eGee

Enabling Grids for E-sciencE

More details on the gLite DMS

Roberto Barbera

University of Catania and INFN

EGEE Tutorial

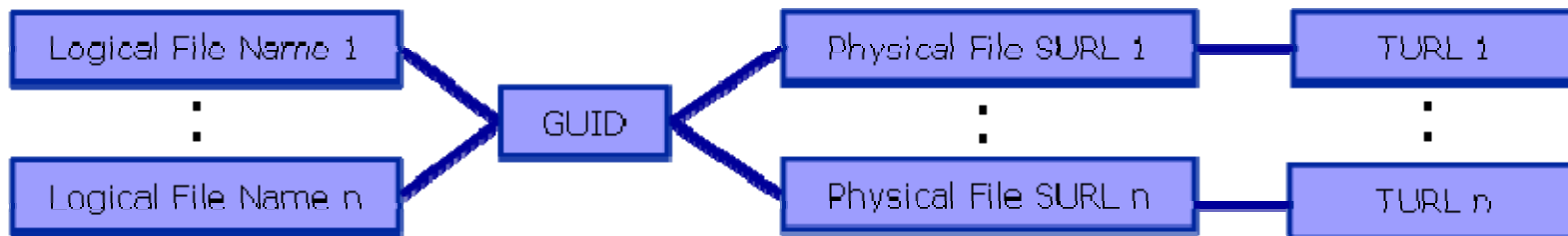*Seoul, 29-30.08.2005*

www.eu-egee.org

Information Society

Enabling Grids for E-sciencE

- **Data Management System**
  - LFC (the present)
  - FiReMan (the future)

LCG File Catalog (LFC)

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:cms/20030203/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

- **File catalogs in LCG:**
  - They keep track of the location of copies (replicas) of Grid files
  - The DM tools and APIs and the WMS interact with them

- **EDG's Replica Location Service (RLS, "old!")**
  - Catalogs in use in LCG-2
  - Replica Metadata Catalog (**RMC**) + Local Replica Catalog (**LRC**)
  - Some performance problems detected during Data Challenges

- **New LCG File Catalog (LFC, "current!")**
  - In production in next LCG release; deployment in January 2005
  - Coexistence with RLS; migration tools provided:

  http://goc.grid.sinica.edu.tw/gocwiki/How_to_migrate_the_RLS_entries_into_the_LCG_File_Catalog_%28LFC%29

  - Accessible by defining: $LCG_CATALOG_TYPE=lfc and $LFC_HOST
  - Better performance and scalability
  - Provides new features: security, hierarchical namespace, transactions...

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

- **RLS:**
  - Stores GUID-SURL mappings
  - Accessible by edg-lrc CLI + API

- **Main weaknesses:**
  - Insecure (anyone can delete catalog entries)
  - Bad performance (java clients…)

The LFC (the present)

Enabling Grids for E-sciencE
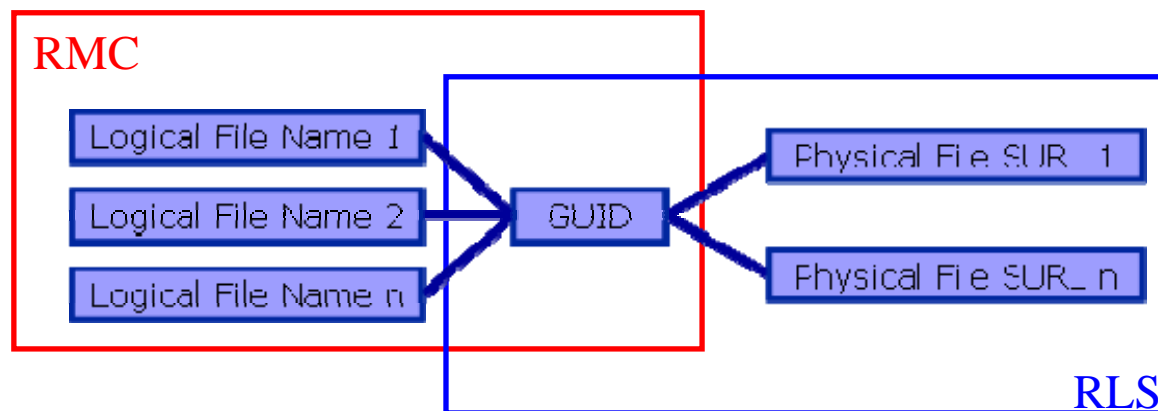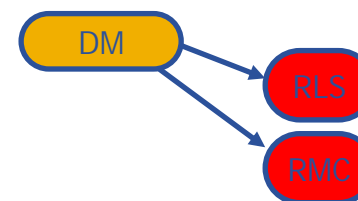
- **One single catalog**
- **LFN acts as main key in the database. It has:**
  - Symbolic links to it (additional LFNs)
  - Unique Identifier (GUID)
  - System metadata
  - Information on replicas
  - One field of user metadata

