Enabling Grids for E-science in Europe

www.eu-egee.org

ESRIN Grid Workshop Tutorial
Introduction to Grid Computing
Frascati, 3 February 2005

Data Services

Presented by Julian Linford

Based on INFN-GRID/EGEE User Tutorial

EGEE is a project funded by the European Union under contract IST-2003-508833

Overview

Enabling Grids for E-science in Europe

- Introduction on Data Management (DM)
  - General Concepts
  - Some details on transport protocols
  - Data management operations
  - Files & replicas: Name Convention
- File catalogs
  - Cataloging requirements and catalogs in egee/LCG
  - RLS file catalog
  - LCG file catalog
- DM tools: overview
- Data Management CLI
  - lcg_utils
- Data Management API
  - lcg_utils
- Advanced concepts
  - Advanced utilities: CLI&APIs
- Conclusions

Data Management: general concepts

Enabling Grids for E-science in Europe

- A uniform approach to Facilitate distribution of data throughout the Grid
  - Provide common tools and services to handle files on the Grid
  - Granularity is at the "file" level (no data "structures")
- Files are stored in appropriate storage resources (large disks, or archive system)
  - Normally associated with a site's computing resources
    - Each CE normally has a 'CloseSE' configured
  - Data Management treats storage device as "black box"
    - hides internals of the storage resource
    - hides details on transfer protocols

Data Management: general concepts

Enabling Grids for E-science in Europe

- A Grid file is READ-ONLY (at least in EGEE/LCG)
  - It can not be modified
  - It can be deleted (so it can be replaced)
  - Files can be any type of data (text, binary, data, programs)
- High-level Data Management tools
  - Standard approach with automation deals with
    - Different transport layer details
    - Different sites storage configurations
- Low-level tools expose these differences
  - More details for the user to handle
    - Details of the transport layer
    - Details of Storage Element implementation
    - Only really useful in “non-standard” situations

Some details on protocols

Enabling Grids for E-science in Europe

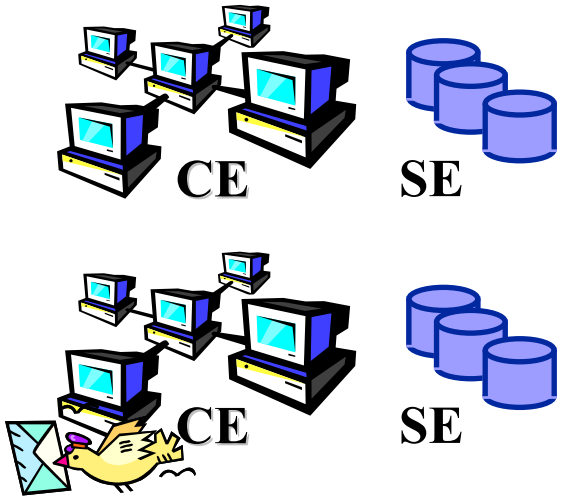
- Basic data transfer protocol between hosts is “gridFTP” (**gsiftp**)
  - secure and efficient data movement
  - extends the standard FTP protocol
  - Public-key-based Grid Security Infrastructure (GSI) support
  - Third-party control of data transfer
  - Parallel data transfer
- Other data access protocols are available
  - file protocol:
    - for local file access
  - rfio protocol
  - gsidcap protocol
- SRM provides standard access to storage devices

Data Management operations

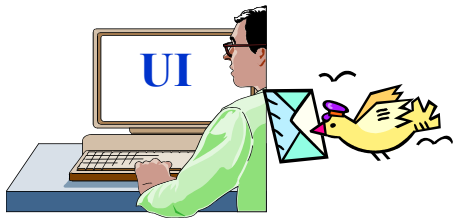
Enabling Grids for E-science in Europe

Upload a file to the grid

- User needs to store data in SE (from a UI)
- Application needs to store data in SE (from a WN)
- User needs to store the application (to be retrieved and run from WN)
  - For small files the InputSandbox can be used (see WMS lecture)



