

L'implementazione ed il deployment di una GRID

luciano gaido (INFN- Sezione di Torino)

Workshop sulle problematiche di calcolo e reti nell'INFN
Isola d'Elba 8.5.02

Le Grid: motivazioni

- da molti anni si stanno sviluppando modelli di calcolo distribuito
- esperimenti futuri:
 - terabyte di dati, potenze di calcolo enorme che non può essere “confinata” in un sito
- “The grid aims to enable communities (the so-called Virtual Organizations) to share geographically distributed resources as they pursue common goals — in the **absence of central control, omniscience**, trust relationships”
(Ian Foster & Carl Kesselmann – CERN jan 2001)

Le Grid: il punto di partenza

Globus

(<http://www.globus.org>)

Il progetto Globus project è sviluppato alla Mathematics and Computer Science Division dei Laboratori Nazionali di Argonne, all'Information Science Institute dell'Università della Southern California's e ai Distributed Systems Laboratory dell'Università di Chicago..

Il **Globus Toolkit** è principalmente una “bag of services,” un set di componenti utili che possono essere utilizzati sia indipendentemente che in modo congiunto per sviluppare delle applicazioni di grid e dei tool vari.

I componenti principali sono:

- Globus Resource Allocation Manager (GRAM)
- Grid Security Infrastructure (GSI)
- Monitoring and Discovery Service (MDS)

Condor

(<http://www.cs.wisc.edu/condor>)

Il progetto Condor Project è sviluppato all'Università del Wisconsin.

L'obiettivo del Condor Project è studiare, sviluppare, implementare, e valutare i meccanismi e le policy che supportano l'High Throughput Computing (HTC) su grandi insiemi di risorse distribuite geograficamente e amministrativamente.

DataGrid

DataGrid è un progetto della durata di 3 anni finanziato dalla Comunità Europea.

L'obiettivo è costruire una infrastruttura di calcolo di nuova generazione che sia in grado di consentire calcolo intensivo e l'analisi di una enorme mole di dati distribuiti e condivisi, dell'ordine dei PetaBytes, nell'ambito di comunità e collaborazioni scientifiche distribuite su scala planetaria.

Data di inizio: 1 gennaio 2001.

<http://www.eu-datagrid.org>

InfnGrid

InfnGrid è fortemente legato a DataGrid. Ha alcune specificità rispetto a Datagrid.

<http://www.infn.it/grid>

Il progetto DataGrid è strutturato in vari WorkPackages (WP).

middleware

- WP1 Grid Work Scheduling
- WP2 Grid Data Management
- WP3 Grid Monitoring Services
- WP4 Fabric Management
- WP5 Mass Storage Management

infrastructure

- WP6 Testbed and Demonstrators
- WP7 Network Services

applications

- WP8 HEP (High Energy Physics) Applications
- WP9 Earth Observation Applications
- WP10 Biology Applications

management

- WP11 Dissemination
- WP12 Project Management

La Grid può essere descritta in vari modi.

Può essere rappresentata come un insieme di **servizi** disponibili per le comunità di utenti.

Un *servizio* è un programma che è continuamente in esecuzione in attesa di ricevere richieste dai clienti.

I clienti fanno le loro richieste attraverso una connessione internet standard.

Ci sono vari tipi di servizio; ognuno può essere composto da vari sottosistemi.

La Grid può essere anche rappresentata come un insieme di **grid elements** (computer, grosso modo) connessi a una rete che forniscono alcuni servizi di grid.

Il Testbed di DataGrid è composto da alcune tipologie di calcolatori chiamati *Grid Elements*:

- **User Interface (UI)**
è il nodo da cui l'utente sottomette job alla grid
- **Computing Element (CE)**
è il gateway verso una farm. Il Resource Broker invia un job al CE che poi lo smista ad uno specifico Worker Node per mezzo di uno scheduler locale. Gli scheduler supportati da DataGrid sono PBS e LSF (+BQS).
- **Worker Node (WN)**
un nodo di una farm.
- **Storage Element (SE)**
è il nodo che fornisce accesso ai dati ed ai servizi per immagazzinare, localizzare e replicare i dati. Inoltre pubblica informazioni sulla disponibilità dei dati.

Una farm è costituita da un Computing Element e un numero variabile di Worker Nodes.

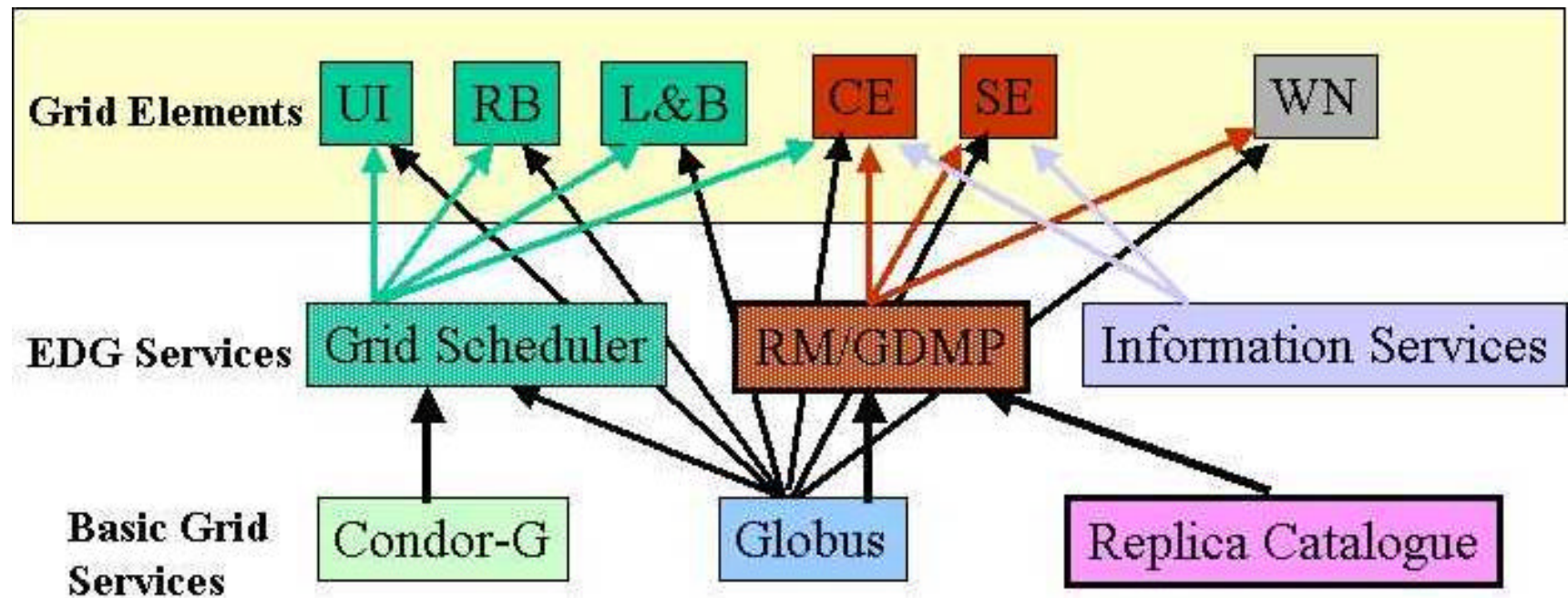
Grid Elements per *'servizi speciali'*:

