egee

Enabling Grids for E-sciencE

EGEE middleware

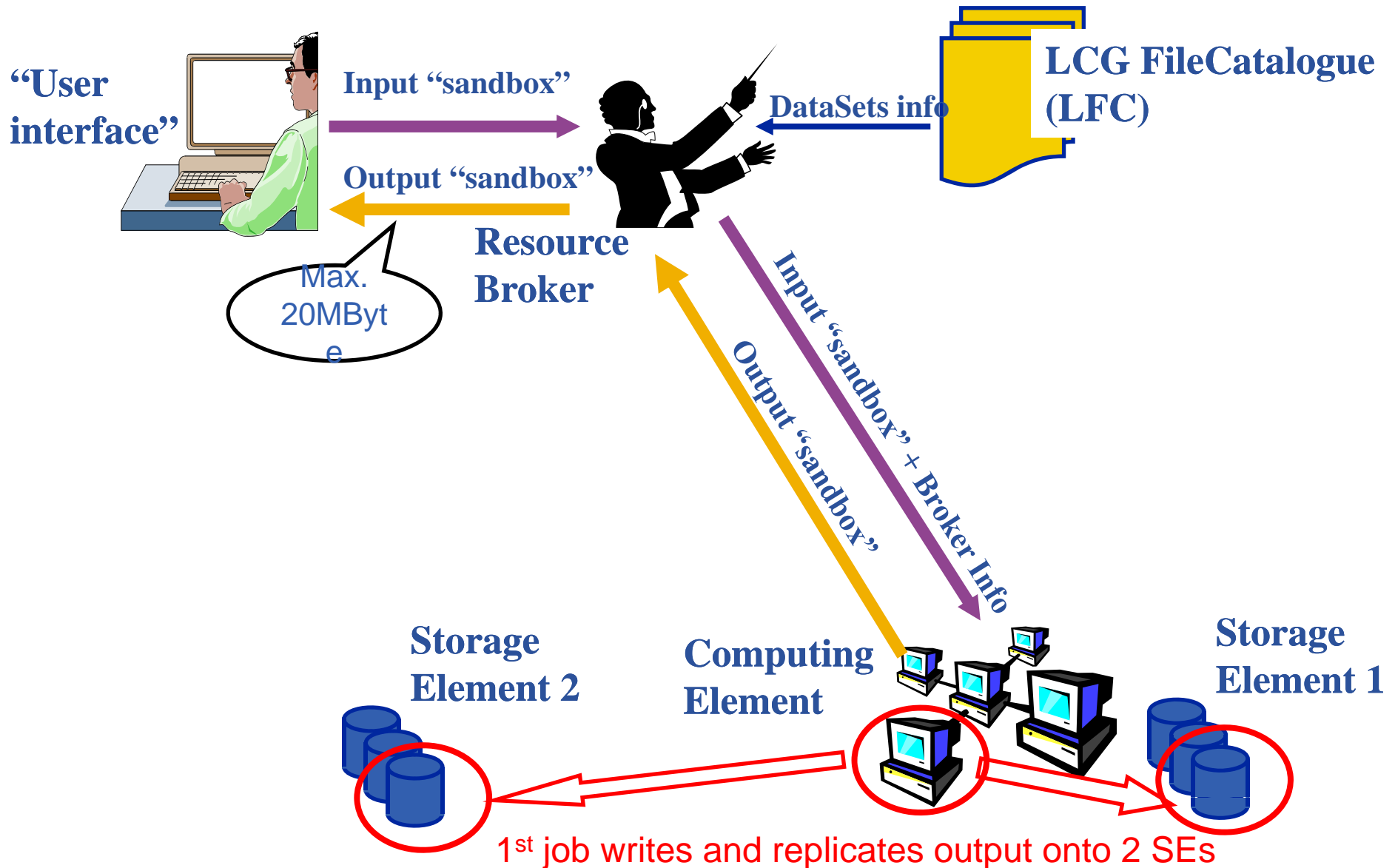