotates weekly between CICs
  - Report problems to ROC/RC
  - ROC is *responsible* for ensuring problem is resolved
  - ROC oversees regional RCs
- **ROCs responsible for organising the operations in a region**
  - Coordinate deployment of middleware, etc
- **CERN coordinates sites not associated with a ROC**
- **Global Grid User Support**

EGEE-II

- **Natural continuation of EGEE**
  - Expanded consortium
  - Emphasis on providing an infrastructure
    - → increased support for applications
    - → interoperate with other infrastructures
    - → more involvement from Industry

SA: service activities

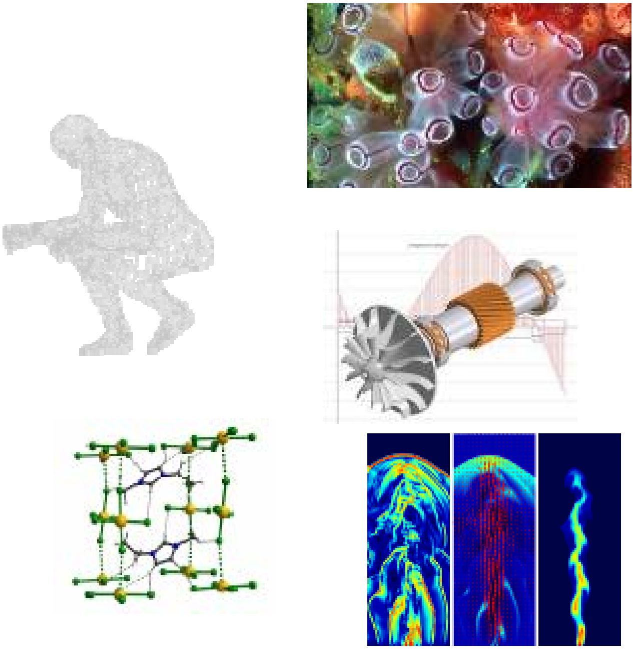
- establishing operations

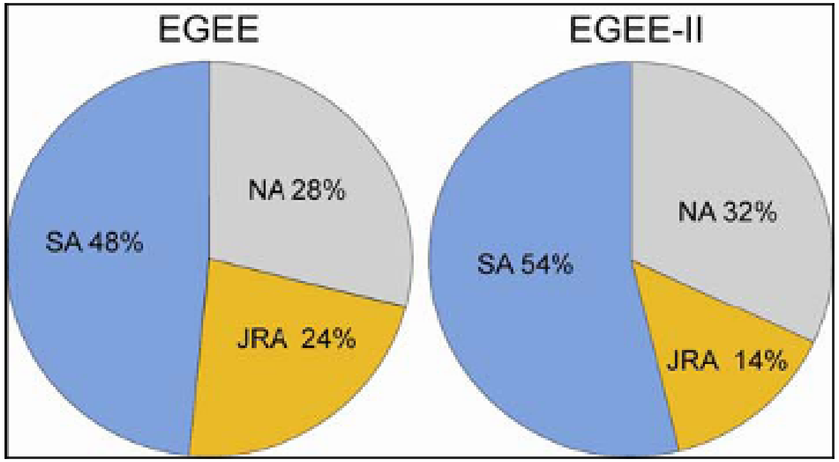
NA: network activities

- supporting VOs

JRA: “joint research activities”

- e.g. hardening middleware

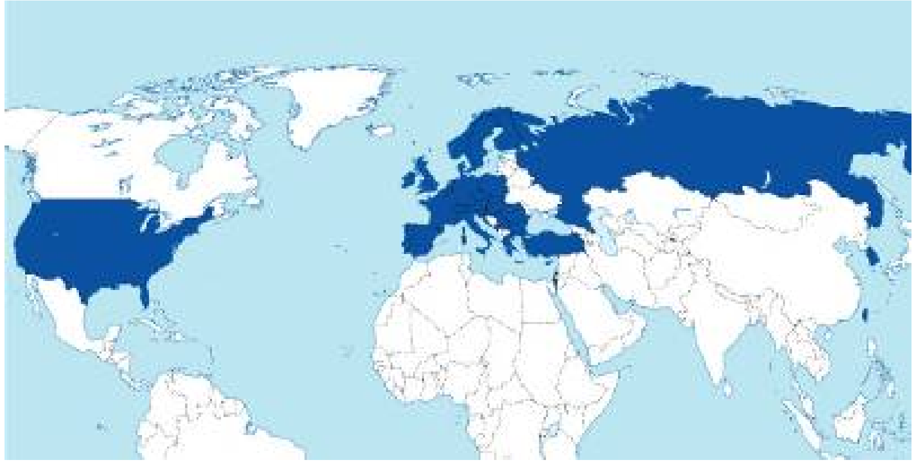




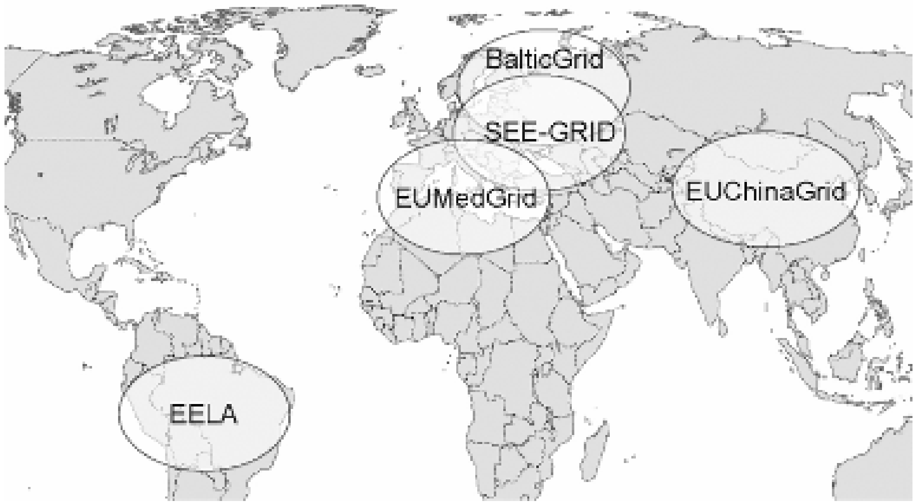
EGEE-II: Expertise & Resources

- More than 90 partners
- 32 countries
- 12 federations

→ Major and national Grid projects in Europe, USA, Asia

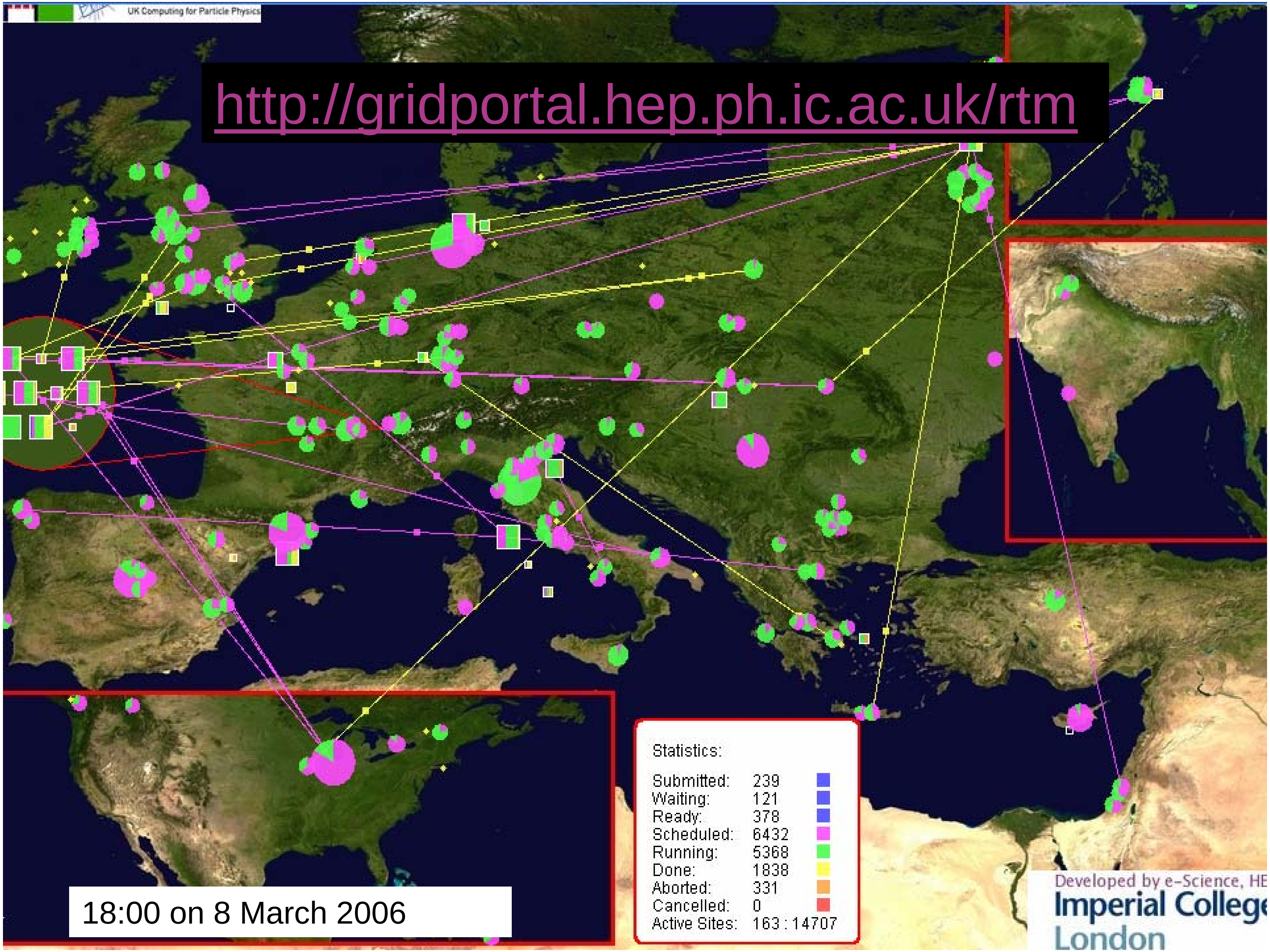


+ 27 countries through related projects:
  - BalticGrid
  - SEE-GRID
  - EUMedGrid
  - EUChinaGrid
  - EELA



Related projects: infrastructure, education, application

Name	*Description*
BalticGrid	EGEE extension to Estonia, Latvia, Lithuania
EELA	EGEE extension to Brazil, Chile, Cuba, Mexico, Argentina
EUChinaGRID	EGEE extension to China
EUMedGRID	EGEE extension to Malta, Algeria, Morocco, Egypt, Syria, Tunisia, Turkey
ISSeG	Site security
eIRGSP	Policies
ETICS	Repository, Testing
BELIEF	Digital Library of Grid documentation, organisation of workshops, conferences
BIOINFOGRID	Biomedical
Health-e-Child	Biomedical – Integration of heterogeneous biomedical information for improved healthcare
ICEAGE	International Collaboration to Extend and Advance Grid Education



http://gridportal.hep.ph.ic.ac.uk/rtm

```
Statistics:

Submitted:     239       🟦
Waiting:       121       🟦
Ready:         378       🟦
Scheduled:     6432      🟪
Running:       5368      🟩
Done:          1838      🟨
Aborted:       331       🟧
Cancelled:     0         🟥
Active Sites:  163 : 14707
```

18:00 on 8 March 2006

Grid services

How can EGEE middleware support collaboration and resource sharing within and between many diverse VO’s ?

Grid Middleware

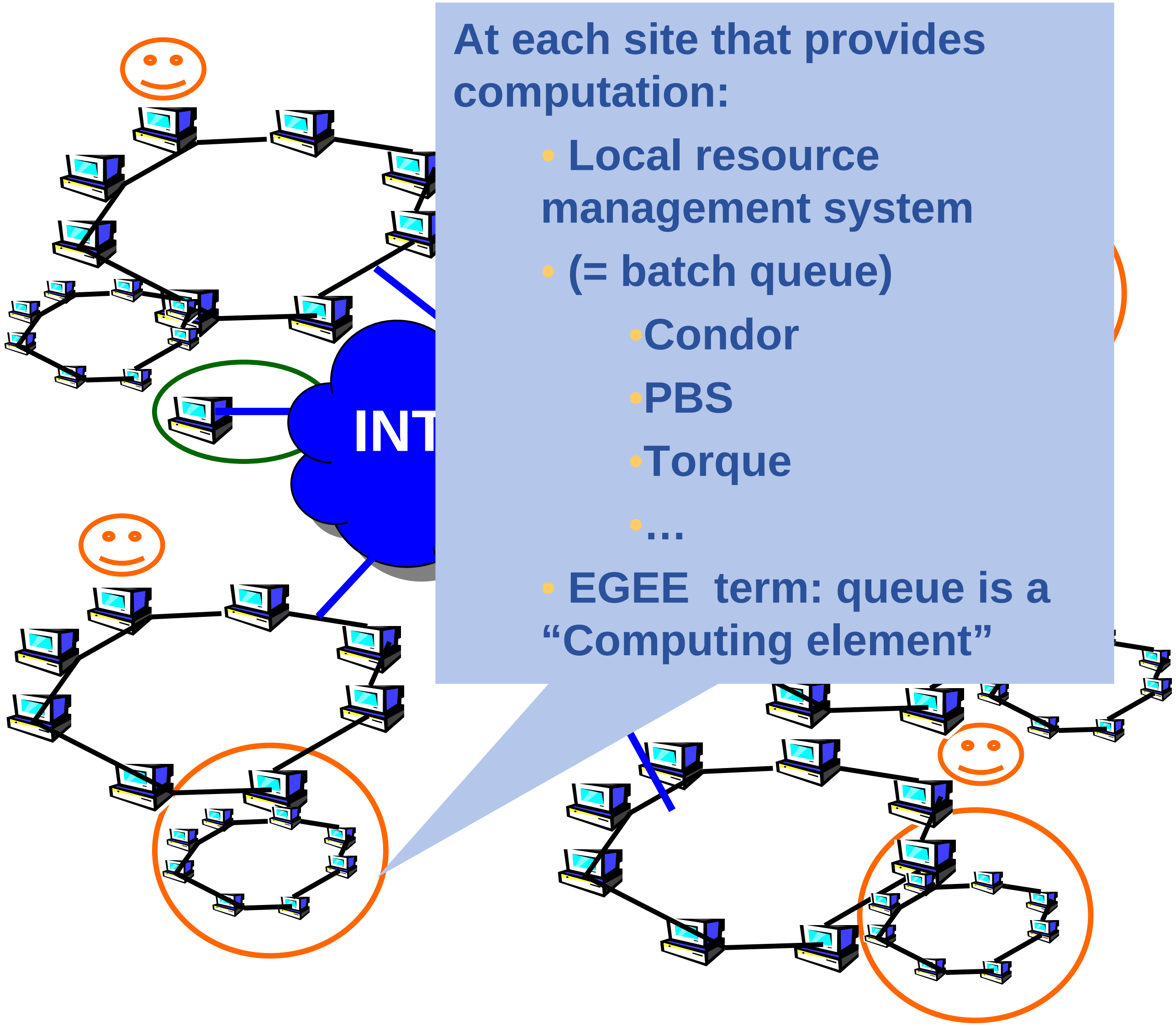
- **When using a PC or workstation you**
  - Login with a username and password (“Authentication”)
  - Use rights given to you (“Authorisation”)
  - Run jobs
  - Manage files: create them, read/write, list directories
- **Components are linked by a bus**
- **Operating system**
- **One admin domain**

- **When using a Grid you**
  - Login with digital credentials (“Authentication”)
  - Use rights given you (“Authorisation”)
  - Run jobs
  - Manage files: create them, read/write, list directories
- **Services are linked by the Internet**
- **Middleware**
- **Many admin domains**

Typical current grid

- Grid middleware runs on each shared resource
  - Data storage
  - (Usually) batch queues on pools of processors
- Users join VO's
- Virtual organisation negotiates with sites to agree access to resources