- **Resource Broker (RB)**
il nodo che effettua il matching tra i requirements del job e le risorse di grid disponibili ed effettua la sottomissione del job, mediante il Job Submission System, al Computing Element “migliore”.
- **Information Index (II)**
il punto di accesso all'Information System (database) che contiene le informazioni relative ai grid elements
- **Logging & Bookkeeping Element (LB)**
il nodo che tiene traccia della storia dell'esecuzione del job; fornisce anche servizi di monitoring.

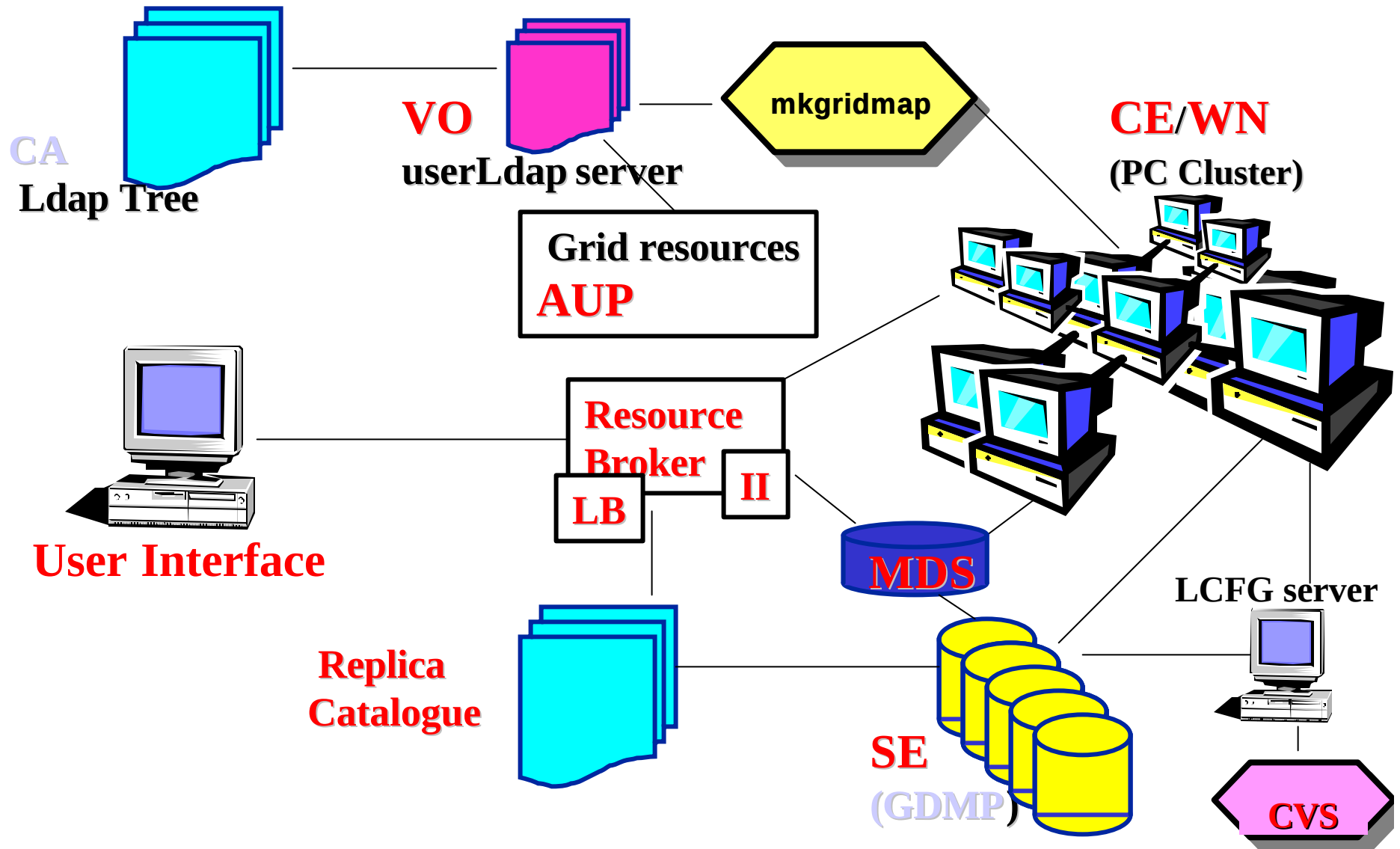
Grid Elements per *'servizi speciali'* (cont.):

- **Replica Catalog (RC)**
il nodo che contiene il database LDAP usato dal Replica Manager (RM);
contiene informazioni sui dati e sulla loro localizzazione.
- **Virtual Organization server (VO)**
il nodo che contiene il database LDAP dove sono registrati i subject dei
certificati degli utenti. Queste informazioni sono utilizzate per
l'autenticazione e l'autorizzazione degli utenti.