Enabling Grids for E-sciencE
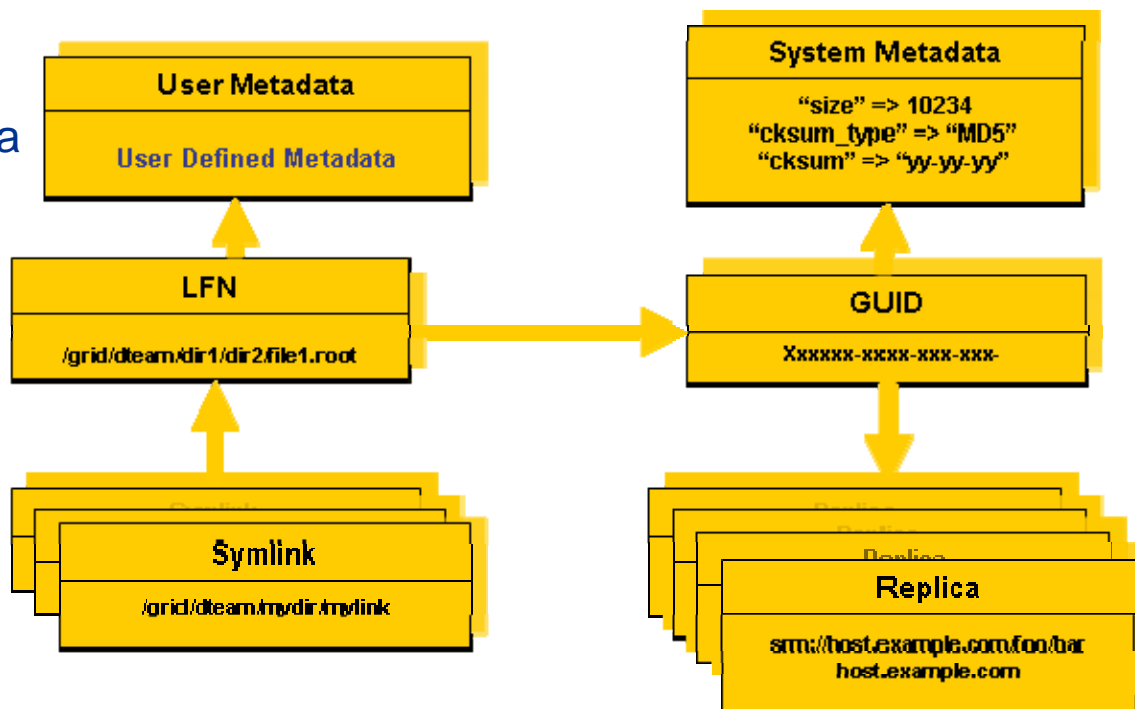
- **Fixes EDG catalogs performance and scalability problems**
  - Cursors for large queries
  - Timeouts and retries from the client

- **Provides more features than the EDG Catalogs**
  - User exposed transaction API (+ auto rollback on failure)
  - Hierarchical namespace and namespace operations (for LFNs)
  - Integrated GSI Authentication + Authorization
  - Access Control Lists (Unix Permissions and POSIX ACLs)
  - Checksums

- **New features will be added soon (requests welcome!)**
  - Integration with VOMS, FiReMan
  - POOL Integration is in progress
  - Sessions
  - Bulk operations

Enabling Grids for E-sciencE

- **LFC client commands**
  - Provide administrative functionality
  - Unix-like
  - LFNs seen as a Unix filesystem (/grid/<VO>/ …)

- **LFC C API**
  - Alternative way to administer the catalog
  - Python wrapper provided

- **Integration with GFAL and lcg_util APIs complete**
  - → lcg-utils access the catalog in a transparent way

- **Integration with the WMS completed**
  - The RB can locate Grid files: allows for data based match-making
  - Using the Data Location Interface
  - Not yet tested in production

- **lcg_utils: lcg-* commands + lcg_* API calls**
  - Provide (all) the functionality needed by the LCG user
  - Transparent interaction with file catalogs and storage interfaces when needed
  - Abstraction from technology of specific implementations

- **Grid File Access Library (GFAL): API**
  - Adds file I/O and explicit catalog interaction functionality
  - Still provides the abstraction and transparency of lcg_utils

- **edg-gridftp tools: CLI**
  - Complete the lcg_utils with low level GridFTP operations
  - Functionality available as API in GFAL
  - May be generalized as lcg-* commands

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**

File Catalog Interaction

lcg-aa	**Add an alias in LFC for a given GUID**
lcg-ra	**Remove an alias in LFC for a given GUID**
lcg-rf	**Registers in LFC a file placed in a SE**
lcg-uf	**Unregisters in LFC a file placed in a SE**
lcg-la	**Lists the alias for a given SURL, GUID or LFN**
lcg-lg	**Get the GUID for a given LFN or SURL**
lcg-lr	**Lists the replicas for a given GUID, SURL or LFN**

Enabling Grids for E-sciencE

Low level methods (many POSIX-like):

lfc_access

lfc_aborttrans

lfc_addreplica

lfc_apiinit

lfc_chclass

lfc_chdir

lfc_chmod

lfc_chown

lfc_closedir

lfc_creat

lfc_delcomment

lfc_delete

lfc_deleteclass

lfc_delreplica

lfc_endtrans

lfc_enterclass

lfc_errmsg

lfc_getacl

lfc_getcomment

lfc_getcwd

lfc_getpath

lfc_lchown

lfc_listclass

lfc_listlinks

lfc_listreplica

lfc_lstat

lfc_mkdir

lfc_modifyclass

lfc_opendir

lfc_queryclass

lfc_readdir

lfc_readlink

lfc_rename

lfc_rewind

lfc_rmdir

lfc_selectsrvr

lfc_setacl

lfc_setatime

lfc_setcomment

lfc_seterrbuf

lfc_setfsize

lfc_starttrans

lfc_stat

lfc_symlink

lfc_umask

lfc_undelete

lfc_unlink

lfc_utime

send2lfc

eGee

Enabling Grids for E-sciencE

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**

Managing ownership and permissions:

> **lfc-chmod**
>
> **lfc-chown**

Managing ACLs:

> **lfc-getacl**
>
> **lfc-setacl**

> Remember that per user mapping can change in every session.
>
> The default is for LFNs and directories to be VO-wide readable.
>
> Consistent user mapping will be added soon.

Renaming:

> **lfc-rename**

Removing:

> **lfc-rm**

> An LFN can only be removed if it has no SURLs associated.
>
> LFNs should be removed by lcg-del, rather than lfc-rm.