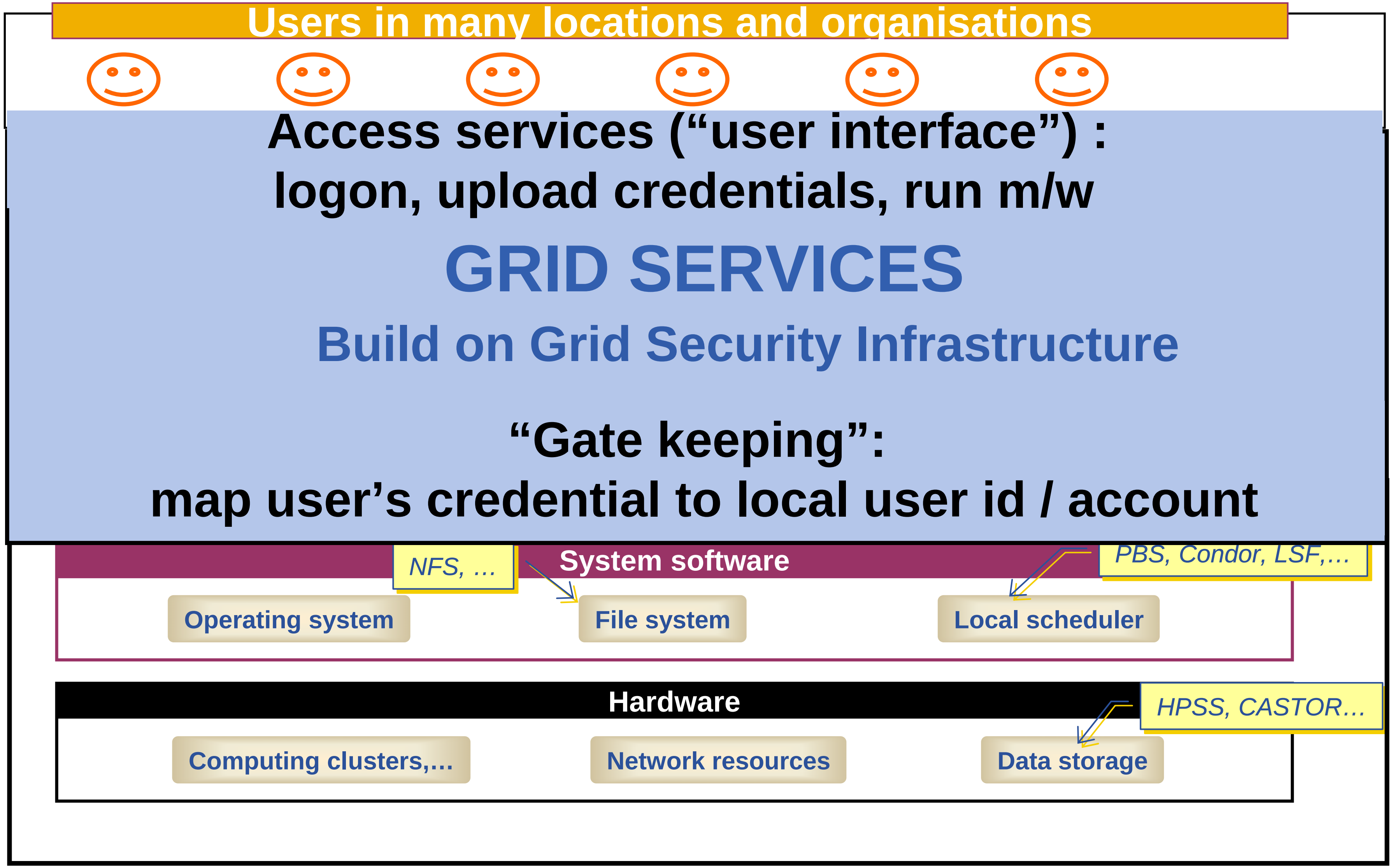
- Distributed services (both people and middleware) enable the grid, allow single sign-on



At each site that provides computation:

- Local resource management system
- (= batch queue)
  - Condor
  - PBS
  - Torque
  - …
- EGEE term: queue is a "Computing element"

Authorisation, Authentication (AA)



Users in many locations and organisations

Access services (“user interface”) :
logon, upload credentials, run m/w

GRID SERVICES

Build on Grid Security Infrastructure

“Gate keeping”:
map user’s credential to local user id / account

System software

NFS, …

PBS, Condor, LSF,…

Operating system

File system

Local scheduler

Hardware

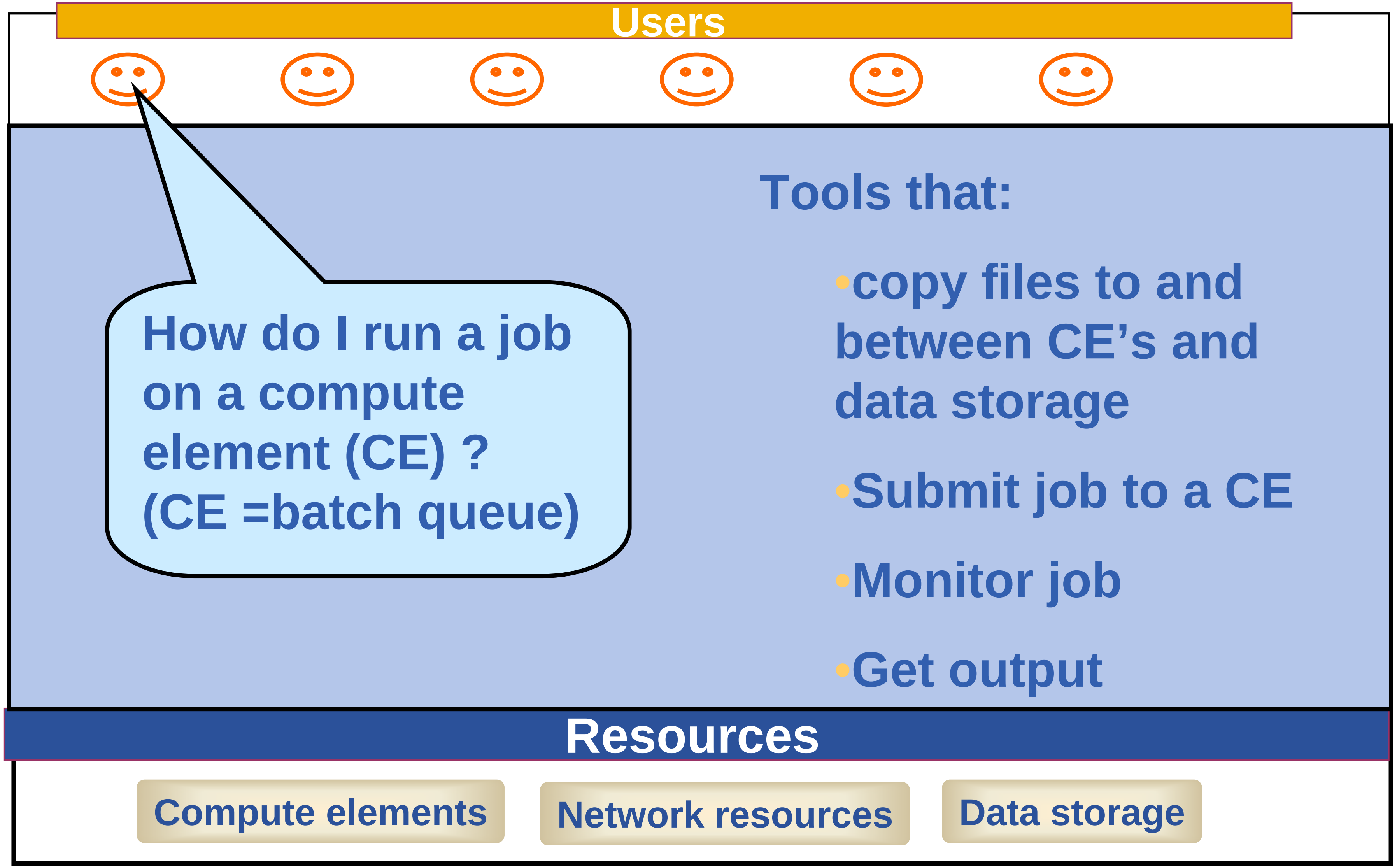
HPSS, CASTOR…

Computing clusters,…

Network resources

Data storage

Basic job submission



Users

How do I run a job on a compute element (CE) ? (CE =batch queue)

Tools that:

- copy files to and between CE’s and data storage
- Submit job to a CE
- Monitor job
- Get output

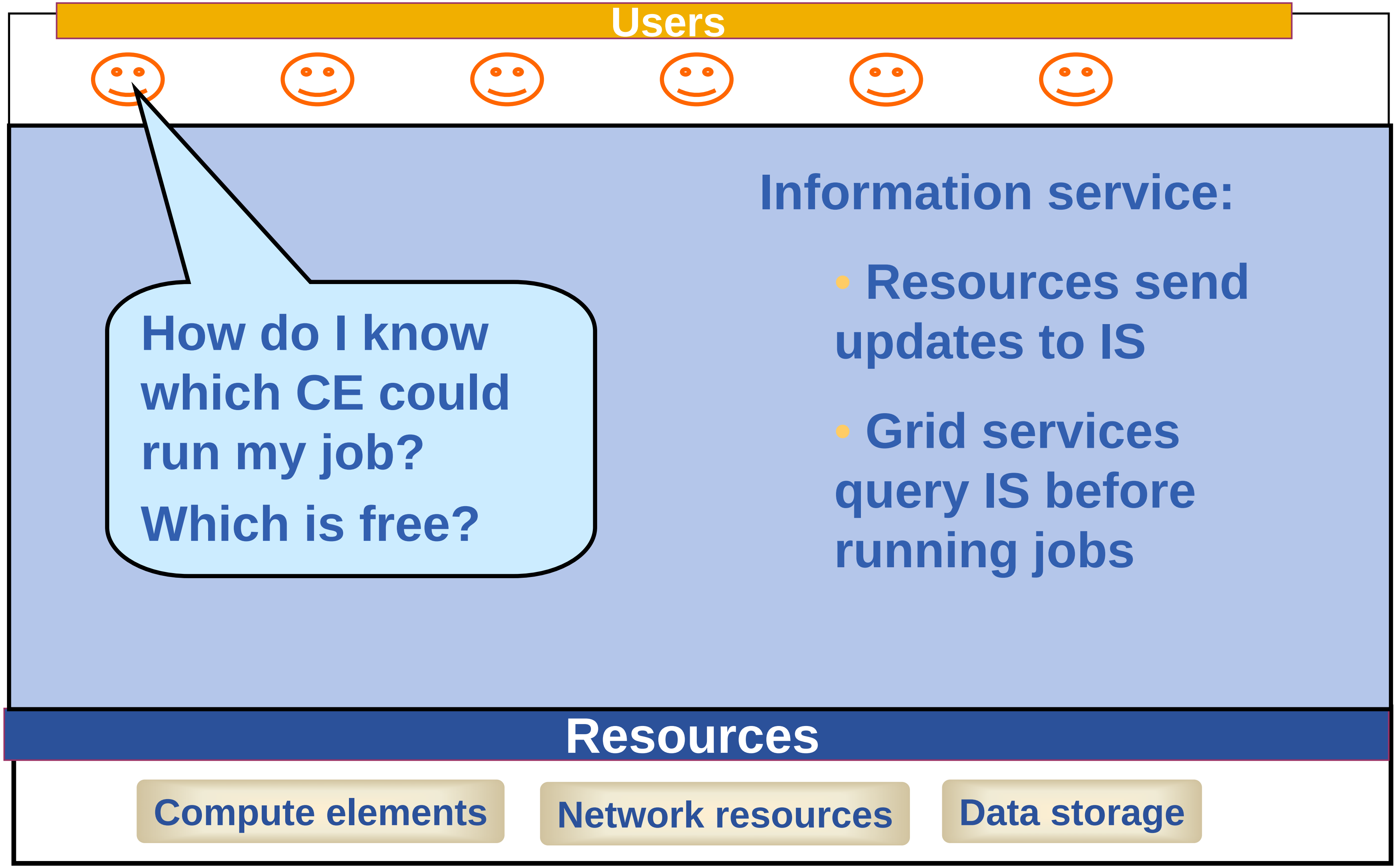
Resources

Compute elements

Network resources

Data storage

Information service (IS)



Users

How do I know which CE could run my job?

Which is free?

Information service:

- Resources send updates to IS
- Grid services query IS before running jobs

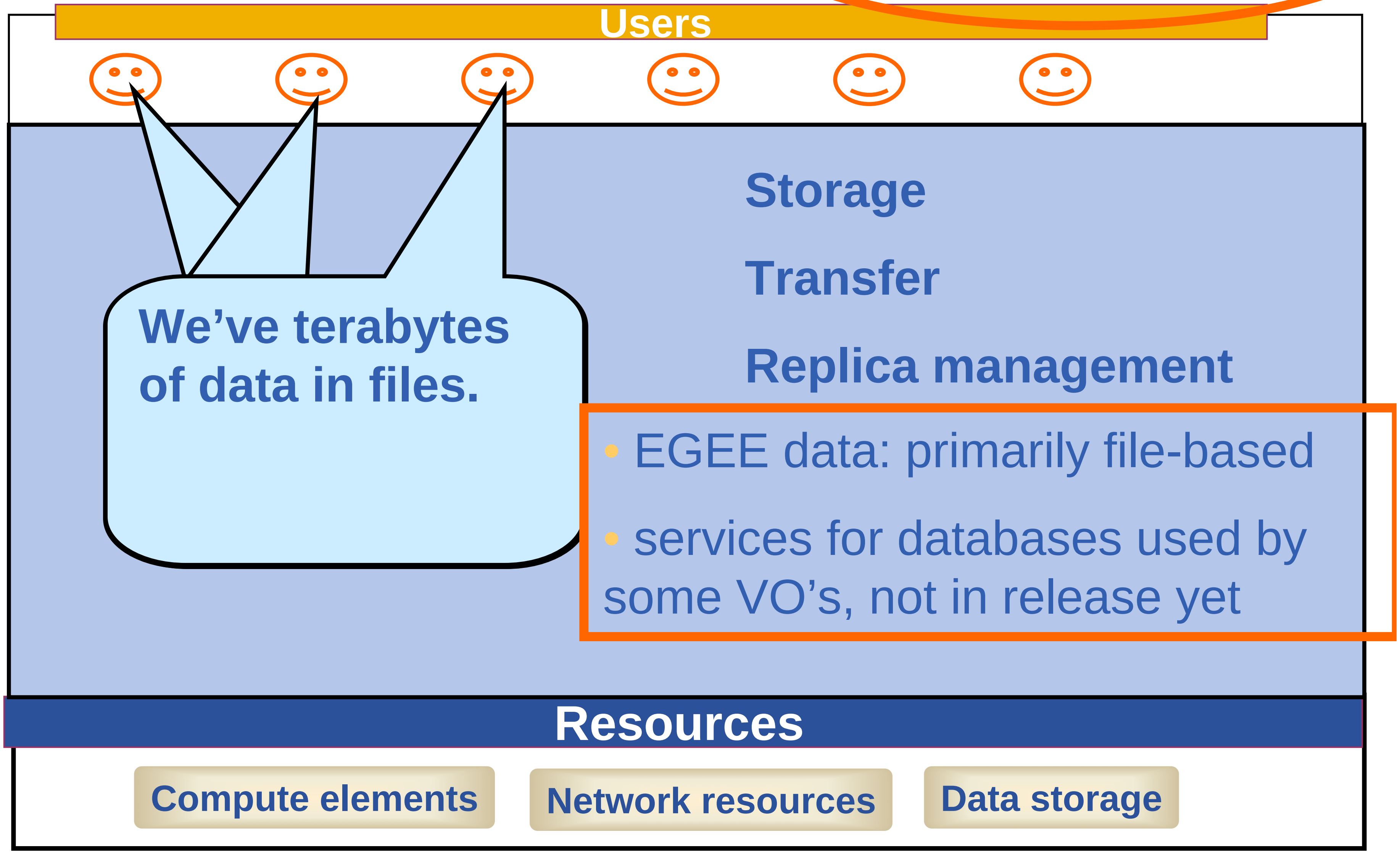
Resources

Compute elements

Network resources

Data storage

File management



Users

We've terabytes of data in files.

Storage

Transfer

Replica management

- EGEE data: primarily file-based
- services for databases used by some VO's, not in release yet

Resources

Compute elements

Network resources

Data storage

Globus Toolkit

- **A software toolkit: a modular "bag of technologies"**
  - Made available under liberal open source license
- ***Not* turnkey solutions, but *building blocks* and *tools* for application developers and system integrators**
- **Tools built on Grid Security Infrastructure to include:**
  - Job submission: run a job on a specific remote compute element
  - Information services: So I know which computer to use
  - File transfer: so large data files can be transferred
    - GridFTP: supports multiple channels for one transfer

- **(Most) production grids are (currently) based on the Globus Toolkit release 2**
- **Globus Alliance: http://www.globus.org/**

Running a job with GT2

- GT2 ***Toolkit***
- **An example of the command line interface:**
  - Job submission – need to know name of a CE to use

```
globus-job-submit grid-data.rl.ac.uk/jobmanager-pbs /bin/hostname -f

https://grid-data.rl.ac.uk:64001/1415/1110129853/

globus-job-status https://grid-data.rl.ac.uk:64001/1415/1110129853/

DONE

globus-job-get-output https://grid-data.rl.ac.uk:64001/1415/1110129853/

grid-data12.rl.ac.uk
```