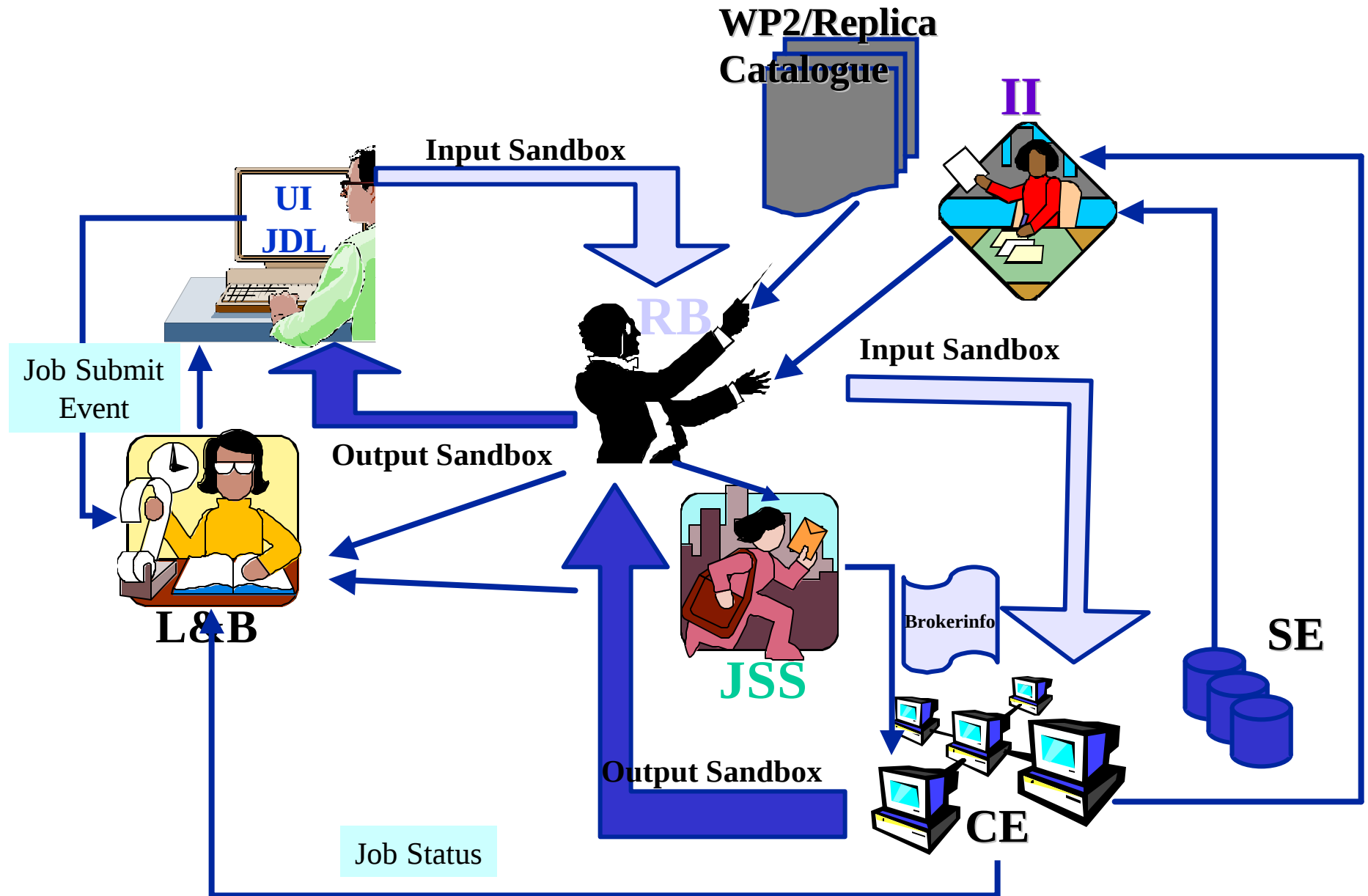
Grid elements and services in the European DataGrid (EDG)



DataGrid Release1 Services



Schema logico della sottomissione di un job



Data Management

Quando un job richiede dei dati il RB interroga il Replica Catalog per localizzarli ed effettuare il match-making.

```
#
# ----- Job Description File -----
#
Executable = "WP1testC";
StdInput = "sim.dat";
StdOutput = "sim.out";
StdError = "sim.err";
InputSandbox = {"/home/wp1/HandsOn-0409/WP1testC", "/home/wp1/HandsOn-0409/file*",
                "/home/wp1/DATA/*"};
OutputSandbox = {"sim.err", "test.out", "sim.out"};
Rank = other.AverageSI00;
Requirements = (other.OpSys == "Linux RH 6.1" || other.OpSys == "Linux RH 6.2") &&
(other.RunTimeEnvironmnet == "CMS3.2");
InputData = "LF:test10096-0009";
ReplicaCatalog = "ldap://sunlab2g.cnaf.infn.it:2010/rc=WP2 INFN Test Replica Catalog,dc=sunlab2g,
                dc=cnaf, dc=infn, dc=it";
DataAccessProtocol = "gridftp";
```



Nell'Information Index le entry relative a CE e SE contengono alcuni attributi relativi all'accessibilità dei dati

Ad esempio:

attributo **close**: l'SE 'vicino, al CE

attributo **seprotocol**: protocolli che possono essere utilizzati dallo SE per trasferire i dati

- file i dati sono su un disco locale (o montato con nfs)
- gridftp protocollo con estensioni all'ftp

I grid elements pubblicano queste informazioni nell'Information Index per mezzo dei loro **GRIS** (Globus Resource Information System), uno dei componenti del Globus MDS (Monitoring and Discovery Service).

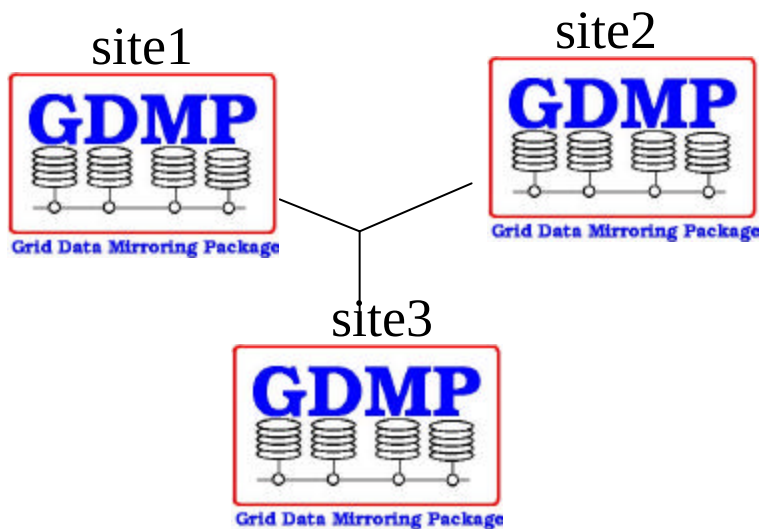
Quando un job crea un nuovo file, questo deve essere registrato nel Replica Catalog che effettua il mapping tra il logical file name e le locazioni fisiche del file.

