

The EGEE Grid and basics of application porting

*Gergely Sipos sipos@sztaki.hu
Training and user support*

*MTA SZTAKI
www.lpds.sztaki.hu*

<http://indico.cern.ch/conferenceDisplay.py?confId=57245>

- **What is EGEE and how to get access**
 - In general
 - To HunGrid
 - To GILDA training infrastructure
- **Executing a simple job**
 - With practical
- **Executing more complex jobs**
 - With practical
- **Working with large datasets**
 - With practical
- **Further information and services for grid users**

- A Grid is the combination of networked resources and the corresponding middleware, which provides services for the user.



- Aim of EGEE:**

“to establish a seamless European Grid infrastructure for the support of the European Research Area (ERA)”

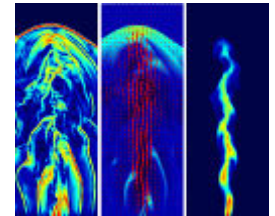
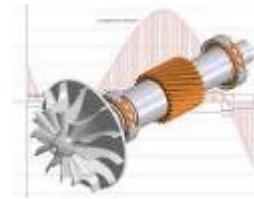
- EGEE**

- 1 April 2004 – 31 March 2006
- 71 partners in 27 countries, federated in regional Grids



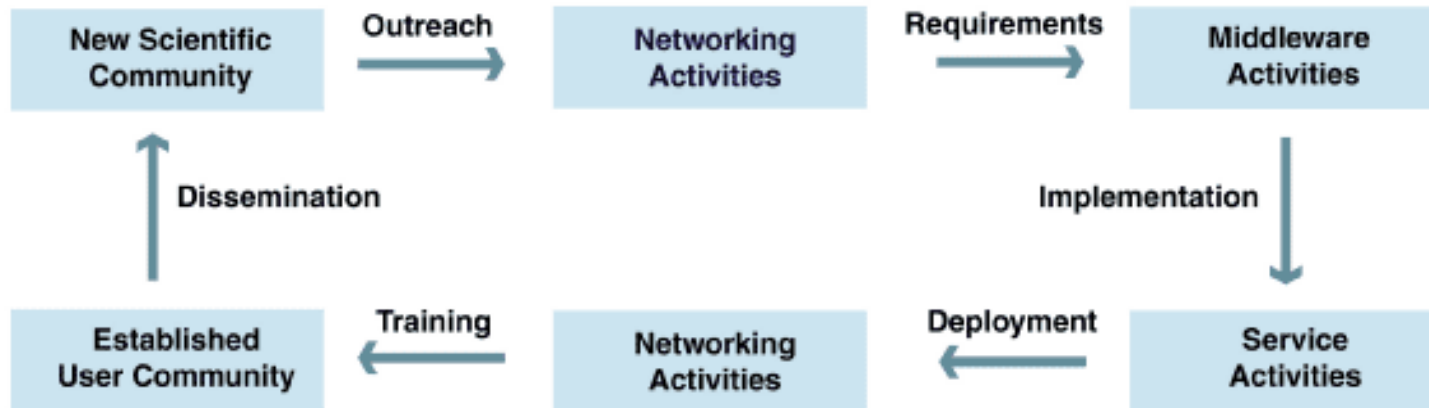
- EGEE-II**

- 1 April 2006 – 30 April 2008
- Expanded consortium



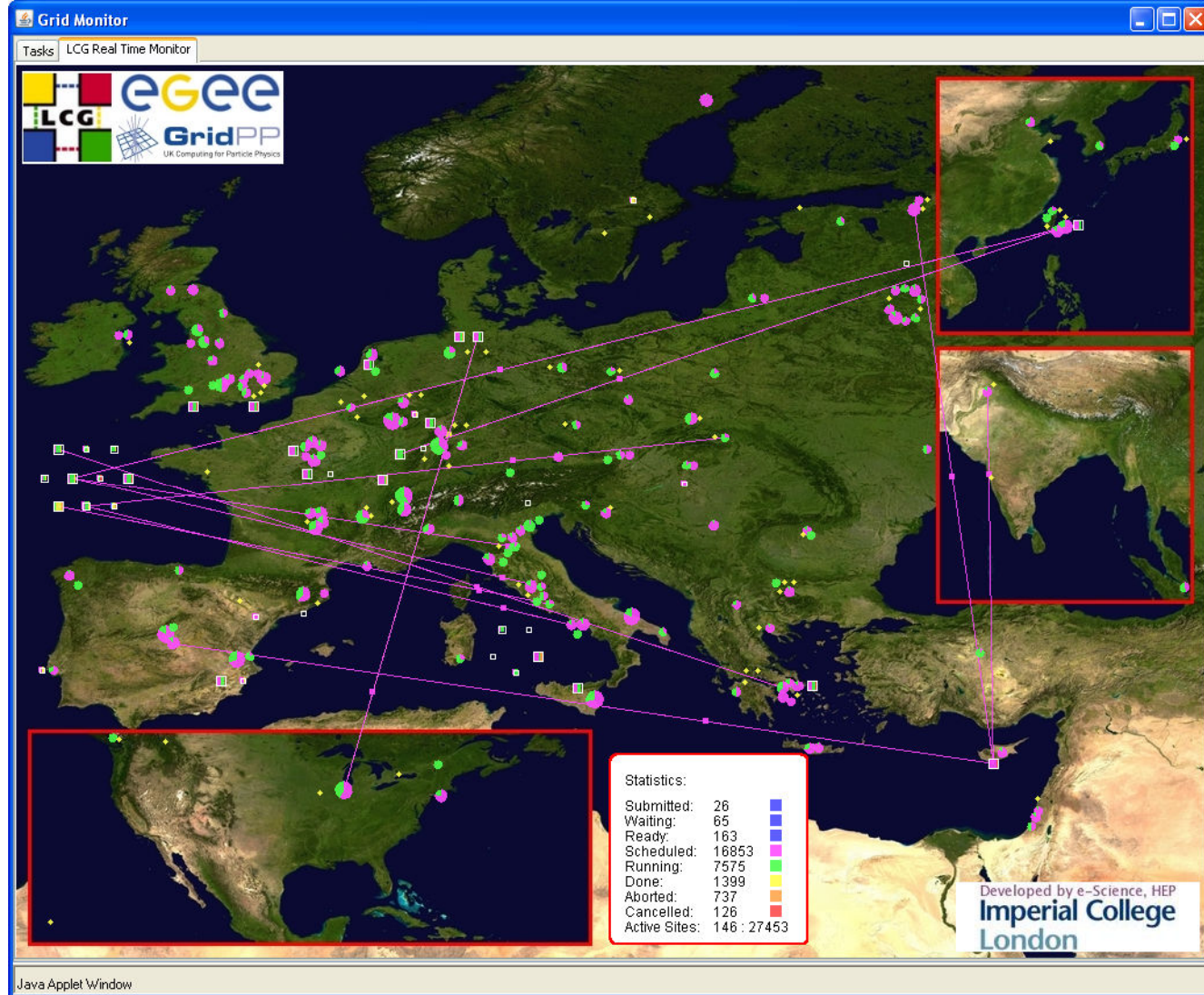
- EGEE-III**

- 1 May 2008 – 30 April 2010
- Transition to sustainable model



Real Time Monitor

- Java tool
- Displays jobs running (submitted through RBs)
- Shows jobs moving around world map in real time, along with changes in status



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

(snapshot 16 January 2007)



- Computing service
- Storage service
- Security services

- grid.ucy.ac.cy
- IBM eServer 326m Machines
 - 2.0GHz dual AMD Opteron
 - 64-bit CPUs
 - 1GB RAM
 - 80 CPUs

- **Production service**
 - Scaling up the infrastructure with resource centres around the globe
 - Running only well-tested and reliable middleware
 - 200 sites, 40 countries, ~38.000 CPUs, ~5 PB storage
 - Separated into ~200 Virtual Organizations
 - Hungrid is one of these!

- **Pre-production service**
 - Run in parallel with the production service (restricted number of sites)
 - First deployment of new versions of the gLite middleware

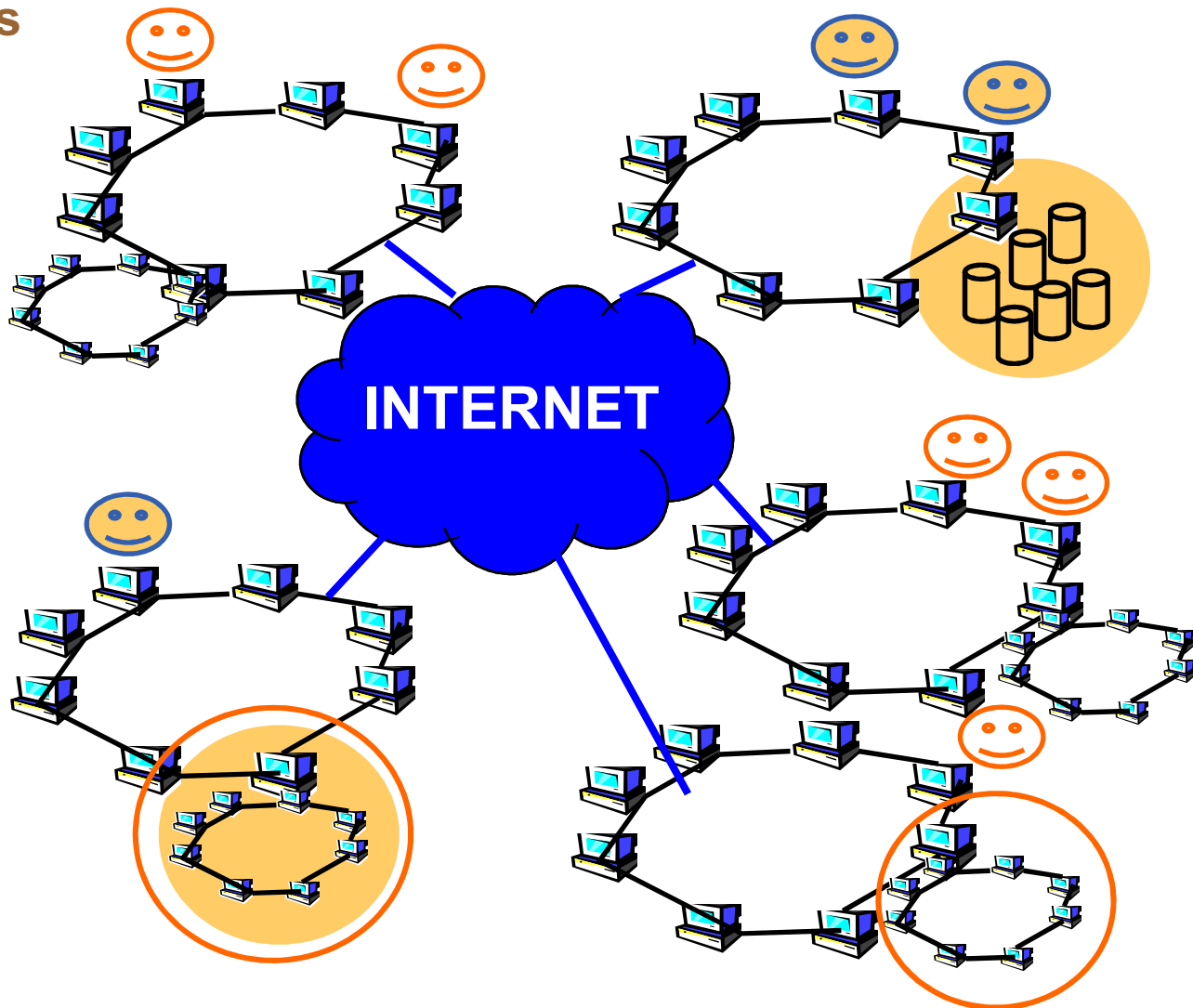
- **T-Infrastructure (Training&Education) - GILDA**
 - Complete suite of Grid elements
 - Everyone can register and use GILDA for training and testing