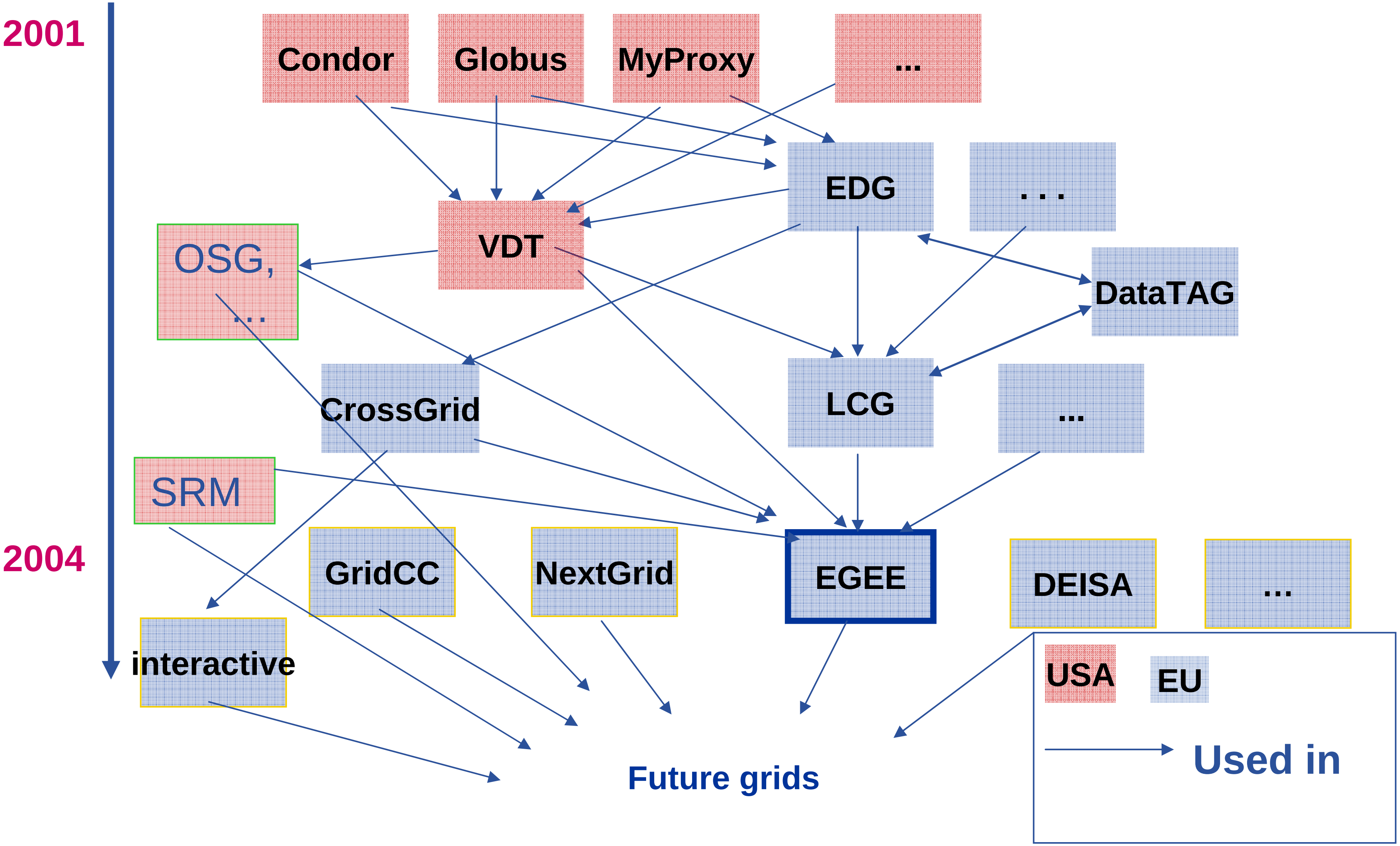
Building a grid

- GT2: a toolkit – not a turnkey solution

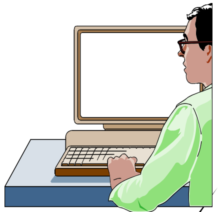
- Need higher level tools including:
  - **Job submission** to “a grid” not a CE
  - **Data management**
  - **Logging** who’s done what, statistics about jobs,…
  - **Monitoring** what’s happening on the grid

- EGEE middleware comprises more than GT2 !

Parts of the Grid "ecosystem"



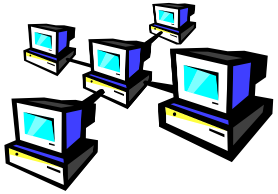
Main components



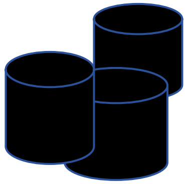
User Interface (UI): The place where users logon to the Grid

Resource Broker (RB): Matches the user requirements with the available resources on the Grid

Information System: Characteristics and status of CE and SE (Uses “GLUE schema”)

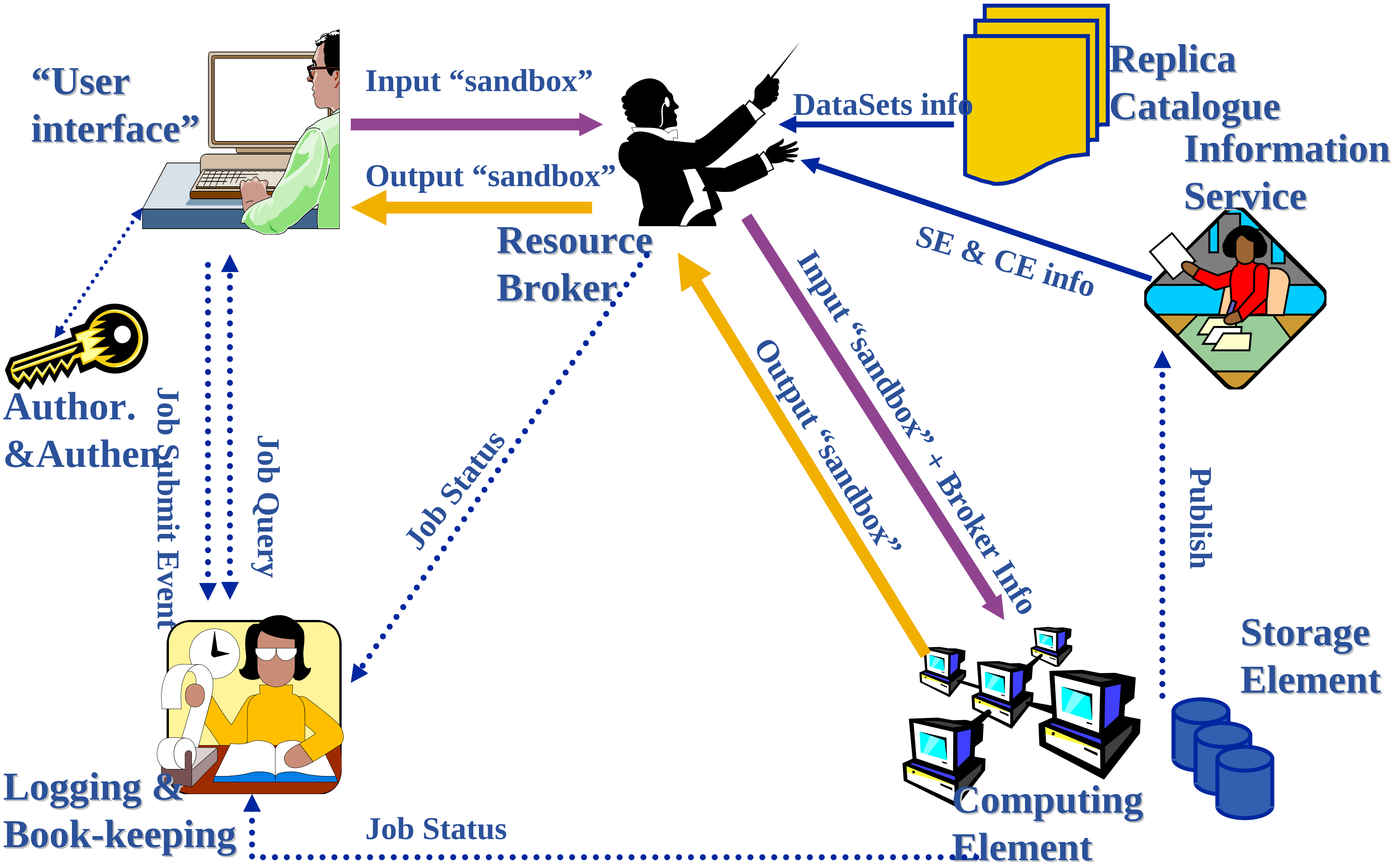


Computing Element (CE): A batch queue on a site’s computers where the user’s job is executed



Storage Element (SE): provides (large-scale) storage for files

Current production middleware



Building on basic tools and Information Service

- Submit job to grid via the “resource broker”,
- edg_job_submit *my.jdl*

Example JDL file

```
Executable = “gridTest”;
StdError = “stderr.log”;
StdOutput = “stdout.log”;
InputSandbox = {“/home/joda/test/gridTest”};
OutputSandbox = {“stderr.log”, “stdout.log”};
…
```

A closer look at the main EGEE grid services

1. Security, Authentication and Authorisation

How does EGEE…

- **How does EGEE build dynamic distributed systems?**
  - For many international collaborations (“virtual organisations”)
  - With n,000 processors in hundreds of independent sites (“administrative domains”)
  - With no prior direct relationship between users and resource providers
  - In a world where public networks are abused by hackers, etc.

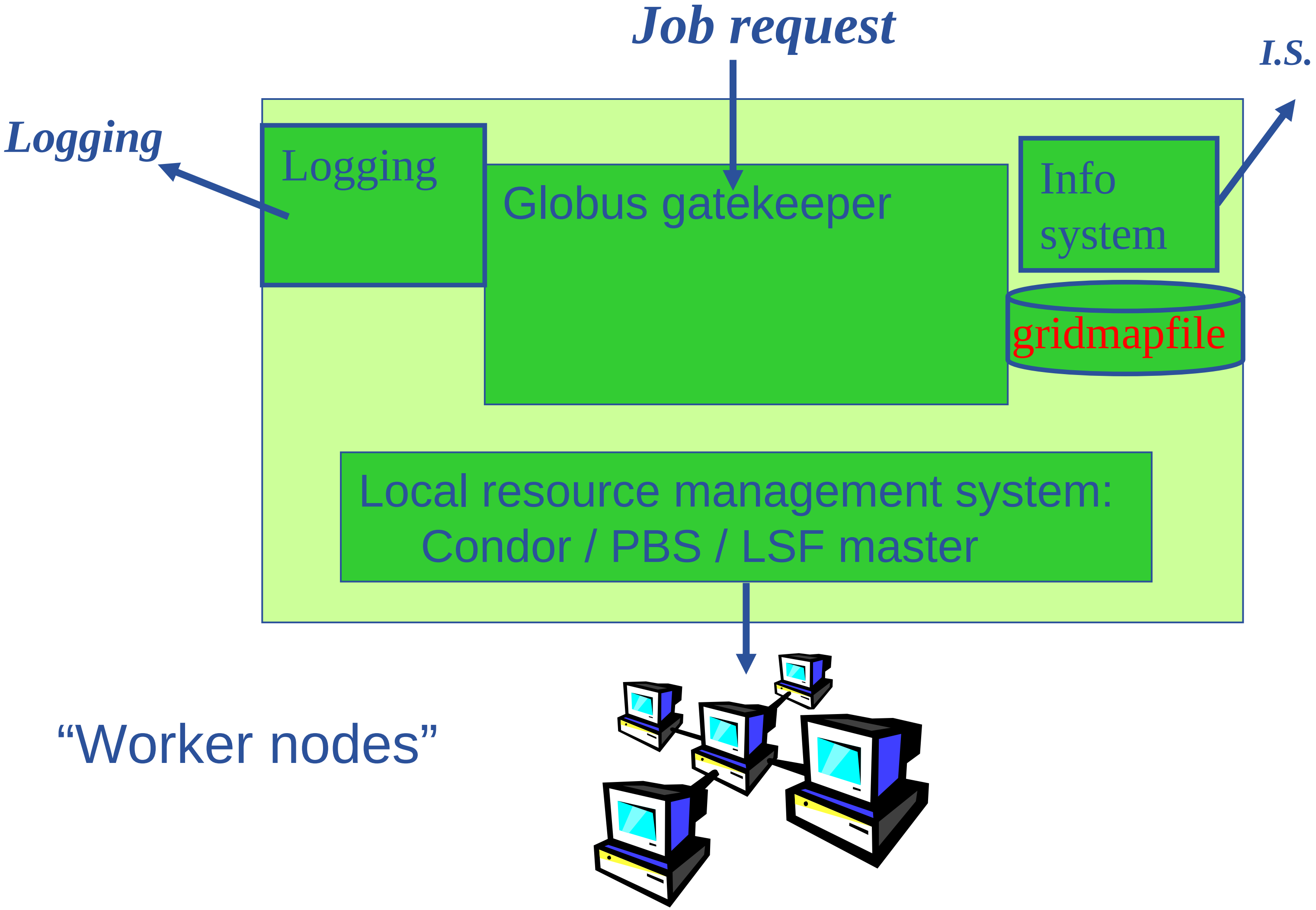
1. **Authentication - communication of identity**

   Basis for

   - Message integrity - so tampering is recognised
   - Message confidentiality, if needed - so sender and receiver only can understand the message
   - Non-repudiation: knowing who did what when – can’t deny it
2. **Authorisation - once identity is known, what can a user do?**
3. **Delegation- A allows service B to act on behalf of A**

- **Based on “X.509 certificates” – next talk!!**

“Compute element”: An LRMS queue



Job request

I.S.

Logging

Logging

Globus gatekeeper

Info system

gridmapfile

Local resource management system: Condor / PBS / LSF master

“Worker nodes”

A closer look at the main EGEE grid services

2. Data services

Data services in EGEE

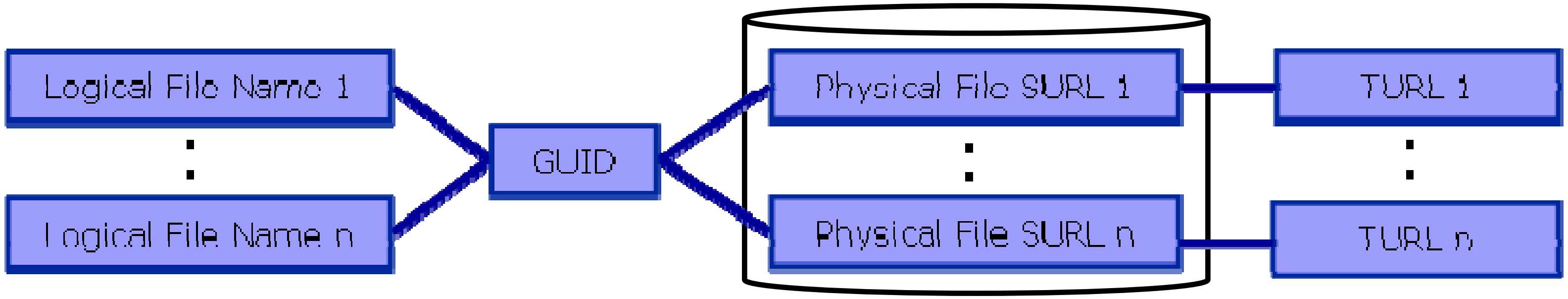
- **Files**
  - File Access Pattern:
    - Write once, read-many
- **3 service types for data**
  - Storage
  - Catalogs
  - Movement

Storage Element

- **Provides**
  - Storage for files
  - Transfer protocol (gsiFTP) ~ GSI based FTP server
  - POSIX-like file access
    - Grid File Access Layer (**GFAL**)
      - *API interface*
      - *To read parts of files too big to copy*
- **Two types**
  - “Classic” SE
    - Massive storage system - disk or tape based
  - “SRM” SE
    - SE’s are virtualised by common interface: “SRMv1”
    - SRM = Storage Resource Manager
    - work in progress to migrate to SRMv2

Name conventions

- **Logical File Name (LFN)**
  - An alias created by a user to refer to some item of data, e.g. "lfn:cms/20030203/run2/track1"
- **Globally Unique Identifier (GUID)**
  - A non-human-readable unique identifier for an item of data, e.g. "guid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6"
- **Site URL (SURL) (or Physical File Name (PFN) or Site FN)**
  - 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)
- **Transport URL (TURL)**
  - Temporary locator of a replica + access protocol: understood by a SE, e.g.
    "rfio://lxshare0209.cern.ch//data/alice/ntuples.dat"



Requirements

If a site acts as a central catalog for several VOs, it can either have:

- One LFC server, with one DB account containing the entries of all the supported VOs. You should then create one directory per VO.
- Several LFC servers, having each a DB account containing the entries for a given VO.

Both scenarios have consequences on the handling of database backups

- Minimum requirements (First scenario)
  - 2Ghz processor with 1GB of memory (not a hard requirement)
  - Dual power supply
  - Mirrored system disk

LFC - Description

The LCG File Catalog fixes the performance and scalability problems of EDG (European Data Grid) file catalogs.

Provides

- Bulk operations.
- Cursors for large queries.
- Timeouts and retries for client operations.

Added features :

- User exposed transaction API.
- Hierarchical namespace and namespace operations.
- Integrated GSI Authentication and Authorization.
- Access Control Lists (Unix Permissions and POSIX ACLs).
- Checksums.