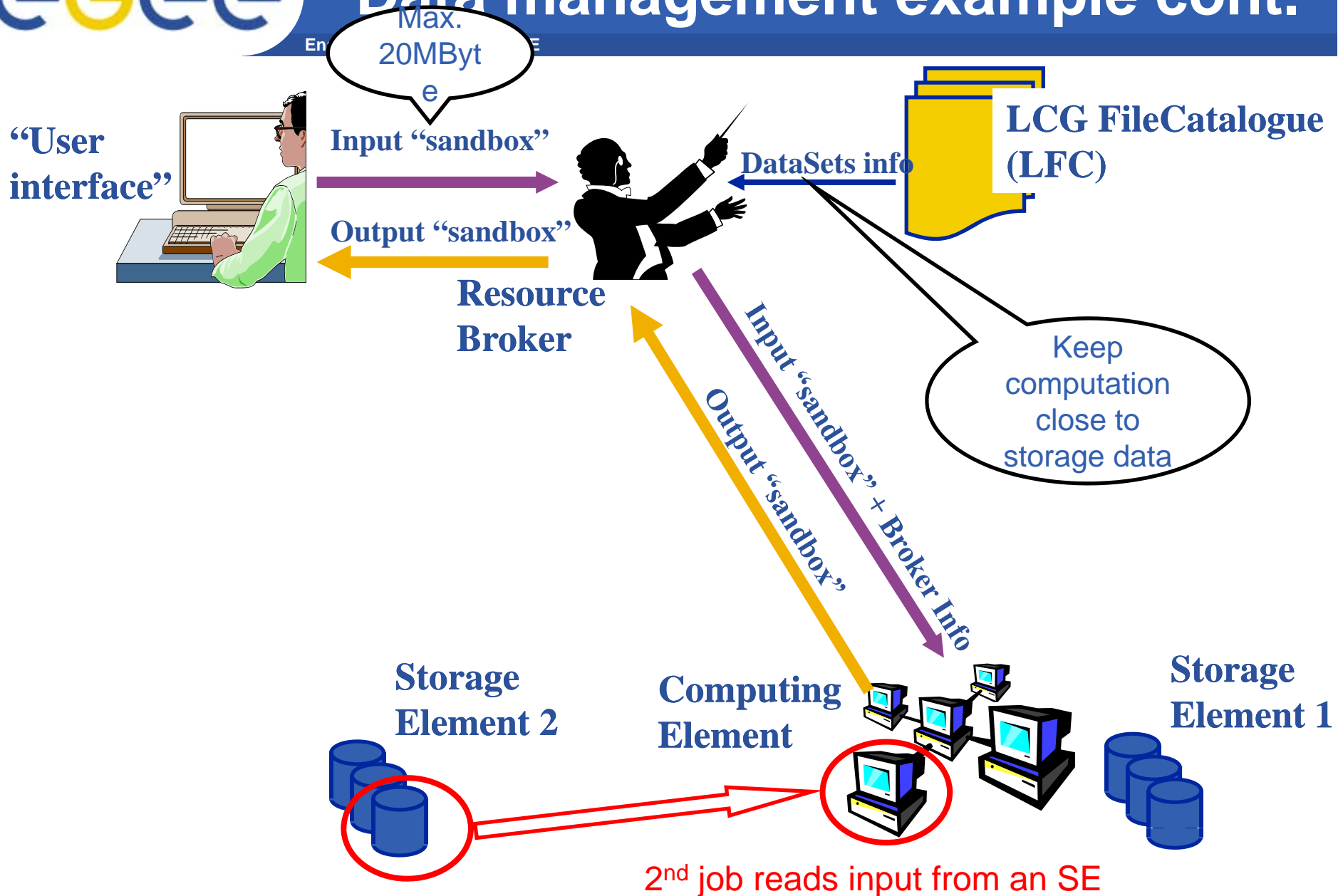## *Data Services in gLite*