Enabling Grids for E-sciencE

- Information on the file catalogs
    - LFC, gfal, lcg-utils:

        "Evolution of LCG-2 Data Management (J-P Baud, J. Casey)"

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

    - LFC installation, administration, migration from RLS:
        - Wiki entries indicated through the presentation:
            - *http://goc.grid.sinica.edu.tw/gocwiki/How_to_set_up_an_LFC_service*
            - *http://goc.grid.sinica.edu.tw/gocwiki/How_to_migrate_the_RLS_entries_into_the_LCG_File_Catalog_%28LFC%29*

    - LFC contacts:
        - Jean-Philippe.Baud@cern.ch
        - Sophie.Lemaitre@cern.ch

File and Replica Management catalog (FiReMan) (the future)

eGee

- **Storage Element**
  - gLite **defines** the SE to have 3 interfaces:
    - Storage Resource Management (SRM) interface
    - Gridftp interface
    - Native I/O interface (rfio, dcap, nfs, ..)
  - LCG only requires the gridftp interface ("classic SE")
- **gLite: SRM is mandatory for each SE**
- **POSIX-like I/O:**

GFAL:

- client-side interaction with the SRM, storage and catalogs
- user certificate is used
- no atomicity guarantee

gLite – I/O:

- provides a server to process SRM, native I/O and catalog interactions
- client delegates user credential to glite I/O server
- glite I/O **owns** files on SE

Enabling Grids for E-sciencE

- **Managed File Transfer**

- **LCG provides command-line utilities through lcg-util to move data. All the operations are performed on the client.**

  – Blocking operation – client has to wait until the copy/replication is done

  – Scaling and Network resource management issue – if every job issues wide-area file movement operations from the worker nodes in a cluster, this will easily clog up the network

- **gLite provides services for asynchronous and bulk data movement**

  – File Transfer

  – File Placement (transfer including catalog registration)

eGee
Enabling Grids for E-sciencE

- **2 independent implementations exist**

Oracle Implementation

- **Catalog Logic lives inside Oracle as Stored Procedures**
- **Tomcat parses credential only, passes operations through to DB**

| TOMCAT5 |
| J2EE |
| Application Server |

| Database **ORACLE** |
| Application Logic |
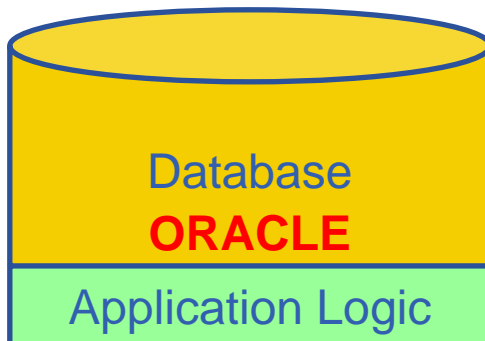
MySQL Implementation

- **Simple Table Structure using InnoDB tables**
- **Credential parsing and all of the logic is in Tomcat**

| TOMCAT5 |
| J2EE |
| Application Server |
| Application Logic |

| Database **MYSQL** |
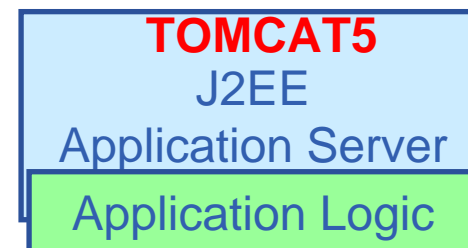
- **Storage Element** – **common interface to storage**
  - Storage Resource Manager   Castor, dCache, DPM, …
  - POSIX-I/O   gLite-I/O, rfio, dcap, xrootd
  - Access protocols   gsiftp, https, rfio, …

- **Catalogs** – **keep track where data is stored**
  - File Catalog
  - Replica Catalog   gLite File and Replica Catalog
  - File Authorization Service   Globus RLS
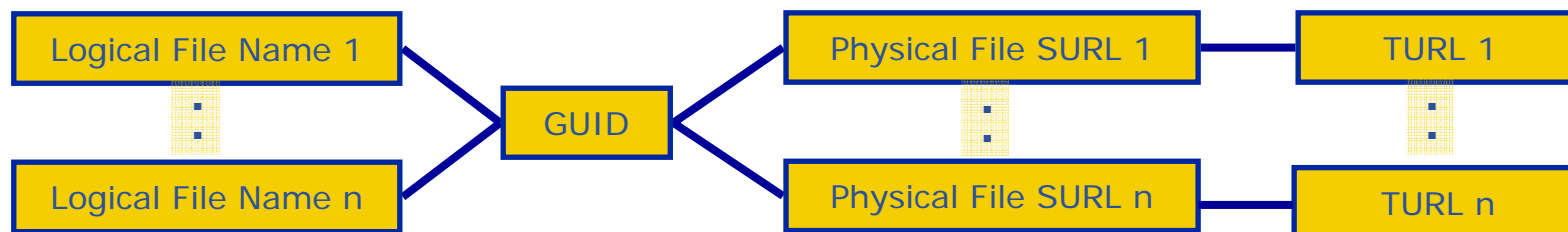 - Metadata Catalog   Application specific catalogs

- **File Transfer** – **scheduled reliable file transfer**
  - Data Scheduler   (only designs exist so far)
  - File Transfer Service   gLite FTS and glite-url-copy;

    (manages physical transfer)   Globus RFT, Stork
  - File Placement Service   gLite FPS

    (FTS and catalog interaction in a transactional way)