- **gLite middleware runs on each shared resource to provide**
 - Data services
 - Computation services
 - Security service

- **Resources and users form Virtual organisations: basis for collaboration**

- **Distributed services (both people and middleware) enable the grid**





A Hungrid Virtuális Szervezet

Magyar English

Főoldal Szolgáltatások Regisztráció Egyéni felhasználók Intézményeknek GyIK Linkek Kapcsolat

A Hungrid Virtuális Szervezet

A Hungrid virtuális szervezet (VO) az EGEE hivatalos magyar virtuális szervezete. A 2005-ben létrehozott magyar kezdeményezésű virtuális szervezet célja, hogy elérhetővé tegye a hazai kutatói közösség számára az EGEE projekt keretében kifejlesztett és üzemeltetett hazai Grid infrastruktúrát.

A VO megalapításával lehetőség nyílik a magyarországi tudományos Grid megteremtésére, kialakítására, amelyet a tudományos élet legkülönbözőbb szakterületein munkálkodó kutatók, oktatók és diákok minél szélesebb rétegei használhatnak. A létrehozott Hungrid egy olyan virtuális szervezet, amely nyitva áll bárki előtt, aki támogatni és használni szeretné az LHC Grid-et akár tudományos, akár oktatási célokra.

A Hungrid VO jelenleg több hazai intézmény (KFKI-RMKI, SZTAKI, BME, ELTE, NIIF) Grid erőforrásait fogja össze és teszi elérhetővé olyan hazai kutatói közösségek számára, melyek nagy számítási és/vagy tárolási kapacitást igényelnek, vagy csak éppen ki akarják próbálni a Grid rendszerek nyújtotta lehetőségeket.

A Hungrid az első 24 órás Grid szolgáltatás Magyarországon. Szekvenciális és paralell programok futtatása mellett a fájlok/programok tárolása, regisztrációja, webes felhasználói felület és részletes információs rendszer is megtalálható a szolgáltatások között.

A Hungrid jelen pillanatban is fejlődik, a felhasználás menete és szabályai pontosítodnak, a rendelkezésére álló erőforrás gyarapszik. Jelenlegi formájában bárki, aki a magyar kutatászférában dolgozik, és egy Grid rendszerre indokoltan szüksége van, igénybe veheti.

Magyar Grid Kompetencia Központ

2003-ban az 5 legaktívabb Grid kutatói közösség létrehozta a Magyar Grid Kompetencia Központot (MGKK) azzal a céllal, hogy a központ a kutatási tevékenységeket, pályázatokat koordinálja, az elért eredményeket pedig minél szélesebb körben publikálja. Az alapítás óta eltelt években az alapító tagok (BME, ELTE, KFKI-RMKI, MTA-SZTAKI, NIIF) számos közös hazai és nemzetközi projektben vettek ill. vesznek részt. A központhoz később csatlakozott a 4D Soft. Számítástechnikai Kft is. Az MGKK két regionális Grid központhoz is csatlakozik, a Közép-Európai Grid Konzorciumhoz (CEGC) és a Dél-Kelet-Európai Grid Konzorciumhoz.

Az MGKK tagintézményeinek eddigi munkája megalapozza azt az elképzelést, miszerint a Nemzeti Grid Iniciatíva (NGI) az MGKK vezetésével jöhetne létre. A tervezett magyar NGI egyik fő platformja a Hungrid lesz.

Az EGEE projekt

Az Európai Unió 6. keretprogramja által támogatott EGEE (Enabling Grids for E-scienceE) projekt keretében létrejött az első igazán világméretű, 24 órás szolgáltatást biztosító Grid rendszer. A ma már közel 40 ország 140 intézetének együttműködésében létrejött hálózatot 2004-ban kezdték fejleszteni. A

Core EGEE services:

- BME
- ELTE
- NIIF
- KFKI-RMKI

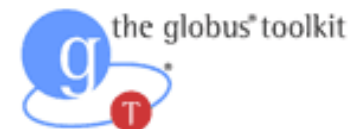
Portal interface:

- SZTAKI
- ~ 110 CPU
- ~ 6 TByte storage

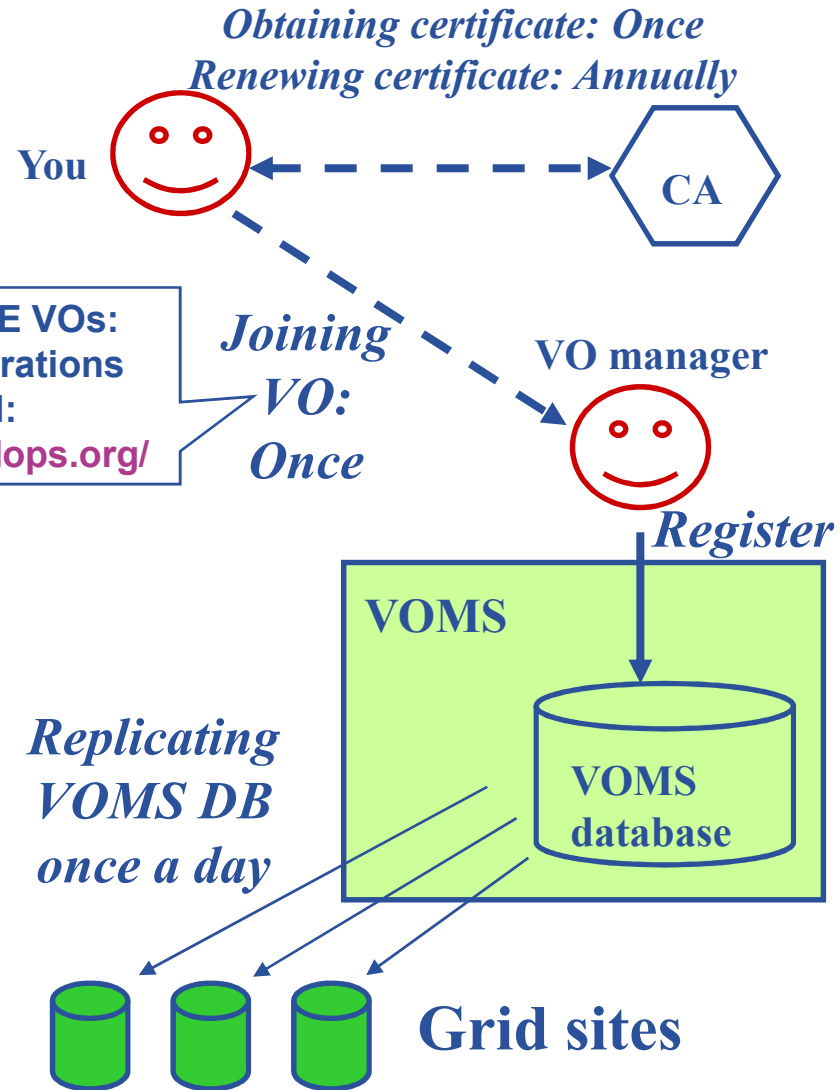
- gLite 3.0, gLite 3.1
- ⇒ Merger of LCG 2.7 and GLite 1.5

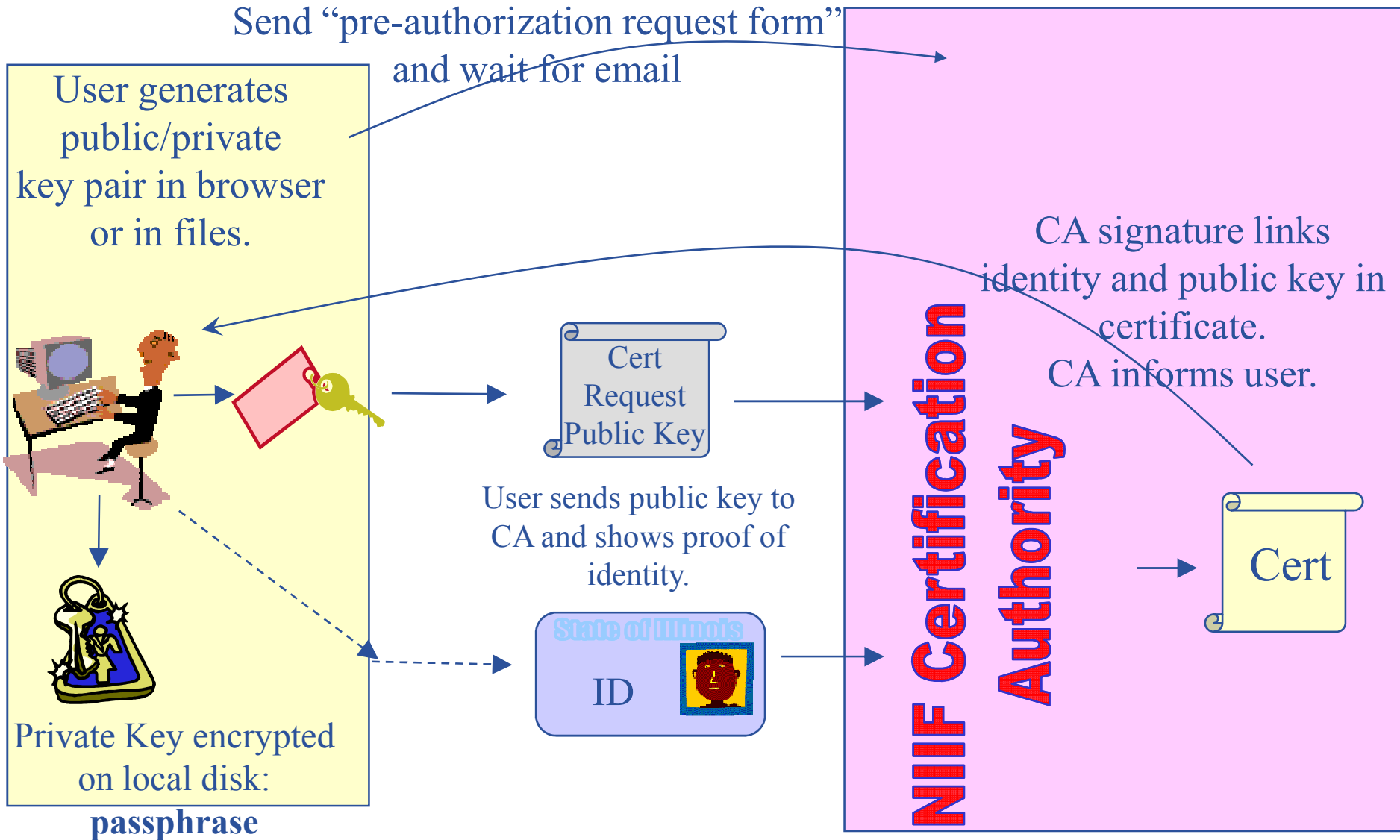


- Exploit **experience and existing components** from VDT (Condor, Globus), EDG/LCG, and others
- Develop a **lightweight stack of generic middleware** useful to EGEE applications (HEP and Biomedics are pilot applications).
 - Should eventually deploy dynamically (e.g. as a globus job)
 - Pluggable components – cater for different implementations
- Focus is on providing a stable and usable infrastructure