Information Society

- **Simple data files on grid-specific storage**

- **Middleware supporting**
  - **Replica files**
    - to be close to where you want computation
    - For resilience
  - **Logical filenames**
  - **Catalogue**: maps logical name to physical storage device/file
  - **Virtual filesystems**, POSIX-like I/O
  - Services provided: storage, transfer, catalogue that maps logical filenames to replicas.

- **Solutions include**
  - **gLite data service**
  - **Globus: Data Replication Service**
  - **Storage Resource Broker**

- **Other data!   e.g. ….**
  - Structured data: RDBMS, XML databases,…
  - Files on project's filesystems
  - Data that may already have other user communities not using a Grid

- Require extendable middleware tools to support
  - Computation near to data
  - Controlled exposure of data *without replication*

- Basis for integration and federation
- **OGSA –DAI**
  - In Globus 4
  - Not (yet...) in gLite

- **<u>Files</u> 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

- **3 service types for data**
  - Storage
  - Catalogs
  - Transfer

"User interface"

Input "sandbox"
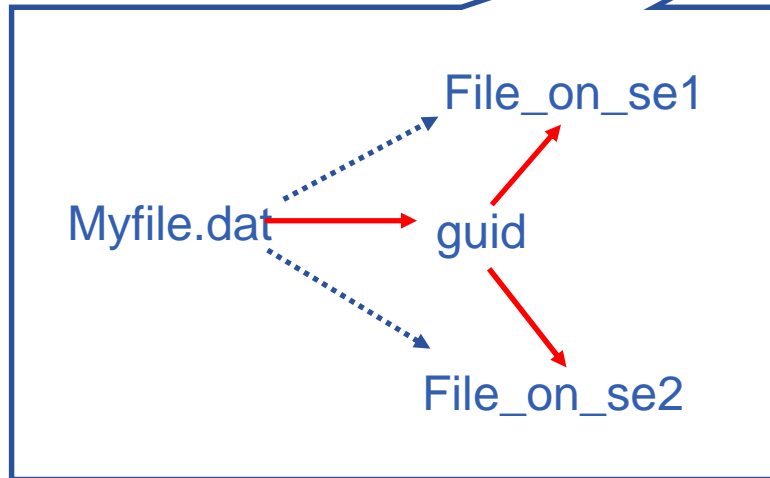
Output "sandbox"

Max. 20MByte

Resource Broker

DataSets info

LCG FileCatalogue (LFC)

Input "sandbox" + Broker Info

Output "sandbox"

Storage Element 2

Computing Element

Storage Element 1

1st job writes and replicates output onto 2 SEs

Data management example cont.

Max. 20MByte

"User interface"

Input "sandbox"

Output "sandbox"

Resource Broker

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

Enabling Grids for E-sciencE

"User interface"

"Myfile.dat"

LCG FileCatalogue (LFC)

File_on_se1

Myfile.dat → guid

File_on_se2

Storage Element 2

Storage Element 1

Content is available on 2 SEs

"User interface"

"Myfile.dat"

LCG FileCatalogue (LFC)

File_on_se1 ("SURL": site URL)

Myfile.dat
"Logical filename"

"GUID" Global Unique Identifier

File_on_se2 ("SURL": site URL)

Storage Element 2

File content cannot change → No need to synchronize replicas
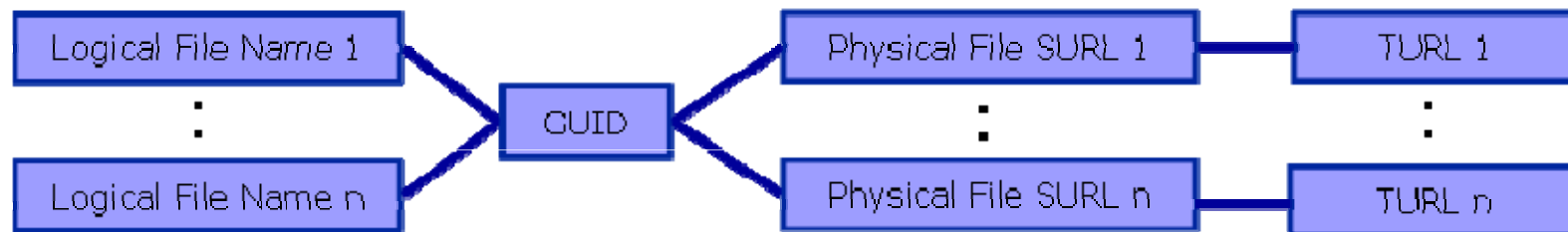
Storage Element 1

