



rijksuniversiteit
 groningen

22-10-2012

Grid Job submission

Fokke Dijkstra

Donald Smits Centre for Information Technology

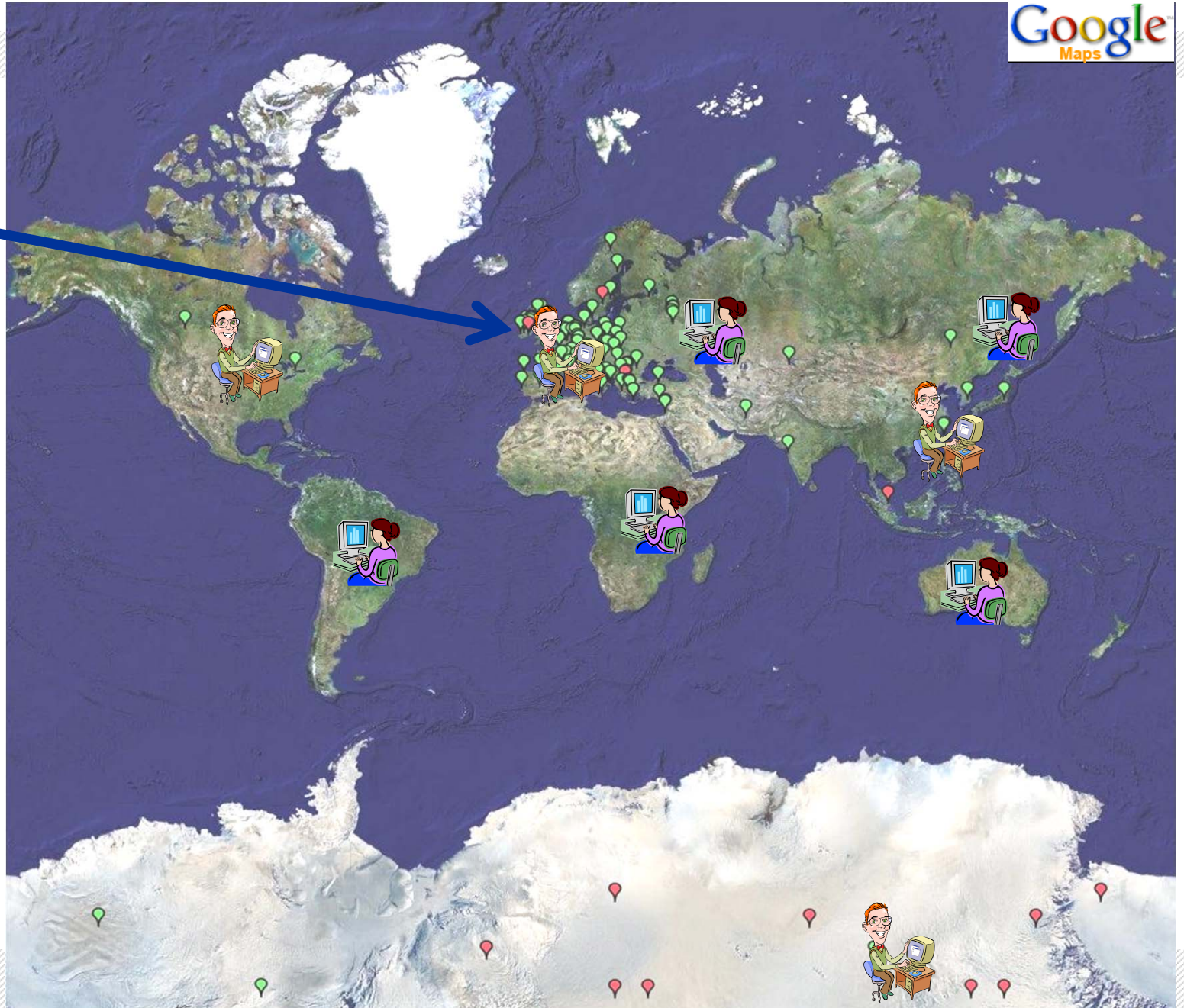


- > The Workload Management System (WMS) in EMI/UMD
- > Job Submission to the EGI Grid
 - Job Preparation
 - A simple example & Job Lifecycle
 - Job Description Language (JDL)
 - Job Submission & Monitoring
 - Some more advanced topics