Several Grid Components

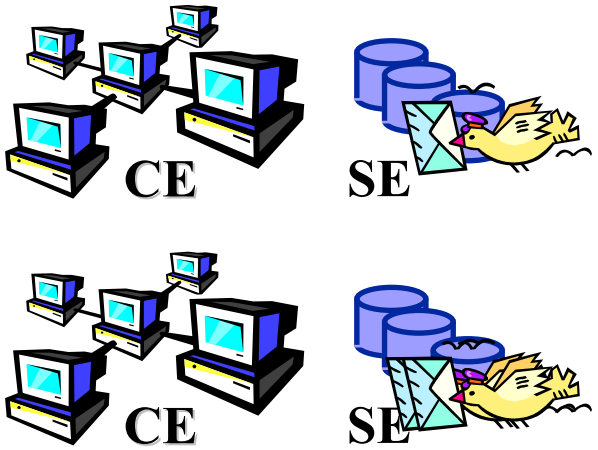


Data Management operations

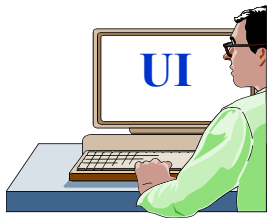
Enabling Grids for E-science in Europe

Download files from the grid

- User needs to retrieve data stored into SE
  - For small files produced in WN the OutputSandbox can be used
- Application needs to copy data locally (into the WN) and use them
- The application itself must be downloaded onto the WN and run



Several Grid Components

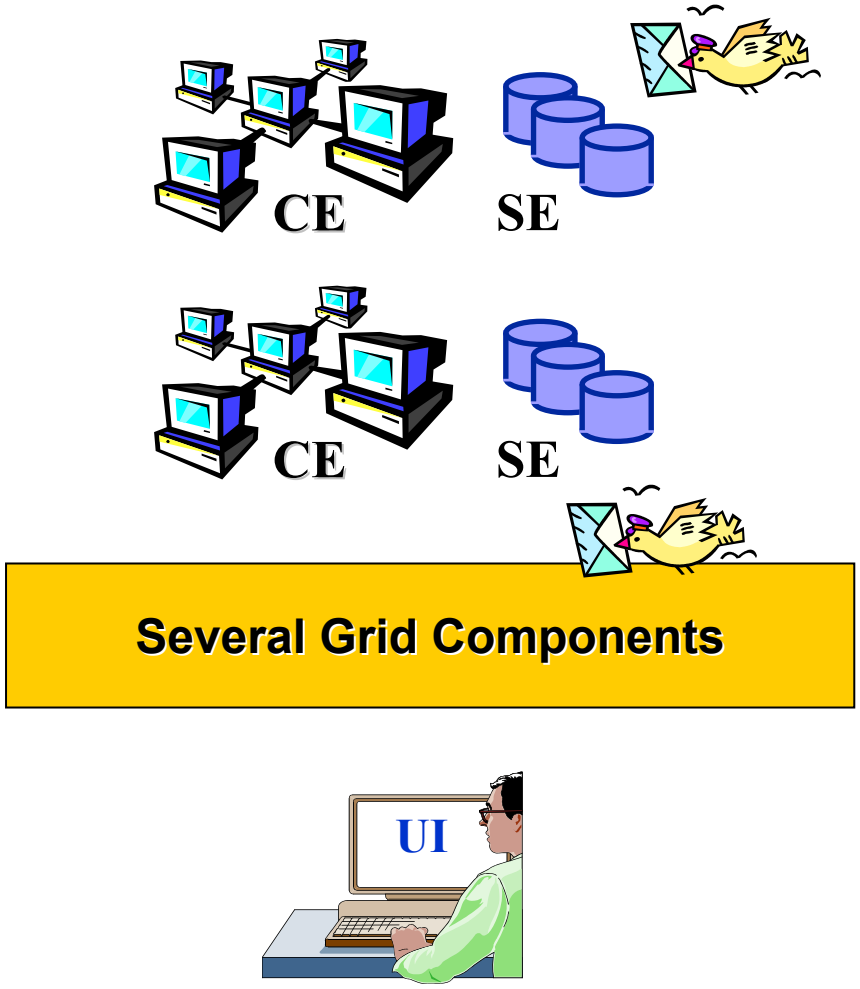


Data Management operations

Enabling Grids for E-science in Europe

Replicate a file on several SEs

- Load balancing of shared computing resources
  - Often a job needs to run at a site where a copy of input data is present
  - JDL InputData attribute allows this
- Performance improvement in data access
  - Several applications might need to access the same file concurrently
- Redundancy of key files provides backup