Content is available on 2 SEs

Enabling Grids for E-sciencE
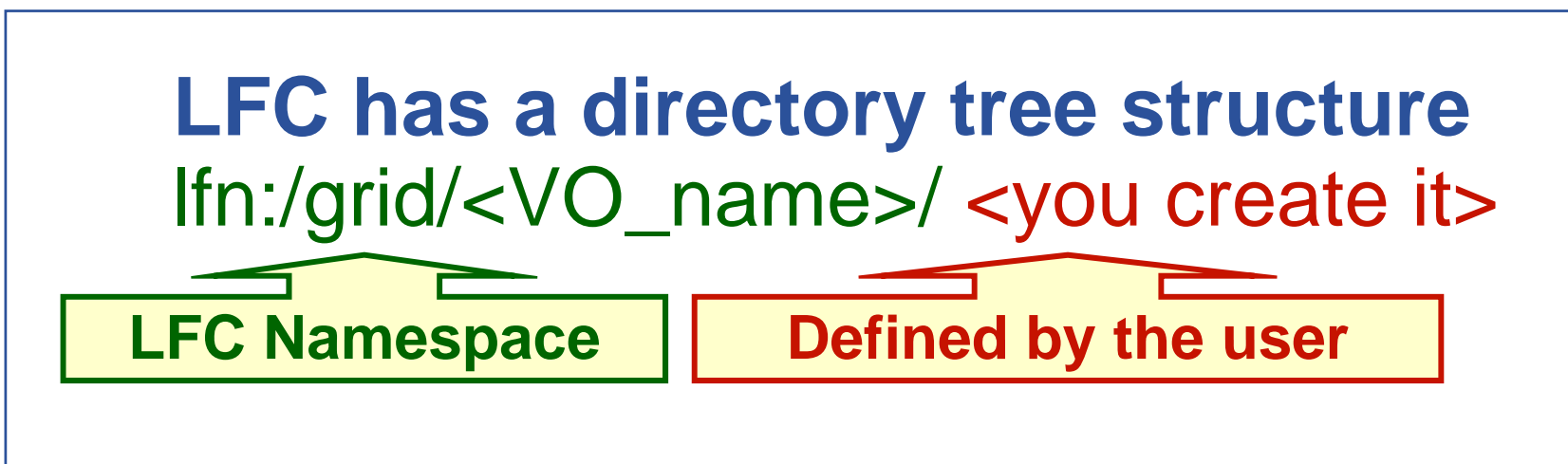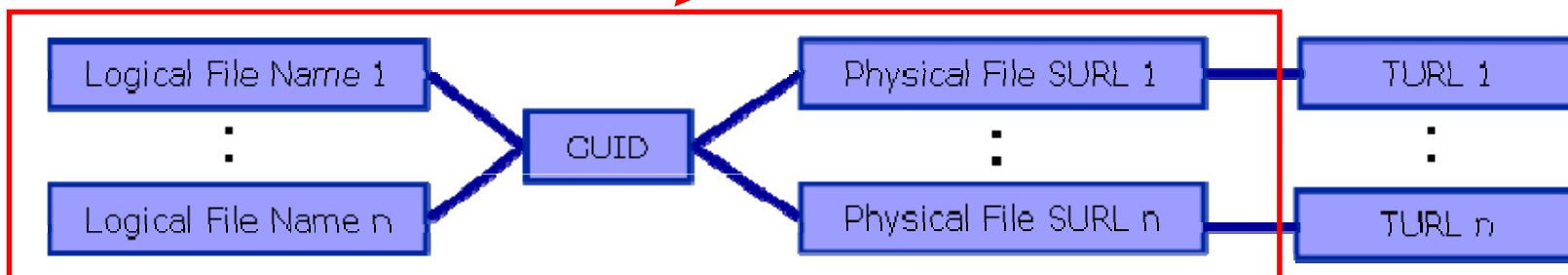
- **Logical File Name (LFN)**
  - An alias created by a user to refer to some item of data, e.g.
    `lfn:/grid/gilda/budapest23/run2/track1`

- **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`

Enabling Grids for E-sciencE

- **Users primarily access and manage files through "logical filenames"**

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

Enabling Grids for E-sciencE

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

LCG FileCatalogue (LFC)

lfn:/grid/gilda/budapest23/run2/

input1

Storage Element 1

sfn://grid005.iucc.ac.il/storage/gilda/generated/2007-06-23/fileb233d43f-5bc6-4ede-a5fe-611d48be2ba5

input2

Storage Element 2

srm://aliserv6.ct.infn.it/dpm/ct.infn.it/home/gilda/generated/2007-06-23/filea21ab3e2-8ff6-4a44-82a7-f2

input3

Storage Element 4

sfn://grid005.iucc.ac.it/flatfiles/SE00/gilda/generated/2007-06-23/filec79a9e3c-2485-4206-a2a5-235f