eGee

Enabling Grids for E-sciencE
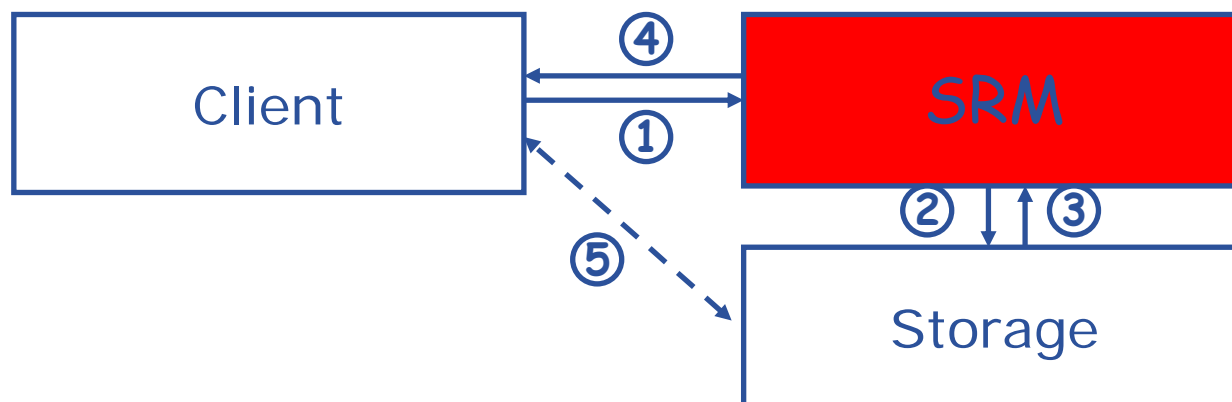
1. The client asks the SRM for the file providing an SURL (Site URL)
2. The SRM asks the storage system to provide the file
3. The storage system notifies the availability of the file and its location
4. The SRM returns a TURL (Transfer URL), i.e. the location from where the file can be accessed
5. The client interacts with the storage using the protocol specified in the TURL
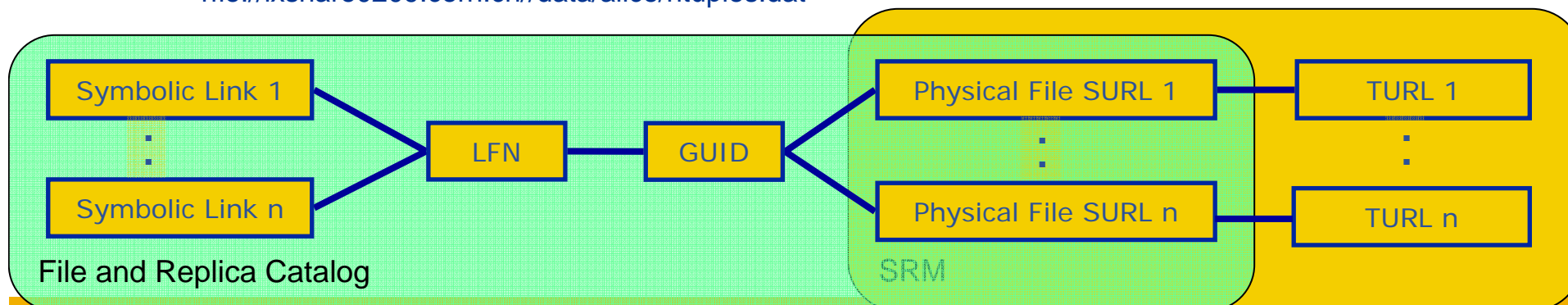
- **Logical File Name (LFN)**
  – An alias created by a user to refer to some item of data, e.g. "lfn:cms/20030203/run2/track1"

- **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"

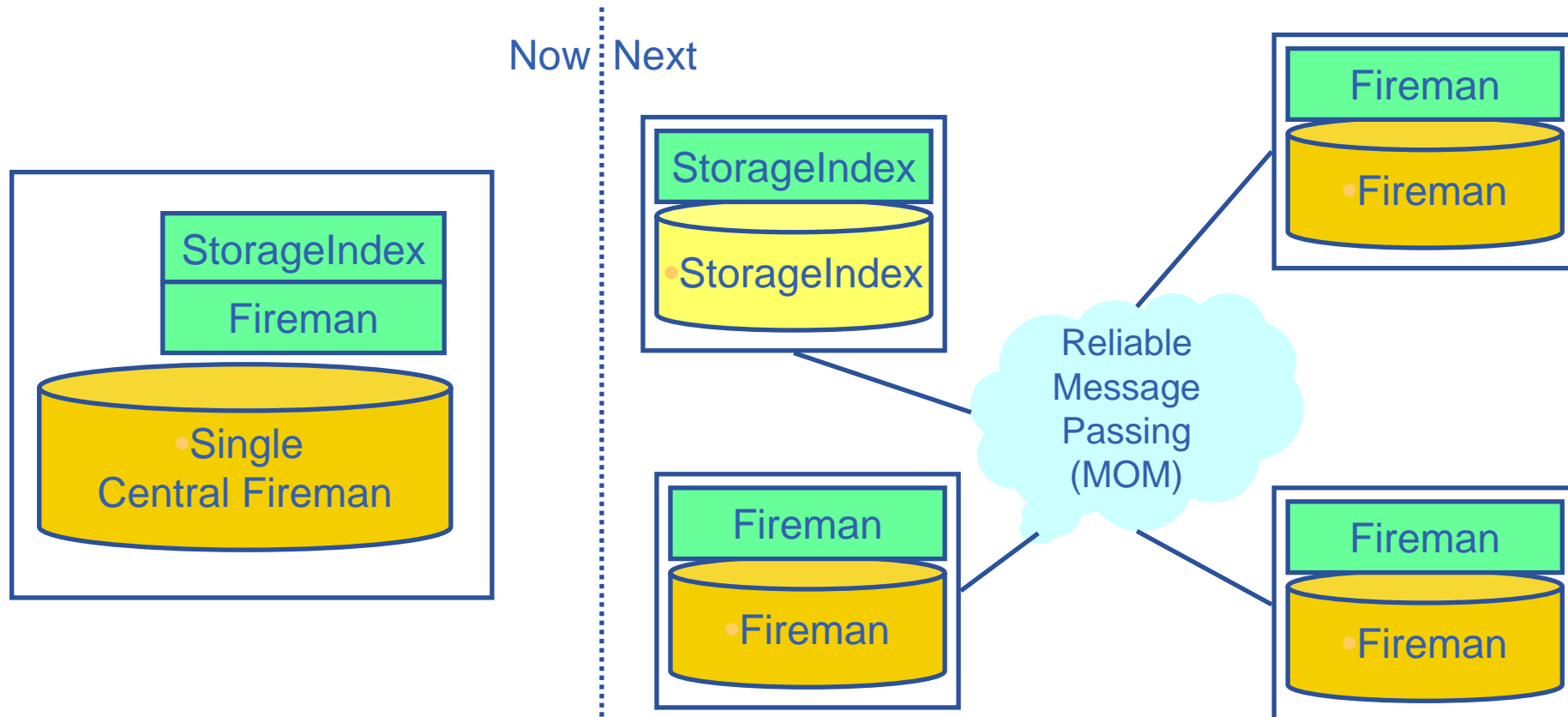| Logical File Name 1		Physical File SURL 1		TURL 1
.		.		.
GUID				
Logical File Name n		Physical File SURL n		TURL n