- Users (and machines) are identified by **grid certificates**.
- VO Membership Service (VOMS) keeps list of people who are registered to a VO
- **Users' steplist**
 - User obtains certificate from Certification Authority
 - <http://www.igtf.net>
 - User registers at the VO
 - usually via a web form
 - VO manager authorizes the user
 - VOMS DB updated
 - The user's identity is replicated onto resources within 24 hours
 - Use the grid through command line or graphical interfaces





- **Keep your private key and certificate secure**
 - if possible *on a USB drive only*
 - *Typically kept*
 - *on the User Interface machine of your VO*
 - *in MyProxy server*
 - *in Web browser*
- ***Different formats exist – transformation may be required!!!***
- **Do not loan your certificate to anyone**
- **Report to your CA if your certificate has been compromised**

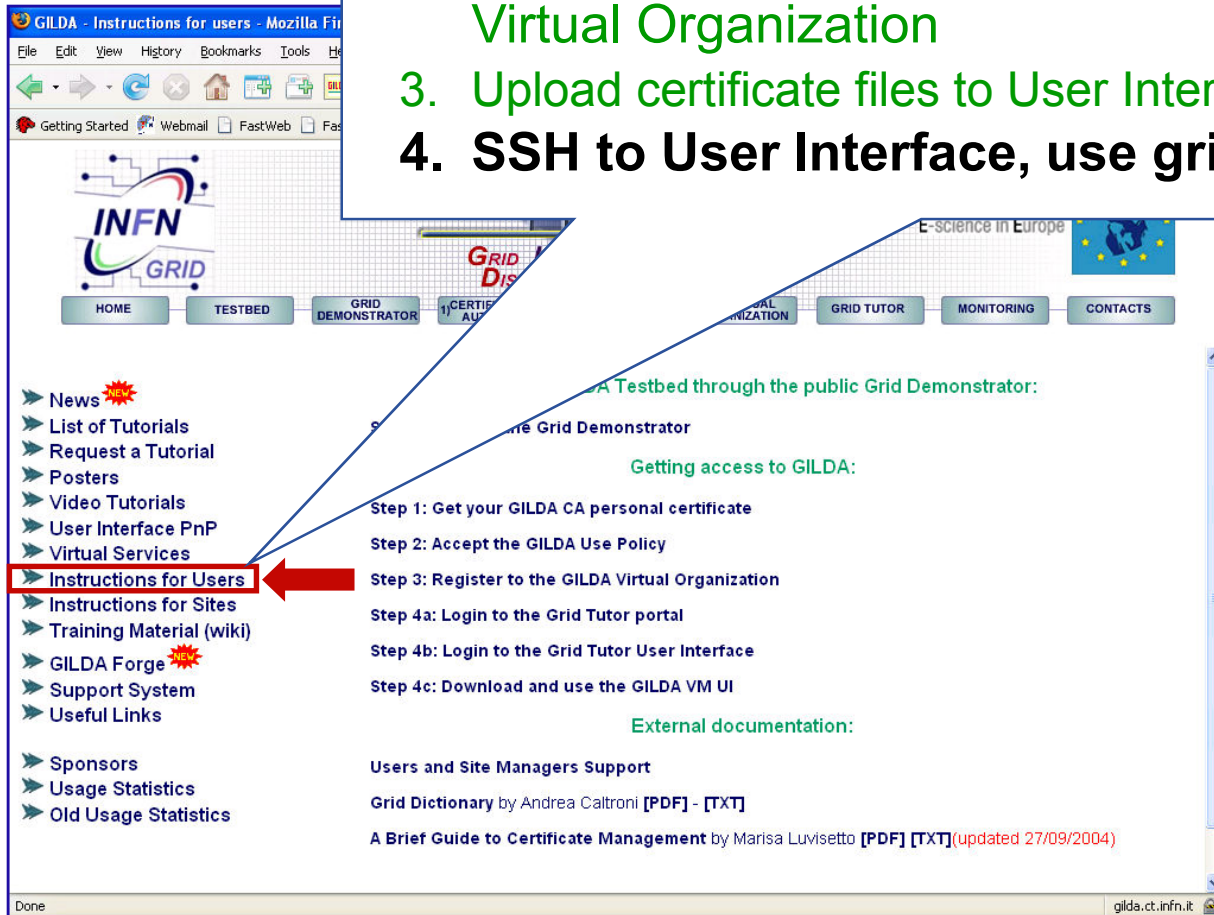
Typical situation on User Interface machine:

```
[sipos@glite-tutor sipos]$ ls -l .globus/
total 8
-rw-r--r--    1 sipos    users    1761 Oct 25  2006 usercert.pem
-r-----    1 sipos    users    951  Oct 24  2006 userkey.pem
```

Someone's identity in EGEE = Subject of certificate:

```
[sipos@glite-tutor sipos]$ grid-cert-info -subject
/C=HU/O=NIIF CA/OU=GRID/OU=NIIF/CN=Gergely Sipos/Email=sipos@sztaki.hu
```

1. Obtain a grid certificate from GILDA CA
2. Accept the GILDA Use Policy, register to GILDA Virtual Organization
3. Upload certificate files to User Interface
4. **SSH to User Interface, use grid services**



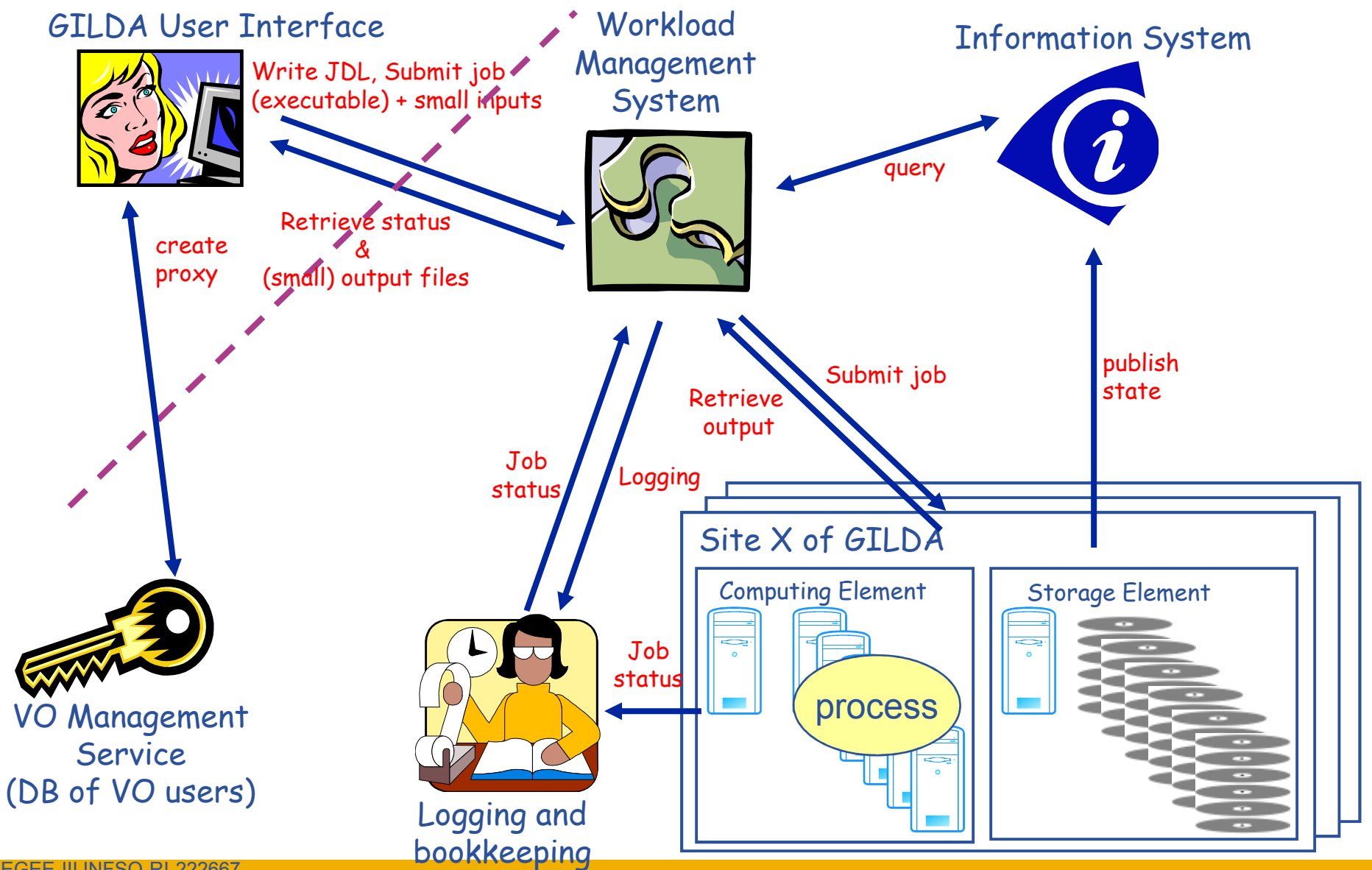
The screenshot shows the GILDA website interface. The top navigation bar includes links for HOME, TESTBED, GRID DEMONSTRATOR, CERTIFICATE AUTHORIZATION, GRID TUTOR, MONITORING, and CONTACTS. A sidebar menu on the left lists various resources, with 'Instructions for Users' highlighted in a red box and pointed to by a red arrow. The main content area contains the following text:

Getting access to GILDA:

- Step 1: Get your GILDA CA personal certificate
- Step 2: Accept the GILDA Use Policy
- Step 3: Register to the GILDA Virtual Organization
- Step 4a: Login to the Grid Tutor portal
- Step 4b: Login to the Grid Tutor User Interface
- Step 4c: Download and use the GILDA VM UI

External documentation:

- Users and Site Managers Support
- Grid Dictionary by Andrea Caltroni [PDF] - [TXT]
- A Brief Guide to Certificate Management by Marisa Luvisetto [PDF] [TXT](updated 27/09/2004)




```
[sipos@glite-tutor sipos]$ voms-proxy-init --voms gilda
Enter GRID pass phrase: *****
Your identity: /C=HU/O=NIIF CA/OU=GRID/OU=NIIF/CN=Gergely
Sipos/Email=sipos@sztaki.hu
Creating temporary proxy ..... Done
Contacting voms.ct.infn.it:15001 [/C=IT/O=INFN/OU=Host/L=Catania/CN=voms.ct.infn.it]
"gilda" Done
Creating proxy ..... Done
Your proxy is valid until Sat Jun 23 04:55:19 2007
```

% voms-proxy-init → login to the Grid

Enter PEM pass phrase: ***** → private key is protected by a password

– Options for voms-proxy-init:

- VO name
- -hours <lifetime of new credential>
- -bits <length of key>
- -help

% voms-proxy-destroy → logout from the grid

Delegated credentials will not be revoked

```
[sipos@glite-tutor sipos]$ nano/vi/etc hostname.jdl
```

```
Type = "Job";
```

```
JobType = "Normal";
```

```
Executable = "/bin/hostname";
```

```
StdOutput = "hostname.out";
```

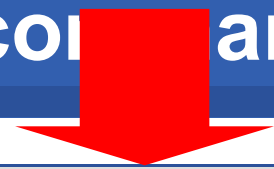
```
StdError = "hostname.err";
```

```
OutputSandbox = {"hostname.err", "hostname.out"};
```

```
Arguments = "-f";
```

```
ShallowRetryCount = 3;
```

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



WMS version	LCG-2 WMS	gLite WMS via NS gLite 3.0	gLite WMS via WMPProxy gLite 3.1+
Delegate proxy		D	glite-wms-job-delegate-proxy -d delegID
Submit	edg-job-submit [-o joblist]jdlfile	glite-job-submit [-o joblist] jdlfile	glite-wms-job-submit [-d delegID] [-a] [-o joblist] jdlfile
Status	edg-job-status [-v verbosity] [-i joblist] jobIDs	glite-job-status [-v verbosity] [-i joblist] jobIDs	glite-wms-job-status [-v verbosity] [-i joblist] jobIDs
Logging	edg-job-get-logging-info [-v verbosity] [-i joblist] jobIDs	glite-job-logging-info [-v verbosity] [-i joblist] jobIDs	glite-wms-job-logging-info [-v verbosity] [-i joblist] jobIDs
Output	edg-job-get-output [-dir outdir] [-i joblist] jobIDs	glite-job-output [-dir outdir] [-i joblist] jobIDs	glite-wms-job-output [-dir outdir] [-i joblist] jobIDs
Cancel	edg-job-cancel [-i joblist] jobID	glite-job-cancel [-i joblist] jobID	glite-wms-job-cancel [-i joblist] jobID
Compatible resources	edg-job-list-match jdlfile	glite-job-list-match jdlfile	glite-wms-job-list-match [-d delegID] [-a] jdlfile

D

E

P

R

E

C

A

T

E

D

GILDA User Interface



`glite-wms-job-delegate-proxy -d delegID`

`glite-wms-job-list-match hostname.jdl`

`glite-wms-job-submit
hostname.jdl → JobID`

`glite-wms-job-status JobID`

`glite-wms-job-output JobID`



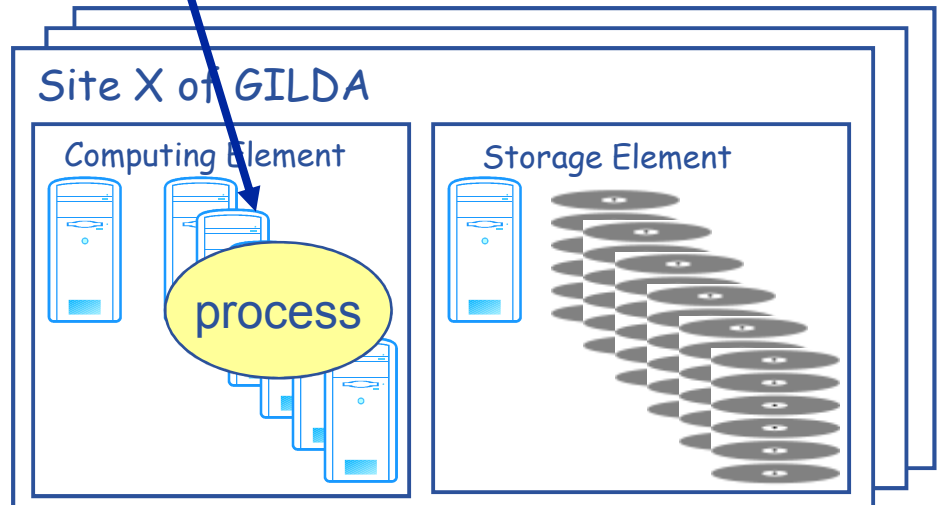
`delegID`

Manage job

`voms-proxy-init --voms gilda`



VO Management
Service
(DB of VO users)



Flag	Meaning
SUBMITTED	submission logged in the Logging & Bookkeeping service
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 Worker Node of the selected CE queue
DONE	job terminated without grid errors
CLEARED	job output retrieved
ABORT	job aborted by middleware, check <i>reason</i>

1. Practical

<https://grid.ct.infn.it/twiki/bin/view/GILDA/AuthenticationAuthorization>

- Investigate your certificate
- Create proxy
- Investigate your proxy

2. Practical

<https://grid.ct.infn.it/twiki/bin/view/GILDA/SimpleJobSubmission>

- Create a simple JDL file
 - *copy&paste JDL file from tutorial into a file. Executable is a server side prg.*
- Delegate proxy (*JobID saved in file*)
- List the CEs that can accept it
- Submit it
- Check its status until its done
- Retrieve output

Extra: <https://grid.ct.infn.it/twiki/bin/view/GILDA/CertificateManagement>

- How to import certificate in a web browser
- Now to convert pkcs12 to pem
- How to send signed email
- How to export a certificate from the web browser

More advanced jobs

```
[sipos@glite-tutor sipos]$ nano/vi/etc hostname.jdl
...
Executable = "/bin/hostname";
...
```

- **Installed on the CE**
 - Standard software in Linux
 - VO specific software: advertised in information system
 - *Use JDL expressions to navigate job to such a site*
- **Or Comes from client side**
 - Part of **InputSandbox**
- **Script**
 - No compilation is necessary
 - Can invoke binary that is statically installed on the CE
- **Or Binary**
 - Must be **compiled on the User Interface** → binary compatibility with CEs
 - Statically linked → to avoid errors caused by different library versions

```
$ cat testsandbox.jdl
Type = "Job";
JobType = "Normal";
Executable = "/bin/sh";
Arguments = "testsandbox.sh";
StdOutput = "testsandbox.out";
StdError = "testsandbox.err";
InputSandbox = "testsandbox.sh";
OutputSandbox = {"testsandbox.err", "testsandbox.out"};
ShallowRetryCount = 1;
```

```
$ cat testsandbox.sh
#!/bin/bash
ls -l
```

```
$ /bin/sh testsandbox.sh
```

```
$ cat yourexe.jdl
Type = "Job";
JobType = "Normal";
Executable = "/bin/sh";
Arguments = "script.sh INSERT_YOUR_NAME";
StdOutput = "script.out";
StdError = "script.err";
InputSandbox = {"script.sh", "myexecutable"};
OutputSandbox = {"script.out", "script.err", "exe.out"};
ShallowRetryCount = 1;
```

Compiled on UI

```
cat script.sh
#!/bin/sh
echo "setting right permissions"
chmod 755 myexecutable
echo "executing program now..."
./myexecutable $1 > exe.out
```

```
$ /bin/sh script.sh Gergely
```

```

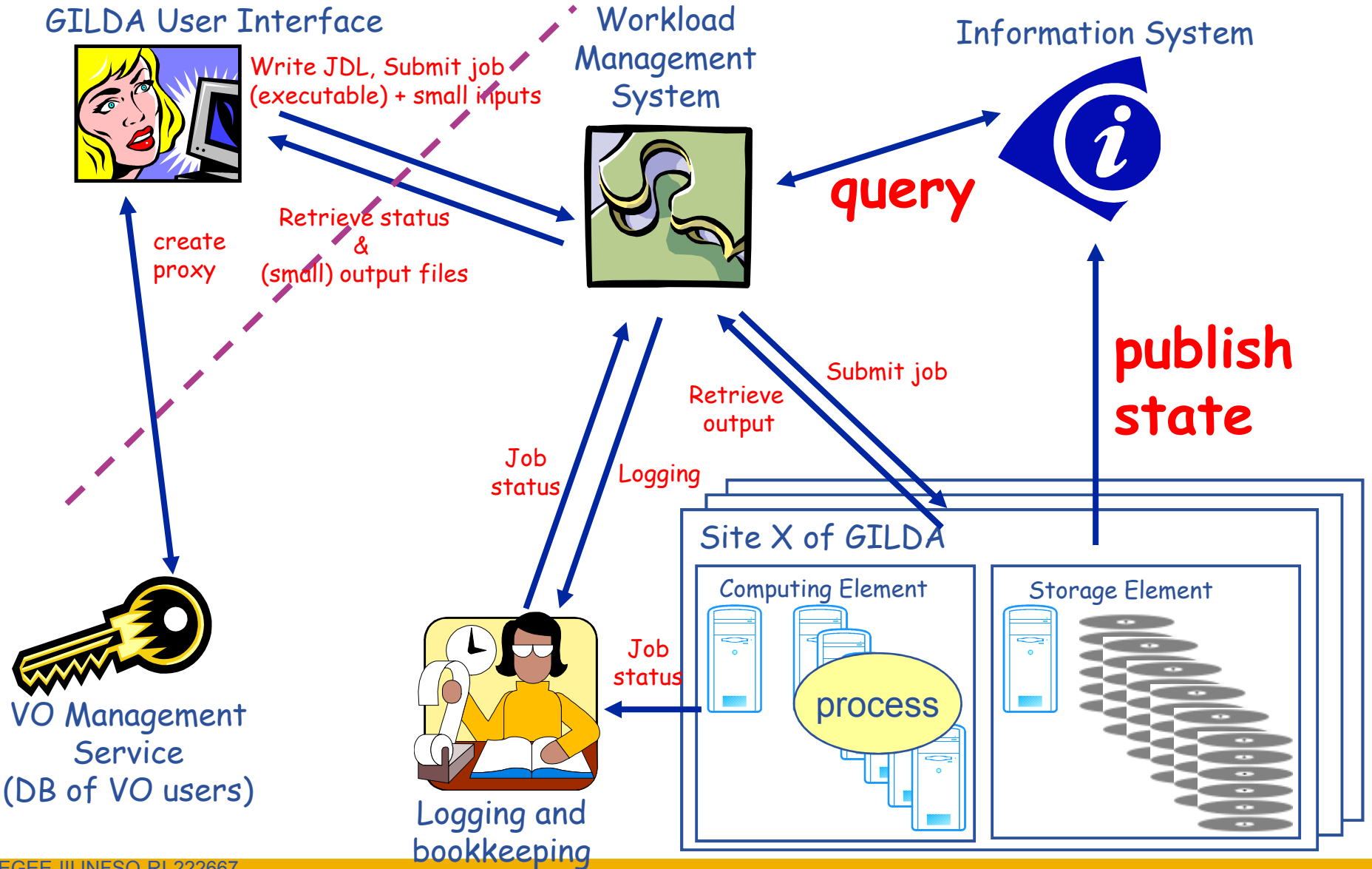
Executable = "gridTest";
StdError = "std...";
StdOutput = "std...";
InputSandbox = "...Test"};
OutputSandbox = { "...log", "...out.log"};

Requirements = other.Architecture=="INTEL" &&
    other.GlueCEInfoTotalCPUs > 480;
Rank = other.GlueCEStateTotalJobs;
    
```