Enabling Grids for E-sciencE
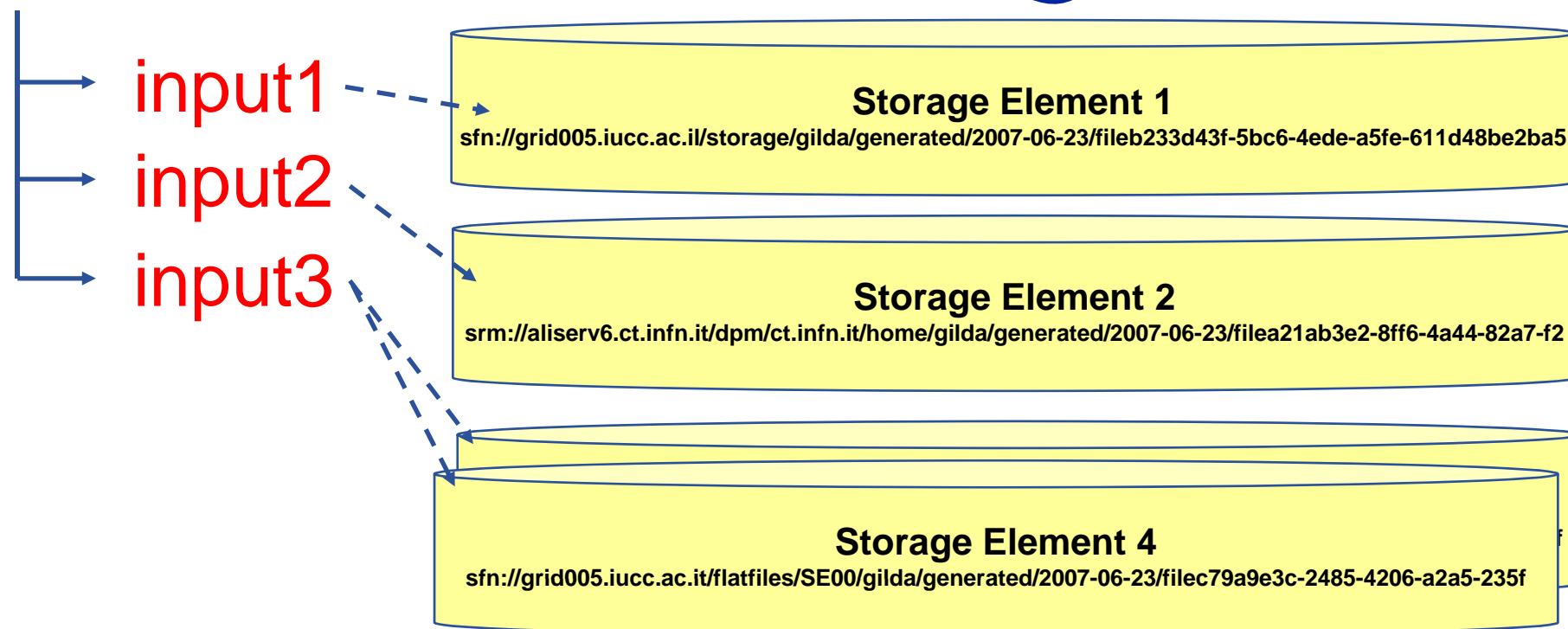
- **LFC = LCG File Catalogue**
    - LCG = LHC Compute Grid
    - LHC = Large Hadron Collider

    – 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-utils**
    – Couples catalogue operations with file management
    - Keeps SEs and catalogue in step!
    – Copy files to/from/between SEs
    – Replicated

Enabling Grids for E-sciencE

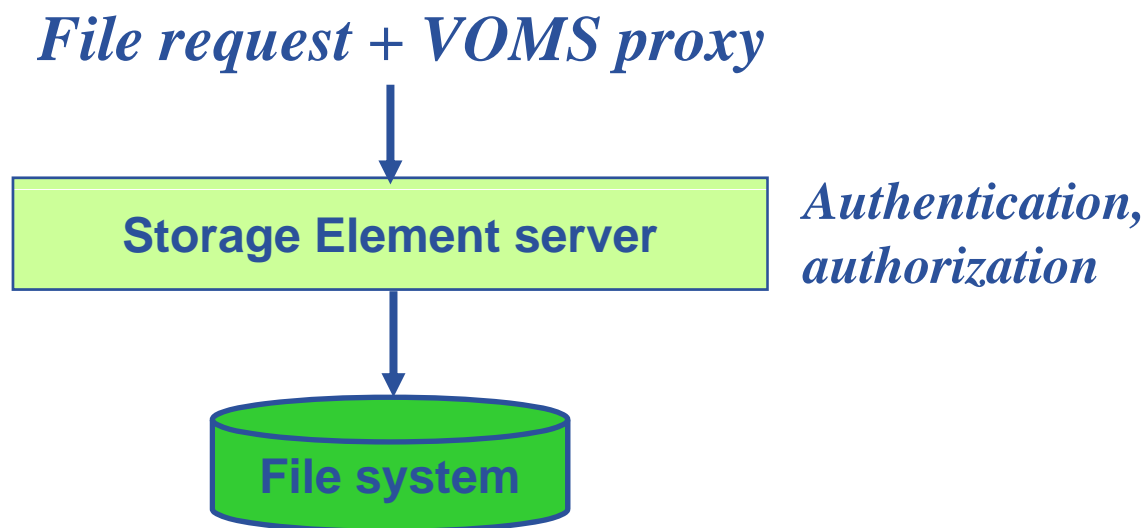## LFC has a directory tree structure

/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

File request + VOMS proxy

Storage Element server

Authentication, authorization

File system

Enabling Grids for E-sciencE

- **GFAL (Grid File Access Library) is a POSIX interface for operation on file on Storage Element**

- **Enable remote handling of files**

- **Libraries are in C and can be included in C/C++ sources (GFAL Java API tomorrow!)**

- **The most common of I/O operations are available, just prefix gfal_ to the function name (open(), read()…)**

- `man gfal` **for further details**

- **The destination SE must provide secure rfio (*classic SEs* don't)**

- **GFAL API Description**
  - http://grid-deployment.web.cern.ch/grid-deployment/documentation/LFC_DPM/gfal/html