Di questo si occupa il Grid Data Mirroring Package (**GDMP**) fornito da WP2. GDMP effettua anche la replica dei dati.

Grid Data Mirroring Package



- **client/server** file replication software system for read-only files
 - initially designed to replicate Objectivity database files
 - now: for any kind of file format



Based on:

- Globus Security Infrastructure
- Globus IO/ Globus Data Conversion
- GridFTP
- Globus Replica Catalogue

La release EDG è costituita da:

- Globus EDG distribution beta 21(basata su Globus beta 2.0) che include:
 - MDS 2.1 (Monitoring and Discovery Service, già Metacomputing Directory Service)
 - gatekeeper
 - GSI (Globus Security Infrastructure)
 - Gridftp, ncftp and wuftp
- pacchetti EDG:
 - sw prodotto da WP1, WP2, WP3, WP4, WP5
 - sw di security
 - pacchetti di configurazione e utilità varie (WP6, WP7, etc.)
 - sw delle applicazioni (WP8, WP9, WP10)
 - sw esterni

I pacchetti sono forniti in formato rpm.

L'[Integration Team](#) riceve tutti i pacchetti software dai vari work packages e Globus, studia le funzionalità e interdipendenze, i requirements per un corretto funzionamento del testbed, pianifica una topologia e produce istruzioni di installazione e configurazione per il deployment.

I grid elements possono essere installati per mezzo di **LCFG**, un tool per l'installazione automatica, prodotto dal Dipartimento di Computer Science dell'Università di Edimburgo.

Per ogni grid element un template LCFG fornisce:

- la lista degli RPM per ciascun package.
- una configurazione tipica di LCFG che deve essere personalizzata per ogni sito del testbed.
- un set specifico di istruzioni sulla configurazione del grid element.

Il Testbed

Gli scopi principali del primo Testbed di DataGrid sono:

- ✍ dimostrare che le componenti sw di DataGrid possono essere integrate in una grid computazionale funzionante
- ✍ permettere ai work package che producono il middleware di valutare l'architettura e le prestazioni della release EDG
- ✍ validare le release sw di DataGrid per mezzo delle applicazioni degli esperimenti
- ✍ fornire la nuova tecnologia agli utenti finali fornendo loro anche la possibilità di farsi un'esperienza pratica
- ✍ facilitare l'interazione ed il feedback tra gli utenti finali e gli sviluppatori

Il Testbed

Utenti:

- DataGrid: comunità HEP, Earth Observation e Biology.
- InfnGrid: oltre agli esperimenti HEP anche Virgo e APE sono coinvolti.

Per raggiungere gli scopi è stata realizzata una infrastruttura di Testbed che coinvolge alcuni paesi Europei (sostanzialmente i partner principali di DataGrid).

Il Testbed1 è il Testbed su cui sono stati implementati i servizi della Release 1 di EDG.

Layout del Testbed1 di DataGrid.

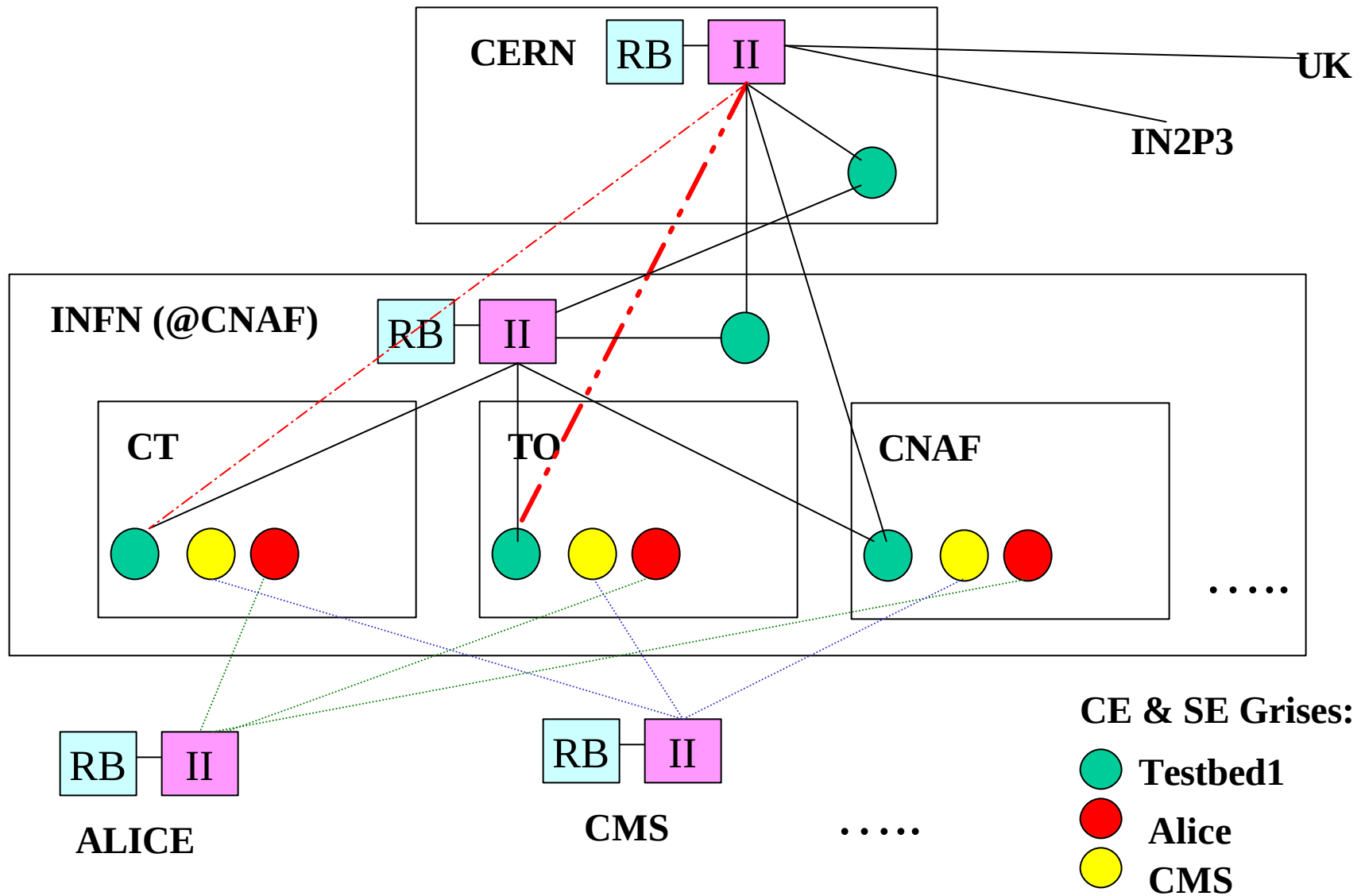
5 partners: CERN, INFN, RAL, IN2P3, Nikhef