WMS uses Information System to find CE

WMS brokering policy :

- Meet CE requirements
- Select CE with highest rank



- **GlueCEUniqueID** – Identifier of a CE
 - Eliminating an erroneous CE:


```
other.GlueCEUniqueID !=
"grid010.ct.infn.it:2119/jobmanager-lcgpbs-long"
```
 - Sending the job to a given CE:


```
other.GlueCEUniqueID ==
"grid010.ct.infn.it:2119/jobmanager-lcgpbs-long"
```
- **GlueCEInfoTotalCPUs** – max number of CPUs at a CE


```
Rank = other. GlueCEInfoTotalCPUs;
```
- **GlueCEStateWaitingJobs** – number of waiting jobs
- **GlueCEPolicyMaxCPUTime** – job will be killed after this number of minutes
- **GlueHostMainMemoryRAMSize** – memory size

<http://glite.web.cern.ch/glite/documentation/> → **JDL specification (submission via WMS WMPProxy)**

- *Rank =*
*(other.GlueCEStateWaitingJobs == 0 ? other.GlueCEStateFreeCPUs :
 -other.GlueCEStateWaitingJobs);*

if there are no waiting jobs,

- then the selected CE will be the one with the most free CPUs
- else the one with the least waiting jobs.

- *Requirements =*
*(Member(„IDL2.1”, other.GlueHostApplicationSoftwareRunTimeEnvironment))
 && (other.GlueCEPolicyMaxWallClockTime > 10000);*

CE where,

- IDL2.1 software is available
- At least 10000s can be spent on the site (waiting + running)

More complex JDLs

<https://grid.ct.infn.it/twiki/bin/view/GILDA/MoreOnJDL>

- Submit a script from client side
 - *Listing work directory of the job*
- Submit a binary from client side with wrapper script
- Requirements, Ranks
 - **Send the job to a particular CE**
 - **Send the job to any CE where “GEANT4-6” is available**
 - **Send a job anywhere but a particular CE (dealing with errors)**

Extra:

Query of the Information System to discover CE and SE characteristics and status:

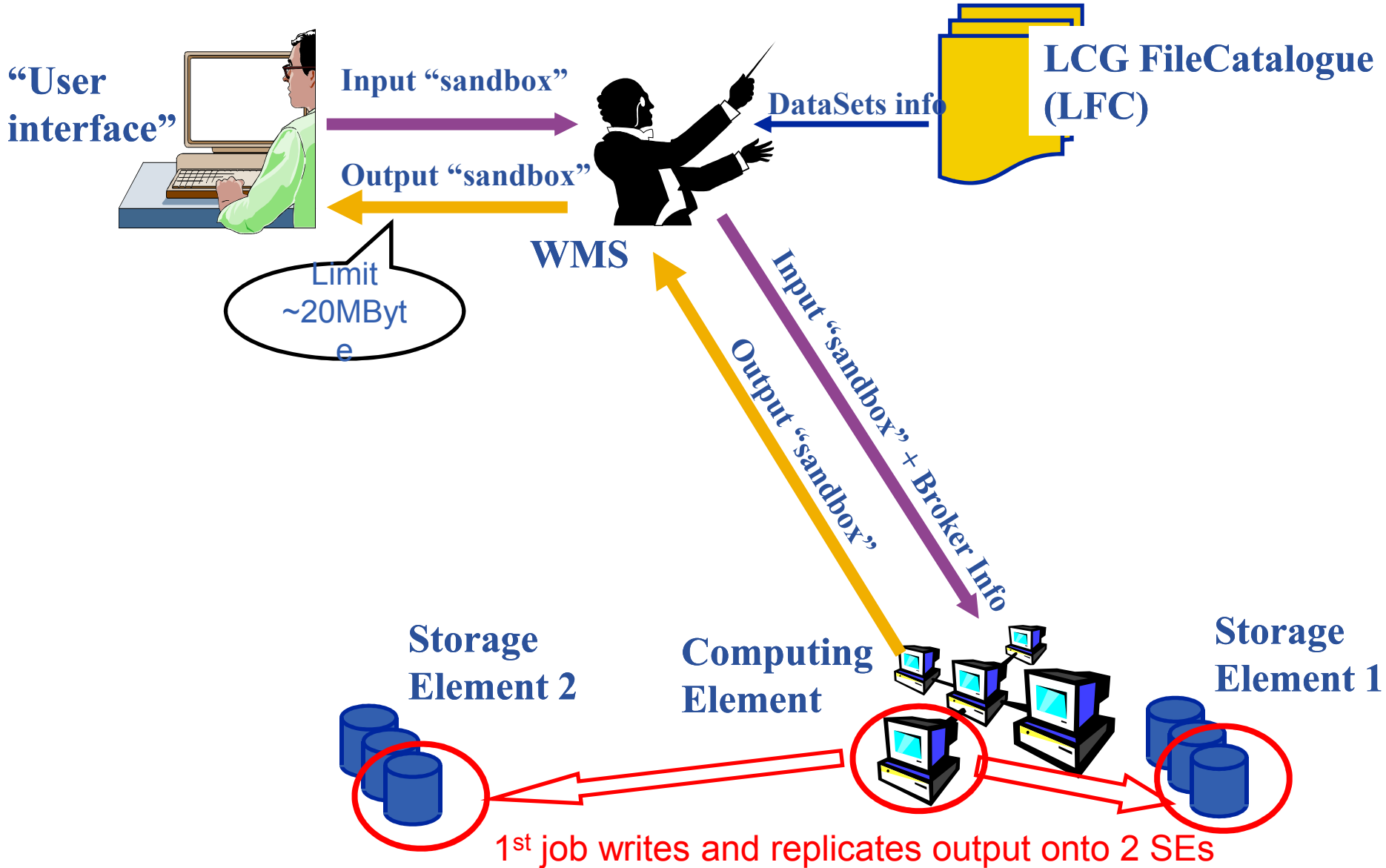
<https://grid.ct.infn.it/twiki/bin/view/GILDA/InformationSystems>

Working with large datasets

- **Files that are write-once, read-many**
 - If users edit files then
 - They manage the consequences!
 - Maybe just create a new filename!
 - No intention of providing a global file management system

- **Services for file management**
 - Storage
 - Catalogs
 - Transfer

Data management example



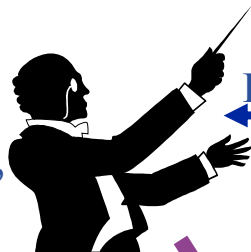
“User interface”



Limit
~20MByte

Input “sandbox”

Output “sandbox”



WMS

DataSets info



LCG FileCatalogue (LFC)

Keep computation close to storage data

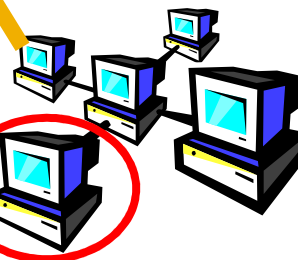
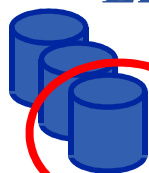
Input “sandbox” + Broker Info

Output “sandbox”

Storage Element 2

Computing Element

Storage Element 1



2nd job reads input from an SE

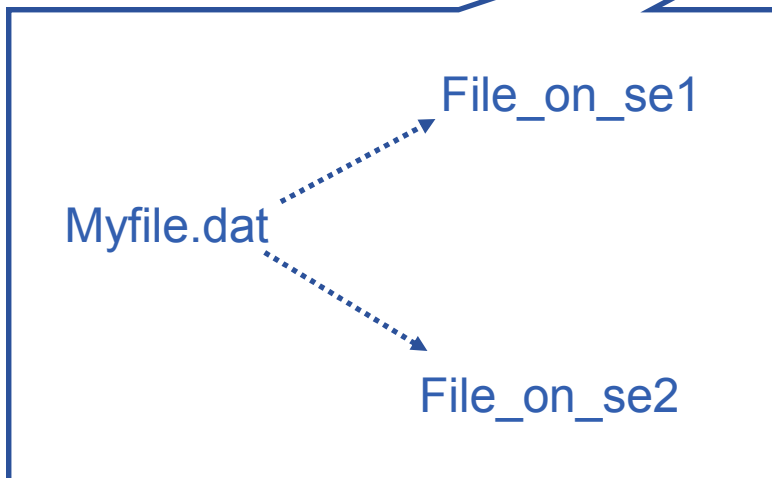
Resolving logical file name

“User interface”

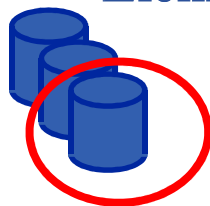


“Myfile.dat”

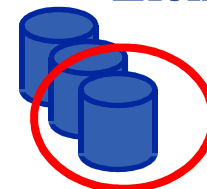
LCG FileCatalogue (LFC)