- **Symbolic Link** in logical filename space

- **Logical File Name (LFN)**
  - An alias created by a user to refer to some item of data, e.g. "lfn:cms/20030203/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"

| Symbolic Link 1				Physical File SURL 1		TURL 1
⋮	LFN	GUID		⋮		⋮
Symbolic Link n				Physical File SURL n		TURL n

File and Replica Catalog

SRM

- **Fireman: Currently only single central catalog implemented**
- **StorageIndex stores information on which SE stores a replica of the files**
- **Next step: Distribution**

Now | Next

StorageIndex

Fireman

• Single Central Fireman

StorageIndex

• StorageIndex

Fireman

• Fireman

Reliable Message Passing (MOM)

Fireman

• Fireman

Fireman

• Fireman

- **Logical File Namespace management**   **FileCatalog**
- **Replica locations**   **ReplicaCatalog**
- **File-based metadata**   **MetaBase**
- **Metadata Management**   **MetaSchema**
- **Authentication and Authorization information (ACLs)**   **FASBase**
- **Service Metadata**   **ServiceBase**
- **WMS interaction and global file location**   **ServiceIndex**

Interface Structure
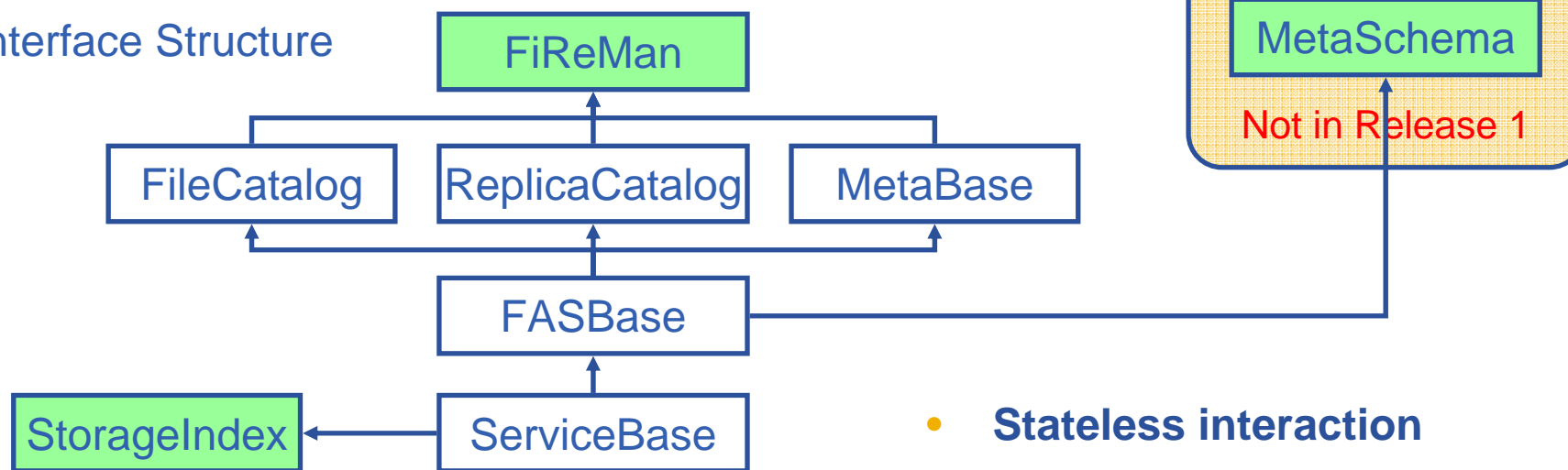
- **Stateless interaction**
- **No transactions outside Bulk**

- **Web-services interface**: Guarantees client support on many platforms and many languages.
- **Standardization effort ongoing**. It is being managed through the EGEE PTF. Are provided:
  - **Linux Command Line tools**
  - **C/C++ API**
  - **Java API**
  - **Perl modules**
  - **JavaScript (for web clients)**
  - **gLite integrated bash (glitesh) – prototype**
- **Security:** Fine-grained ACL support with minimal performance penalty.
  - DNs own the files
  - VOMS group support
  - Basic Unix security (ugo rwx)
  - Additional ACLs for setPermission, list, remove, setMetadata, getMetadata

egee

Summary of the Fireman Catalog commands

glite-catalog-chmod **glite-catalog-setacl** **glite-catalog-setdefacl** **glite-catalog-setdefperm**	**Change access mode of the Fireman file/directory. Set the ACL, the default ACL and the default permission**
glite-catalog-stat **glite-catalog-getguid**	**List the details of a file – all attributes, replicas. Or just the associated GUID.**
glite-catalog-setattr **glite-catalog-getattr** **glite-catalog-setschema**	**Set/get metadata attribute and set the metadata schema of a given directory**
glite-catalog-getacl **glite-catalog-getdefacl**	**Get file/directory access control lists and default ACL**
glite-catalog-symlink	**Make a symbolic link to a file. Directory symlinks are not supported by design.**

egee

Enabling Grids for E-sciencE

Summary of the Fireman Catalog commands

glite-catalog-ls	List file/directory entries in a directory
glite-catalog-mkdir	Create a directory
glite-catalog-mv	Rename a file/directory
glite-catalog-rm glite-catalog-rmdir	Remove a file/directory
glite-catalog-getreplica	Get all replicas associated with a file/GUID
glite-catalog-touch glite-catalog-create	Create a new entry in the catalog/update the modification time
glite-catalog-find	Find entries based on their name pattern
glite-seindex-list	List all SEs having a replica of the given files