Data management operations

Enabling Grids for E-science in Europe

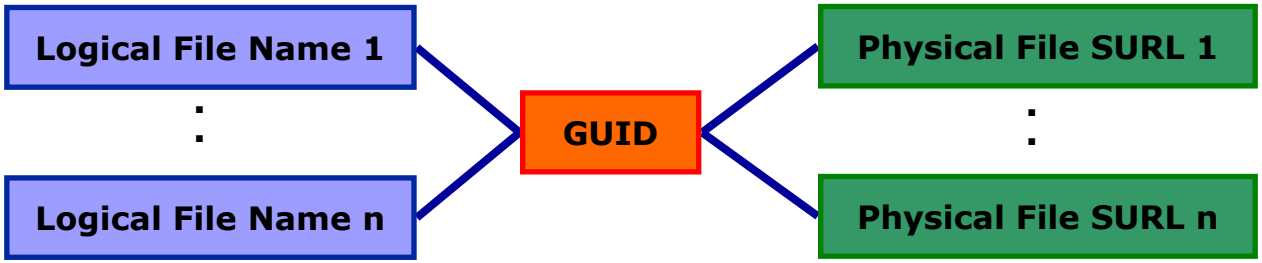
- Data Management means movement and replication of files across/on grid elements
- Grid DM tools/applications/services can be used for all kinds of files

HOWEVER

- Data Management focuses on "large" files
  - large means greater than ~20MB
  - Typically on the order of hundreds of MB
- Tools/applications/services are optimized to deal with large files
- Small files can be efficiently transferred using WM
  - User can send programms & data to the WN using the InputSandbox
  - User can retrieve data generated by a job (on the WN) using the OutputSandbox

Files & replicas: Name Convention

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



Replica Manager

Enabling Grids for E-science in Europe

- The Replica Manager allows to keep track of files on the Grid storage resources
- To track our files on the Grid we use a Replica Catalogue
- Potentially, millions of files need to be registered and located
  - Requirement for performance
- Distributed architecture might be desirable
  - scalability
  - prevent single-point of failure
  - Site managers need to change autonomously file locations