Il Testbed1 di InfnGrid è costituito da:

- 6 siti principali (Catania, Cnaf, Milano, Padova, Pisa e Torino) che partecipano anche al TB1 di DataGrid
- 13 siti ‘minori’: Bari, Bologna, Cagliari, Ferrara, Genova, Lecce, Legnaro, Napoli, Parma, Pavia, Roma1, Roma3, Trieste.

I servizi centrali (RB, II, RC, VO LDAP server sono stati implementati al CNAF).

RB – II- GRIS layout



Una parte fondamentale nel processo di esecuzione di un job riguarda le politiche di accesso.

Per eseguire un programma sulle risorse di DataGrid gli utenti devono:

- essere autenticati
- essere autorizzati

Questi processi sono costituiti da vari step.

Gli amministratori locali hanno il completo controllo sulle loro risorse.

Autenticazione

Ogni utente della grid deve essere autenticato per mezzo di un certificato X509 rilasciato da una Certification Authority (CA).

Quasi tutti i partner di DataGrid hanno la loro CA:

CA	CA Certificate	CA CRL
CERN	c35c1972.0	c35c1972.r0
Czech Republic - CESNET	ed99a497.0	ed99a497.r0
France - CNRS	cf4ba8c8.0	cf4ba8c8.r0
France - CNRS-Projets	34a509c3.0	34a509c3.r0
France - CNRS Datagrid-fr	6b4ddd18.0	6b4ddd18.r0
Ireland - Grid-Ireland	1e43b9cc.0	1e43b9cc.r0
Italy - INFN	df312a4e.0 <i>PhP script</i>	df312a4e.r0 <i>DER format</i>
Netherlands - NIKHEF	16da7552.0	16da7552.r0
Nordic countries - NorduGrid	1f0e8352.0	1f0e8352.r0
Portugal - LIP	41380387.0	41380387.r0
Russia - Russian DataGRID	d64ccb53.0	d64ccb53.r0
Spanish - DATAGRID-ES	90e2484f.0	90e2484f.r0
United Kingdom - GridPP	0ed6468a.0	0ed6468a.r0

Autorizzazione degli utenti.

Per poter utilizzare le risorse della grid gli utenti devono essere autorizzati.
Il processo di autorizzazione può essere suddiviso in due fasi:

- ✍ 'sign-on' sulla UI mediante il comando `grid-proxy-init`
Mediante il meccanismo dei proxy certificate un solo sign-on è sufficiente
- ✍ autorizzazione all'uso delle risorse sul Resource Broker, sui Computing Elements e gli Storage Elements

L'autenticazione è realizzata utilizzando la Globus Security Infrastructure:
l'utente devono collocare il loro certificato in formato PEM ([usercert.pem](#) e [userkey.pem](#)) nella sottodirectory `.globus`.

- `usercert.pem` contiene la chiave pubblica (deve essere aperta in lettura a tutti)
- `userkey.pem` contiene la chiave privata (deve essere leggibile solo dall'utente)

Autorizzazione degli utenti.

Il secondo step (l'autorizzazione vera e propria) è effettuata per mezzo di un file speciale (**gridmapfile**) sui grid elements.

Per ogni utente il gridmapfile mappa il subject del certificato ad un account locale (worker account) che sarà utilizzato dal sistema operativo per eseguire il job.

Sul **Resource Broker** il subject del certificato di ogni grid user è mappato sull'account locale usato dal processo server del broker (ad es. wp1)

Per facilitare la ricostruzione di chi ha fatto cosa nel processo di esecuzione dei job (questi dati devono essere forniti all'Accounting System) si è deciso che sui **Computing Element** e sugli **Storage Element** il subject del certificato di ciascun utente deve essere mappato su un utente locale diverso.

Gli account locali possono essere statici o dinamici (creati al volo, secondo le necessità, oppure pool di account)

Il meccanismo dei pool di account è quello più utilizzato in DataGrid.

Esempio di gridmapfile

```
#####  
## InfnGrid Testbed1 users (experiment-independent people)  
#####  
"/C=IT/O=INFN/L=Torino/CN=Andrea Guarise/Email=Andrea.Guarise@to.infn.it" user01  
"/C=IT/O=INFN/L=Milano/CN=Francesco Prelz/Email=francesco.prelz@mi.infn.it" user03  
"/C=IT/O=INFN/L=CNAF/CN=Francesco Giacomini/Email=Francesco.Giacomini@cnafe.infn.it" user04  
"/C=IT/O=INFN/L=Padova/CN=Massimo Sgaravatto/Email=massimo.sgaravatto@pd.infn.it" user05  
"/C=IT/O=INFN/L=Catania/CN=Carlo Rocca/Email=Carlo.Rocca@ct.infn.it" user06  
#####  
## ALICE Testbed1 users  
#####  
O=dutchgrid/O=users/O=nikhef/CN=Michiel Botje" user32  
"/C=FR/O=CNRS/OU=SUBATECH/CN=Yves Schutz/Email=schutz@in2p3.fr" user33  
"/O=Grid/O=CERN/OU=cern.ch/CN=Federico Carminati" user36  
"/O=Grid/O=CERN/OU=cern.ch/CN=Predrag Buncic" user37  
"/C=IT/O=INFN/L=Catania/CN=Roberto Barbera/Email=roberto.barbera@ct.infn.it" user38  
"/C=IT/O=INFN/OU=Personal Certificate/L=Torino/CN=Chiara \  
Oppedisano/Email=Chiara.Oppedisano@to.infn.it" user40  
"/C=IT/O=INFN/OU=Personal Certificate/L=Bari/CN=Domenico Di Bari \  
/Email=domenico.dibari@ba.infn.it" user41  
"/C=IT/O=INFN/L=Torino/CN=Piergiorgio Cerello/Email=Piergiorgio.Cerello@to.infn.it" user42
```

Creazione dei gridmapfile

Per facilitare la creazione dei gridmafile i subject dei certificati degli utenti sono stati raccolti in opportune repository, gestite dalla Virtual Organization (esperimento, Istituzione, collaborazione) a cui gli utenti appartengono.