Enabling Grids for E-sciencE

API level methods:

glite_catalog_free
glite_catalog_get_endpoint
glite_catalog_get_errclass
glite_catalog_get_error
glite_catalog_new
glite_catalog_set_default_perm
glite_catalog_set_error
glite_catalog_get_verror
glite_catalog_aclentry_clone
glite_catalog_aclentry_free
glite_catalog_aclentry_freearray
glite_catalog_aclentry_new
glite_catalog_attribute_clone
glite_catalog_attribute_free
glite_catalog_attribute_freearray
glite_catalog_attribute_new
glite_catalog_fcentry_clone
glite_catalog_fcentry_free
glite_catalog_fcentry_freearray
glite_catalog_fcentry_new
glite_catalog_fcentry_setguid
glite_catalog_fcentry_update
glite_catalog_fcentry_addsurl
glite_catalog_frcentry_clone
glite_catalog_frcentry_free
glite_catalog_frcentry_frearray
glite_catalog_frcentry_new
glite_catalog_frcentry_setchecksum
glite_catalog_frcentry_setguid
glite_catalog_guidstat_clone
glite_catalog_guidstat_copy
glite_catalog_guidstat_free
glite_catalog_guidstat_freearray

glite_catalog_guidstat_new
glite_catalog_guidstat_setchecksum
glite_catalog_lfnstat_clone
glite_catalog_lfnstat_copy
glite_catalog_lfnstat_free
glite_catalog_lfnstat_freearray
glite_catalog_lfnstat_new
glite_catalog_permission_addaclentry
glite_catalog_permission_clone
glite_catalog_permission_delaclentry
glite_catalog_permission_free
glite_catalog_permission_freearray
glite_catalog_permission_new
glite_catalog_permission_setgroupname
glite_catalog_permission_setusername
glite_catalog_rcentry_addsurl
glite_catalog_rcentry_clone
glite_catalog_rcentry_free
glite_catalog_rcentry_freearray
glite_catalog_rcentry_new
glite_catalog_rcentry_setchecksum
glite_catalog_stat_clone
glite_catalog_stat_free
glite_catalog_stat_freearray
glite_catalog_stat_new
glite_catalog_surlentry_clone
glite_catalog_surlentry_free
glite_catalog_surlentry_freearray
glite_catalog_surlentry_new
glite_fireman_expand_path
glite_fireman_get_locate_limit
glite_fireman_get_query_limit
glite_fireman_get_readdir_limit

glite_fireman_getinterfaceversion
glite_fireman_getschemaversion
glite_fireman_getservicemetadata
glite_fireman_getversion
glite_fireman_checkpermission
glite_fireman_getpermission
glite_fireman_setpermission
glite_fireman_createfile
glite_fireman_getfilecatalogentry
glite_fireman_getguidforlfn
glite_fireman_getlfnforguid
glite_fireman_locate
glite_fireman_mkdir
glite_fireman_mv
glite_fireman_readdir
glite_fireman_rmdir
glite_fireman_symlink
glite_fireman_unlink
glite_fireman_updatemodifytime
glite_fireman_updatevaliditytime
glite_fireman_addguidreplica
glite_fireman_clearattributes
glite_fireman_createguid
glite_fireman_getatributes
glite_fireman_getdefaultglobalpermission
glite_fireman_getdefaultprincipalpermission
glite_fireman_getguidforsurl
glite_fireman_getguidstat
glite_fireman_getmasterreplica
glite_fireman_getsurlstat
glite_fireman_hasguid
glite_fireman_listattributes
glite_fireman_listreplicasbyguid
glite_fireman_listsurlsbyguid
glite_fireman_query
glite_fireman_removeguid
glite_fireman_removeguidreplica

glite_fireman_setattributes
glite_fireman_setdefaultglobalpermission
glite_fireman_setdefaultprincipalpermission
glite_fireman_setmasterreplica
glite_fireman_updateguidstat
glite_fireman_updatestatus
glite_fireman_updatesurlstat
glite_fireman_addreplica
glite_fireman_associatedirwithschema
glite_fireman_create
glite_fireman_getstat
glite_fireman_listlfn
glite_fireman_listreplicas
glite_fireman_remove
glite_fireman_removereplica
glite_seindex_getinterfaceversion
glite_seindex_getschemaversion
glite_seindex_getversion
glite_seindex_listsebyguid
glite_seindex_listsebylfn
glite_conf_value
glite_config_file
glite_discover_endpoint
glite_freestringarray
glite_location
glite_location_log
glite_location_var
glite_pkg_var
glite_tmp
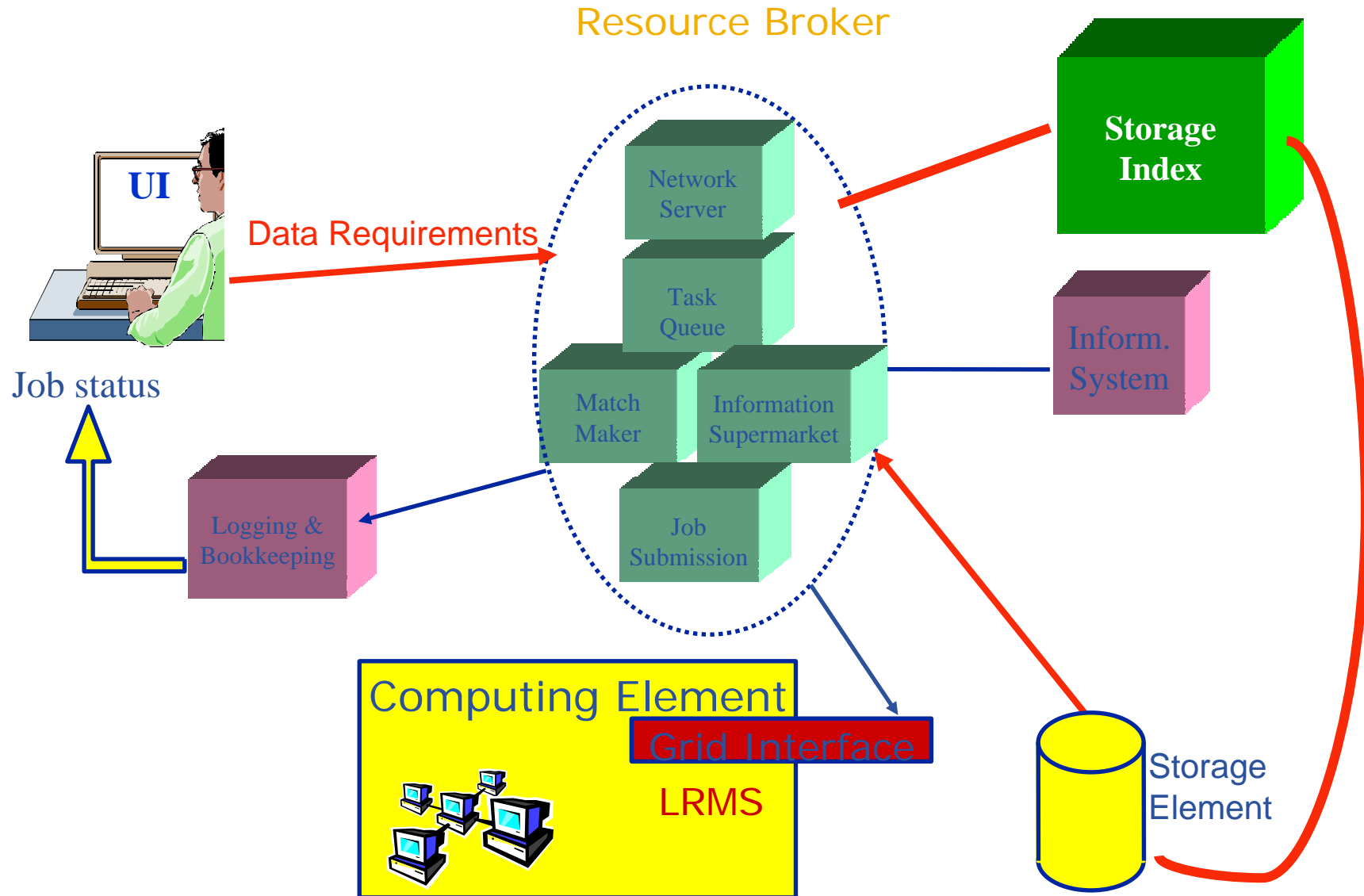glite_uri_free
glite_uri_new