rijksuniversiteit
groningen

WMS

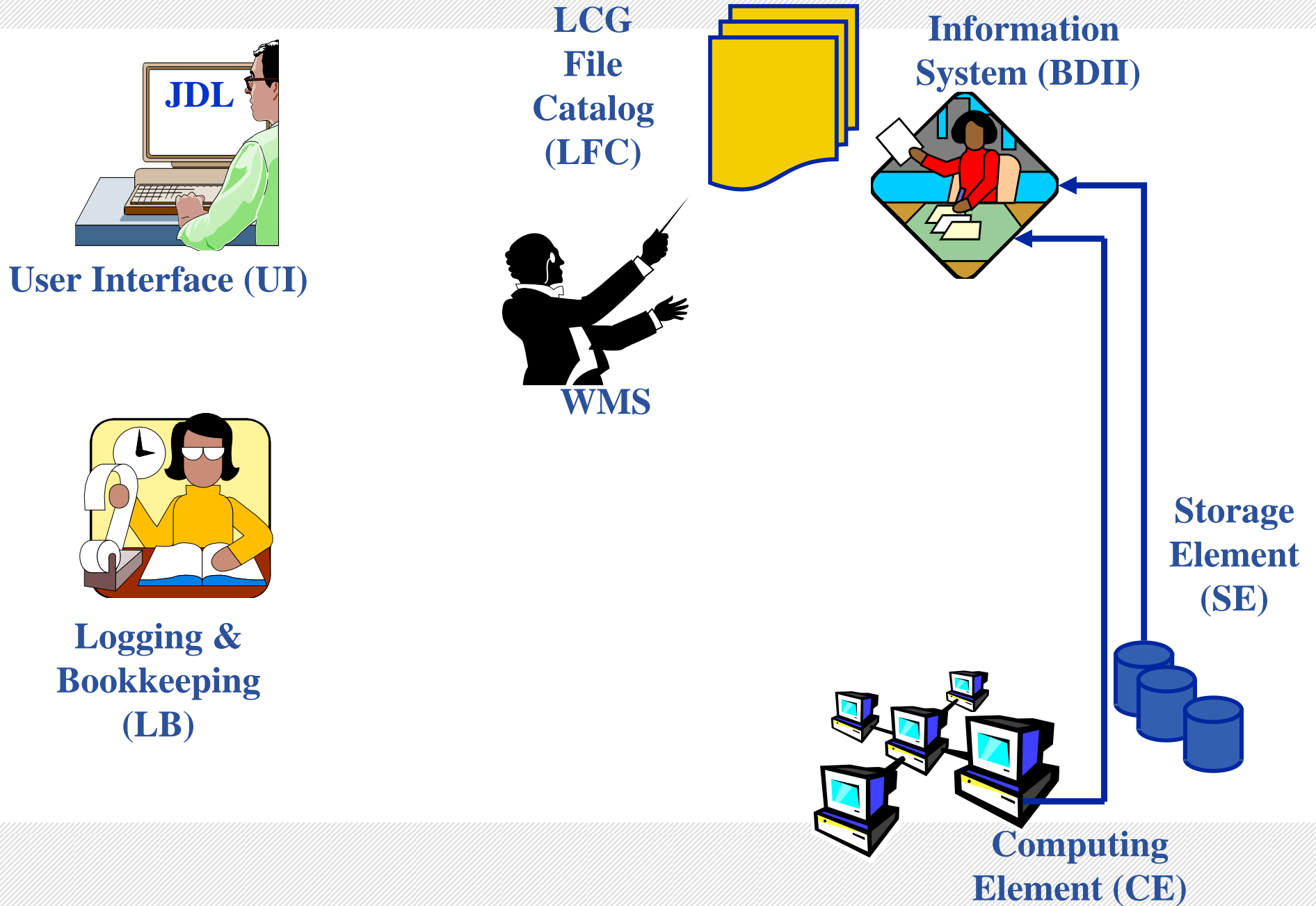




- › The user submits jobs via the Workload Management System
- › The Goal of WMS is the distributed scheduling and resource management in a Grid environment.
- › What does it allow Grid users to do?
 - To submit their jobs
 - To execute them
 - To get information about their status
 - To retrieve their output
- › **The WMS tries to**
 - **Optimize the usage of resources**
 - **Execute user jobs as fast as possible**