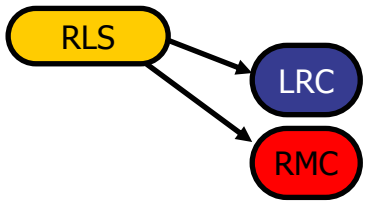
Replica Catalogs in EGEE/LCG

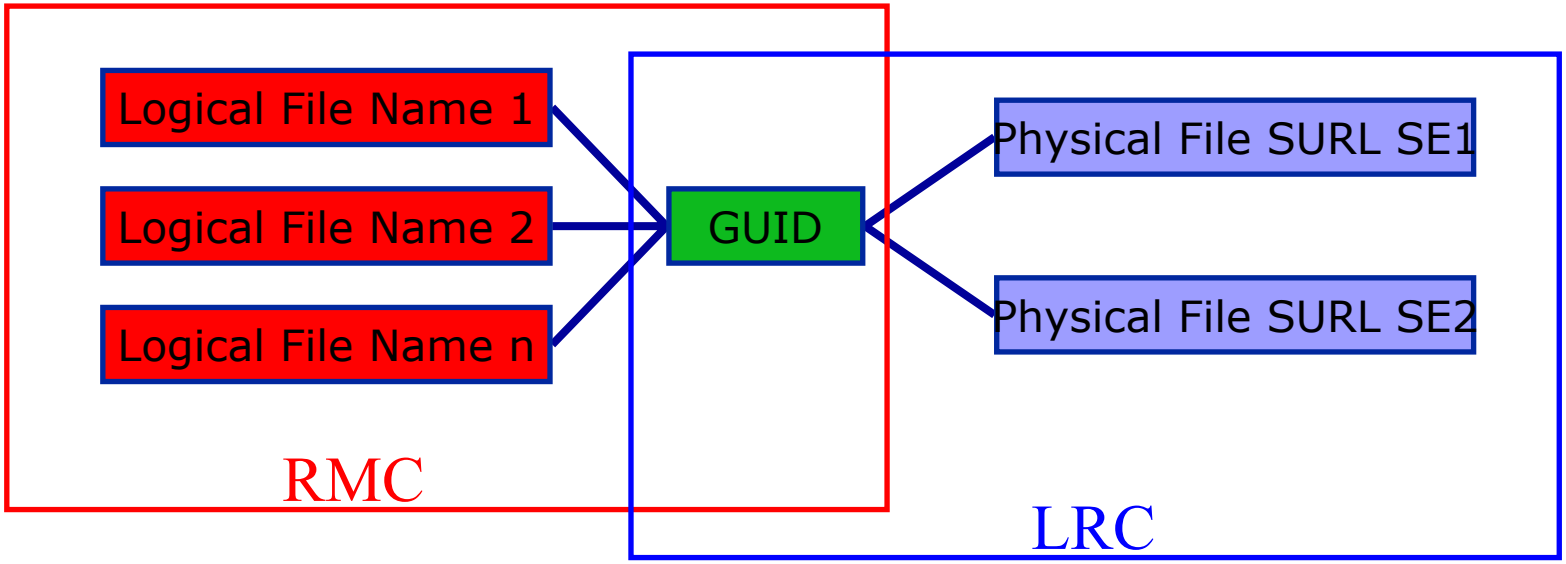
Enabling Grids for E-science in Europe

- Access to the file catalog
  - The DM tools and APIs and the WMS interact with the catalog
    - Hide catalogue implementation details
  - Lower level tools allow direct catalogue access
- Replica Location Service (RLS)
  - Catalogs in use in LCG-2
  - Replica Metadata Catalog (RMC) + Local Replica Catalog (LRC)
  - Some performance problems detected during LCG Data Challenges
- New LCG File Catalog (LCF)
  - deployment in January 2005
  - Coexistence with RLS and migration tools provided
  - Better performance and scalability
  - Provides new features: security, hierarchical namespace, transactions...

File Catalogs: The RLS

- **LRC**:
  - Stores GUID-SURL mappings
  - Accessible by edg-lrc CLI + API
- **RMC**:
  - Stores LFN-GUID mappings
  - Accessible by edg-rmc CLI + API





Possible Improvements

Enabling Grids for E-science in Europe

- Fix performance and scalability problems
  - Progress indicators for large queries
  - Timeouts and retries from the client
- More features
  - User exposed transaction API (+ auto rollback on failure of mutating method call)
  - Hierarchical namespace and namespace operations (for LFNs)
  - Integrated GSI Authentication + Authorization
  - Access Control Lists (Unix Permissions and POSIX ACLs)
  - Checksums
- Interaction with other components
  - Support Oracle and MySQL database back ends
- Security
  - VOMS will be integrated

Data management tools

- Replica manager: lcg-* commands + lcg_* API
  - Provide (all) the functionality needed by the EGEE/LCG user
  - Combine file transfer and cataloging as an **atomic transaction**
  - Insure consistent operations on catalogues and storage systems
  - Offers high level layer over technology specific implementations
  - Based on the Grid File Access Library (GFAL) API
    - Discussed in SE section

DM CLIs & APIs: Old EDG tools

Enabling Grids for E-science in Europe

- Old versions of EDG CLIs and APIs still available
- File & replica management
  - edg-rm
    - Implemented (mostly) in java
- Catalog interaction (only for EDG catalogs)
  - edg-lrc
  - edg-rmc
    - Java and C++ APIs
- Use discouraged
  - Worse performance (slower)
  - New features added only to lcg_utils
  - Less general than GFAL and lcg_utils

lcg_utils: Replica mgm. commands

lcg-cp	Copies a Grid file to a local destination
lcg-cr	Copies a file to a SE and registers the file in the LRC
lcg-del	Deletes one file (either one replica or all replicas)
lcg-rep	Copies a file from SE to SE and registers it in the LRC
lcg-sd	set file status to “Done” in a specified request

lcg_utils: Catalog interaction cmd's

lcg-aa	Adds an alias in RMC for a given GUID
lcg-gt	Gets the TURL for a given SURL and transfer protocol
lcg-la	Lists the aliases for a given LFN, GUID or SURL
lcg-lg	Gets the GUID for a given LFN or SURL
lcg-lr	Lists the replicas for a given LFN, GUID or SURL
lcg-ra	Removes an alias in RMC for a given GUID
lcg-rf	Registers a SE file in the LRC (optionally in the RMC)
lcg-uf	Unregisters a file residing on an SE from the LRC