RED methods also have bulk versions

Summary of the gLite I/O command line tools

glite-get	Retrieve a file from the Grid using LFN or GUID
glite-put	Put a local file into the Grid, assigning LFN
glite-rm	Remove a file (replica!) from the Grid using LFN or GUID

Summary of the gLite I/O API calls (C only)

glite_open	glite_posix_open
glite_read	glite_posix_read
glite_write	glite_posix_write
glite_creat	glite_posix_creat
glite_fstat	glite_posix_fstat
glite_lseek	glite_posix_lseek
glite_close	glite_posix_close
glite_unlink	glite_posix_unlink
glite_error	glite_filehandle
glite_strerror	

Resource Broker

UI

Data Requirements

Job status

Network Server

Task Queue

Match Maker

Information Supermarket

Job Submission

Storage Index

Inform. System

Logging & Bookkeeping

Computing Element

Grid Interface

LRMS

Storage Element

```
[
Executable = "helloCSC.sh";
StdOutput = "Message.txt";
StdError = "stderr.log";
StorageIndex = "http://lxb2028.cern.ch:8080/EGEE/glite-data-
catalog-service-fr/services/SEIndex";
InputData = "lfn:///tmp/testCSC";
DataAccessProtocol = "gridftp,gliteio";
InputSandbox = {"helloGet.sh"};
OutputSandbox = {"Message.txt","stderr.log", "testfile.txt"};
]
```

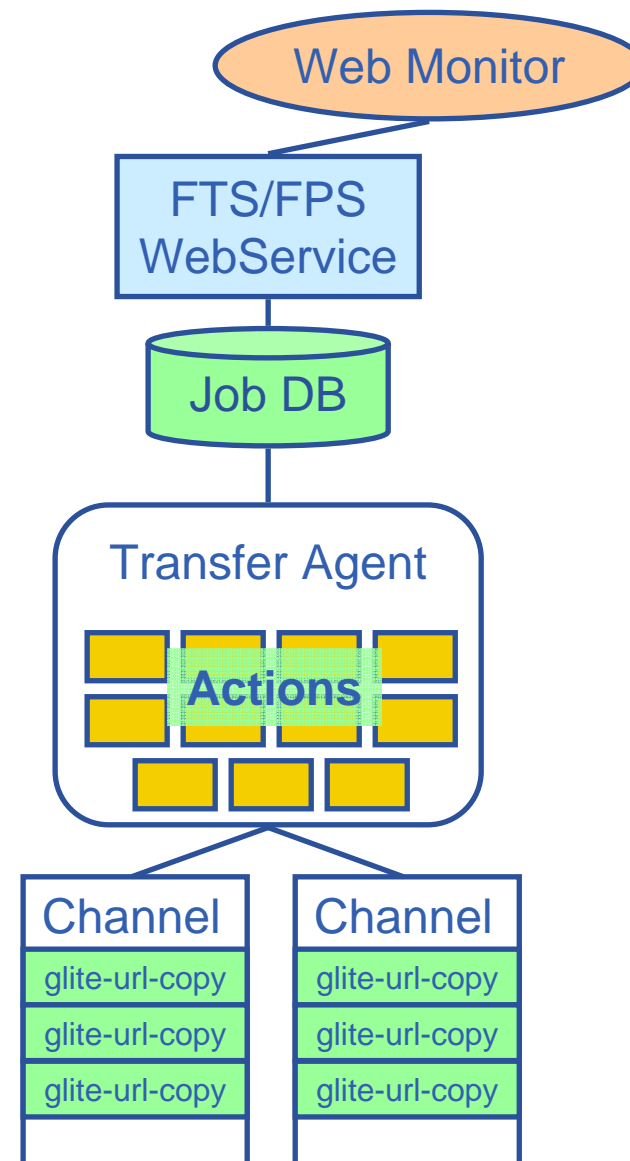Endpoint of the Catalog (StorageIndex interface)

LFN of the file needed

Access protocol used

Enabling Grids for E-sciencE

- **Many Grid applications will distribute a LOT of data across the Grid sites**
- **Need efficient and easy to manage File movement service**

- **gLite File Transfer Service FTS**
  – Manage the network and the storage at both ends
  – Define the concept of a CHANNEL: a link between two SEs
  – Channels can be managed by the channel administrators, i.e. the people responsible for the network link and storage systems
  – These are potentially different people for different channels
  – Optimize channel bandwidth usage – lots of parameters that can be tuned by the administrator
  – VOs using the channel can apply their own internal policies for queue ordering (i.e. professor's transfer jobs are more important than student's)
- **gLite File Placement Service**
  – It **IS** an FTS with the additional catalog lookup and registration steps, i.e. LFNs and GUIDs can be used to perform replication. Could've been called File Replication Service. (**replica = managed/catalogued copy**)

