

A File in the DataGrid

A Brief Review on File Management in DataGrid

Isaac Iglesias Faustino, IT Summer Student

Introduction

- This document describes how to manage files within the **European DataGrid**
- Three single operations:
 - How to **put** a file in the Grid
 - How to **copy** (duplicate) it
 - How to **erase** it
- Two different tools:
 - **GDMP** (Grid Data Mirroring Package)
 - **EDG-REPLICA-MANAGER**

Main Concepts

- **Grid:**
 - Emerging infrastructure
 - Totally change the actual way of computing
 - Allows its users not only to exchange information (as the Internet)...
 - ... but also *every* computational resource
- **DataGrid:**
 - Project funded by European Union
 - Objective: to build the next generation computing infrastructure providing:
 - Intensive computation
 - Analysis of shared large-scale databases

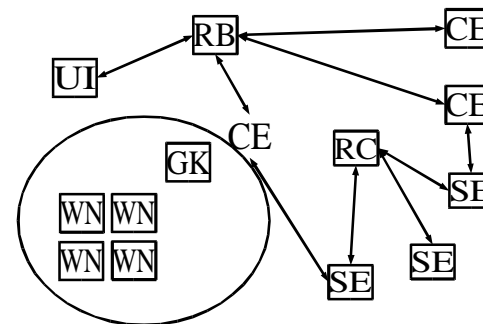
Elements in a Grid (I)

- There are several different elements in a Grid that must be introduced first
 - **UI (User Interface):**
 - Allows the end-user to interact with the Grid
 - The machine the end-user logs into
 - **RB (Resource Broker):**
 - Accepts jobs
 - Match its requirements to available resources
 - Dispatches them
 - **CE (Computing Element):**
 - Consists of a gatekeeper node and one or more worker nodes
 - Provides computational resources to the user

Elements in a Grid (II)

- **GK (GateKeeper):**
 - The front-end of a CE
 - Handles interaction with the rest of the Grid by accepting, dispatching for execution or returning jobs to the output
- **WN (Worker Nodes):**
 - Nodes where user computation is actually performed.
 - Managed by the GK
- **SE (Storage Element):**
 - Provides uniform access to large storage spaces...
 - ... with help from the Replica Catalog
- **RC (Replica Catalogue):**
 - Maintains a database of the locations of master copies of files and the locations of any replicas
 - They do not hold any data, but only describe it
 - Connected to several CE's

Elements in a Grid (III)



Grid Data Mirroring Package

- **Globus:** The Globus Project is developing fundamental technologies needed to build computational grids. Those are...
 - ... persistent environments
 - integrate instruments, displays, computational and information resources,
 - that are managed by diverse organizations in widespread locations.
- The **GDMP** client-server software system
 - A generic file replication tool that replicates files securely and efficiently in a DataGrid environment
 - Uses several Globus Grid tools.
 - A collaboration between the European DataGrid project and Particle Physics DataGrid (PPDG).
- **RM (Replica Manager):**
 - Client tools that operate using GDMP software
 - We will see it in depth later on.

GDMP vs. EDG-RM

GDMP

- Replication between SE's only
- Replicates set of files
- Provides MSS interface
- Client-server
- Logical file attributes
- Subscription model
- Event notification
- CRC file size check
- Support for Objectivity/DB
- Automatic retries
- Support for multiple VOs

EDG-RM

- Replication between SEs, UI* or CE to SE
- Replicates single files
- Uses GDMP's MSS interface
- Client side only

NOTE: The documentation say so but actually is not possible unless it runs grid ftp.

GDMP: Catalogues (I)

- There are three different file catalogues that GDMP utilizes to handle the files:

- Local File Catalogue:

- Contains all the files that are locally managed by the node.
- Its content ought to be published for outer nodes to copy (import) files.

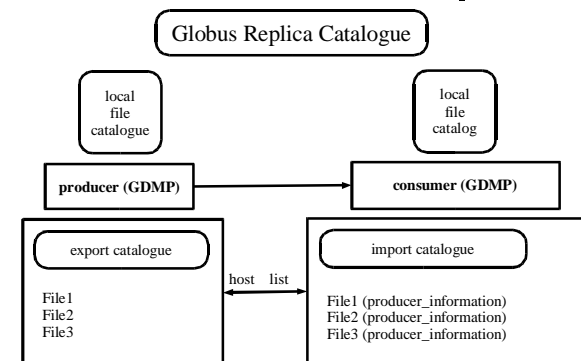
- Export Catalogue:

- Contains all the file that a node offers to other nodes to copy
- The files in the local catalogue are registered in this one at the moment of publishing

- Import Catalogue:

- Contains the files that are wanted to be copied (imported) by a machine.
- Apart from the file, it contains some extra information on the producer
- If a node is subscribed to another one, it will automatically receive import information every time a file is published

GDMP: Catalogues (II)



GDMP: How to Put a File

- There are three main steps to put a file in the Grid

- First of all, it is necessary to place it somehow in the Storage Element. We use one of the Globus commands.