Supported database backends: Oracle and MySQL

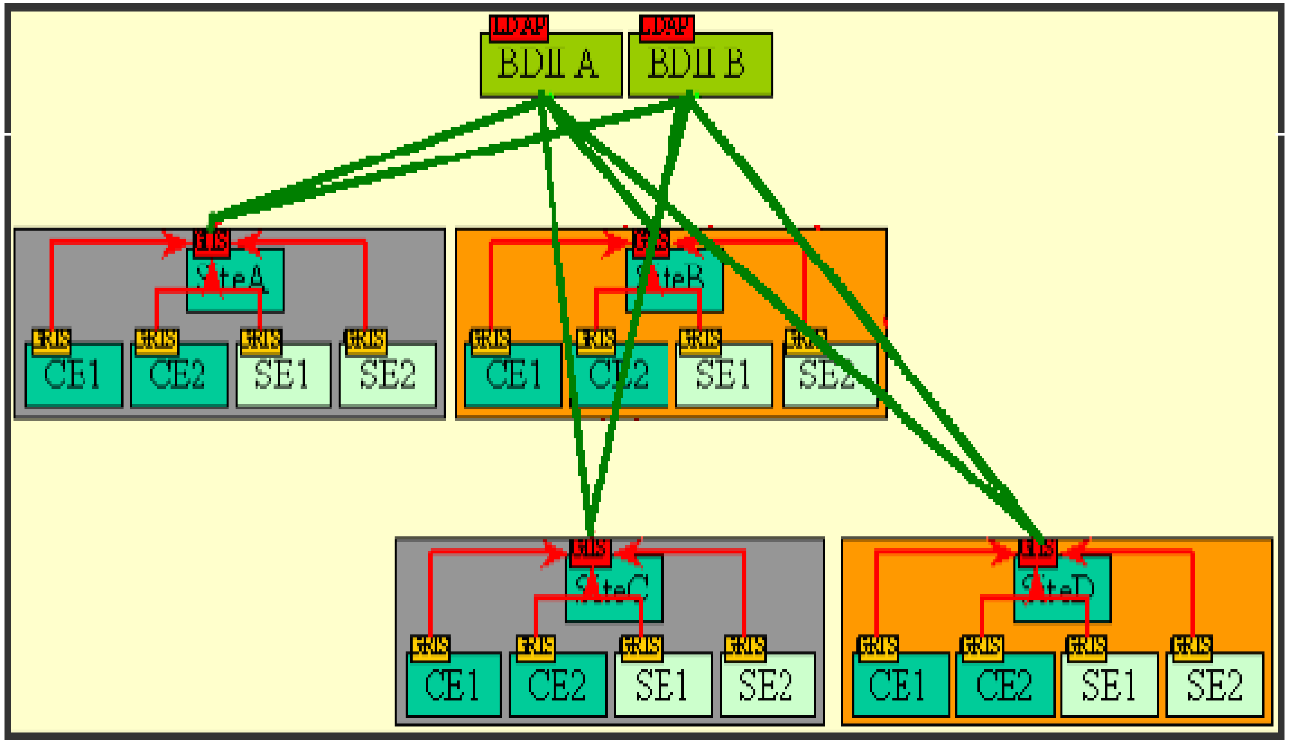
GFAL integration and support to lcg-* done by Grid Deployment group

A closer look at the main EGEE grid services

3. Information services

1st Information System: “BDII”

- Users can interrogate BDII servers by 2 sets of commands
  - lcg-infosites
  - lcg-info



- LDAP (Lightweight Directory Access Protocol)
- Glue Schema.

The 2nd Information System: R-GMA

- **Relational Grid Monitoring Architecture (R-GMA)**
  - Developed as part of the EuropeanDataGrid Project (EDG)
  - Now as part of the EGEE project.
  - Based on the Grid Monitoring Architecture (GMA)

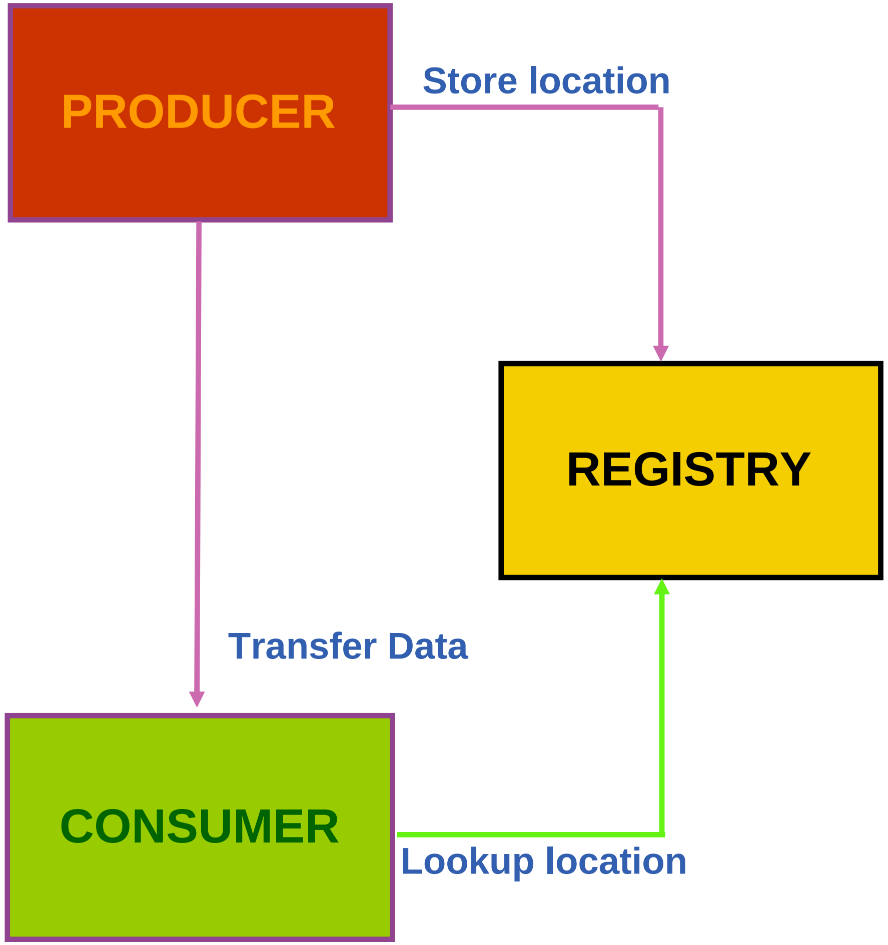
- **Uses a relational data model.**
  - Data are viewed as a table.
  - Data structure defined by the columns.
  - Each entry is a row (tuple).
  - Queried using Structured Query Language (SQL).

name	ID	birth	Group
Tom	4	1977-08-20	HR

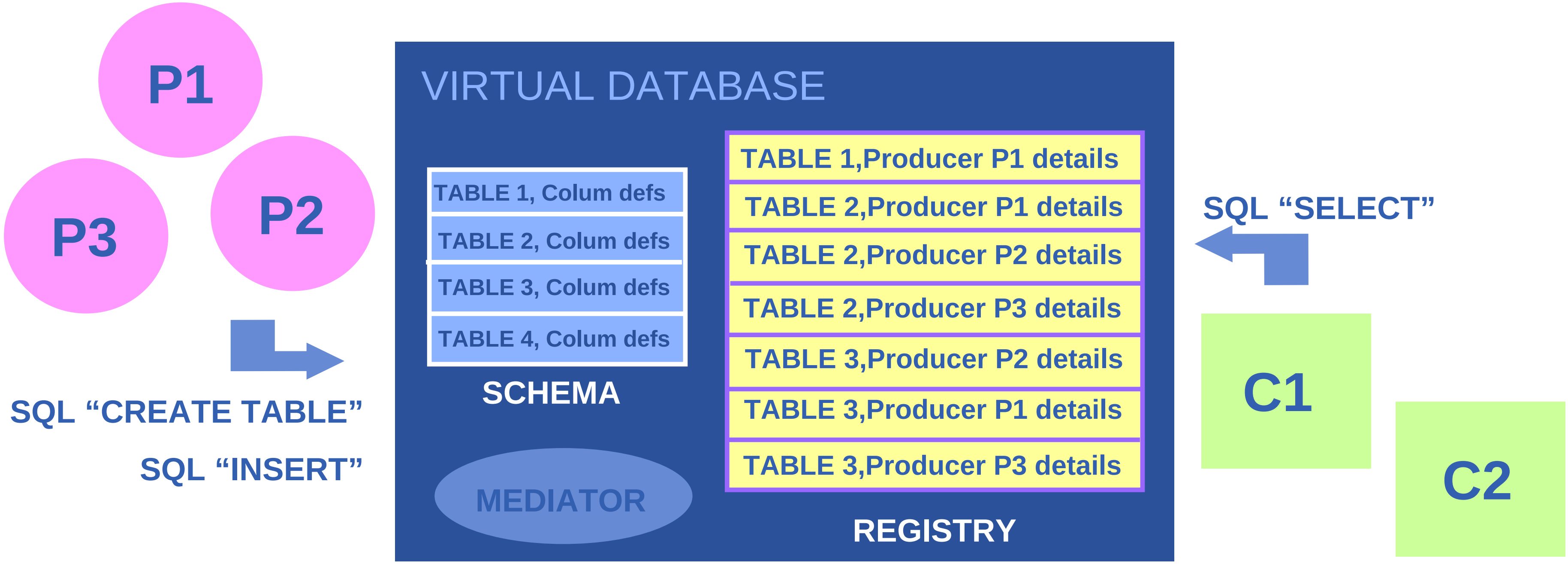
SELECT * FROM people WHERE group=‘HR’

Service orientation

- The Producer stores its location (URL) in the Registry.
- The Consumer looks up producer URLs in the Registry.
- The Consumer contacts the Producer to get all the data or the Consumer can listen to the Producer for new data.



R-GMA


P1
P2
P3
SQL “CREATE TABLE”
SQL “INSERT”
VIRTUAL DATABASE
TABLE 1, Colum defs
TABLE 2, Colum defs
TABLE 3, Colum defs
TABLE 4, Colum defs
SCHEMA
MEDIATOR
TABLE 1,Producer P1 details
TABLE 2,Producer P1 details
TABLE 2,Producer P2 details
TABLE 2,Producer P3 details
TABLE 3,Producer P2 details
TABLE 3,Producer P1 details
TABLE 3,Producer P3 details
REGISTRY
SQL “SELECT”
C1
C2

There is no central repository!!! There is only a “*Virtual Database”.*

Schema is a list of table definitions: additional tables/schema can be defined by applications

Registry is a list of data producers with all its details.

Producers publish data – from sites, from applications

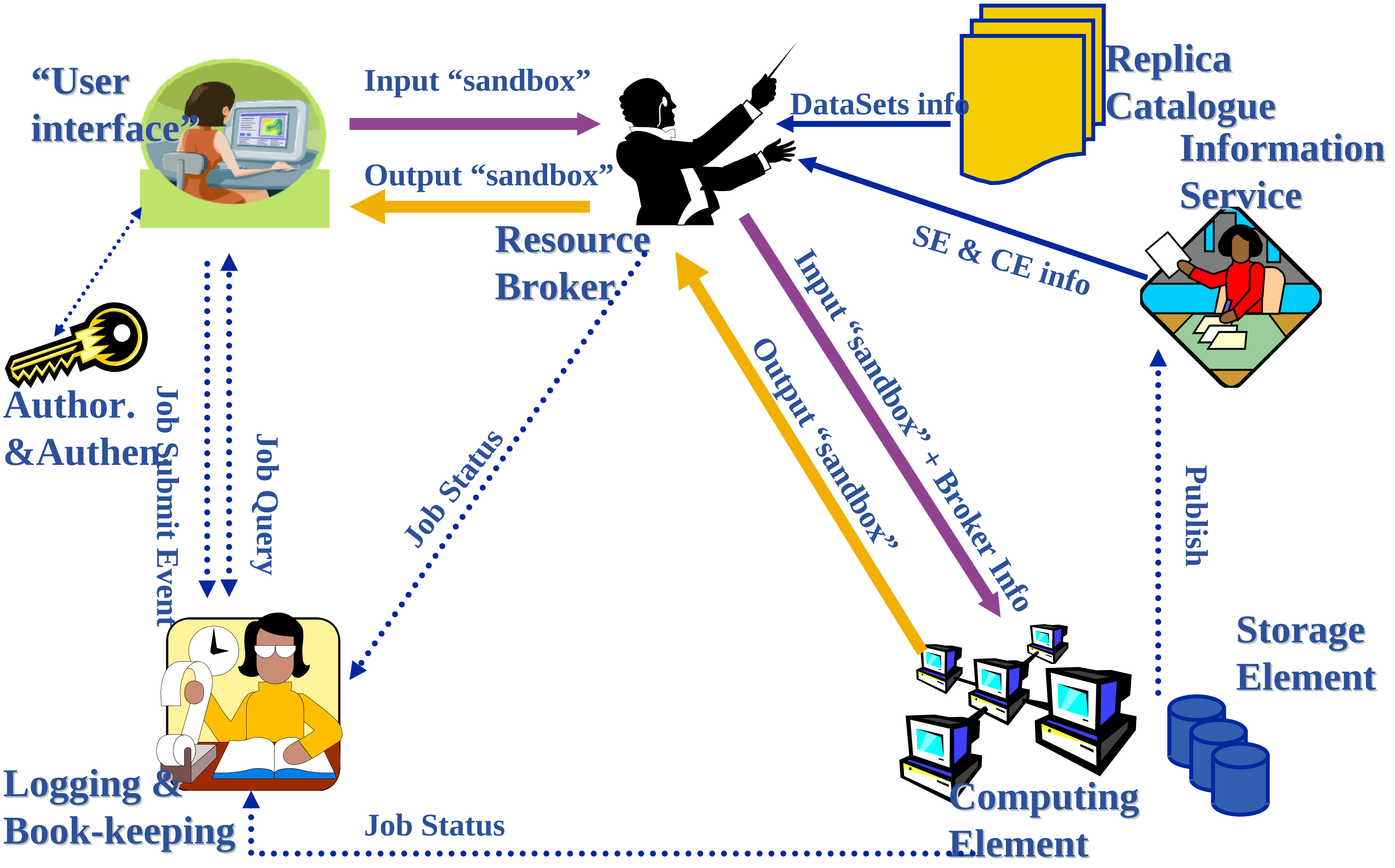
Consumer read data published.

A closer look at
the main EGEE grid services

4. Job submission

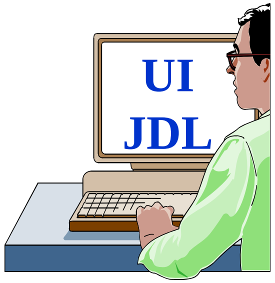
Current production middleware

Enabling Grids for E-sciencE



User Interface node

- **The user's interface to the Grid**
- **Command-line interface to**
  - Create/Manage proxy certificates
  - Job operations
    - To submit a job
    - Monitor its status
    - Retrieve output
  - Data operations
    - Upload file to SE
    - Create replica
    - Discover replicas
  - Other grid services
- **Also C++ and Java APIs**



- **To run a job user creates a JDL (Job Description Language) file**

Building on basic tools and Information Service

- **Submit job to grid via the “resource broker (RB)”,**
- **edg_job_submit *my.jdl***

Returns a “job-id” used to monitor job, retrieve output

Example JDL file

```
Executable = “gridTest”;
StdError = “stderr.log”;
StdOutput = “stdout.log”;
InputSandbox = {“/home/joda/test/gridTest”};
OutputSandbox = {“stderr.log”, “stdout.log”};
InputData = “lfn:testbed0-00019”;
DataAccessProtocol = “gridftp”;
Requirements = other.Architecture==“INTEL” && \
             other.OpSys==“LINUX” && other.FreeCpus >=4;
Rank = “other.GlueHostBenchmarkSF00”;
```

Building on basic tools and Information Service

- Submit job to grid via the “resource broker”,

- edg_job_submit *my.jdl*

Returns a “job-id” used to monitor job, retrieve output

Example JDL file

```
Executable = “gridTest”;
StdError = “stderr.log”;
StdOutput = “stdout.log”;
InputSandbox = {“/home/joda/test/gridTe
OutputSandbox = {“stderr.log”, “stdout.log”};
InputData = “lfn:testbed0-00019”;
DataAccessProtocol = “gridftp”;
Requirements = other.Architecture==“INTEL” && \
             other.OpSys==“LINUX” && other.FreeCpus >=4;
Rank = “other.GlueHostBenchmarkSF00”;
```

lfn: logical file name

RB uses Catalog to find replica locations

Building on basic tools and Information Service

- Submit job to grid via the “resource broker”,

- edg_job_submit *my.jdl*

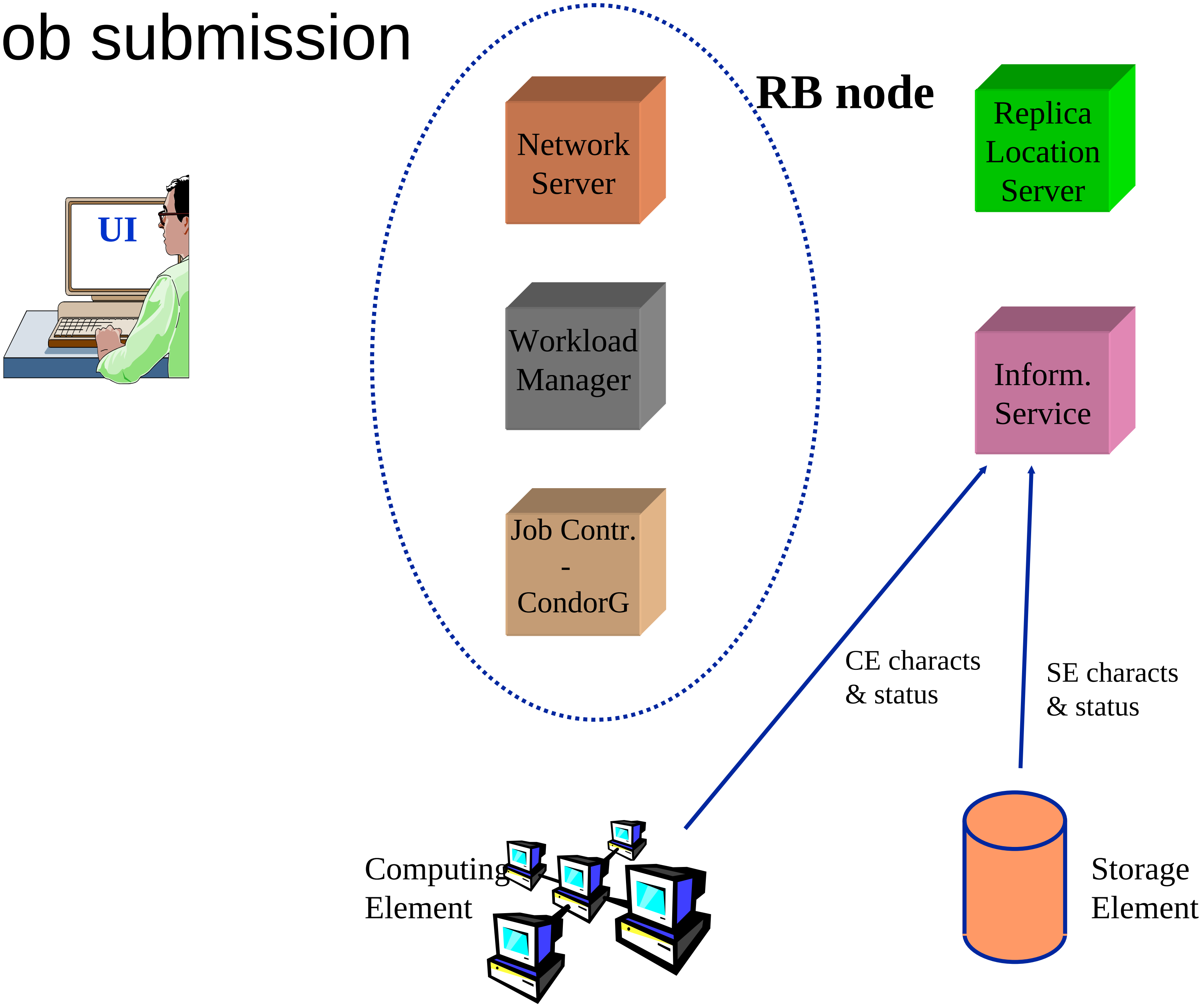
Returns a “job-id” used to monitor job, retrieve output