Gathering informations: *lcg-infosites*

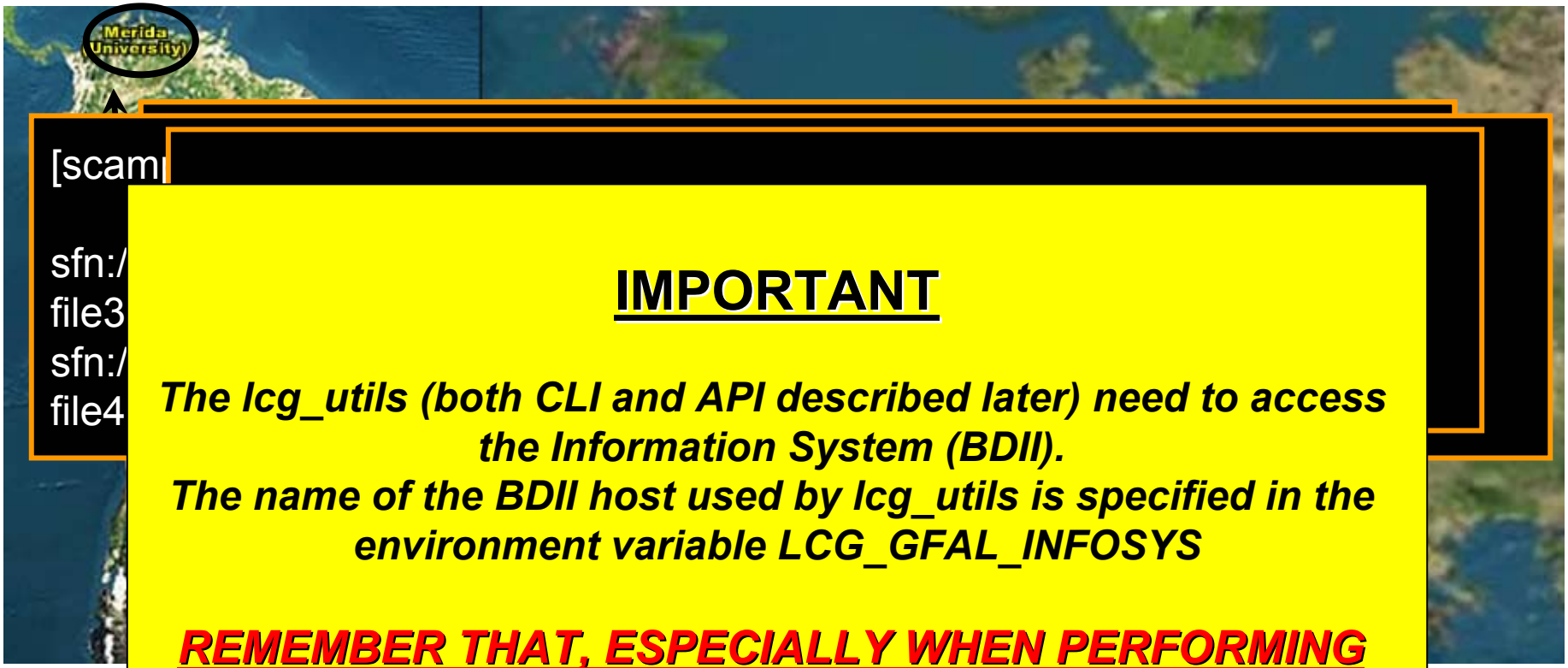
```
[scampana@grid019:~]$ lcg-infosites --vo gilda se

*****************************************************
These are the related data for gilda: (in terms of SE)
*****************************************************

Avail Space(Kb) Used Space(Kb)        SEs
-------------------------------------------------------
1570665704      576686868             grid3.na.astro.it
225661244       1906716               grid009.ct.infn.it
523094840       457000                grid003.cecalc.ula.ve
1570665704      576686868             testbed005.cnaf.infn.it
15853516        1879992               gilda-se01.pd.infn.it
```

lcg_utils CLI : usage example

Enabling Grids for E-science in Europe



[scam

sfn:/
file3
sfn:/
file4

IMPORTANT

The lcg_utils (both CLI and API described later) need to access the Information System (BDII).
The name of the BDII host used by lcg_utils is specified in the environment variable LCG_GFAL_INFOSYS