- However, this is not enough. There is a catalogue every SE keeps which contains all the files it manages. The next step should be registering it in the mentioned catalogue.

- The third and the only optional step would be:

- Checking if everything went right,
- by obtaining the status of the registering job and finally
- listing the files in the local catalogue

GDMP: How to Put a File

- There are three main steps to put a file in the Grid

- **First of all, it is necessary to place it somehow in the Storage Element. We use one of the Globus commands.**

- However, this is not enough. There is a catalogue every SE keeps which contains all the files it manages. The next step should be registering it in the mentioned catalogue.

- The third and the only optional step would be:

- Checking if everything went right,
- by obtaining the status of the registering job and finally
- listing the files in the local catalogue

GDMP: How to Put a File (I)

- First step to put a file in the Grid for all the users to see would be physically copying it to the SE.
- We will use three different elements:
 - a Globus command named *globus-url-copy*,
 - the well-known *file* protocol
 - and *gsiftp* which is an enhanced version of ftp with some Grid authentication.

```
[UI] /path $ globus-url-copy file://pathUI/filename.txt gsiftp://pathSE/filename.txt
```

- The file is now copied to the Storage Element we have chosen

GDMP: How to Put a File

- There are three main steps to put a file in the Grid

- First of all, it is necessary to place it somehow in the Storage Element. We use one of the Globus commands.

- However, this is not enough. There is a catalogue every SE keeps which contains all the files it manages. The next step should be registering it in the mentioned catalogue.

- The third and the only optional step would be:
 - Checking if everything went right,
 - by obtaining the status of the registering job and finally
 - listing the files in the local catalogue.

GDMP: How to Put a File (I)

- To update the mentioned catalogue, the next step to take would be putting the file in the catalogue.
- We use the command *gdmp_register_local_file*.
 - We use the *-p* parameter to specify a file...
 - ... or the *-d* parameter to register a whole directory
 - We could also specify server, port or Virtual Organization

```
[UI] /path $ gdmp_register_local_file -p /pathSE/filename.txt
Server Message [SE]: A client has been started to register the requested files.
Message: Server Log ID=SE_1058971661_1
```

GDMP: How to Put a File

- There are three main steps to put a file in the Grid

- First of all, it is necessary to place it somehow in the Storage Element. We use one of the Globus commands.

- However, this is not enough. There is a catalogue every SE keeps which contains all the files it manages. The next step should be registering it in the mentioned catalogue.

- The third and the only optional step would be:
 - Checking if everything went right,
 - by obtaining the status of the registering job and finally
 - listing the files in the local catalogue.

GDMP: How to Put a File (II)

- Now, the registration process has been started. There is a job ID that we need to check if it worked. We use `gdmp_job_status`.

```
[UI] /path $ gdmp_job_status -f job-id
Message: Obtaining file attribs e.g size, timestamp, checksum etc for /pathSE/filename.txt
Message: File attribs for /pathSE/filename.txt obtained.
Message: Registering file:SE.path.filename_.txt:session.txt:986:2432916819:1058970986
Message: Registered file:SE.path.filename_.txt:session.txt:986:2432916819:1058970986
Message: Out of 1 file(s) 1 are registered.
==>END LOGGING OUTPUT FOR PROCESS-SE_26682_1058971661_1
```

- The report says that everything went the way it was supposed to go, to check if the file really is in the local catalogue, we use the same command as above but with `-c` parameter.

```
[UI] /path $ gdmp_job_status -c local_file_catalogue
SE_path:filename.txt:986:2432916819:1058970986
```

GDMP: How to Copy a File (I)

- Once we have already a registered file, we are ready to copy it from a CE to another.
- Now two other two catalogues enter the game: the **import catalogue** and the **export catalogue**.
- The **export catalogue** contains the files that a site wants to publish for the Grid to use.
- The **import catalogue** contains all the files that are published by the producer but not yet transferred to the consumer site.
- A host can subscribe to any other host in the Grid in order to be notified when new files are published in the remote host.
- Once two hosts are subscribed to one another, whenever a file is published, it can be copied (replicated) with `gdmp_replicate_get` and `gdmp_replicate_put`
- We will see it all in detail

GDMP: How to Copy a File

- We now have four steps to follow to copy a file:

- First of all, it is necessary to register the receiving node in the sending node so the moment that the latter one publishes the files, the first one is able to receive it.
- Next step to take is to properly publish the above-mentioned file, for both catalogues (export and import) to be the way we need.
- A third optional step would be checking whether both catalogues contain what they are supposed to.
- The fourth and most important step can now be carried out: copying the file.