Example JDL file

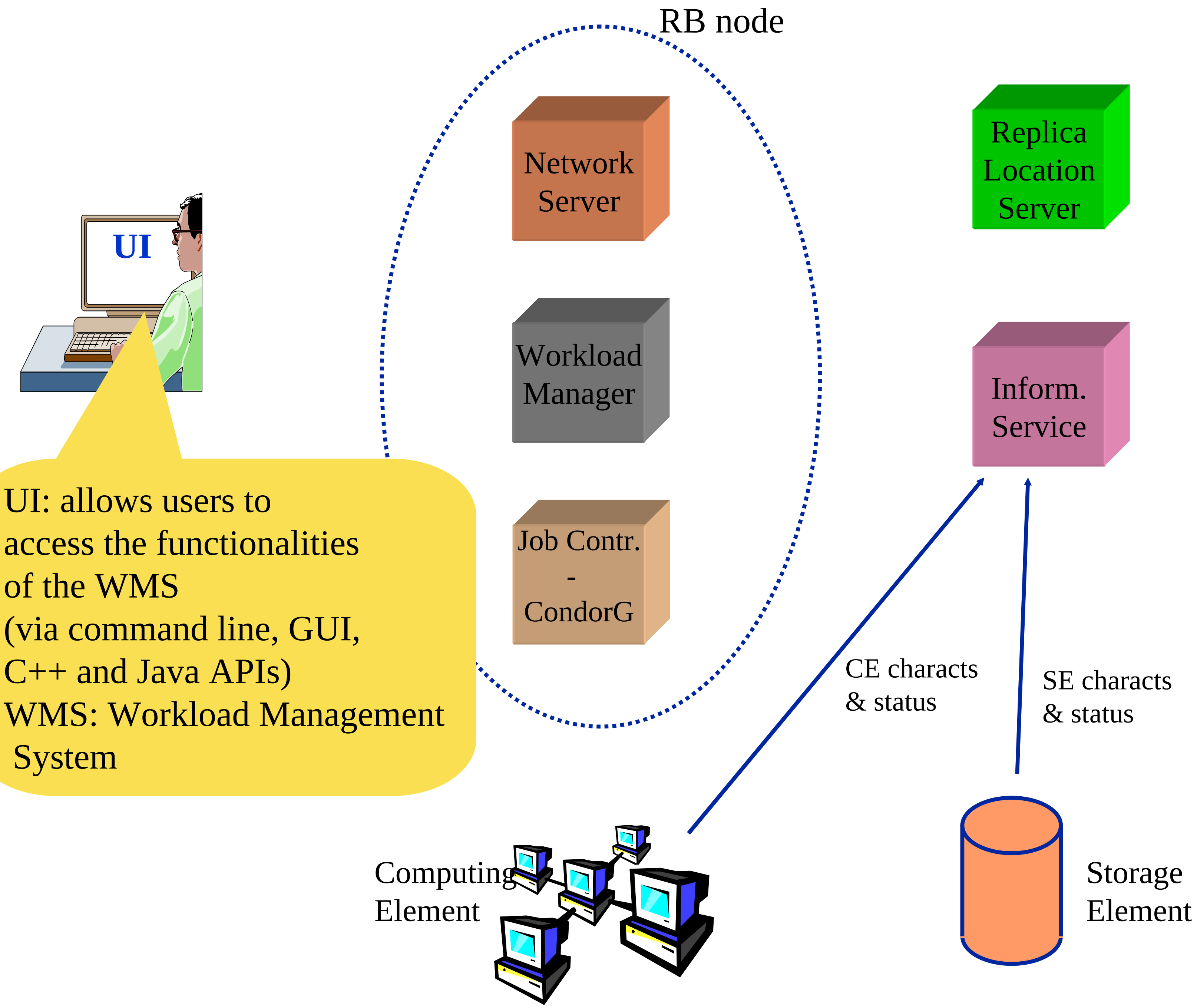
```
Executable = “gridTest”;
StdError = “stderr.log”;
StdOutput = “stdout.log”;
InputSandbox = {“/home/joda/test/c       est”};
OutputSandbox = {“stderr.log”, “    out.log”};
InputData = “lfn:testbed0-0001   ;
DataAccessProtocol = “gridft  ”;
Requirements = other.Architecture==“INTEL” && \
             other.OpSys==“LINUX” && other.FreeCpus >=4;
Rank = “other.GlueHostBenchmarkSF00”;
```

Uses BDII Information System

Job submission



Job Status



UI: allows users to access the functionalities of the WMS (via command line, GUI, C++ and Java APIs)
WMS: Workload Management System

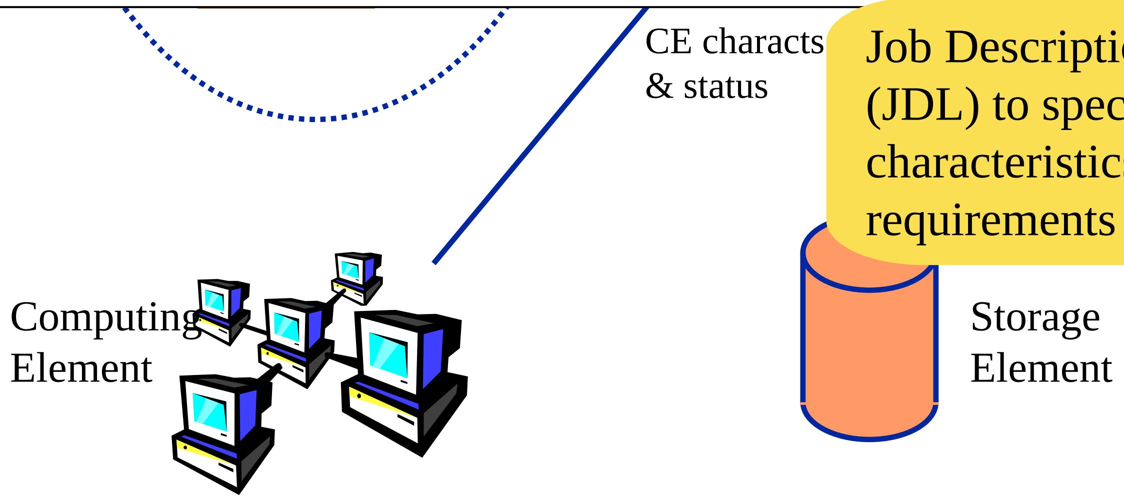
edg-job-submit myjob.jdl

Job Status

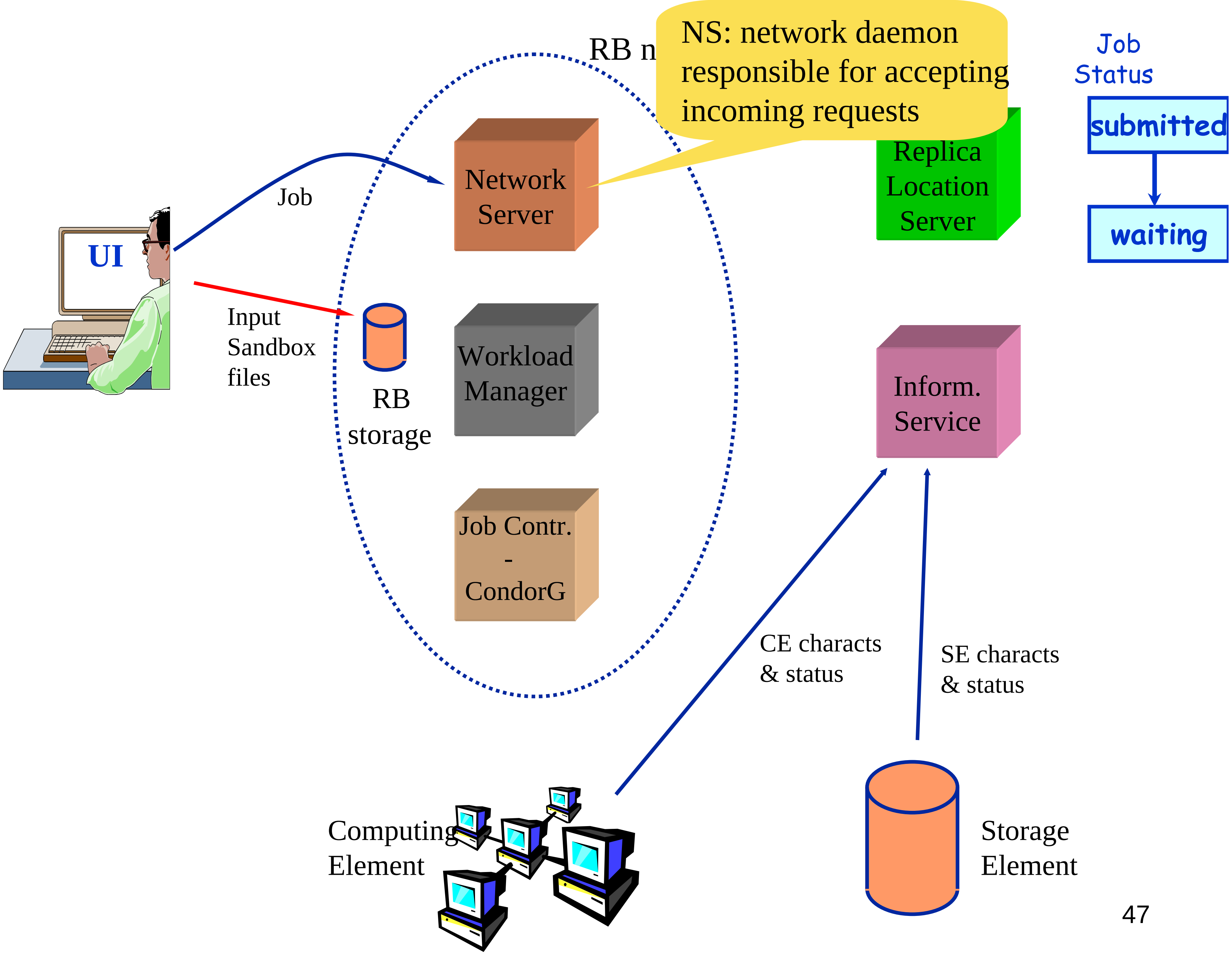
submitted

Myjob.jdl

```
JobType = “Normal”;
Executable     = "$(CMS)/exe/sum.exe";
InputSandbox = {"/home/user/WP1testC","/home/file*”,
"/home/user/DATA/*"};
OutputSandbox = {“sim.err”, “test.out”, “sim.log"};
Requirements   = other. GlueHostOperatingSystemName ==
“linux" &&
other. GlueHostOperatingSystemRelease == "Red Hat 7.3“
&& other.GlueCEPolicyMaxCPUTime > 10000;
Rank  = other.GlueCEStateFreeCPUs;
```