Examples in gLite3 User Guide (Appendix F)

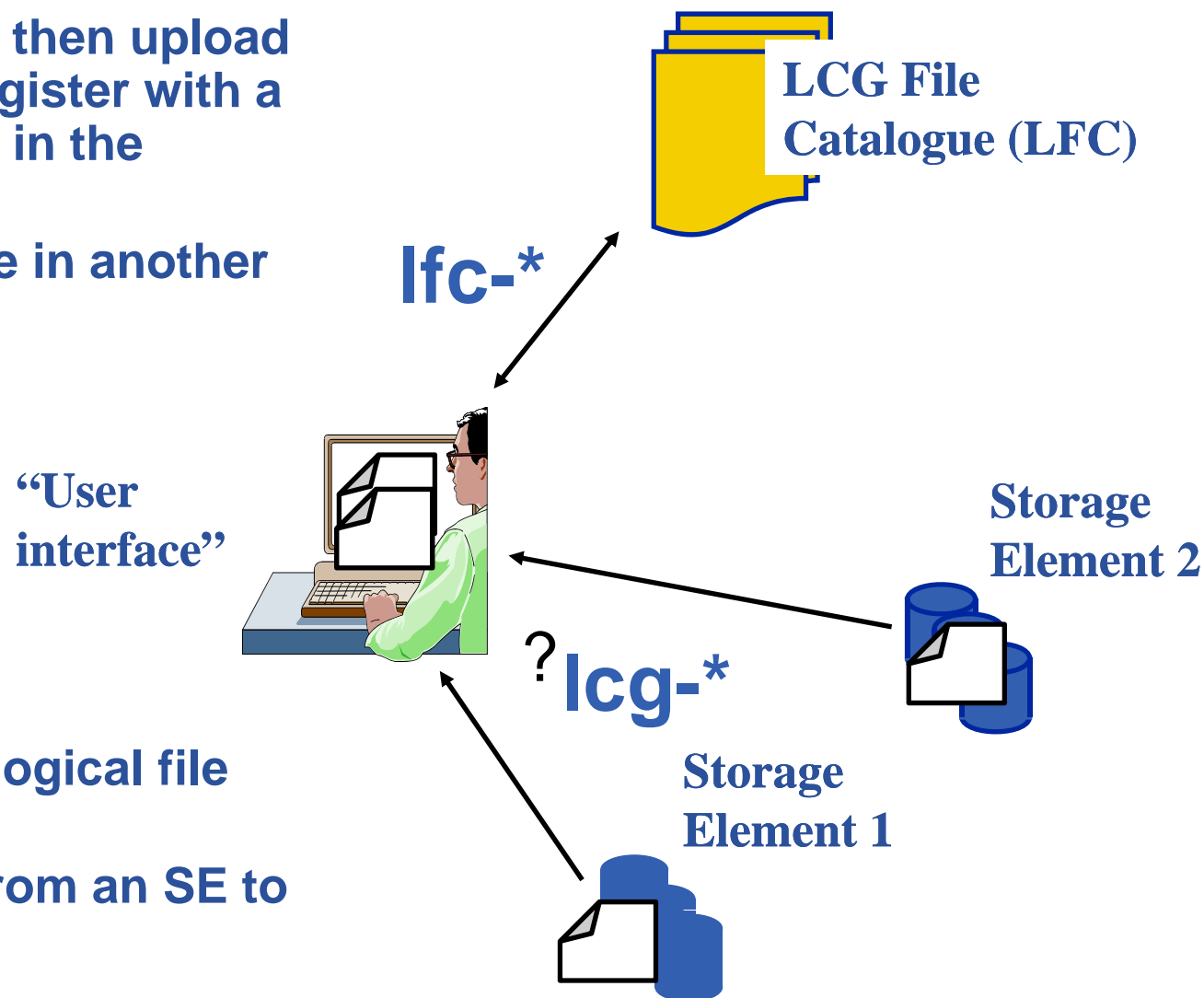– https://edms.cern.ch/file/722398//gLite-3-UserGuide.pdf

```
int fd;
struct stat remote_file_stat;


fd = gfal_open(file_ref, O_RDONLY, 0644);
cod_ex = gfal_stat(file_ref, &file_stat)
...
cod_ex = gfal_read(fd, buffer, file_stat.st_size));
...
cod_ex = gfal_close(fd);
```

- **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**

lfc-*

LCG File Catalogue (LFC)