GDMP: How to Copy a File

- We now have four steps to follow to copy a file:

- **First of all, it is necessary to register the receiving node in the sending node so the moment that the latter one publishes the files, the first one is able to receive it.**
- Next step to take is to properly publish the above-mentioned file, for both catalogues (export and import) to be the way we need.
- A third optional step would be checking whether both catalogues contain what they are supposed to.
- The fourth and most important step can now be carried out: copying the file.

GDMP: How to Copy a File (I)

- We assume that we have already registered our file: *filename.txt*, in one SE (A) and we want to copy it into another (B). First, we subscribe SE-B to SE-A.

```
[SE-B] /path $ gdmf_host_subscribe -r SE-A.desy.de
```

- The next step would be publishing the local file catalogue in SE-A, so their files go to its export catalogue and the import catalogue in SE-B, as it is subscribed.

```
[SE-A] /path $ gdmf_publish_catalogue
```

GDMP: How to Copy a File

- We now have four steps to follow to copy a file:

- First of all, it is necessary to register the receiving node in the sending node so the moment that the latter one publishes the files, the first one is able to receive it.
- **Next step to take is to properly publish the above-mentioned file, for both catalogues (export and import) to be the way we need.**
- A third optional step would be checking whether both catalogues contain what they are supposed to.
- The fourth and most important step can now be carried out: copying the file.

GDMP: How to Copy a File (I)

- We assume that we have already registered our file: *filename.txt*, in one SE (A) and we want to copy it into another (B). First, we subscribe SE-B to SE-A.

```
[SE-B] /path $ gdmf_host_subscribe -r SE-A.desy.de
```

- The next step would be publishing the local file catalogue in SE-A, so their files go to its export catalogue and the import catalogue in SE-B, as it is subscribed.

```
[SE-A] /path $ gdmf_publish_catalogue
```

GDMP: How to Copy a File

- We now have four steps to follow to copy a file:

- First of all, it is necessary to register the receiving node in the sending node so the moment that the latter one publishes the files, the first one is able to receive it.
- Next step to take is to properly publish the above-mentioned file, for both catalogues (export and import) to be the way we need.
- **A third optional step would be checking whether both catalogues contain what they are supposed to.**
- The fourth and most important step can now be carried out: copying the file.

GDMP: How to Copy a File (II)

- We can check if both catalogues we need are all right using the already known command `gmdp_job_status` with the `-c` parameter and either `import_catalogue` or `export_catalogue`.

```
[SE_A] /path $ gmdp_job_status -c import_catalogue  
SE_A.path:filename.txt +some producer information
```

```
[SE_B] /path $ gmdp_job_status -c export_catalogue  
SE_B.path:filename.txt
```

GDMP: How to Copy a File

- We now have four steps to follow to copy a file:

- First of all, it is necessary to register the receiving node in the sending node so the moment that the latter one publishes the files, the first one is able to receive it.
- Next step to take is to properly publish the above-mentioned file, for both catalogues (export and import) to be the way we need.
- A third optional step would be checking whether both catalogues contain what they are supposed to.
- **The fourth and most important step can now be carried out: copying the file.**

GDMP: How to Copy a File (III)

- Now, having the file in each catalogue we can replicate it from the receiving SE using `gmdp_replicate_get` or...

```
[SE-B] /path $ gmdp_replicate_get
```

- ...we can also copy it from the sending SE using another complementary command, than even uses internally the one above: `gmdp_replicate_put`.

```
[SE-A] /path $ gmdp_replicate_put
```

GDMP: How to Remove a File

- Removing a file is much easier than copying it
- We use the command `gmdp_remove_local_file...`
- which not only removes the file from disk,
- but also from all possible catalogues.

```
[SE-B] /path $ gmdp_remove_local_file -p /path/filename.txt  
Message: Deleting /path/filename.txt of type file.  
Message: /path/filename.txt deleted.
```

EDG-REPLICA-MANAGER-2.0

- **EDG-REPLICA-MANAGER** is a piece of software responsible for replicating files and updating the Globus Replica Catalogue.
 - It uses the GDMP commands we have already dwelled into before
 - Therefore, EDG-RM operates in a layer that is placed on top of the GDMP layer.
- All of the three actions described before (putting a file in the Grid, replicating it or removing it) can be also carried about by using EDG-RM.
- We will see it in detail in the following slides at the same time we try to show the relationship between EDG-RM and GDMP.
- EDG-RM is implemented with two different interfaces.
 - The Core-API interface (coded in C++)
 - The command-line interface (the one we concentrate in, in this review).