Storage Element 2



Storage Element 1



Content is available on 2 SEs

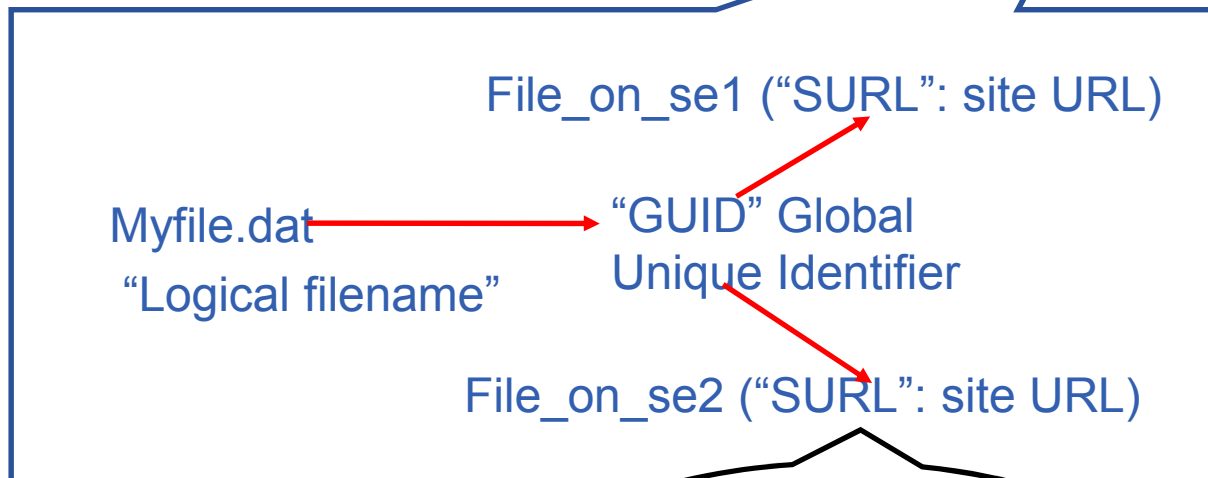
Resolving logical file name

“User interface”

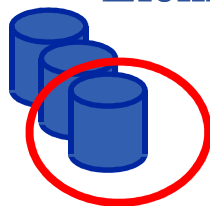


“Myfile.dat”

LCG FileCatalogue (LFC)

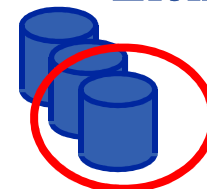


Storage Element 2



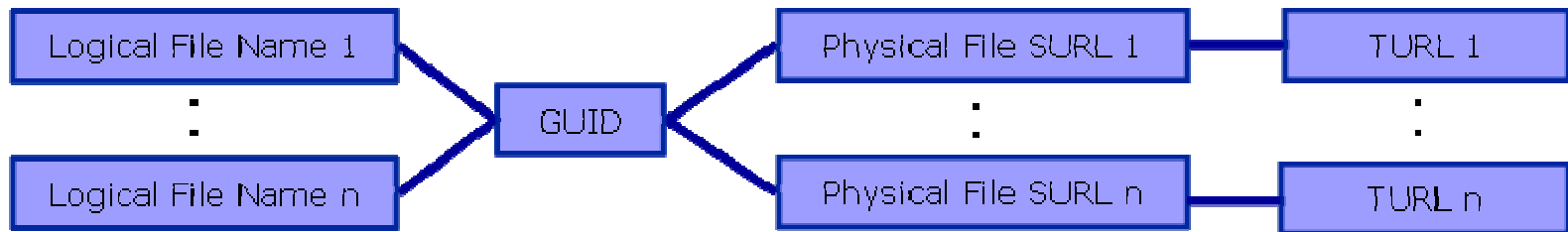
File content cannot change → No need to synchronize replicas

Storage Element 1



Content is available on 2 SEs

- **Logical File Name (LFN)**
 - An alias created by a user to refer to some item of data, e.g.
`lfn:/grid/gilda/budapest23/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`



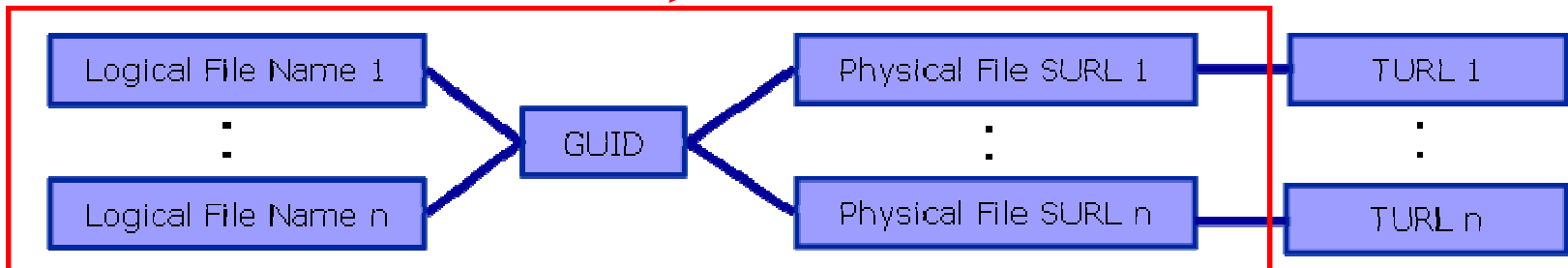
- Users primarily access and manage files through “logical filenames” - LFC

LFC has a directory tree structure
lfn:/grid/<VO_name>/ <you create it>

LFC Namespace

Defined by the user

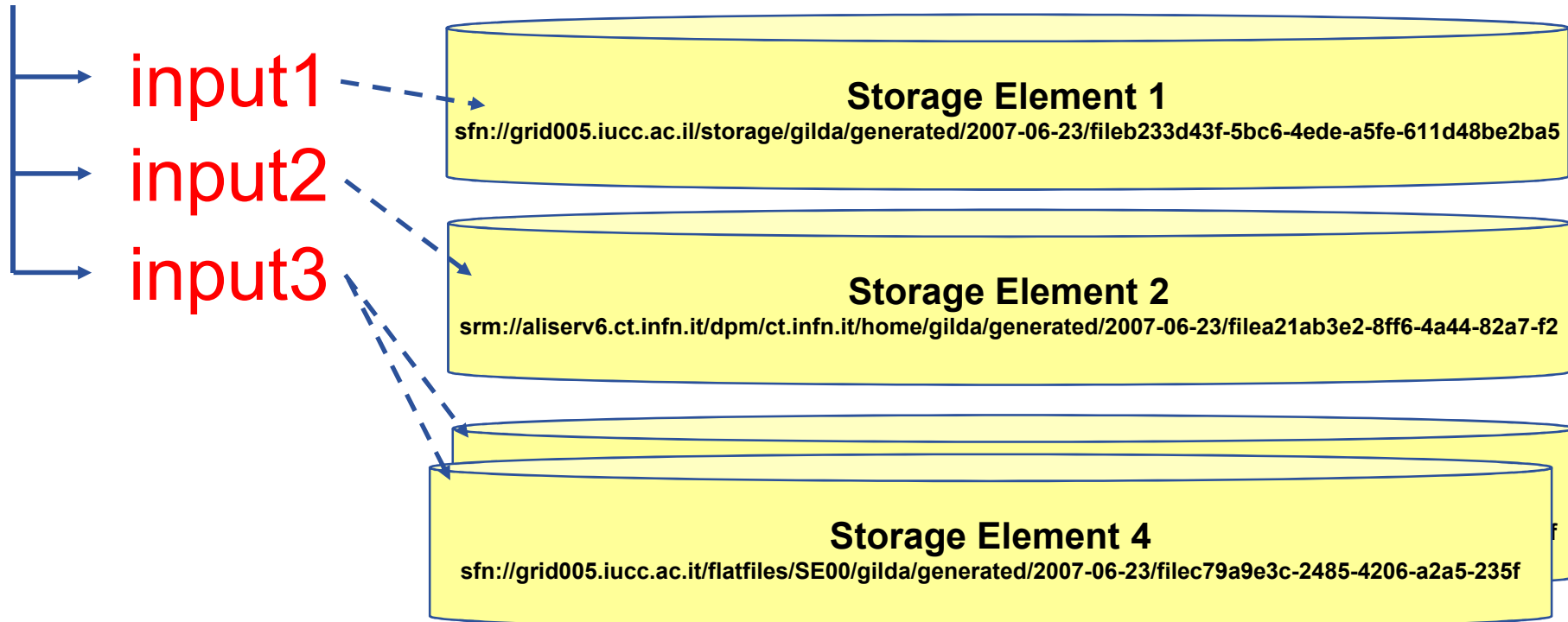
- Mapping by the “LFC” catalogue server



`lfn:/grid/gilda/sipos/run2/`



LCG FileCatalogue (LFC)



- **LFC directories = virtual directories**
 - Each entry in the directory may be stored on different SEs

- **lfc-***

LFC = LCG File Catalogue

- LCG = LHC Compute Grid
- LHC = Large Hadron Collider

- Use LFC commands to interact with the catalogue only

- To create catalogue directory
- List files

- Used by you, your application and by lcg-utils (see below)

- **lcg-***

- Couples catalogue operations with file management

- Keeps SEs and catalogue in step!

