



Enabling Grids for  
E-science in Europe

[www.eu-egee.org](http://www.eu-egee.org)

*Introduction à EGEE, 22-23/03/2005  
Clermont-Ferrand*

# Gestion des données

**David Bouvet**  
CCIN2P3

Présentation faite à partir des slides de Jeff Templon



**IN2P3**

INSTITUT NATIONAL DE PHYSIQUE NUCLÉAIRE  
ET DE PHYSIQUE DES PARTICULES

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

# Table des matières

- Problématique
- Introduction aux outils de Data Management
- Divers scénarios de gestion des données sur grille
- RLS



# Problématique : Comment connecter Utilisateur ↔ Programmes ↔ Données ?

- Utilisateur
  - connecté à une UI, ou
  - connecté à sa machine de bureau
- Programmes
  - sur sa machine
  - sur l'UI
  - sur des machines de la grille... quelque part...
- Données
  - Peut avoir besoin de fournir des données (grille ou non grille) aux programmes
  - Programme peut générer des données, avoir besoin de les sauvegarder
  - Comment récupère-t-on ces données ?

# Outils de gestion des données de grille

- Edg-replica-manager (RM) est le premier outil
  - Replica Location Service (RLS) garde la trace de la localisation des diverses copies des fichiers grille
  - Transfert de données utilise surtout *gsiftp*
    - Comme le bon vieux FTP, utilise l'authentification grille
    - Pas de mot de passe !
    - Peut aussi utiliser des flots multiples pour un transfert plus rapide
  - RM couvre l'interaction entre *gsiftp* et le RLS pour faciliter l'instanciation, l'enregistrement et la réplication des données grille
- Resource Broker
  - peut envoyer (un petit ensemble) de données aux/depuis les jobs
  - peut utiliser le RLS pour trouver vos données et les envoyer à votre job, si vos données sont dans le RLS et que vous le spécifiez au RB

# Commandes de base du RM (I)

- Mettre des données sur la grille
  - Mettre le fichier `/home/myhome/toto` (de l'ordinateur local) sur le *storage element* `leSEchoisi01.in2p3.fr` et l'enregistrer avec le *nom de fichier logique* `edgrm.test`
  - `lcg-cr -d leSEchoisi.in2p3.fr -l lfn:edgrm.test \`  
`--vo gilda file:/home/myhome/toto`
- **Storage Element** – machine accessible par la grille pour le stockage de données
- **Logical File Name** – nom de fichier symbolique avec lequel vous pouvez faire référence à un fichier grille sans préciser sa localisation actuelle
- La commande ci-dessus retourne le “GUID” :
  - `guid:76373236-b4c7-11d8-bb5e-eba42b5000d0`
- Les GUID sont permanents, les LFN non !

# Commandes de base du RM (II)

- Trouvez vos données : la méthode listReplicas (lr)
  - `lcg-lr --vo gilda lfn:edgrm.test # via LFN`
    - `sfn://leSEchoisi.in2p3.fr/grid/gilda/tutorial/data/generated/2004-11-10/file7115df45-b4c7-11d8-bb5e-eba42b5000d0`
  - `lcg-lr --vo gilda \ # via GUID`  
`guid:76373236-b4c7-11d8-bb5e-eba42b5000d0`
    - `sfn://leSEchoisi.in2p3.fr/grid/gilda/tutorial/data/generated/2004-11-10/file7115df45-b4c7-11d8-bb5e-eba42b5000d0`
- “replicas” car quelqu’un (ou un programme) peut faire une copie sur un SE différent – les LFN et GUID font référence à toutes les copies

# Commandes de base du RM (III)

- Trouver l'information concernant le RLS ou le "DMS" (Data Management System)
  - Comment connaît-on que `leSEchoisi.in2p3.fr` était un SE ?

- `lcg-infosites --vo atlas se`

- ```
*****
***
These are the related data for atlas: (in terms of SE)
*****
***
Avail Space (Kb)  Used Space (Kb)          SEs
-----
725228544        5741568          clrldgse01.in2p3.fr
190257628        38750756         clrldgse02.in2p3.fr
```

- Emplacements de toutes les ressources

# Fonctions standards de gestion des données grille

- Traiter les données générées par votre job
  - Récupérer les données sur votre machine de bureau
  - Mettre les données sur la grille
- Récupérer des données pour votre job
  - Soumettre des données avec votre job
  - Mettre vos données sur la grille (depuis l'extérieur)
  - Adresser vos données grille à votre job
- Déplacer des données sur la grille
- Comment trouver vos données si vous ne vous rappelez plus où vous les avez mises ?