Il repository della VO è un server LDAP con una struttura gerarchica che consente anche la creazione di gruppi di utenti.

Ogni server LDAP è amministrato da un VO manager che può inserire, modificare, cancellare e raggruppare i subject di certificati (e quindi utenti).

Sono stati sviluppati alcuni tools dall'Authorization Group per la gestione del processo di autorizzazione sulla grid.

Mkgridmap, ad es., consente di ottenere il subject dei certificati di un gruppo di utenti dal server della VO e creare il gridmapfile.

N.B. Gli amministratori locali hanno il controllo completo sul grid-mapfile

Virtual Organization Authorization servers

In questo momento esistono 8 VO LDAP servers:

ALICE	grid-vo.nikhef.nl	o=alice,dc=eu-datagrid,dc=org
ATLAS	grid-vo.nikhef.nl	o=atlas,dc=eu-datagrid,dc=org
CMS	grid-vo.nikhef.nl	o=cms,dc=eu-datagrid,dc=org
LHCb	grid-vo.nikhef.nl	o=lhcb,dc=eu-datagrid,dc=org
WP6	marianne.in2p3.fr	o=testbed,dc=eu-datagrid,dc=org
InfnGrid	grid-vo.infn.it	o=infngrid,dc=eu-datagrid,dc=org
EarthOb	grid-vo.nikhef.nl	o=earthob,dc=eu-datagrid,dc=org
Biomedical	grid-vo.nikhef.nl	o=biomedical,dc=eu-datagrid,dc=org

Bug reporting

Ogni problema o bug incontrato durante le fasi di test e validazione della grid può essere riportato ad un Bug-tracking System.

In DataGrid si usa [Bugzilla](http://marianne.in2p3.fr/datagrid/bugzilla).

[\[http://marianne.in2p3.fr/datagrid/bugzilla\]](http://marianne.in2p3.fr/datagrid/bugzilla)

Il sistema è stato pensato per facilitare il report e la risoluzione dei bug nel codice di Globus e di EDG.

I bug relativi a Globus sono riportati al Globus Team, mentre quelli di EDG al WP di DataGrid di pertinenza.



DataGrid Bug Tracking

Bug Reports

New Bug	Enter a new bug report.
Summary	Get summaries of reported bugs.
Search Form	Search existing bug reports.
Quick Search	Show

Account Information

Logout	Forget cookie identifying you to Bugzilla.
Preferences	Change user password or preferences.
New Account	Open a new Bugzilla account.

The European DataGrid uses [Bugzilla](#), the [Mozilla.org](#) bug system, to track bugs in the DataGrid software.

General Information

The bug-tracking system is intended to facilitate the reporting and fixing of bugs in the **European DataGrid** software. This includes the DataGrid's distribution of the Globus2 system; confirmed bugs in Globus will be forwarded to the Globus team.

This system is **not** intended to track bugs in application software, that is, user and experimental software running on the grid. For these types of problems, please refer to the list of application support contacts.

Please try to submit **concise bug reports**. Stripped down test cases and detailed explanations are greatly appreciated and will speed-up resolution of the reported bug. ([guidelines](#))