Job Description Languag
(JDL) to specify job
characteristics and
requirements



NS: network daemon responsible for accepting incoming requests

RB n

UI

Job

Input Sandbox files

RB storage

Network Server

Workload Manager

Job Contr. - CondorG

Replica Location Server

Inform. Service

CE characts & status

SE characts & status

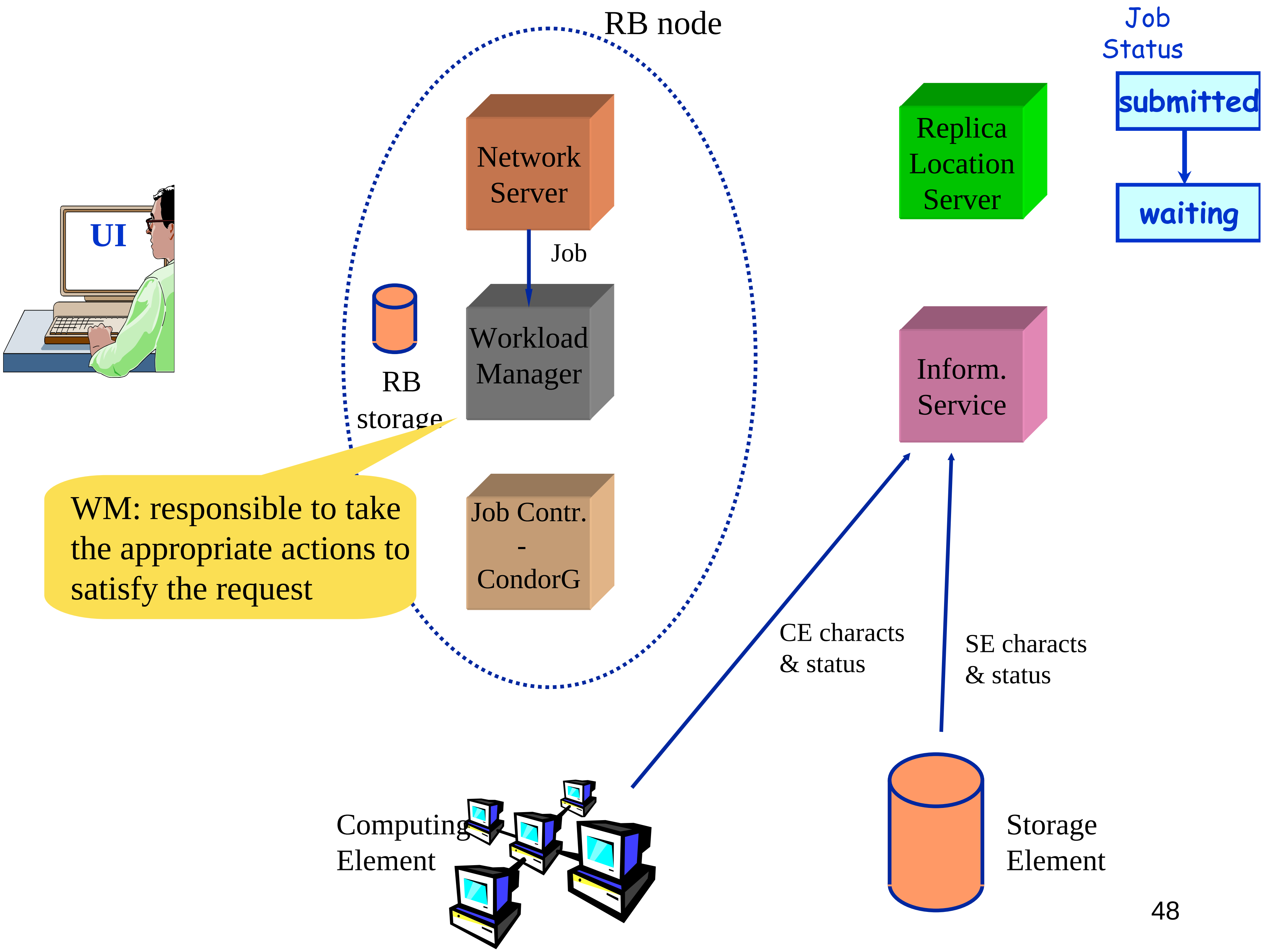
Computing Element

Storage Element

Job Status

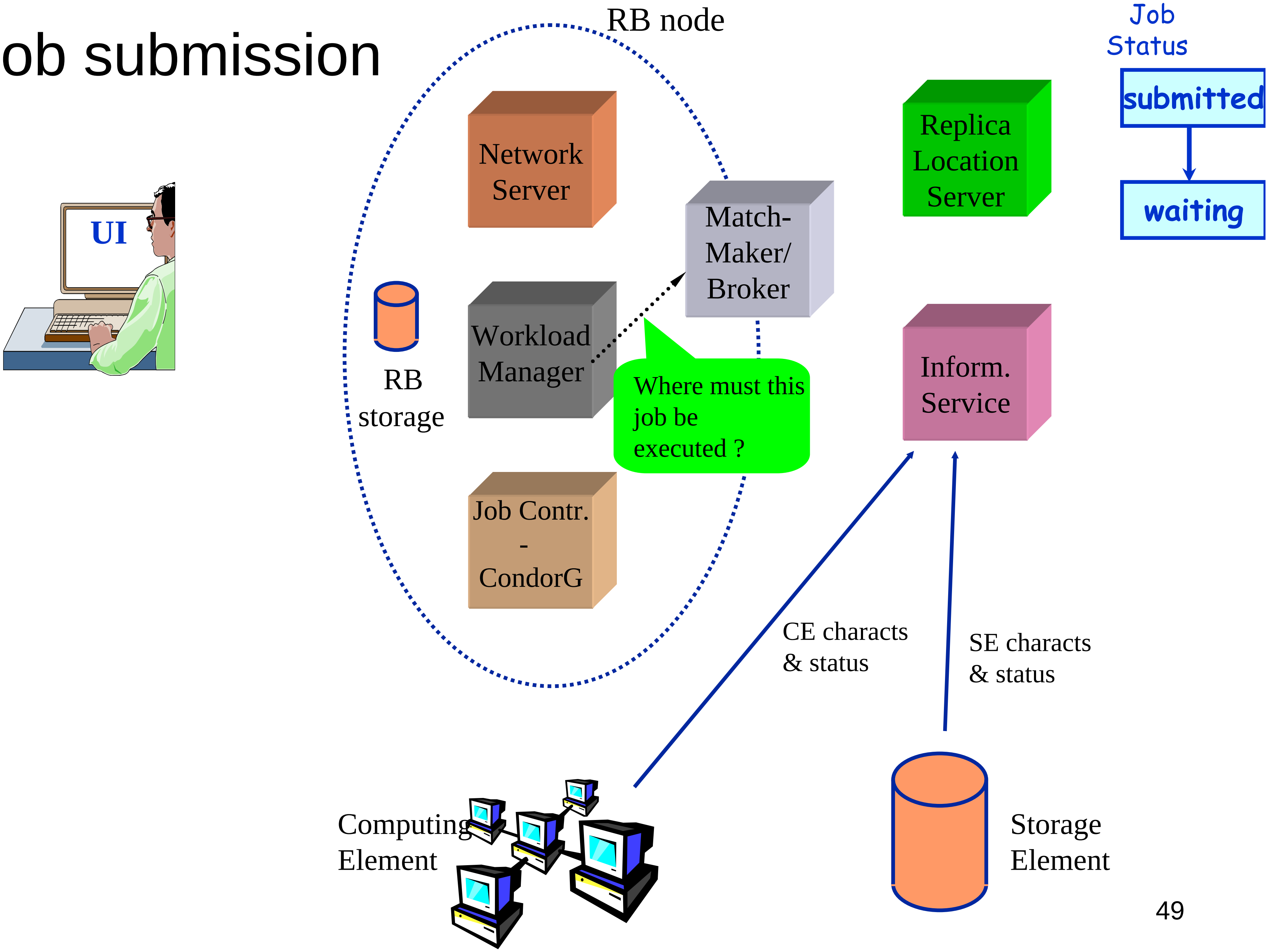
submitted

waiting

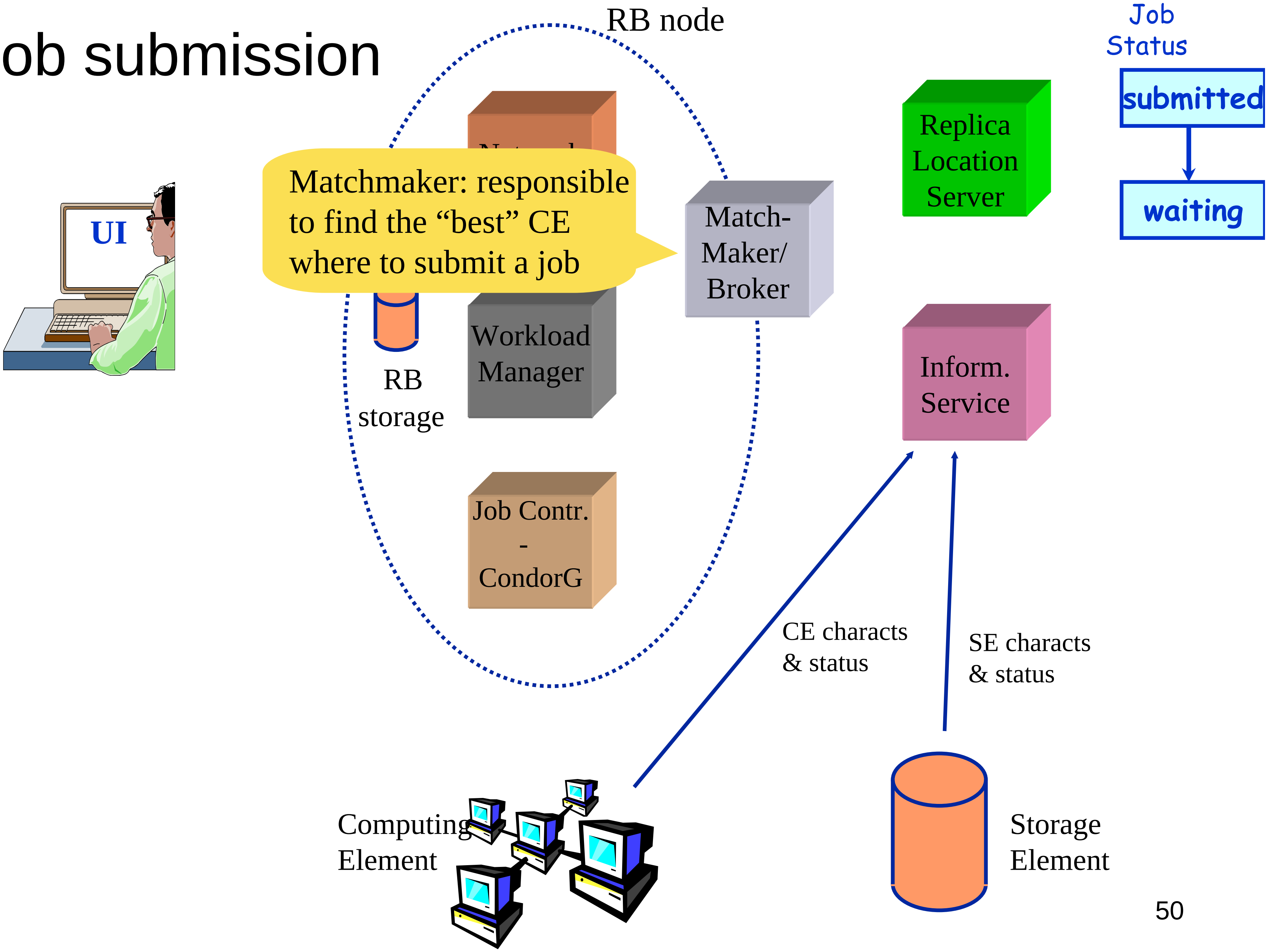


WM: responsible to take the appropriate actions to satisfy the request

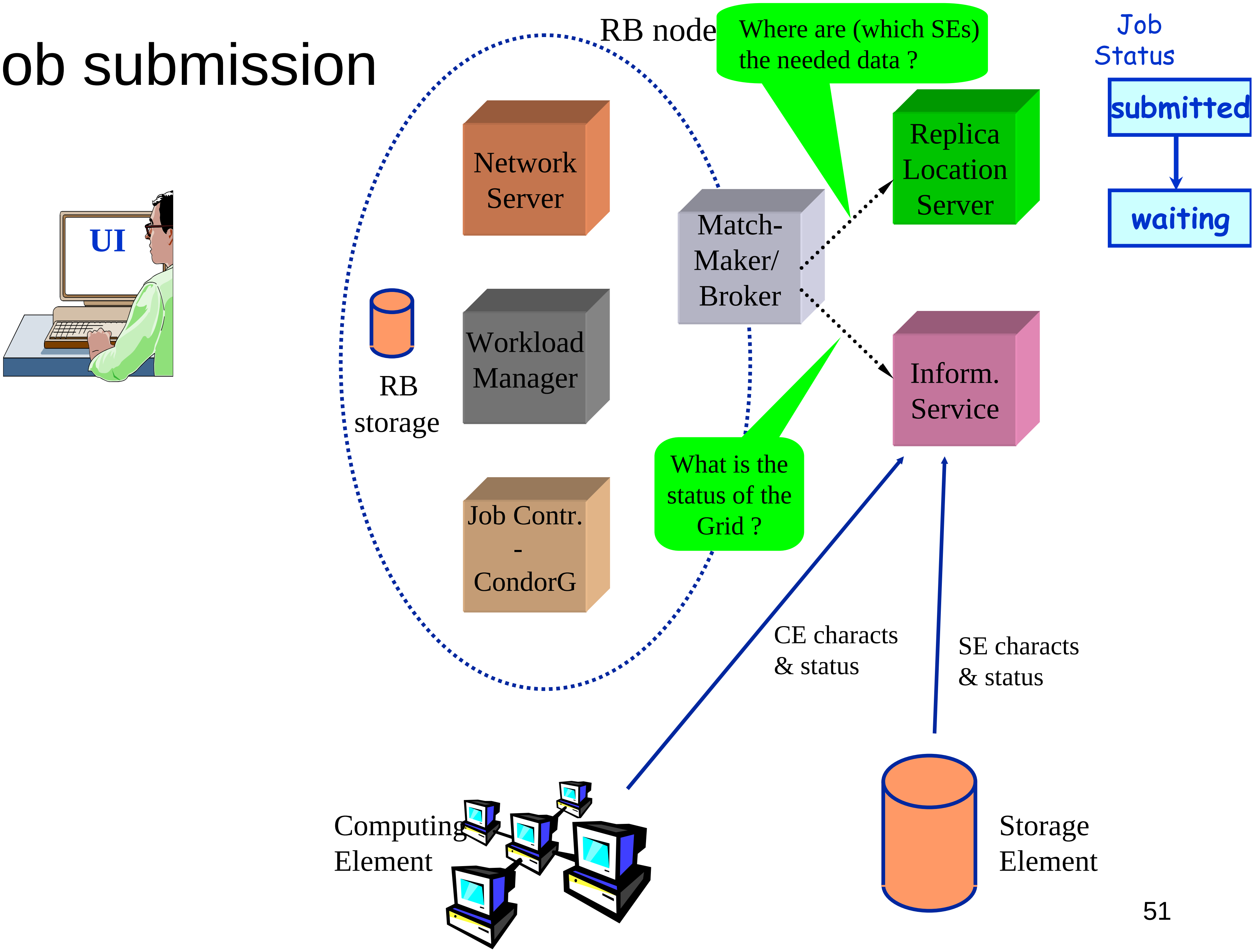
Job submission



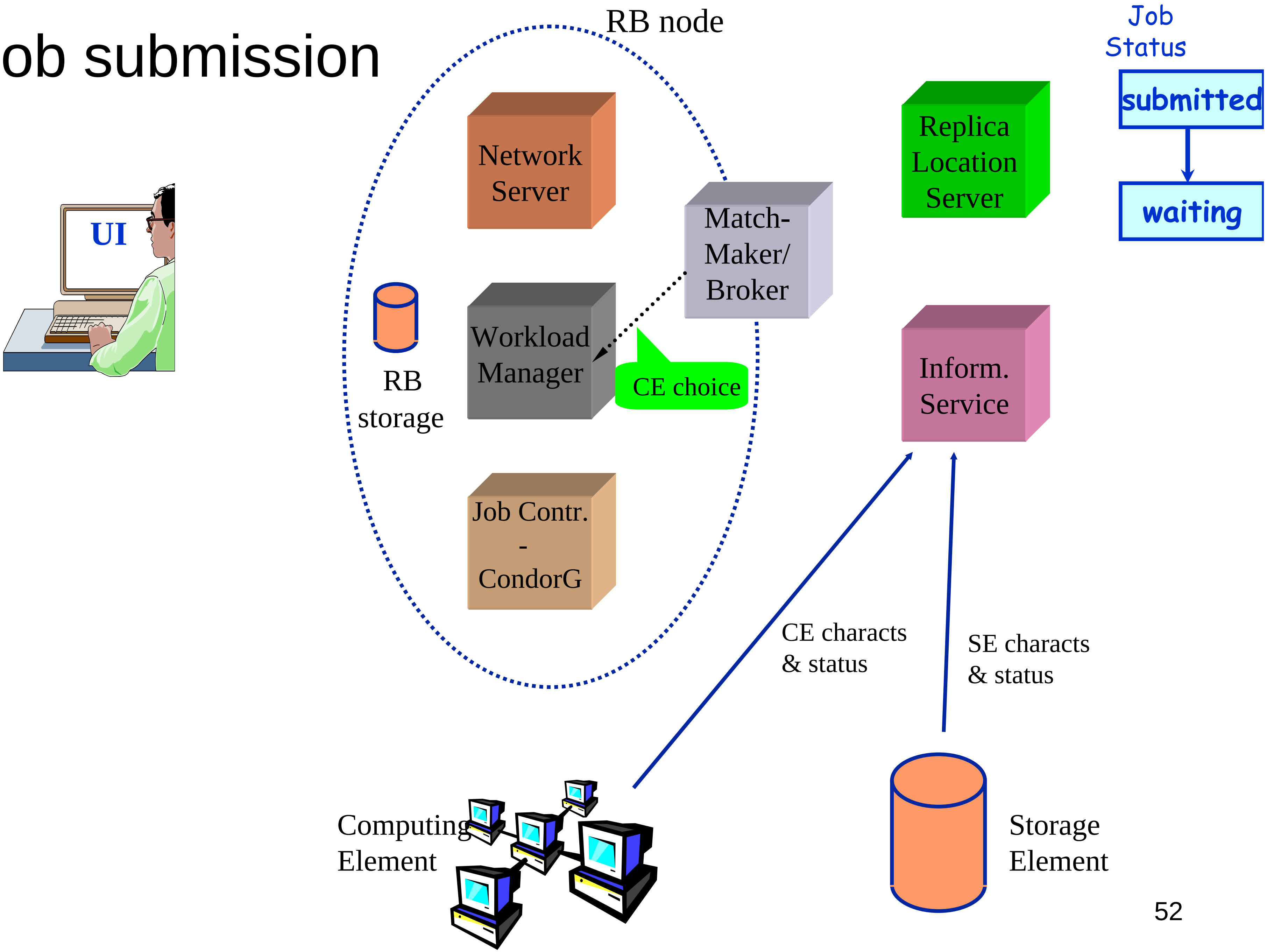
Job submission



Job submission



Job submission



UI

RB node

Network Server

Workload Manager

Job Contr. - CondorG