# Grille → Données sur votre machine de bureau

- Vous pouvez définir votre job pour récupérer les données
  - Le job génère des données dans le répertoire de travail du WN
  - A la fin du job, les données sont placées dans l'espace de stockage temporaire du RB
  - Vous les récupérez via "edg-job-get-output"
- **Éléments clé :**
  - Vous devez connaître les noms des fichiers que vous souhaitez récupérer
  - **OutputSandbox = {"higgs.root", "graviton.HDF"};**
  - Méthode non destinée pour de gros fichiers (> centaine MB)  
→ espace de stockage temporaire limité sur le RB

# Programme grille → données sur grille (I)

- Votre programme génère des données dans un fichier local
- Le programme doit connaître quel est le nom du fichier local
- Le programme utilise les commandes **lcg-\*** pour :
  - Mettre les données sur un SE de la grille
  - Enregistrer les données comme un jeu de données grille
- Deux options utiles :
  - Sur quel SE les données doivent-elles être sauvegardées (ou même dans quel répertoire de quel SE !) ?  
→ Par défaut: SE "local"
  - Un nom de fichier logique.  
→ **Par défaut: pas de LFN !**

# Programme grille → données sur grille (II)

- Rappels :
  - Si vous voulez un SE spécifique, vous pouvez le trouver en utilisant la commande :

```
lcg-infosites --vo <yourvo>
```

- Mettez un fichier dans le système de stockage de la grille (dans le RLS et sur un SE) en utilisant :

```
lcg-cr --vo <yourvo>
```

(cf. transparent 5)

# Prog. grille → données sur grille (III)

## Laissez faire le WMS

- L'attribut `OutputData` du JDL spécifie où les fichiers doivent aller
  - Si aucun LFN n'est précisé, WMS en choisit un
  - Si aucun SE n'est précisé, le SE le plus proche est choisi
- A la fin du job, les fichiers sont effacés du WN et enregistrés
- Un fichier contenant le résultat de cette opération est créé et ajouté à la sandbox : `DSUpload_<unique jobstring>.out`
- ```
OutputData = { [  
    OutputFile = "toto.out" ;  
    StorageElement = "adc0021.cern.ch" ;  
    LogicalFileName = "lfn:theBestTotoEver" ; ],  
  [  
    OutputFile = "toto2.out" ;  
    StorageElement = "adc0021.cern.ch" ;  
    LogicalFileName = "lfn:theBestTotoEver2" ; ]  
};
```

# Soumission de données par le job

- C'est facile : utiliser le champ **InputSandbox**
- Attention : on ne peut pas utiliser ce champ pour de trop grand transfert (plusieurs megabytes)
- `InputSandbox = {"input-ntuple.root", "job.sh"};`

# Mettre des données sur la grille depuis l'extérieur

- Mettre des données sur la grille (rappel transparent 5)
  - Mettre le fichier `/home/myhome/toto` (de l'ordinateur local) sur le *storage element* `leSEchoisi01.in2p3.fr` et enregistrer-le avec le *nom de fichier logique* `edgrm.test`
  - `lcg-cr -d leSEchoisi.in2p3.fr -l lfn:edgrm.test \`  
`--vo gilda file:/home/myhome/toto`
- Storage Element – machine accessible par la grille pour le stockage de données
- Logical File Name – nom de fichier symbolique avec lequel vous pouvez faire référence à un fichier grille sans préciser sa localisation actuelle
- La commande ci-dessus retourne le “GUID” :
  - `guid:76373236-b4c7-11d8-bb5e-eba42b5000d0`
- Les GUID sont permanents, les LFN non !

# Adresser vos données grille à votre job

- Il faut des données sur la grille == enregistrées dans le RLS
- Spécifiez les données grille au job (JDL) :
  - **InputData = "lfn:myfile.dat"**
- *(Le Resource Broker met des infos concernant la correspondance avec les données dans le fichier .BrokerInfo sur le noeud d'exécution distant)*
- Dans le script d'exécution de votre job, utilisez la commande *edg-brokerinfo* & les commandes *lcg-\** pour avoir une copie locale au job

# Dupliquer les données

- Vous pouvez dupliquer vos données sur d'autres SE à partir du LFN du fichier
  - `lcg-rep --vo gilda lfn:edgrm.test -d \`  
`autreSEchoisi.in2p3.fr`
- Si vous listez ces données avec la méthode `listReplicas`, vous trouvez une nouvelle liste :
  - `sfn://leSEchoisi.in2p3.fr/grid/gilda/tutorial/data`  
`/generated/2004-11-10/file7115df45-b4c7-11d8-bb5e-`  
`eba42b5000d0`  
`sfn://autreSEchoisi.in2p3.fr/grid/gilda/tutorial/d`  
`ata/generated/2004-11-10/file3498dg93-h3b6-38s2-`  
`kf7d-ksr38h8379q7`



# RLS : Replica Location Service

- Le RLS a deux composants
  - Local Replica Catalog (LRC)
    - maintient le catalogue des GUID:(fichiers physiques)
    - Attention : les noms des fichiers “physiques” peuvent avoir besoin d’un traitement supplémentaire (voir la documentation de la méthode “edg-rm getTurl”)
  - Replica Metadata Catalog (RMC)
    - maintient le catalogue des LFN:GUID
    - peut aussi maintenir les méta-données sur les LFN
- lcg-\* interagit avec les deux

# Autres commandes

- edg-rm (cr, lr, rep...)
  - anciennes commandes lcg-\*
- Outils bas niveau pour copier les données distribuées et obtenir des infos
  - globus-url-copy → copie sécurisée sans enregistrement RLS
  - edg-gridftp-ls et commandes associées
- Interaction avec les composants RLS
  - edg-lrc (local replica catalog)
  - edg-rmc (replica metadata catalog, search on metadata)