Grid components





- › You need to provide
 - A complete (enough) job description
 - What program?
 - What data?
 - Any requirements on OS, installed software, ??
 - Possibly a program
 - You're submitting in *unknown territory!*
 - Program portably!
 - Don't rely on hard-coded paths or special locations
 - The program you send may not even be in \$HOME!
 - Perhaps some input data
 - Perhaps instructions on what to do with the output



- › Here is a minimal job description (call it hello.jdl)

```
Executable = "/bin/echo" ;  
Arguments = "Goedemiddag" ;  
StdError = "stderr.log" ;  
StdOutput = "stdout.log" ;  
OutputSandbox = { "stderr.log", "stdout.log" } ;
```

- › We specified
 - The program to run and its arguments
 - Directed the standard error and output streams to files
 - Told it what to do with the output



- > User issues a *voms-proxy-init*
 - enters his certificate's password
 - Receives a valid X509 proxy
- > User issues a: *glite-wms-job-submit -a mytest.jdl*
and gets back from the system a unique Job Identifier (JobId)
- > User issues a: *glite-wms-job-status JobId*
to get logging information about the current status of his Job
- > When the "OutputReady" status is reached, the user can issue a
glite-wms-job-output JobId
and the system returns the name of the temporary directory where the
job output can be found on the UI machine.



```
$ voms-proxy-init --voms tutor
Enter GRID pass phrase:
Your identity: /DC=org/DC=egee-ne/O=Training Services/OU=users/CN=Fokke Dijkstra
Creating temporary proxy .....
Done
Contacting voms.grid.sara.nl:30007 [/O=dutchgrid/O=hosts/OU=sara.nl /CN=voms.grid.sara.nl]
"tutor" Done
Creating proxy ..... Done

Your proxy is valid until Tue Oct 16 22:21:03 2012

$ $ glide-wms-job-submit -a HelloWorld.jdl

Connecting to the service https://wms4.grid.sara.nl:7443/glide_wms_wmproxy_server

===== glide-wms-job-submit Success =====

The job has been successfully submitted to the WMPProxy
Your job identifier is:
https://wms4.grid.sara.nl:9000/EzeP-VIol4yf8g5wLB0QzA
```

JobId



\$ \$ glite-wms-job-status https://wms4.grid.sara.nl:9000/EzeP-VI ol 4yf8g5wLB0QzA

===== glite-wms-job-status Success =====

BOOKKEEPING INFORMATION:

Status info for the Job : https://wms4.grid.sara.nl:9000/EzeP-VI ol 4yf8g5wLB0QzA

Current Status: Done (Success)

Logged Reason(s):

- job completed
- Job Terminated Successfully

Exit code: 0

Status Reason: Job Terminated Successfully

Destination: cygnus.grid.rug.nl:8443/cream-pbs-short

Submitted: Tue Oct 16 10:22:03 2012 CEST

=====



submitted

22-10-2012

waiting

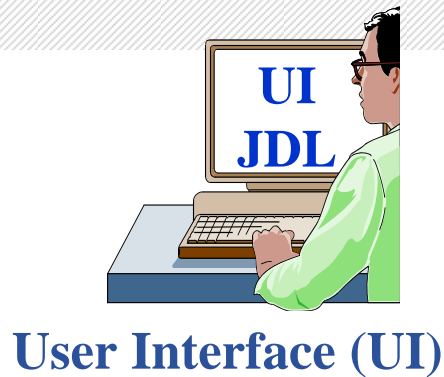
scheduled

running

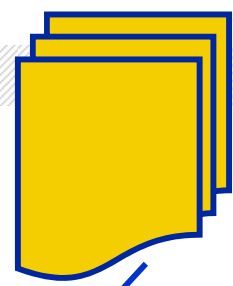
done

Storage Element

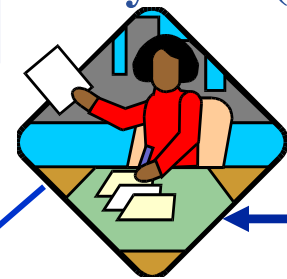
outputready



LCG
File
Catalog
(LFC)