RB storage

Match-Maker/ Broker

CE choice

Replica Location Server

Inform. Service

Job Status

submitted

waiting

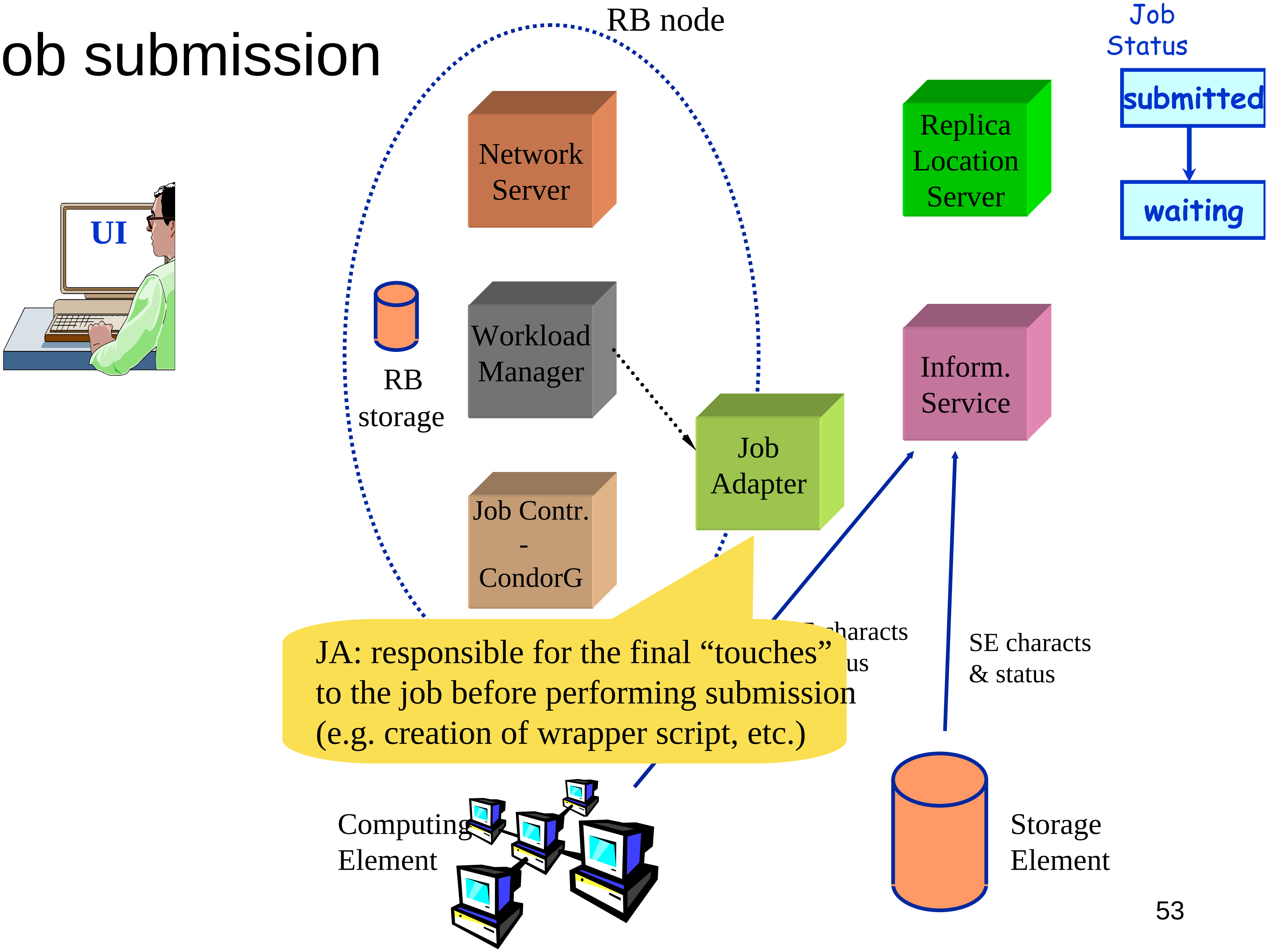
CE characts & status

SE characts & status

Computing Element

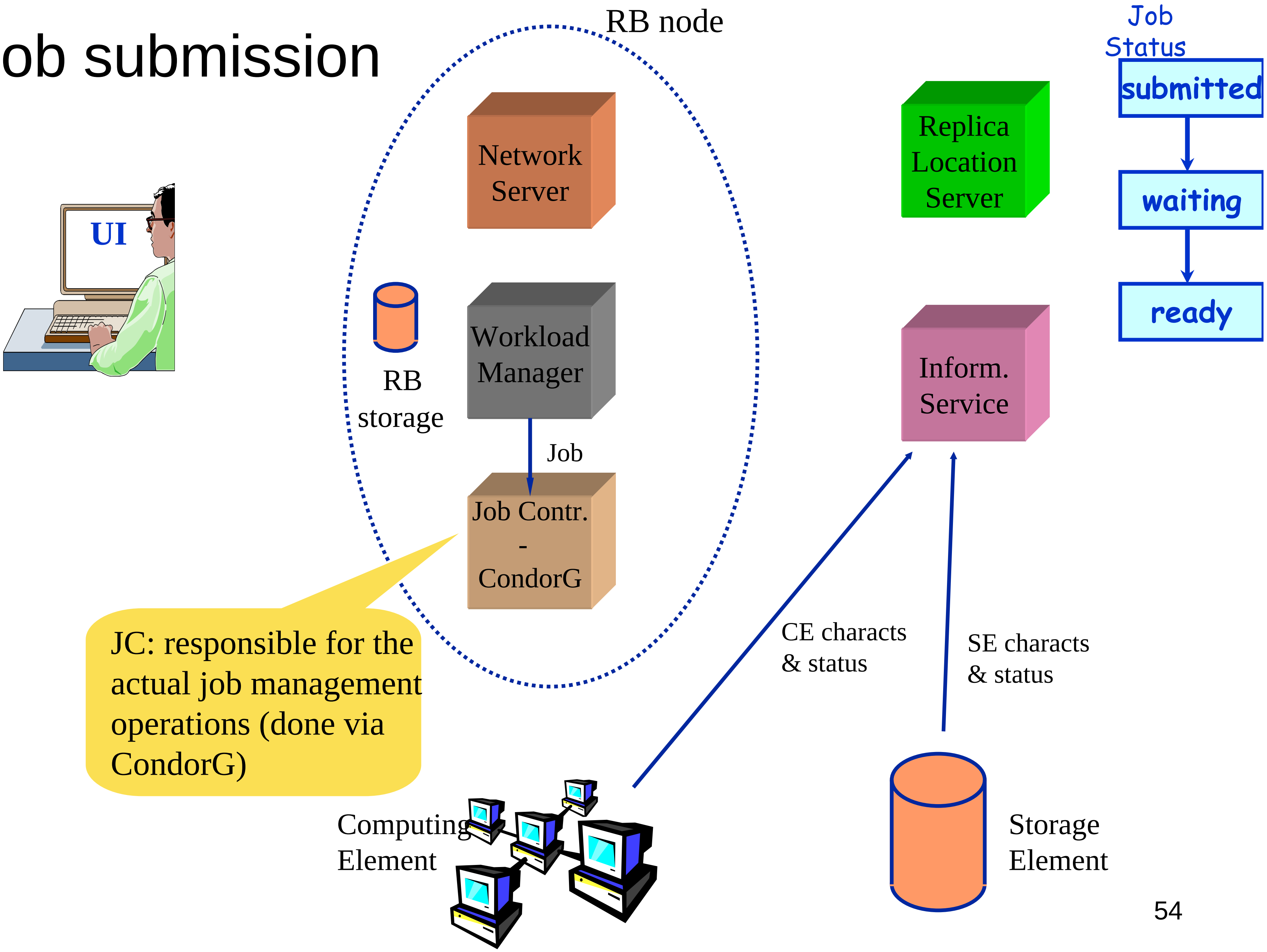
Storage Element

Job submission



JA: responsible for the final “touches” to the job before performing submission (e.g. creation of wrapper script, etc.)

Job submission



UI

RB node

Network Server

RB storage

Workload Manager

Job

Job Contr. - CondorG

Replica Location Server

Inform. Service

CE characts & status

SE characts & status

Computing Element

Storage Element

Job Status

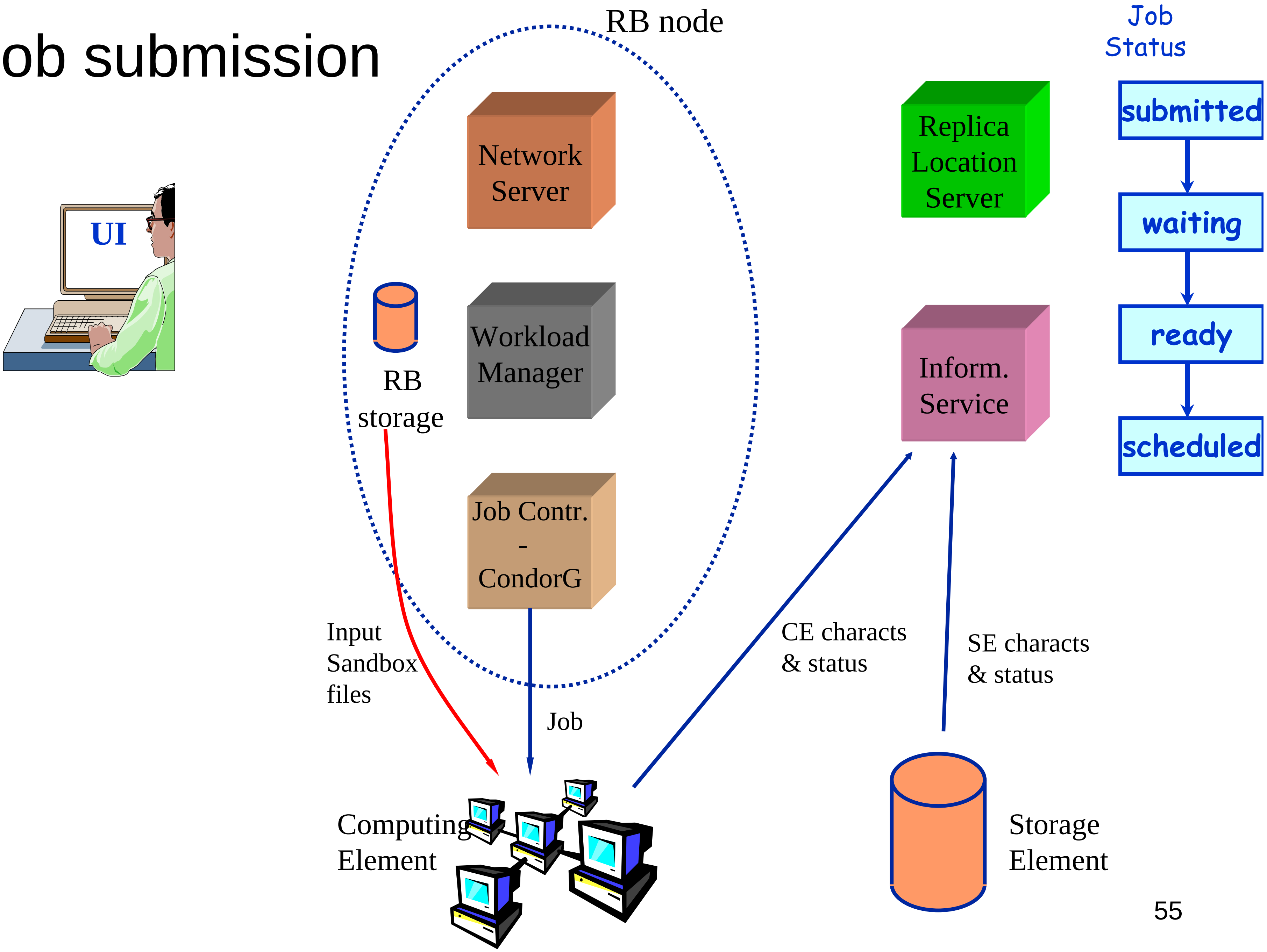
submitted

waiting

ready

JC: responsible for the actual job management operations (done via CondorG)