- **Data transfer and access protocol for** secure and efficient **data movement**

- **Standardized in the Global Grid Forum**

- extends **the standard** FTP **protocol**
  - Public-key-based Grid Security Infrastructure (GSI) or Kerberos support (both accessible via GSS-API
  - Third-party control of data transfer
  - Parallel data transfer
  - Striped data transfer Partial file transfer
  - Automatic negotiation of TCP buffer/window sizes
  - Support for reliable and restartable data transfer
  - Integrated instrumentation, for monitoring ongoing transfer performance

- **GridFTP is the basis of most transfer systems**

- **Retry functionality is limited**
    - Only retries in case of network problems; no possibility to recover from GridFTP a server crash

- **GridFTP handles one transfer at a time**
    - No possibility to do bulk optimization
    - No possibility to schedule parallel transfers

- **Need a layer on top of GridFTP that provides reliable scheduled file transfer**
    - FTS/FPS
    - Globus RFT (layer on top of single gridftp server)
    - Condor Stork

- **File Transfer/Placement Service (FTS,FPS)**
  - Transfer Job Database
  - Exposes the Transfer Web Service Interface to which user clients talk (submit, cancel, status capability)
  - Has a Web Interface
  - Manages Catalog updates if necessary
- **Transfer Agent**
  - Basic Actions
    - Get transfer jobs from Transfer Job Database
    - Manages transfer over many channels
    - Monitors transfer status and updates Transfer Job Database
  - Extensible with user-defined custom actions
    - Retry Policy
- **Transfer Service (glite-url-copy)**
  - Actually performs transfer: SRM – SRM, gsiftp – SRM, gsiftp – gsiftp
  - Monitor capability, including gsiftp performance markers

Web Monitor

FTS/FPS WebService

Job DB

Transfer Agent

Actions

Channel
glite-url-copy
glite-url-copy
glite-url-copy

Channel
glite-url-copy
glite-url-copy
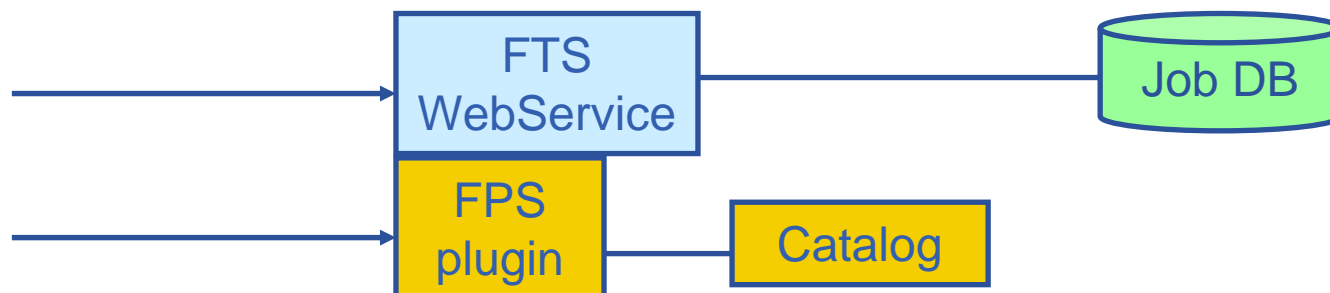glite-url-copy

Enabling Grids for E-sciencE

- **File Transfer Service (FTS)**
  - Acts only on SRM SURLs or gsiftp URLs
  - `submit(source-SURL, destination-SURL)`

- **File Placement Service (FPS)**
  - A plug-in into the File Transfer that allows to act on logical file names (LFNs)
  - Interacts with replica catalogs (similar to gLite-I/O)
  - Registers replicas in the catalog
  - `submit(transferJobs) (transferJob = sourceLFN,`
    `destinationSE)`

```
                        ┌──────────────┐                    ┌──────────┐
  ─────────────────────▶│     FTS      │────────────────────│  Job DB  │
                        │  WebService  │                    └──────────┘
                        ├──────────────┤
                        │     FPS      │        ┌──────────┐
  ─────────────────────▶│    plugin    │────────│ Catalog  │
                        └──────────────┘        └──────────┘
```

File Transfer commands

Enabling Grids for E-sciencE

Summary of the FTS/FPS commands

glite-transfer-submit	**Submit a transfer job, consisting of source/target pairs.**
glite-transfer-cancel	**Cancel an existing job**
glite-transfer-status	**Retrieve the status of a transfer job**
glite-transfer-list	**List jobs**
glite-transfer-channel	**Get all replicas associated with a file/GUID**
glite-catalog-touch **glite-catalog-create**	**Create a new entry in the catalog/update the modification time**
glite-catalog-find	**Find entries based on their name pattern**
glite-seindex-list	**List all SEs having a replica of the given files**

API is also available in C and Java (WSDL-autogenerated)
Simple C API is in the works, will be available in gLite 1.2.x

- **Using the File Transfer Service (FTS)**
  - Lookup source SURL in replica catalog
  - Initiate and monitor transfer
  - After successful transfer register new replica in the catalog

- **Using the File Placement Service (FPS)**
  - Initiate and monitor transfer
  - Plugin takes care of catalog interactions

- **FTS and FPS offer the same interface**
  - Difference only in input parameters to the submit command
    - SURLs vs. LFNs
  - Different configuration
    - FPS requires catalog endpoint

Enabling Grids for E-sciencE

- **gLite homepage**
  - http://www.glite.org

- **DM subsystem documentation**
  - http://egee-jra1-dm.web.cern.ch/egee-jra1-dm/doc.htm

- **FiReMan catalog user guide**
  - https://edms.cern.ch/file/570780/1/EGEE-TECH-570780-v1.0.pdf

- **gLite-I/O user guide**
  - https://edms.cern.ch/file/570771/1.1/EGEE-TECH-570771-v1.1.pdf

- **FTS/FPS user guide**
  - https://edms.cern.ch/file/591792/1/EGEE-TECH-591792-Transfer-CLI-v1.0.pdf