REMEMBER THAT, ESPECIALLY WHEN PERFORMING DATA MANAGEMENT OPERATIONS FROM THE WN

Catania

JDL Data Management Attributes

Enabling Grids for E-science in Europe

- **InputSandbox (optional)**
  - List of files on the UI will be automatically sent to the WN before execution

InputSandbox={“myscript.sh”,”/tmp/cc,sh”};

- **OutputSandbox (optional)**
  - List of files on the WN will be automatically retrieved by the “edg-job-get-output” command on the UI

OutputSandbox ={ “std.out”, ”std.err”, “image.png”};

JDL Replica Manager Attributes

Enabling Grids for E-science in Europe

- **InputData (optional)**
  - This is a string or a list of strings representing the *Logical File Name (LFN)* or *Grid Unique Identifier (GUID)*
  - The Resource Broker will select the “best” CE (i.e. which has the replicas stored on a ‘close’ SE)

InputData = {“lfn:mytestfile”,

“guid:135b7b23-4a6a-11d7-87e7-9d101f8c8b70”};

JDL Replica Manager Attributes

- **DataAccessProtocol** (**mandatory if InputData has been specified**)
  - The protocol which the job running on the WN will use for accessing files listed in *InputData*

- Supported protocols are currently **gridftp**, **file** and **rfio**.

DataAccessProtocol = {“file”,“gridftp”,“rfio”};

JDL Replica Manager Attributes

- **OutputSE** (**optional**)
  - URI of a **Storage Element (SE)**
  - The Resource Broker will select the “best” CE (i.e. that has OuputSE as ‘close’ SE)

OutputSE = “grid009.ct.infn.it”;

JDL Data Management Attributes

Enabling Grids for E-science in Europe

- **OutputData** (**optional**)
  - This attribute allows the user to ask for the automatic upload and registration of datasets produced by the job on the **Worker Node (WN)**.
  - This attribute contains the following three attributes:
    - ***OutputFile***
    - ***StorageElement***
    - ***LogicalFileName***

JDL OutputData Attributes

- **OutputFile** (**mandatory if OutputData has been specified**)
  - Name of the file on the WN
- **StorageElement** (**optional**)
  - URI of the target Storage Element
- **LogicalFileName** (**optional**)
  - LFN to be associated with the specified file

JDL OutputData Example

```
OutputData = {
        [
            OutputFile = “dataset1.out”;
            LogicalFileName = “lfn:test-result1”;
        ],
        [
            OutputFile = “dataset2.out”;
            LogicalFileName = “lfn:test-result2”;
            StorageElement = “grid009.ct.infn.it”;
        ],
        [
            OutputFile = “dataset3.out”;
        ]
};
```

JDL without Replica Management

```
Executable = "script.sh";
Arguments = "Hello World";
StdOutput = "stdout";
StdError = "stderr";
InputSandbox = {"script.sh"};
OutputSandbox = {"stderr", "stdout"};
```

JDL with Replica Management

```
Executable = "script.sh";
Arguments = "Hello World";
StdOutput = "stdout";
StdError = "stderr";
InputSandbox = {"script.sh"};
OutputSandbox = {"stderr", "stdout"};
InputData = "lfn:myoutdata.1";
DataAccessProtocol = {"gridftp", "rfio"};
```