Job submission



UI

RB node

Network Server

Workload Manager

Job Contr. - CondorG

RB storage

Replica Location Server

Inform. Service

Input Sandbox files

Job

CE characts & status

SE characts & status

Computing Element

Storage Element

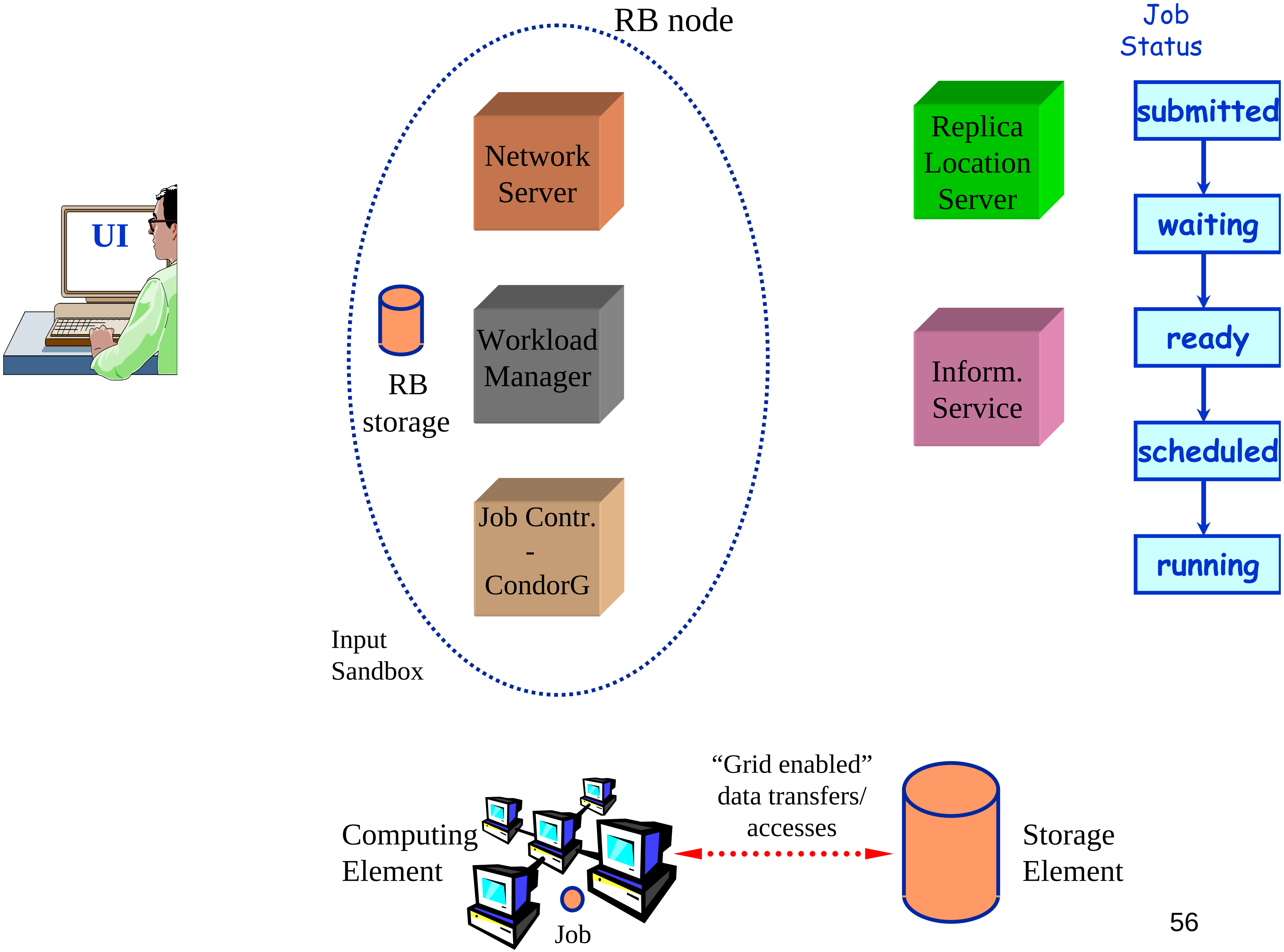
Job Status

submitted

waiting

ready

scheduled



UI

RB node

Network Server

Workload Manager

Job Contr. - CondorG

RB storage

Input Sandbox

Replica Location Server

Inform. Service

Job Status

submitted

waiting

ready

scheduled

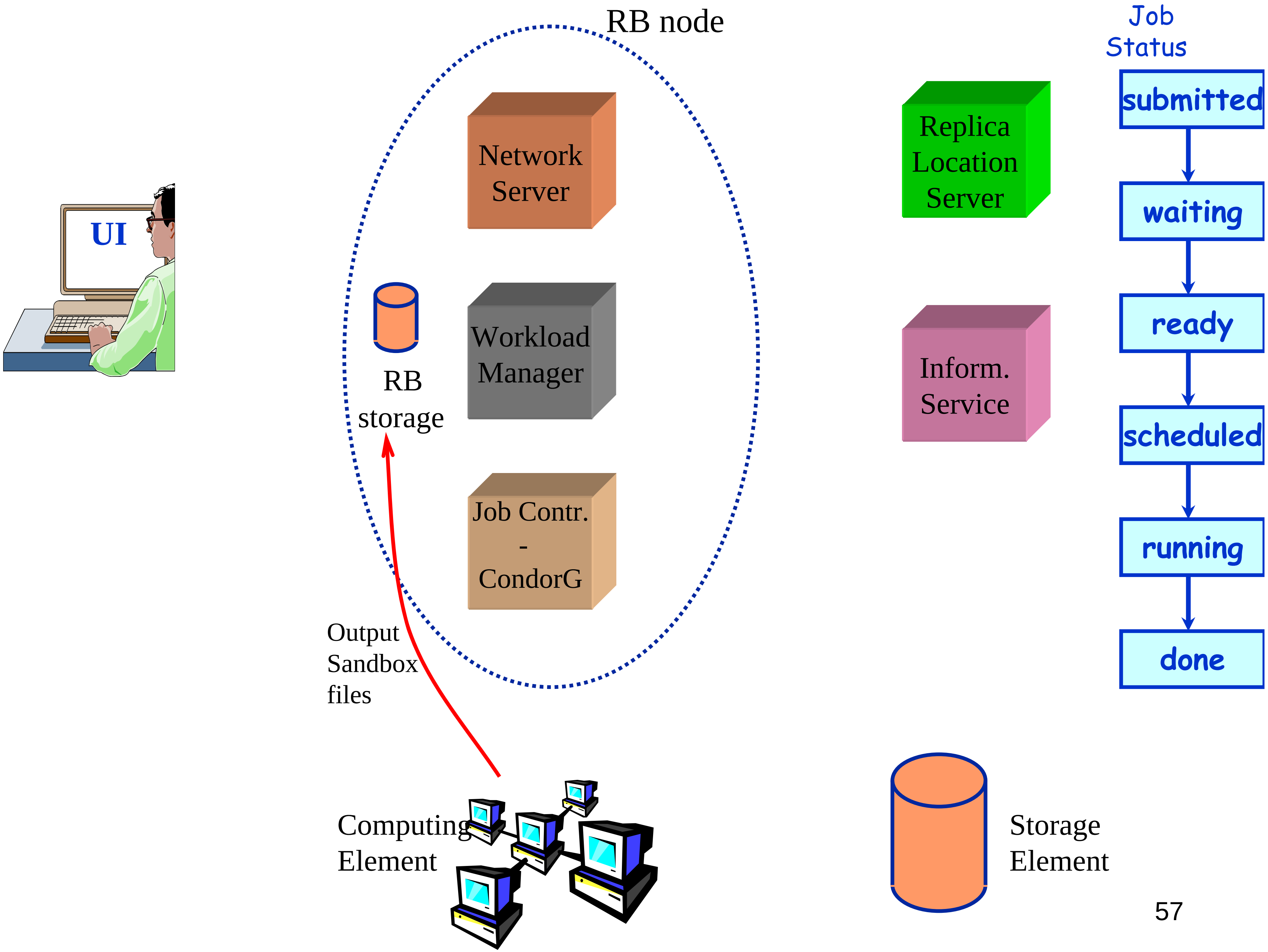
running

Computing Element

Job

“Grid enabled” data transfers/ accesses

Storage Element



UI

RB node

Network Server

Workload Manager

Job Contr. - CondorG

RB storage

Output Sandbox files

Computing Element

Replica Location Server

Inform. Service

Storage Element

Job Status

submitted

waiting

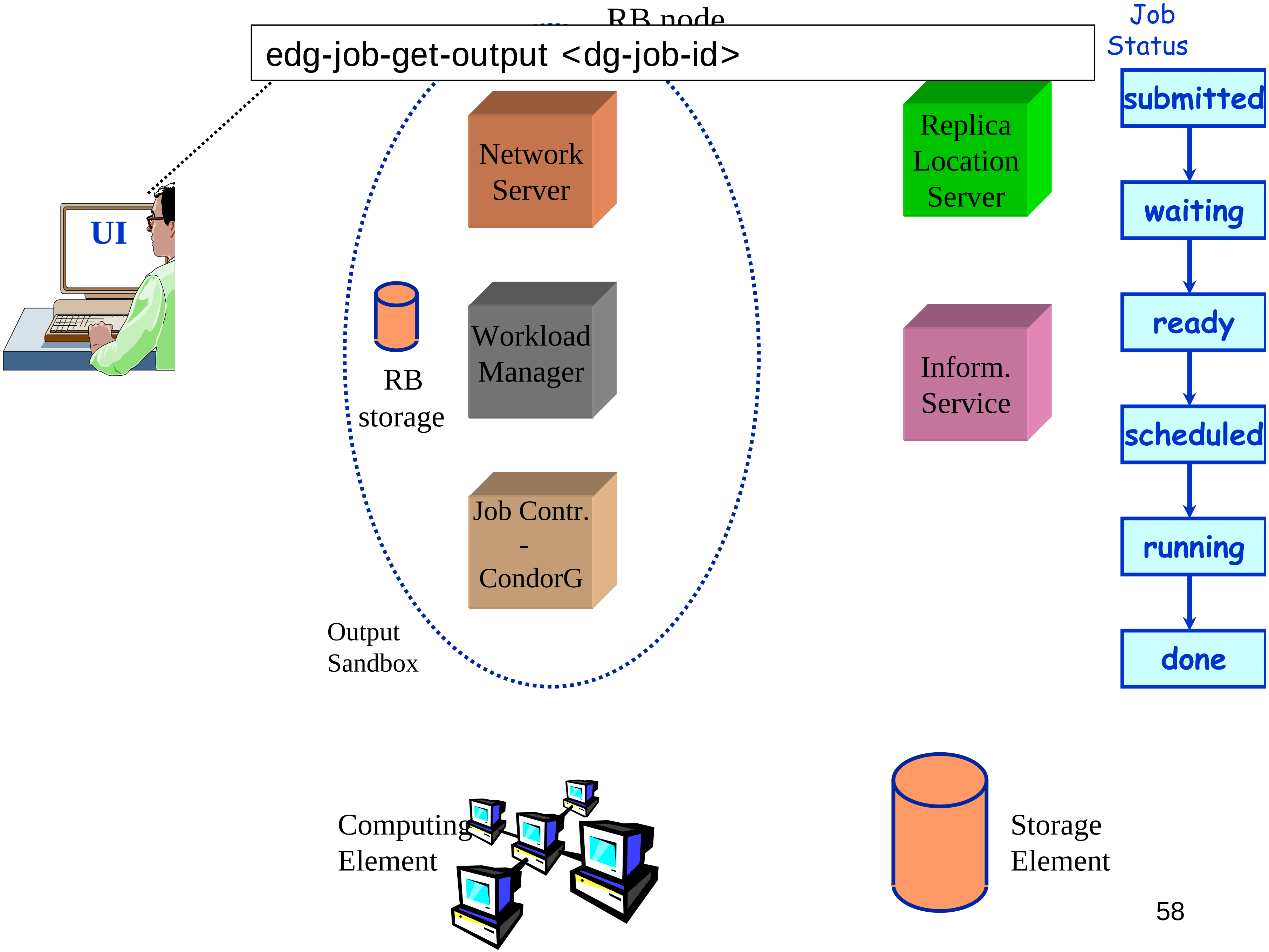
ready

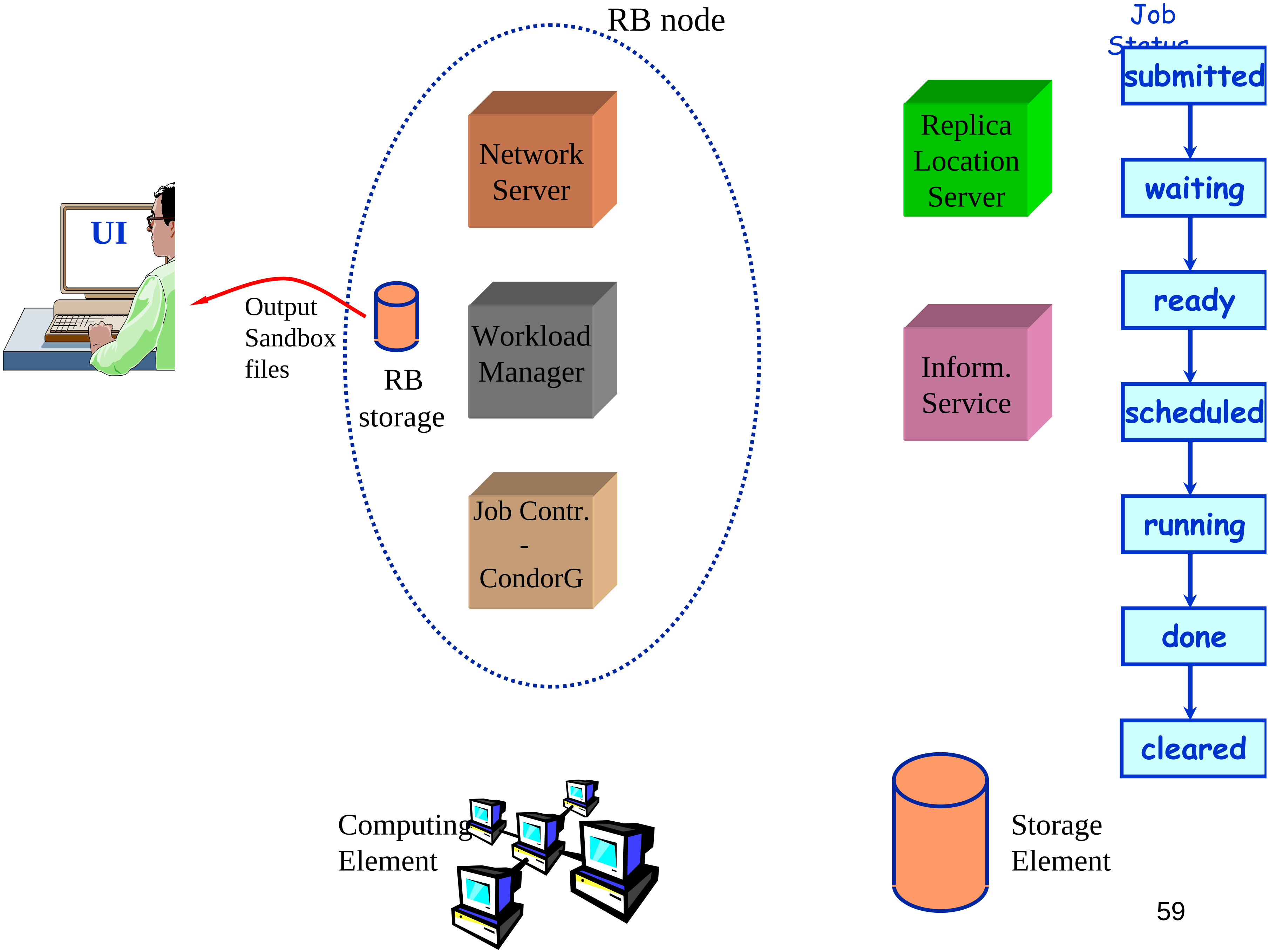
scheduled

running

done

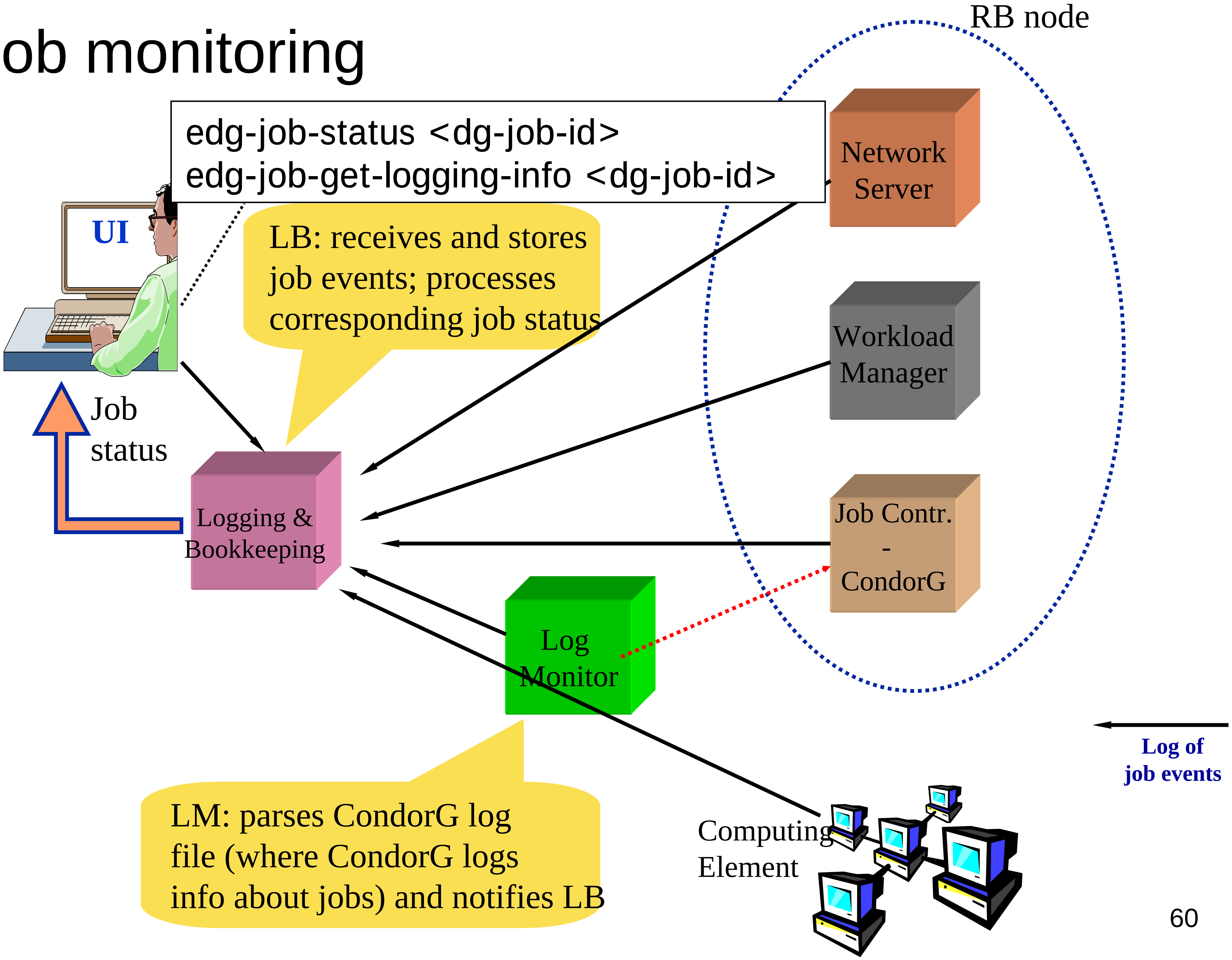
```
edg-job-get-output <dg-job-id>
```




RB node
UI
Output Sandbox files
RB storage
Network Server
Workload Manager
Job Contr. - CondorG
Replica Location Server
Inform. Service
Job Status
submitted
waiting
ready
scheduled
running
done
cleared
Computing Element
Storage Element

Job monitoring

```
edg-job-status <dg-job-id>
edg-job-get-logging-info <dg-job-id>
```



RB node

UI

Network Server

Workload Manager

Job Contr. - CondorG

Log Monitor

Logging & Bookkeeping

Job status

Computing Element

LB: receives and stores job events; processes corresponding job status

LM: parses CondorG log file (where CondorG logs info about jobs) and notifies LB

Log of job events

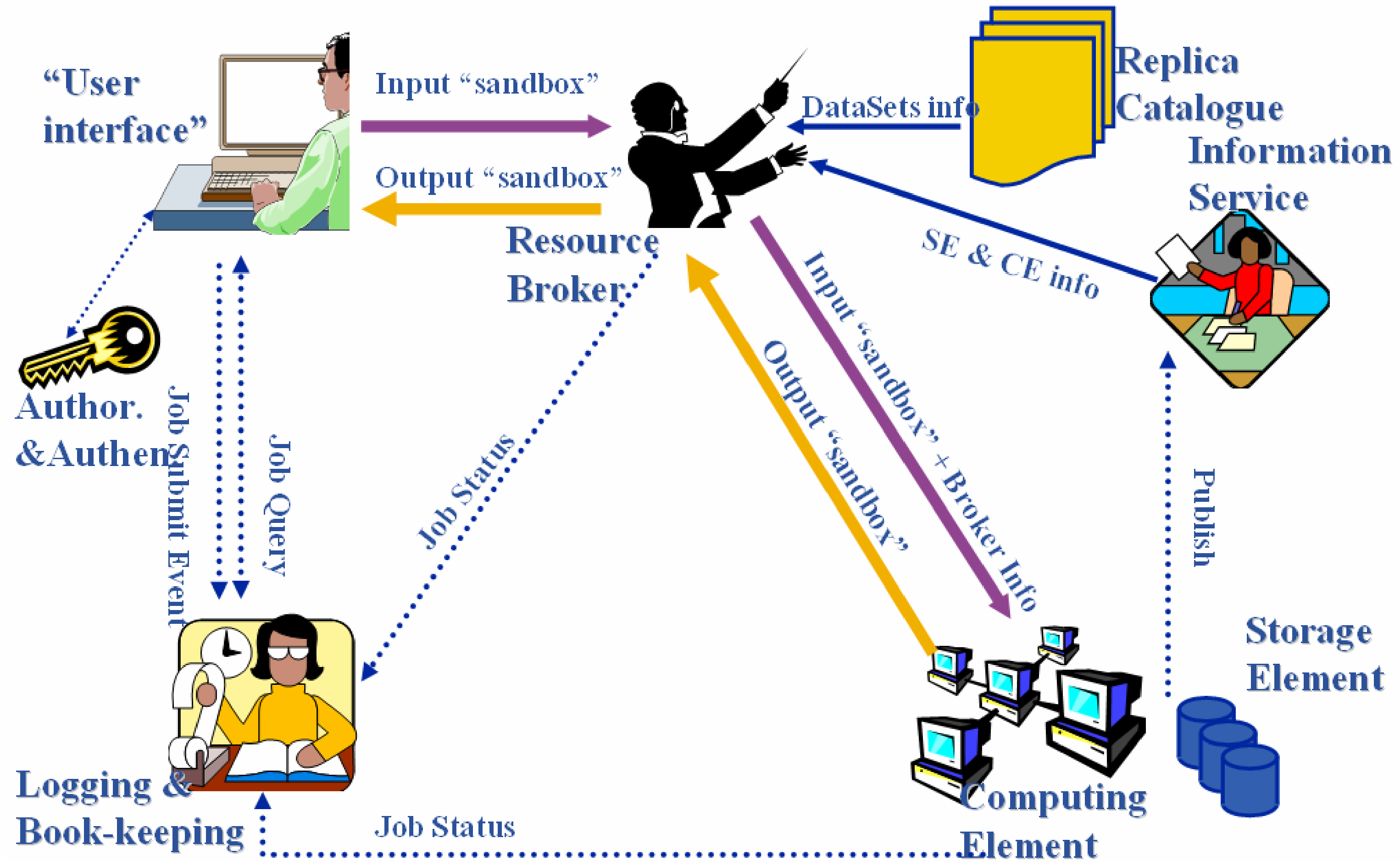
Possible job states

Flag	Meaning
SUBMITTED	**submission logged in the LB**
WAIT	**job match making for resources**
READY	**job being sent to executing CE**
SCHEDULED	**job scheduled in the CE queue manager**
RUNNING	**job executing on a WN of the selected CE queue**
DONE	**job terminated without grid errors**
CLEARED	**job output retrieved**
ABORT	**job aborted by middleware, check** *reason*

Summary

- **From the rich grid ecosystem emerged the EGEE production middleware**
  - **Built on tools for**
    - Authorisation and authentication
    - Job submission (direct to a Computing Element)
    - File transfer
  - …**with higher level services**
    - Job submission to "a grid" (via resource broker)
    - Data management
    - Information Systems
  - ..**and upon these can be built toolkits and services for new application communities**
    - Workflow
    - Portals: e.g. P-GRADE Portal – www.lpds.sztaki.hu/pgportal
- **Authorisation and authentication underpin the middleware**
  - resource-sharing across organisations, without centralised control

Summary


"User interface"
Input "sandbox"
Output "sandbox"
Resource Broker
DataSets info
Replica Catalogue
Information Service
SE & CE info
Author. &Authen.
Job Submit Event
Job Query
Job Status
Input "sandbox" + Broker Info
Output "sandbox"
Publish
Storage Element
Computing Element
Logging & Book-keeping
Job Status

Further information

- EGEE www.eu-egee.org
- EGEE: 1st user Forum http://egee-intranet.web.cern.ch/egee-intranet/User-Forum

- LCG http://lcg.web.cern.ch/LCG/
- LCG User Guide https://edms.cern.ch/file/454439//LCG-2-UserGuide.pdf

- User Scenario https://edms.cern.ch/file/498081//UserScenario2.pdf

- JDL Attributes
  http://server11.infn.it/workload-grid/docs/DataGrid-01-TEN-0142-0_2.pdf
  https://edms.cern.ch/document/590869/1

- Global Grid Forum http://www.gridforum.org/
- Globus Alliance http://www.globus.org/
- VDT http://www.cs.wisc.edu/vdt/

- EGEE digital library: http://egee.lib.ed.ac.uk/

NEW!!!

further Further Information

- 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