"User interface"

lcg-*

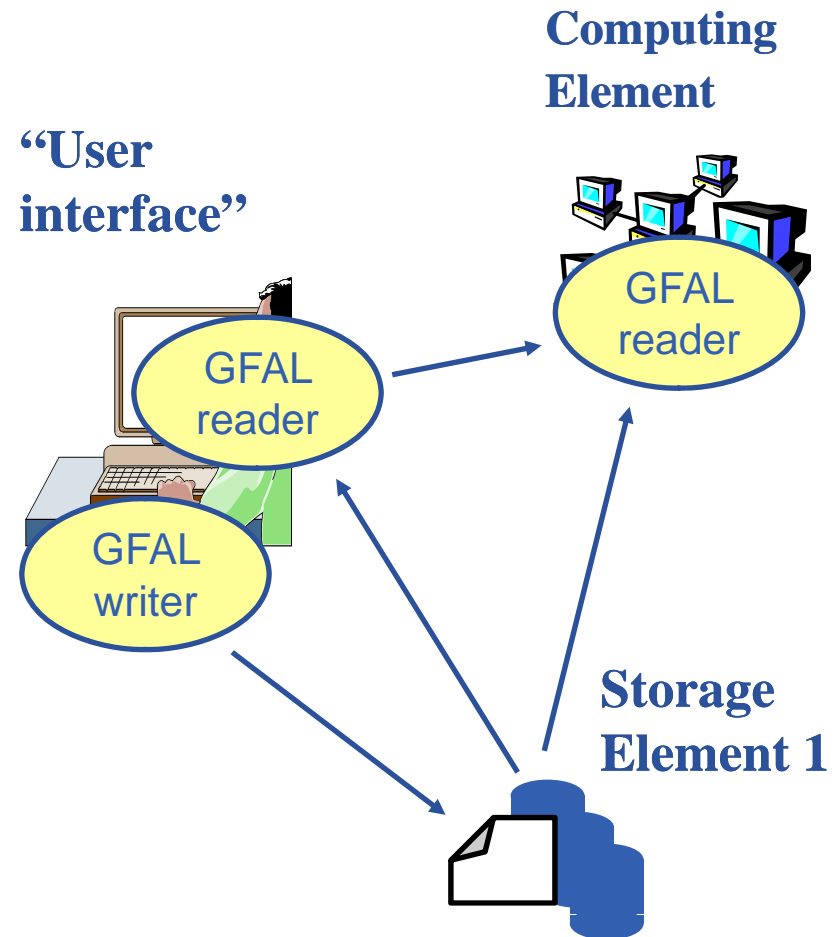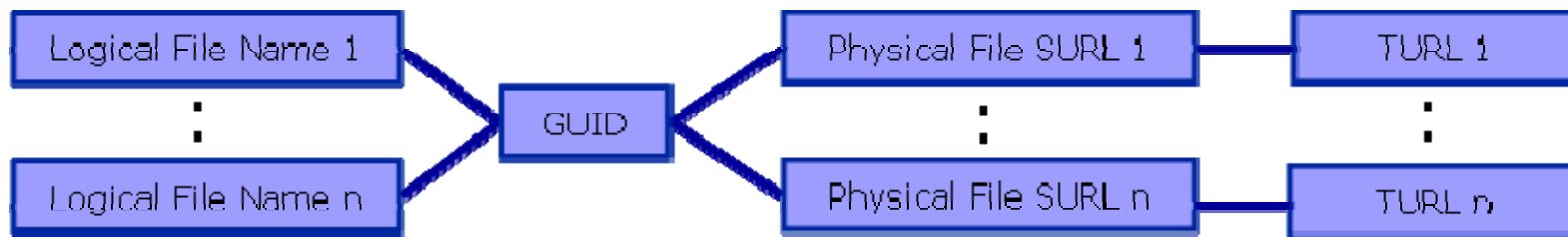**Storage Element 2**

Storage Element 1

- **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**

LCG File Catalogue (LFC)

lfc-*

"User interface"