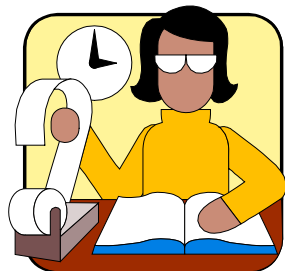
Information
System (IS)



Input Sandbox



Resource
Broker (RB)

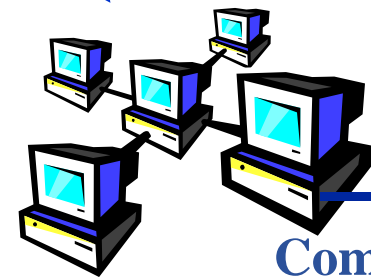


Logging &
Bookkeeping
(LB)

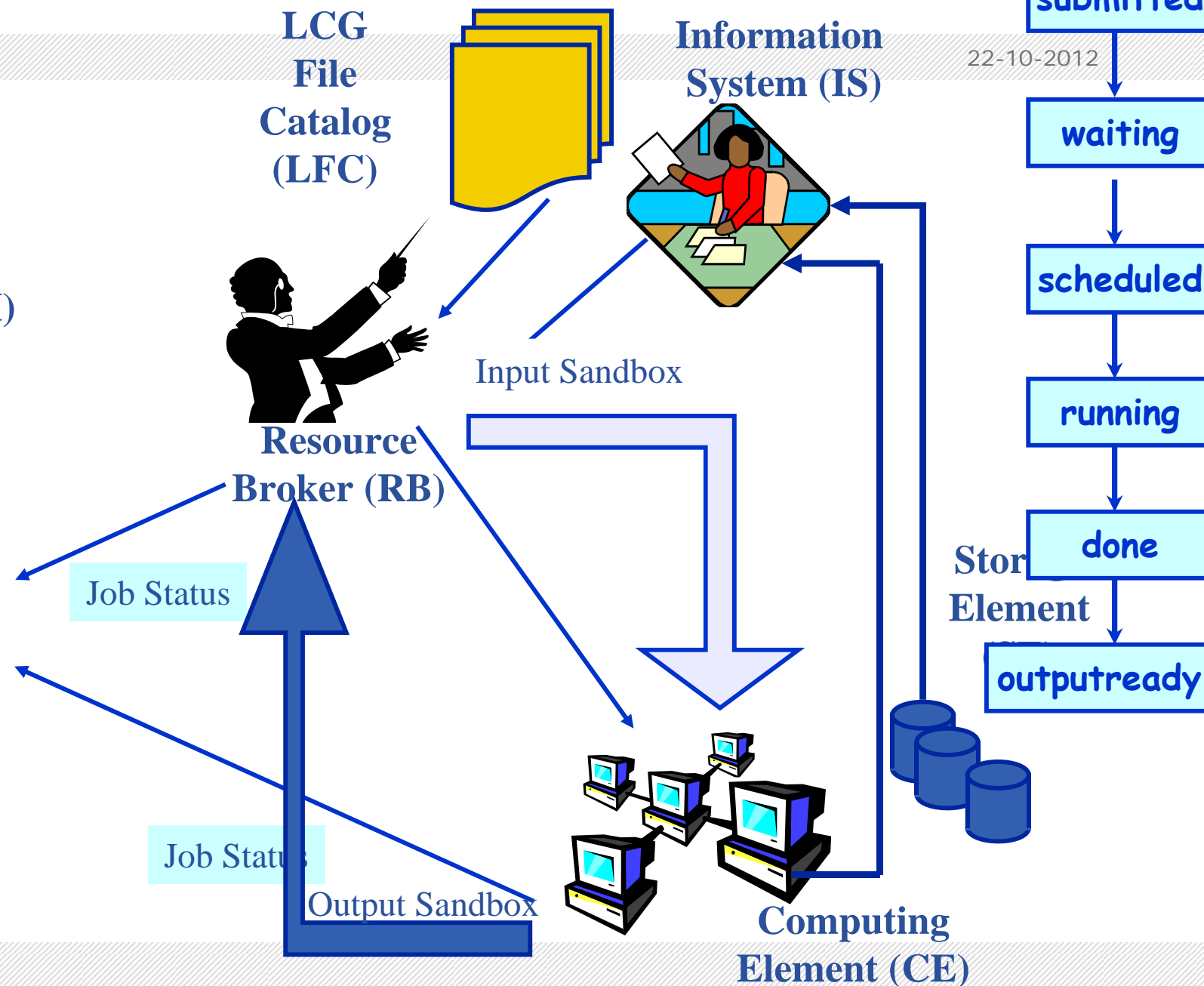
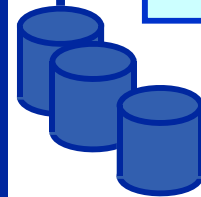
Job Status