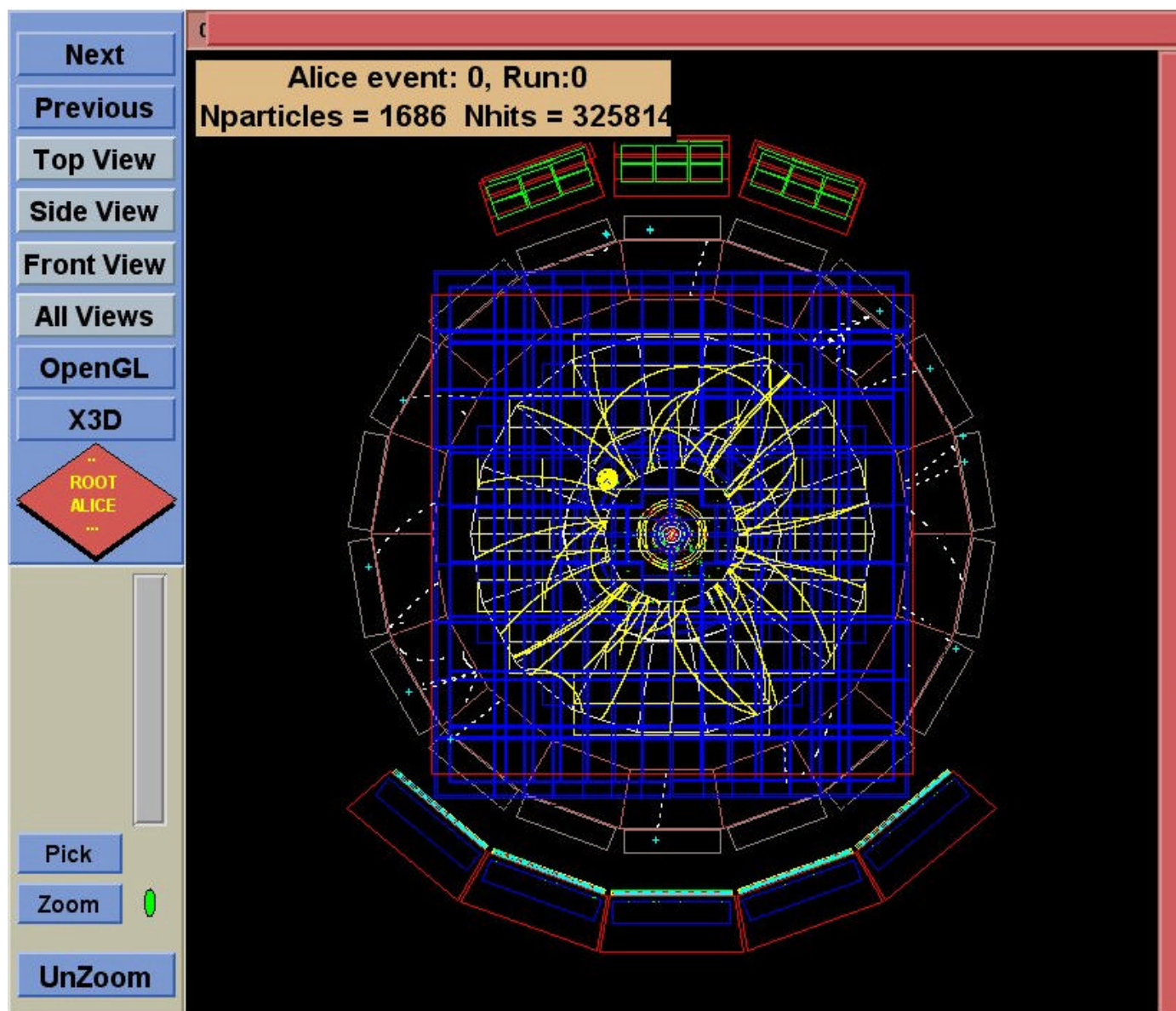
User Support

DataGrid Bugs	Submit bug report.
Operational Problems	Contact the site administrator (see contact list or try mailing <a href="mailto:grid.support@<site>">grid.support@<site>)
Application Problems	Try support contact person for your application or experiment.
User's Guide	Documentation regarding use of the Testbed 1 DataGrid.

System Administrator Support

DataGrid Bugs	Submit bug report.
Installation	Documentation regarding installation and configuration of the DataGrid software. Contact the Integration Team for guidance.
Incidents	Hardware incidents which disrupt the DataGrid should be reported via the bug-tracking system. This is intended to allow us to categorize faults and improve the fault-tolerance of the DataGrid software.

The first ALICE event produced on the DataGrid Testbed1



Documentation

European DataGrid

<http://www.eu-datagrid.org>

InfnGrid

<http://www.infn.it/grid>

DataGrid Testbed

<http://marianne.in2p3.fr>

InfnGrid Testbed

<http://www.infn.it/testbed-grid>

E' questo è tutto..... o quasi.

DataGrid Testbed1 usage

The following slides will show how to:

- submit a job
- verify the status of a job
- retrieve the output
- check which resources are available for the job
- submit a job to a specific resource
- cancel a job

Simple job submission example

Prerequisite: user authentication on the UI.

Each grid user needs a user account on the User Interface.
The user certificate's keys must be stored in the .globus directory.

```
[gaido@grid006 ~]$ grid-proxy-init
```

```
Your identity: /C=IT/O=INFN/L=Torino/CN=Luciano Gaido/Email=luciano.gaido@to.infn.it
```

```
Enter GRID pass phrase for this identity:
```

```
Creating proxy ..... Done
```

```
Your proxy is valid until Tue Feb 12 22:33:42 2002
```

Simple job submission example

A simple “Hello” job

```
[gaido@grid006 ~]$ more hello.jdl
```

```
#####
```

```
#
```

```
#-----Job Description File :
```

```
#    print "Hello World"
```

```
#
```

```
#####
```

```
Executable  = "/bin/echo";
```

```
Arguments   = "This job has been executed on CE: `/bin/hostname`: and submitted from UI: \ngrid006.to.infn.it";
```

```
StdOutput   = "message.txt";
```

```
StdError    = "stderr.log";
```

```
OutputSandbox = "message.txt";
```

Job submission

```
[gaido@grid006 ~]$ dg-job-submit hello.jdl
```

```
Connecting to host grid004f.cnaf.infn.it, port 7771
```

```
Lookup:InputSandbox...No
```

```
Logging to host grid004f.cnaf.infn.it, port 15830
```

```
*****
```

JOB SUBMIT OUTCOME

```
The job has been successfully submitted to the Resource Broker.
```

```
Use dg-job-status command to check job current status. Your job identifier (dg_jobId) is:
```

```
- https://grid004f.cnaf.infn.it:7846/192.135.19.46/095228232744527?grid004f.cnaf.infn.it:7771
```

```
*****
```

Checking the status of a job

```
[gaido@grid006 ~]$ dg-job-status
```

```
"https://grid004f.cnaf.infn.it:7846/192.135.19.46/100042232905186?grid004f.cnaf.infn.it:7771"
```

Retrieving Information from server.

Please wait: this operation could take some seconds.

```
*****
```

BOOKKEEPING INFORMATION:

Printing status info for the Job :

```
https://grid004f.cnaf.infn.it:7846/192.135.19.46/100042232905186?grid004f.cnaf.infn.it:7771
```

Some bookkeeping information has not reached the LB server yet.

Missing information should come from GlobusJobmanager

```
---
```

```
dg_JobId      =
```

```
https://grid004f.cnaf.infn.it:7846/192.135.19.46/100042232905186?grid004f.cnaf.infn.it:7771
```

```
Status        = Done 
```

```
Last Update Time (UTC) = Tue Feb 12 10:03:47 2002
```

```
Job Destination    = dell10.cnaf.infn.it:2119/jobmanager-pbs/workq 
```

```
Status Reason      = terminated 
```

```
Job Owner          = /C=IT/O=INFN/L=Torino/CN=Luciano Gaido/Email=luciano.gaido@to.infn.it
```

```
Status Enter Time (UTC) = Tue Feb 12 10:00:56 2002
```

```
*****
```

Retrieving the output

```
[gaido@grid006 ~]$ dg-job-get-output  
"https://grid004f.cnaf.infn.it:7846/192.135.19.46/100042232905186?grid004f.cnaf.infn.it:7771"
```

Retrieving OutputSandbox files...