- Copy files to/from/between SEs

- Replicated

LFC has a directory tree structure

/grid/<VO_name>/ <you create it>



LFC Namespace

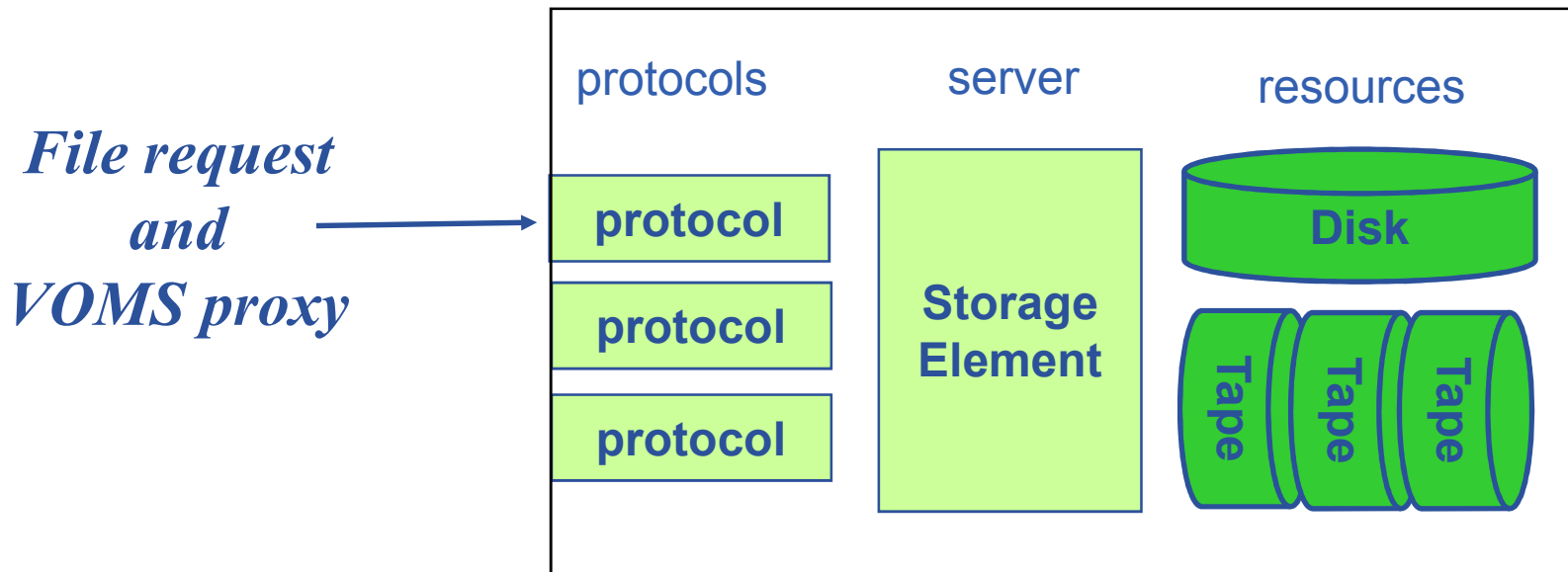


Defined by the user

- All members of a given VO have read-write permissions in their directory
- Commands look like UNIX with “lfc-” in front (often)

- **Provides**

- Storage for files : massive storage system - disk or tape based
- Transfer protocol (gsiFTP) ~ GSI based FTP server
 - Striped file transfer – cluster as back-end

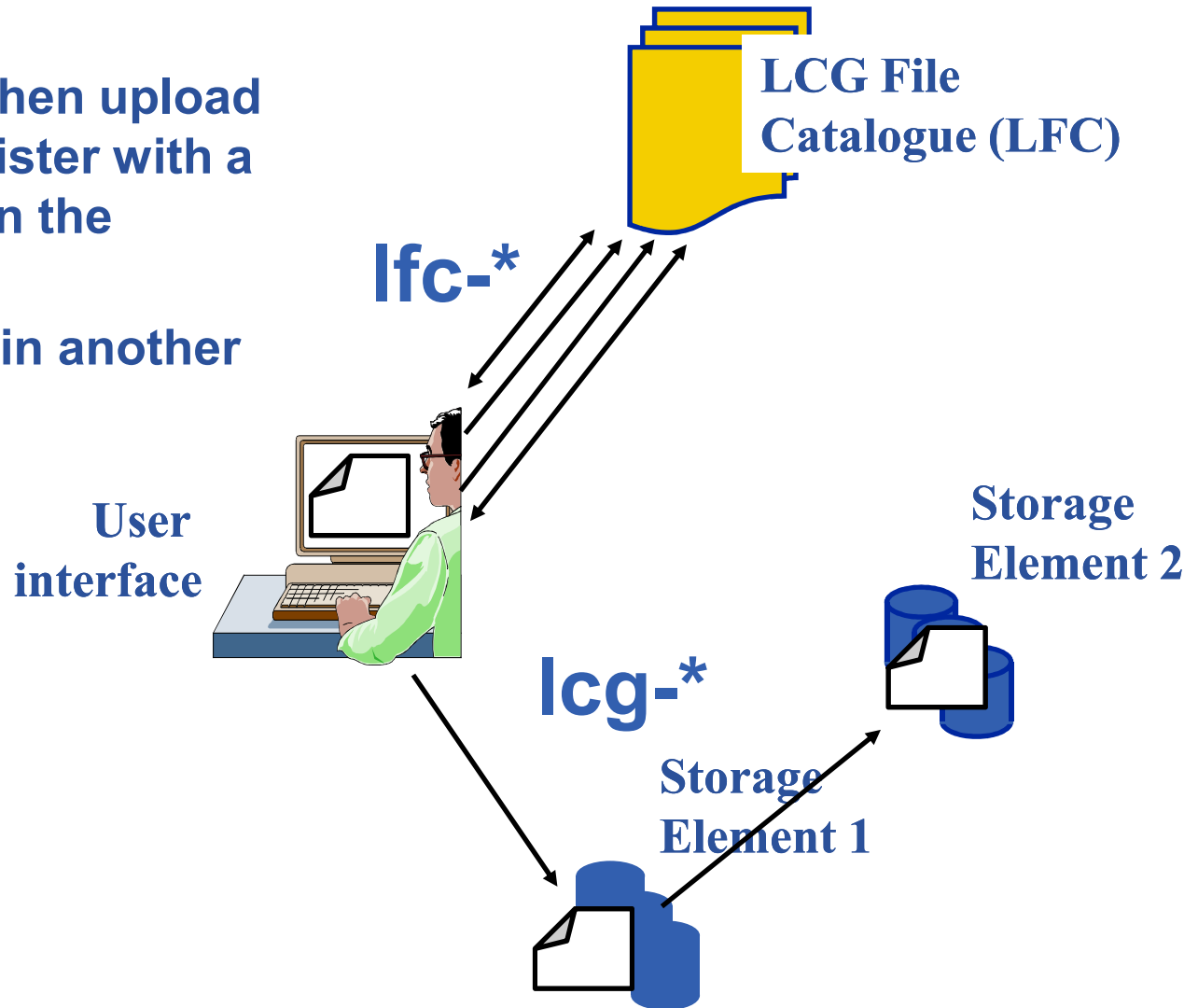


*Authentication,
authorization*

Type	Resources	File transfer	File I/O	SRM
Classic SE	Disk server	GSIFTP	insecure RFIO	No
DPM	Disk pool	GSIFTP	secure RFIO	Yes
dCache	Disk pool/MSS	GSIFTP	gsidcap	Yes
CASTOR	MSS	GSIFTP	insecure RFIO	Yes

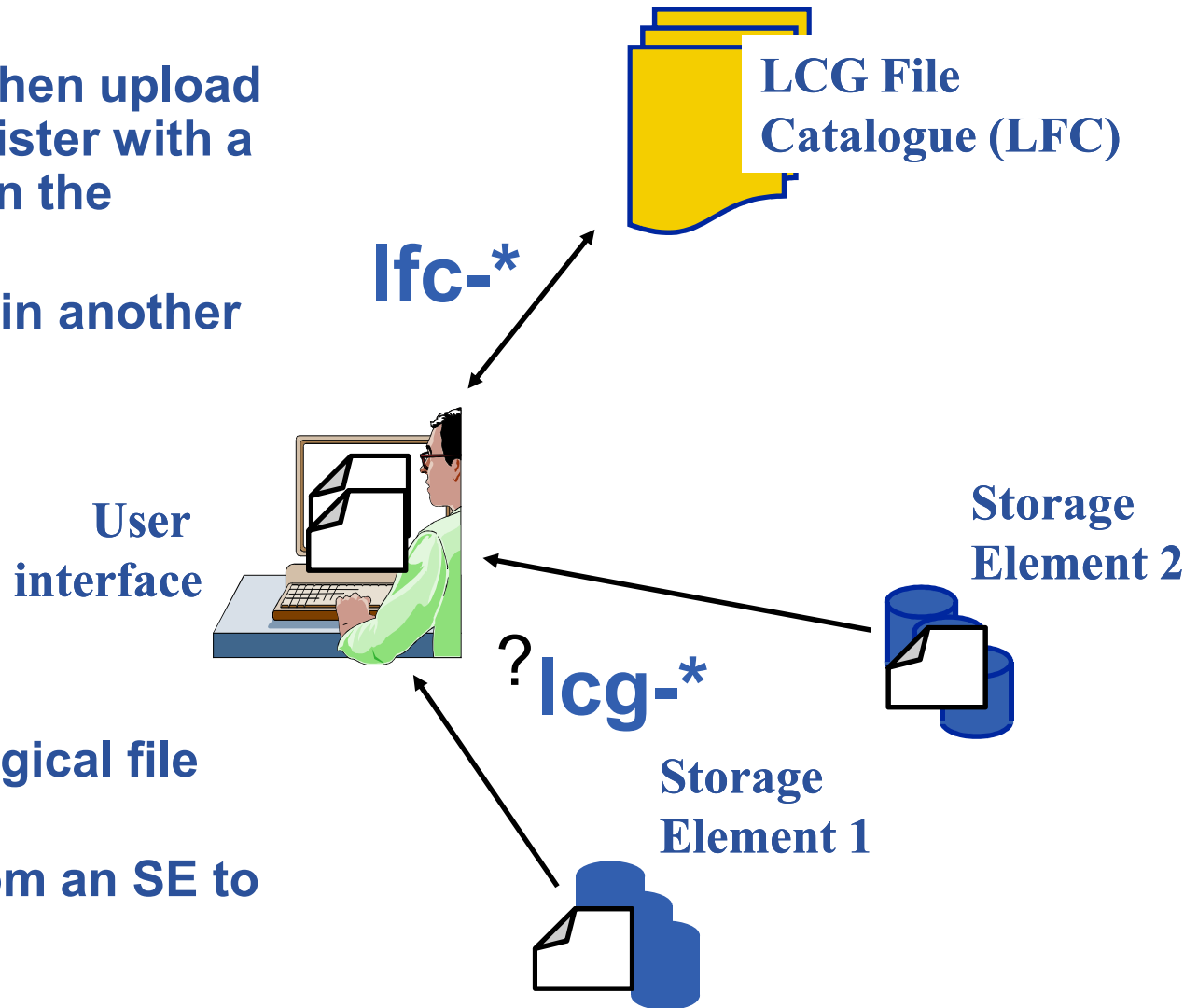
http://www.lpds.sztaki.hu/~sipos/egEE/tutorials/File_Management-lfc_and_lcg_commands.htm

- List directory
- Create a local file then upload it to an SE and register with a logical name (lfn) in the catalogue
- Create a duplicate in another SE
- List the replicas



http://www.lpds.sztaki.hu/~sipos/egEE/tutorials/File_Management-lfc_and_lcg_commands.htm

- List directory
 - Create a local file then upload it to an SE and register with a logical name (lfn) in the catalogue
 - Create a duplicate in another SE
 - List the replicas
-
- Create a second logical file name for a file
 - Download a file from an SE to the UI



Find exercises here:

<https://grid.ct.infn.it/twiki/bin/view/GILDA/JobDataWMS>

Exercise one:

1. Upload a file to SE, register it in LFC
2. Send wrapper script to CE as job
3. Send real executable in sand-box
4. Script starts executable on CE
5. Executable downloads file from CE
6. Executable processes file, produces result on STDOUT