Low-Level Commands

Enabling Grids for E-science in Europe

- globus-url-copy <sourceURL> <destURL>
  - low level file transfer
  - URL may have **file** or **gsiftp** as protocol
- Interaction with RLS components
  - edg-lrc command (actions on LRC)
  - edg-rmc command (actions on RMC)
  - C++ and Java API for all catalog operations
    - http://edg-wp2.web.cern.ch/edg-wp2/replication/docu/r2.1/edg-lrc-devguide.pdf
    - http://edg-wp2.web.cern.ch/edg-wp2/replication/docu/r2.1/edg-rmc-devguide.pdf
- Avoid using low level CLI and API where possible
  - Risk: loose consistency between SEs and catalogues
  - REMEMBER: a file is in Grid if it is BOTH:
    - stored in a Storage Element
    - registered in the file catalog

lcg_utils API

- lcg_utils API:
  - High-level data management C API
  - Same functionality as lcg_util command line tools
- Single shared library
  - liblcg_util.so
- Single header file
  - lcg_util.h

    (+ linking against libglobus_gass_copy_gcc32.so)

lcg_utils: Replica management

Enabling Grids for E-science in Europe

int **lcg_cp** (char *src_file, char *dest_file, char *vo, int nbstreams, char * conf_file, int insecure, int insecure);

int **lcg_cr** (char *src_file, char *dest_file, char *guid, char *lfn, char *vo, char *relative_path, int nbstreams, char *conf_file, int insecure, int verbose, char *actual_guid);

int **lcg_del** (char *file, int aflag, char *se, char *vo, char *conf_file, int insecure, int verbose);

int **lcg_rep** (char *src_file, char *dest_file, char *vo, char *relative_path, int nbstreams, char *conf_file, int insecure, int verbose);

int **lcg_sd** (char *surl, int regid, int fileid, char *token, int oflag);

lcg_utils: Catalog interaction

Enabling Grids for E-science in Europe

int **lcg_aa** (char *lfn, char *guid, char *vo, char *insecure, int verbose);

int **lcg_gt** (char *surl, char *protocol, char **turl, int *regid, int *fileid, char **token);

int **lcg_la** (char *file, char *vo, char *conf_file, int insecure, char ***lfns);

int **lcg_lg** (char *lfn_or_surl, char *vo, char *conf_file, int insecure, char *guid);

int **lcg_lr** (char *file, char *vo, char *conf_file, int insecure, char ***pfns);

int **lcg_ra** (char *lfn, char *guid, char *vo, char *conf_file, int insecure);

int **lcg_rf** (char *surl, char *guid, char *lfn, char *vo, char *conf_file, int insecure, int verbose, char *actual_guid);

int **lcg_uf** (char *surl, char *guid, char *vo, char *conf_file, int insecure);

Bibliography

- General egee/LCG information
  - EGEE Homepage
    http://public.eu-egee.org/
  - EGEE's NA3: User Training and Induction
    http://www.egee.nesc.ac.uk/
  - LCG Homepage
    http://lcg.web.cern.ch/LCG/
  - LCG-2 User Guide
    https://edms.cern.ch/file/454439//LCG-2-UserGuide.html
  - GILDA
    http://gilda.ct.infn.it/
  - GENIUS (GILDA web portal)
    http://grid-tutor.ct.infn.it/

Bibliography

- Information on Data Management middleware
  - LCG-2 User Guide (chapters 3rd and 6th)

    https://edms.cern.ch/file/454439//LCG-2-UserGuide.html
  - Evolution of LCG-2 Data Management. J-P Baud, James Casey.

    http://indico.cern.ch/contributionDisplay.py?contribId=278&sessionId=7&confId=0
  - Globus 2.4

    http://www.globus.org/gt2.4/
  - GridFTP

    http://www.globus.org/datagrid/gridftp.html

Bibliography

- Information on egee/LCG tools and APIs
  - Manpages (in UI)
    - lcg_utils: lcg-* (commands), lcg_* (C functions)
  - Header files (in $LCG_LOCATION/include)
    - lcg_util.h
  - CVS developement (sources for commands)

    http://isscvs.cern.ch:8180/cgi-bin/cvsweb.cgi/?hidenonreadable=1&f=u&logsort=date&sortby=file&hideattic=1&cvsroot=lcgware&path

- Information on other tools and APIs
  - EDG CLIs and APIs

    http://edg-wp2.web.cern.ch/edg-wp2/replication/documentation.html
  - Globus

    http://www-unix.globus.org/api/c/ , ...globus_ftp_client/html , ...globus_ftp_control/html