JOB GET OUTPUT OUTCOME

Output sandbox files for the job:

- https://grid004f.cnaf.infn.it:7846/192.135.19.46/100042232905186?grid004f.cnaf.infn.it:7771
have been successfully retrieved and stored in the directory:
/tmp/100042232905186

```
[gaido@grid006 ~]$ cd /tmp/100042232905186
```

```
[gaido@grid006 100042232905186]$ ls -l
```

total 4

```
-rw-r--r--  1 gaido  tbusers   97 Feb 12 11:03 message.txt
```

```
[gaido@grid006 100042232905186]$ cat message.txt
```

This job has been executed on CE: dell10.cnaf.infn.it: and submitted from UI: grid006.to.infn.it

Checking which resources are available for this job (aka match the user's requirements)

```
[gaido@grid006 ~]$ dg-job-list-match hello.jdl
```

```
Connecting to host grid004f.cnaf.infn.it, port 7771
```

```
*****
```

COMPUTING ELEMENT IDs LIST

The following CE(s) matching your job requirements have been found:

- dell10.cnaf.infn.it:2119/jobmanager-pbs-workq
- grid001.ct.infn.it:2119/jobmanager-pbs-workq
- grid001.pd.infn.it:2119/jobmanager-lsf-grid01pd
- grid001.pd.infn.it:2119/jobmanager-lsf-grid02pd
- grid002.to.infn.it:2119/jobmanager-pbs-workq

```
*****
```

Forcing the submission of the job to a specific resource

```
[gaido@grid006 ~]$ dg-job-submit hello.jdl \  
-resource grid001.pd.infn.it:2119/jobmanager-lsf-grid01pd
```

Connecting to host grid004f.cnaf.infn.it, port 7771

Lookup:InputSandbox...No

Logging to host grid004f.cnaf.infn.it, port 15830

JOB SUBMIT OUTCOME

The job has been successfully submitted to the Resource Broker.

Use dg-job-status command to check job current status. Your job identifier (dg_jobId) is:

- <https://grid004f.cnaf.infn.it:7846/192.135.19.46/103714233509253?grid004f.cnaf.infn.it:7771>

```
[gaido@grid006 ~]$ dg-job-status \  
" https://grid004f.cnaf.infn.it:7846/192.135.19.46/103714233509253?grid004f.cnaf.infn.it:7771"
```

Retrieving Information from server.

Please wait: this operation could take some seconds.

BOOKKEEPING INFORMATION:

Printing status info for the Job :

<https://grid004f.cnaf.infn.it:7846/192.135.19.46/103714233509253?grid004f.cnaf.infn.it:7771>

Some bookkeeping information has not reached the LB server yet.

Missing information should come from GlobusJobmanager

dg_JobId =

<https://grid004f.cnaf.infn.it:7846/192.135.19.46/103714233509253?grid004f.cnaf.infn.it:7771>

Status = Done

Last Update Time (UTC) = Tue Feb 12 10:40:22 2002

Job Destination = grid001.pd.infn.it:2119/jobmanager-lsf/grid01pd

Status Reason = terminated

Job Owner = /C=IT/O=INFN/L=Torino/CN=Luciano

Gaido/Email=luciano.gaido@to.infn.it

Status Enter Time (UTC) = Tue Feb 12 10:40:17 2002



Cancelling a job

```
[gaido@grid006 ~]$ dg-job-submit hello.jdl
```

```
Connecting to host grid004f.cnaf.infn.it, port 7771
```

```
Lookup:InputSandbox...No
```

```
Logging to host grid004f.cnaf.infn.it, port 15830
```

```
*****
```

JOB SUBMIT OUTCOME

```
The job has been successfully submitted to the Resource Broker.
```

```
Use dg-job-status command to check job current status. Your job identifier (dg_jobId) is:
```

```
- https://grid004f.cnaf.infn.it:7846/192.135.19.46/162132237024641?grid004f.cnaf.infn.it:7771
```

```
*****
```

Cancelling a job

```
[gaido@grid006 ~]$ dg-job-status
```

```
"https://grid004f.cnaf.infn.it:7846/192.135.19.46/162132237024641?grid004f.cnaf.infn.it:7771"
```

```
.....
```

```
Printing status info for the Job :
```

```
https://grid004f.cnaf.infn.it:7846/192.135.19.46/162132237024641?grid004f.cnaf.infn.it:7771
```

```
dg_JobId          =
```

```
https://grid004f.cnaf.infn.it:7846/192.135.19.46/162132237024641?grid004f.cnaf.infn.it:7771
```

```
Status           = Scheduled
```

```
Last Update Time (UTC) = Tue Feb 12 16:21:18 2002
```

```
Job Destination    = dell10.cnaf.infn.it:2119/jobmanager-pbs/workq
```

```
Status Reason      = initial
```

```
.....
```

```
[gaido@grid006 ~]$ dg-job-cancel \
```

```
https://grid004f.cnaf.infn.it:7846/192.135.19.46/162132237024641?grid004f.cnaf.infn.it:7771
```

```
Are you sure you want to remove specified job(s)? [y/n]n:y
```

```
Cancel request submitted to RB "grid004f.cnaf.infn.it". Waiting for job(s) cancellation results....
```

Cancelling a job

```
[gaido@grid006 ~]$ dg-job-status \  
"https://grid004f.cnaf.infn.it:7846/192.135.19.46/162132237024641?grid004f.cnaf.infn.it:7771"
```

Retrieving Information from server.
Please wait: this operation could take some seconds.

```
*****  
BOOKKEEPING INFORMATION:
```

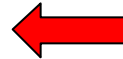
Printing status info for the Job :

```
https://grid004f.cnaf.infn.it:7846/192.135.19.46/162132237024641?grid004f.cnaf.infn.it:7771
```

```
.....
```

```
dg_JobId      =  
https://grid004f.cnaf.infn.it:7846/192.135.19.46/162132237024641?grid004f.cnaf.infn.it:7771
```

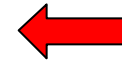
```
Status        = Done (Cancelled)
```



```
Last Update Time (UTC) = Tue Feb 12 16:24:40 2002
```

```
Job Destination = dell10.cnaf.infn.it:2119/jobmanager-pbs/workq
```

```
Status Reason   = cancelled on ResourceBroker/grid004f.cnaf.infn.it
```



```
Job Owner       = /C=IT/O=INFN/L=Torino/CN=Luciano
```

```
Gaido/Email=luciano.gaido@to.infn.it
```

```
Status Enter Time (UTC) = Tue Feb 12 16:21:47 2002
```

```
*****
```