Storage Element 2

? lcg-*

Storage Element 1
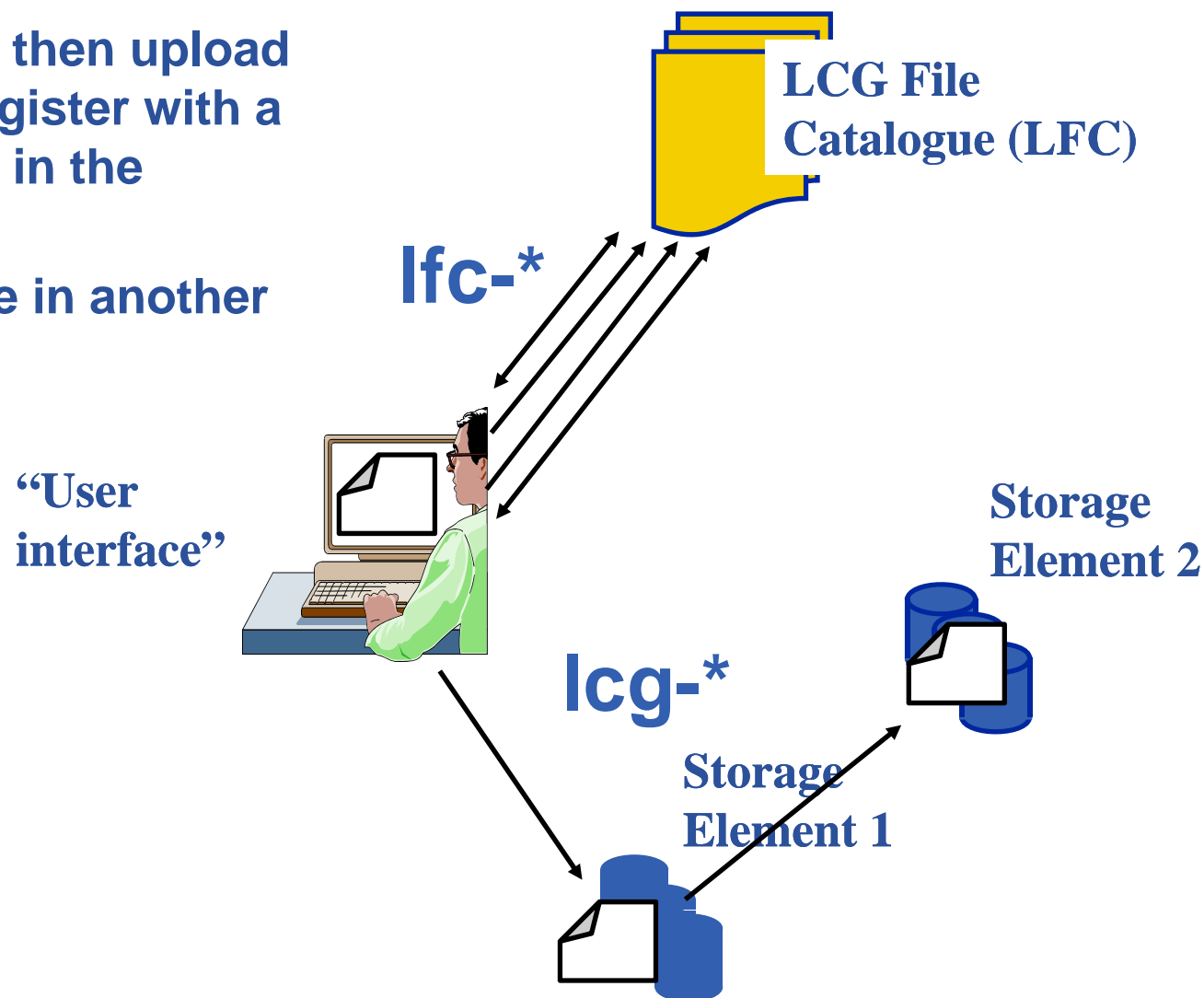
- **Create a second logical file name for a file**
- **Download a file from an SE to the UI**

- **Write a file to an SE**

- **Read a file from an SE**

- **Submit the reader code as a job into the GILDA, read the file remotely**

Computing Element

"User interface"

GFAL reader

GFAL reader

GFAL writer

Storage Element 1

Please go to the web page for this practical

- **Spare slides follow – could be used after the practical**

Summary of the LFC Catalog commands

lfc-chmod	Change access mode of the LFC file/directory
lfc-chown	Change owner and group of the LFC file-directory
lfc-delcomment	Delete the comment associated with the file/directory
lfc-getacl	Get file/directory access control lists
lfc-ln	Make a symbolic link to a file/directory
lfc-ls	List file/directory entries in a directory
lfc-mkdir	Create a directory
lfc-rename	Rename a file/directory
lfc-rm	Remove a file/directory
lfc-setacl	Set file/directory access control lists
lfc-setcomment	Add/replace a comment

Enabling Grids for E-sciencE

Replica Management

lcg-cp	**Copies a grid file to a local destination**
lcg-cr	**Copies a file to a SE and registers the file in the catalog**
lcg-del	**Delete one file**
lcg-rep	**Replication between SEs and registration of the replica**
lcg-gt	**Gets the TURL for a given SURL and transfer protocol**
lcg-sd	**Sets file status to "Done" for a given SURL in a SRM request**

FTS client

glite-transfer-submit	Submit a transfer job : needs at least source and destination SURL
glite-transfer-status	Given one or more job ID, query about their status
glite-transfer-cancel	Delete the transfer with the give Job ID
glite-transfer-list	Query about status of all user's jobs; support options for query restrictions
glite-transfer-channel-list	Show all available channel; detailed info only if user has admin privileges

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**