GDMP vs. EDG-RM

GDMP

- Replication between SE's only
- Replicates set of files
- Provides MSS interface
- Client-server
- Logical file attributes
- Subscription model
- Event notification
- CRC file size check
- Support for Objectivity/DB
- Automatic retries
- Support for multiple VOs

EDG-RM

- Replication between SEs, UI* or CE to SE
- Replicates single files
- Uses GDMP's MSS interface
- Client side only

NOTE: The documentation say so but actually is not possible unless it runs grid ftp.

EDG-RM: How to Put a File

- There are two ways of putting a file in the Grid
 - By one way, the file can be put in two steps the way we did with GDMP:
 - We first physically copy the file into the place we want
 - and then we register it right after.
 - Nonetheless, EDG-RM brings a more powerful command that can perform both tasks together. We will now see each one of them in detail.

EDG-RM: How to Put a File

- There are two ways of putting a file in the Grid
 - **By one way, the file can be put it two steps the way we did with GDMP:**
 - **We first physically copy the file into the place we want**
 - **and then we register it right after.**
 - Nonetheless, EDG-RM brings a more powerful command that can perform both tasks together. We will now see each one of them in detail.

EDG-RM: How to Put a File (I)

- To copy the file we want to put, we use the *edg-replica-manager-copyFile* command.
- Two parameters: source and destination.
- Both hosts must run *gsiftp* server, otherwise, the lower *globus_url_copy* should be used.

```
[UI] /data/jnowak $ edg-rm-cp -s host/path/source -d host/path/destination
configuration file: /path/fileconf.conf
source:            host/path/source
destination:      host/path/destination
protocol:          gsiftp
The program was successfully executed.
```

EDG-RM: How to Put a File (II)

- To register the file, we use *edg-replica-manager-registerEntry* indicating the logical filename and, of course, the source file to register.

```
[UI] /data/jnowak $ edg-rm-reg -l filename -s host/path/filename
configuration file: /path/fileconf.conf
logical file name: filename
source:            host/path/filename
protocol:          gsiftp
SASL/GSI-GSSAPI authentication started
SASL SSF: 56
SASL installing layers
The program was successfully executed.
```

EDG-RM: How to Put a File

- There are two ways of putting a file in the Grid
- By one way, the file can be put it two steps the way we did with GDMP:
 - We first physically copy the file into the place we want
 - and then we register it right after.
- **Nonetheless, EDG-RM brings a more powerful command that can perform both tasks together. We will now see each one of them in detail.**

EDG-RM: How to Put a File (III)

- We can copy and register the file using the *edg-replica-manager-copyAndRegisterFile* command.
- It makes sense that we pass the same parameters.

```
[UI] /data/jnowak $ edg-rm-creg -l filename -s host/path/filename -d host/path/filename
configuration file: /path/fileconf.conf
logical file name: filename
source:            host/path/filename
destination:      host/path/filename
protocol:          gsiftp
VO:
SASL/GSI-GSSAPI authentication started
SASL SSF: 56
SASL installing layers
The program was successfully executed.
```

EDG-RM: How to Replicate a File

- To replicate one file from one SE to another we use `edg-replica-manager-replicateFile`.
- It is really similar to the one before but there are two main differences.
 - Both hosts have to use `gsiftp`, and be SE's as well.
 - Besides, this command deletes the copy already made if the `Replica-Catalog-Update` process fails. The `copyAndRegister` command does not do so.
- Apart from that, syntax is exactly the same as in `copyAndRegister` and
- The answer echoed by the command is also exactly alike.

EDG-RM: How to Delete a File

- EDG-RM deleting is a really easy thing to do: we have to use the `deleteFile` command indicating:
 - the physical file name of the file we want to erase
 - as well as its logical file name.
- It will automatically remove an entry from the Replica Catalog and the file.
- If we want all replicas to be deleted we just type `-a` as a parameter.

```
[grid003] /path/ $ edg-rm-d -l filename -s host/path/filename SASL installing layers
configuration file: /path/fileconf.conf SASL/GSI-GSSAPI authentication started
logical file name: filename SASL SSF: 56
source: host/path/filename SASL installing layers
protocol: gsiftp SASL/GSI-GSSAPI authentication started
SASL/GSI-GSSAPI authentication started SASL SSF: 56
SASL installing layers
The program was successfully executed.
```

Summary

- We have now an overview on how the file management in the grid is about from the lower GDMP side and the upper EDG-RM side
- EDG-RM is much
 - Safer
 - More powerful (API core)
 - Simple (higher level)
- Disadvantage:
 - Some operations cannot be done from EGM-RM
 - ought to be used from GMDP, i.e., transferring files without `gsiftp` protocol.

Bibliography

- DataGrid: EDG User's Guide
- EDG-REPLICA-MANAGER-2.0: Installation and User's Guide
- Grid Data Mirroring Package: User Guide for GDMP 2.0

Acknowledgments

- Dr. Andreas Gellrich
- Mr. Jacek Nowak