Job Status

Output Sandbox



Computing
Element (CE)





```
$ glide-wms-job-output https://wms4.grid.sara.nl:9000/EzeP-VIol4yf8g5wLB0QzA
```

```
Connecting to the service https://wms4.grid.sara.nl:7443/glide_wms_wmproxy_server
```

```
=====
```

```
JOB GET OUTPUT OUTCOME
```

```
Output sandbox files for the job:
```

```
https://wms4.grid.sara.nl:9000/EzeP-VIol4yf8g5wLB0QzA
```

```
have been successfully retrieved and stored in the directory:
```

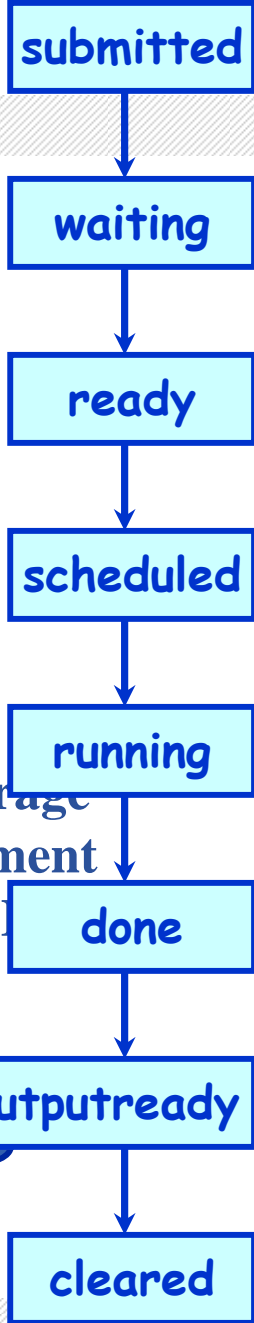
```
/tmp/jobOutput/fdi jkstra_EzeP-VIol4yf8g5wLB0QzA
```

```
=====
```

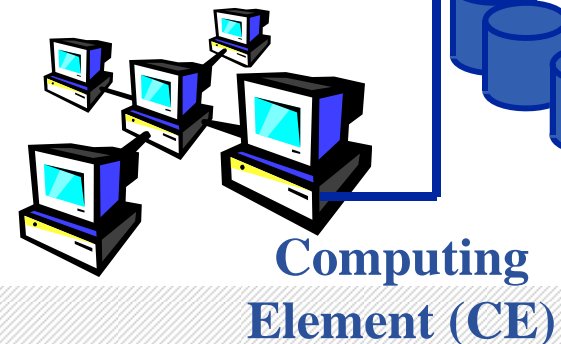


rijksuniversiteit
groningen

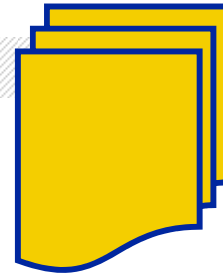
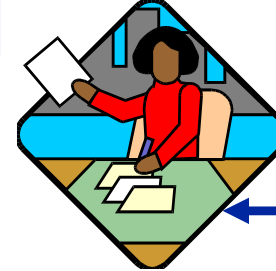
Job Status



Storage
Element
(S)



Information
System (IS)



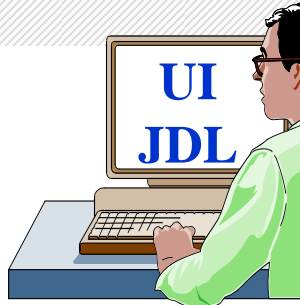
LCG
File
Catalog
(LFC)