Exercise two:

1. Send wrapper script to CE as job
2. Send real executable in sand-box
3. Script starts executable on CE
4. Executable produces result in a local file
5. Wrapper uploads file to CE, registers in LFC
6. Result file is available for download with lcg-cp



Enabling Grids for E-science

Services and resources for EGEE Users

www.eu-egee.org



- Expand portfolio of training materials, train users and admins

- Training material repositories:

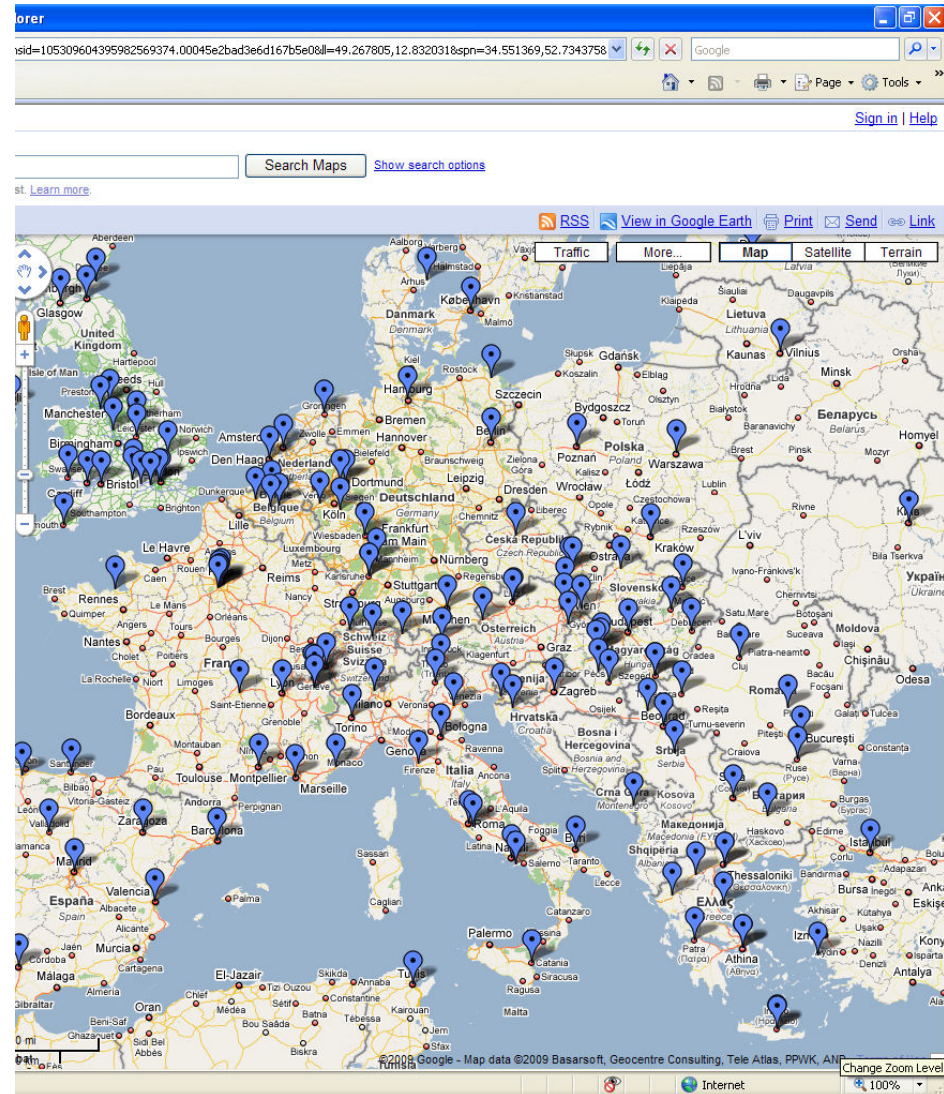
- <http://egee.lib.ed.ac.uk/>
 - Videos, MP3 talks on grid computing
- <https://grid.ct.infn.it/twiki/bin/view/GILDA/UserTutorials>
 - Tutorials for GILDA users

- Forthcoming and past training events

- <http://www.egee.nesc.ac.uk/>

- EGEE-EDGeS Summer School:

- www.egee.hu/grid09
- Budapest, June 29 – July 4



- **Support the large and diverse EGEE user community:**
 - **Promote dialog:** Users' Forums & EGEE Conferences
 - **Technical Aid:** Porting support, procedural issues
 - **Liaison:** Software and operational requirements

- **Main support services:**
 - **Application porting support:** www.lpds.sztaki.hu/gasuc
 - XIII, Victor Hugo utca 18-22.
 - Peter Kacsuk, Gergely Sipos {kacsuk, sipos}@sztaki.hu
 - **Direct user support:** www.ggus.org
 - **VO support:** <http://cic.gridops.org/>

- **Main application clusters:**
 - HEP, Life sciences, Astronomy & astrophysics, Earth science, Computational chemistry, Fusion, Grid observatory

- **GILDA support system:** <http://gilda-support.ct.infn.it/>
 - If you have problem with any of the GILDA services
 - Registration required
- **GGUS:** <http://www.ggus.org/>
 - If you have problems with any of the EGEE Production VOs (e.g. Hungrid sites, WMS, UI, ...)
 - Your grid certificate must be in your browser (See [this tutorial](#) on how to do it)
- **SZTAKI GASuC:** <http://www.lpds.sztaki.hu/gasuc/>
 - If you don't know how to port your application to EGEE

- **gLite manuals, documentation**
 - <http://glite.web.cern.ch/glite/documentation/>
(gLite user guide)
- **EGEE**
 - <http://www.eu-egee.org/>
- **gLite middleware**
 - <http://www.glite.org>

- **Friday, Saturday:**
 - Introduction to HEP applications and simulations
 - Porting simulations to GILDA
 - Submission of applications as jobs
 - Using pre-installed binaries?
- **Executing your own simulations on GILDA**
 - Until the 1st of May
 - Accounts and certificates expire!
- **Executing your own simulations on Hungrid**
 - Obtain personal grid certificates from NIIF CA
 - Port simulations from GILDA to Hungrid

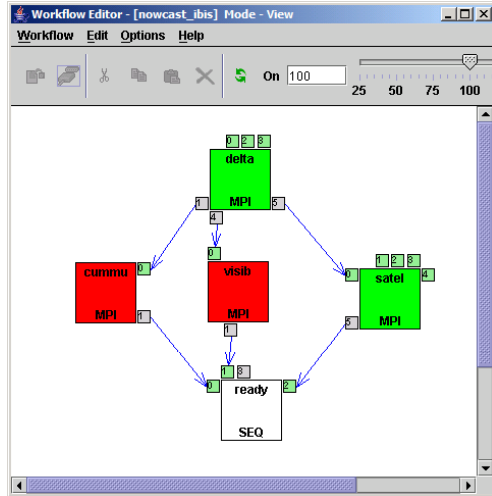
APPLICATION DESIGN



CERTIFICATE MANAGEMENT



FILE MANAGEMENT EXECUTION
PERF. ANALYSIS ON GRIDS



Certificate Manager

Used certificate

Downloaded from: cvs.lpdcs.sztaki.hu
 Issued by: O=HunGrid,O=SZTAKI HPCC,OU=lpds.sztaki.hu,CN=Workflow demo user,CN=proxy
 Timeleft: 9:56:40
 Description:

Issuer	Time Left	Status	[Actions]
O=HunGrid,O=SZTAKI HPCC,OU=lpds.sztaki.hu,CN=Nemeth Csaba,CN=proxy	27:16:7		Details Use this Delete
O=HunGrid,O=SZTAKI HPCC,OU=lpds.sztaki.hu,CN=Workflow demo user,CN=proxy	9:56:40	[used]	Details

Download (Download certificate from MyProxy server.) Upload (Upload authentication data to MyProxy server.)
 Message: [Press a button.]

Workflow Manager

Workflow	Job	Gridname	Hostname	Status	[Logs]	[Output]	[Visualization]	[All]	[Abort]
LM_9_DEMO_TOTAL	INIT	SEE-GRID	ce01.grid.acad.bg	finished	-	N/A	-	-	-
	LM_P	SEE-GRID	n40.hpcc.sztaki.hu	running	-	-	Visualize	-	-
	LM_P.2	SEE-GRID	n40.hpcc.sztaki.hu	finished	-	-	Visualize	-	-
	LM_S	SEE-GRID	grid-oe.il.edu.mk	finished	-	-	Visualize	-	-
	LM_S.2	SEE-GRID	grid1.irb.hr	finished	-	-	Visualize	-	-
	LM_S.3	SEE-GRID	grid1.metmode.ecce.univ-gr	finished	-	-	Visualize	-	-
	LM_S.4	SEE-GRID	grid1.irb.hr	finished	-	-	Visualize	-	-
	LM_S.5	SEE-GRID	testbed001.grid.ki.ro	finished	-	-	Visualize	-	-
	LM_S.6	HUNGRID	chemgrid3.chemres.hu	finished	-	-	Visualize	-	-
	TIFF	HUNGRID	grid109.kfki.hu	init	-	-	-	-	-

Message: Job list refreshed.

Grid Monitor

Select Grid: EGEE-Testzone
 Select VO: EGEE-Testzone

Site Name	Computing Element				Storage Element				
	Total	Free	Usage	Running	Waiting	Job	Total	Available	Usage
BEIJING-LCG2	6	6	0%	0	0	0%	469.278 GB	451.833 GB	4%
BG-INRNE	40	36	10%	2	0	0%	74.709 GB	74.645 GB	0%
BGO1-IPP	54	9	83%	5	28	85%	609.554 GB	513.6 GB	16%
BGO2-IPP	12	8	33%	1	0	0%	240.177 GB	167.654 GB	30%
BHAM-LCG2	100	100	0%	0	0	0%	897.462 GB	873.093 GB	3%
BUDAPEST	210	54	74%	132	0	0%	N/A	N/A	N/A
CAVENDISH-LCG2	19	0	0%	0	0	0%	59.458 GB	133.15 GB	16%
CEBSA-EGEE	30	24	20%	2	0	0%	22.416 GB	16.224 GB	28%
CGG-LCG2	201	0	100%	67	26	28%	427.446 GB	331.503 GB	22%
CINNAT-LCG2	42	42	0%	0	0	0%	138.204 GB	28.348 GB	79%
EGEE-Testzone	N/A	N/A	N/A	N/A	N/A	N/A	1.411.736 TB	8.046.74 TB	33%

Tracefile visualization

workflow: nowcast_ibis

Trace View Info

Width: [600]
 Height: [350]
 OK

The visualization shows a network diagram of nodes and their execution times on a timeline from 0s to 5m30s.

www.portal.p-grade.hu

Thank you