WMS

Output Sandbox

Job Status



UI
JDL



Logging &
Bookkeeping
(LB)



- › Based upon Condor's *CLASSified ADvertisement language (ClassAd)*
- › ClassAd is an extensible language
- › Sequence of attributes (key,value pairs) separated by semi-colons.

```
Executable = "/bin/echo";  
Arguments = "Goedemiddag";  
StdError = "stderr.log";  
StdOutput = "stdout.log";  
OutputSandbox = { "stderr.log",  
"stdout.log" };
```



- › The supported attributes are grouped in two categories:
 - *Job*
 - Define the job itself
 - **Resources**
 - Taken into account by the RB for carrying out the matchmaking algorithm
 - *Computing Resource (Attributes)*
 - Used to build expressions of Requirements and/or Rank attributes by the user
 - Have to be prefixed with “**other.**”
 - *Data and Storage resources (Attributes)*
 - Input data to process, SE where to store output data, protocols spoken by application when accessing SEs



- › Executable (mandatory)
 - The command name
- › Arguments (optional)
 - Job command line arguments
- › StdInput, StdOutput, StdErr (optional)
 - Standard input/output/error of the job
- › Environment (optional)
 - List of environment settings
- › InputSandbox (optional)
 - List of files on the UI local disk needed by the job for running
 - The listed files are staged from the UI to the remote CE
- › OutputSandbox (optional)
 - List of files, generated by the job, which have to be retrieved



> Requirements

- Job requirements on computing resources
- Specified using attributes of resources published in the Information System
- If not specified, default value defined in UI configuration file is considered
 - Default: `other.GlueCEStateStatus == "Production"` (the resource has to be in the Production grid)

> Rank

- Expresses preference (how to rank resources that have already met the Requirements expression)
- Specified using attributes of resources published in the Information Service
- If not specified, default value defined in the UI configuration file is considered
 - Default: `- other.GlueCEStateFreeCPUs` (the highest number of free CPUs)



```
Executable = "gridTest";
StdError = "stderr.log";
StdOutput = "stdout.log";
InputSandbox = {"/home/joda/test/gridTest"};
OutputSandbox = {"stderr.log", "stdout.log"};
Requirements = other.Architecture=="INTEL" && \
               other.OpSys=="CentOS" && other.FreeCpus >=4;
Rank = "other.GlueHostBenchmarkSF00";
```



```
> glite-wms-job-submit [-a] [-d delegationid][-r  
<res_id>] [-c <config file>] [-o <output file>]  
<job.jdl>
```

-a Perform automatic delegation

-d Use an existing delegation identified by delegationid

-r the job is submitted by the WMS directly to the computing element identified by *<res_id>*

-c the configuration file *<config file>* is used by the UI instead of the standard configuration file

-o the generated edg_jobId is written in the *<output file>*

Useful for other commands, e.g.:

```
edg-job-status -i <input file> (or edg_jobId)
```

-i the status information about edg_jobId contained in the *<input file>* are displayed

--vo the VO under which the job will be run



- > `glite-wms-job-list-match`
 - Lists resources matching a job description**
 - Performs the matchmaking without submitting the job**
- > `glite-wms-job-cancel`
 - Cancels a given job**
- > `glite-wms-job-status`
 - Displays the status of the job**
- > `glite-wms-job-output`
 - Returns the job-output (the OutputSandbox files) to the user**
- > `glite-wms-job-logging-info`
 - Displays logging information about submitted jobs (all the events “pushed” by the various components of the WMS)**
 - Very useful for debug purposes**



- > The WMS has to find the best suitable computing resource (CE) where the job will be executed
- > It interacts with Data Management service and Information System
They supply RB with all the information required for the resolution of the matches
- > The CE chosen by the WMS has to match the job requirements (e.g. runtime environment, data access requirements, and so on)
- > If 2 or more CEs satisfy all the requirements, the one with the best Rank is chosen.



rijksuniversiteit
groningen

Further Information

- › The gLite User Guide!

<https://edms.cern.ch/document/722398/>