

# Gestion des Données

*David Bouvet, David Weissenbach*

*CCIN2P3 16/03/10*

- **Les principaux systèmes**
- **Les commandes de base**
  - Mettre des données sur la grille
  - Retrouver ces données
  - Trouver l'information
  - Dupliquer les données
  - Effacer les données
- **La gestion des données au cours d'un job**
  - Données sur votre bureau
  - Données sur la grille
- **Le catalogue LFC**



- **Les principaux systèmes mis en jeu**
- Les commandes de base
- La gestion des données pendant un job sur la grille
- Le catalogue LFC

- **Gestion de l'espace de noms logiques : le LCG File Catalog (LFC)**
  - garde la trace de la localisation physique des diverses copies d'un fichier sur la grille
  - fournit le GUID (Global Unique Identifier) du fichier, et aussi le SURL (Storage URL)
  - présente une arborescence et des commandes façon « système de fichiers »
  - c'est l'interface avec l'utilisateur
- **Gestion de l'espace de noms physiques : le Storage Resource Manager (SRM)**
  - utilise le GUID (ou le SURL) et nous fournit le TURL (Transport URL)
  - c'est l'interface avec la DB et les systèmes de stockage
    - disques, bandes, MSS, ...
- **Les protocoles de transfert de données : rfiio, gsiftp, xroot**
  - utilisent le TURL

- Les principaux systèmes mis en jeu
- **Les commandes de base**
- La gestion des données pendant un job sur la grille
- Le catalogue LFC

- Copier le fichier `/home/myhome/toto` (ordinateur local) sur le *Storage Element* `leSEchoisi.in2p3.fr` et l'enregistrer avec le *nom de fichier logique* `macopie.test`
    - `lcg-cr -d leSEchoisi.in2p3.fr \ -`  
`l lfn:/grid/gilda/myname/macopie.test \`  
`--vo gilda file:/home/myhome/toto`
  - La commande ci-dessus retourne le “GUID” :
    - `guid:76373236-b4c7-11d8-bb5e-eba42b5000d0`
    - les GUID sont permanents, les LFN non !
  - **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 physique
- Remarque : les LFNs peuvent comporter des dossiers et sous-dossiers et l'utilisateur est libre de les organiser à sa guise
- Il y a plein d'autres options disponibles: '`lcg-cr -help`'

- Localisez vos données en utilisant le LFN :
  - `lcg-lr --vo gilda \`  
`lfn:/grid/gilda/macopie.test`
- Ou bien le GUID :
  - `lcg-lr --vo gilda \`  
`guid:76373236-b4c7-11d8-bb5e-eba42b5000d0`
- Ces commandes retournent le SURL :
  - `sfn://leSEchoisi.in2p3.fr/grid/gilda/tutorial/data/generate`  
`d/2004-11-10/file7115df45-b4c7-11d8-bb5e-eba42b5000d0`
- `lcg-lr (listReplicas)` : “replicas” car on peut avoir plusieurs copies sur des SE différents – les LFN et GUID font référence à toutes les copies
- Exercice inverse : obtenir une copie locale de vos données
  - `lcg-cp --vo gilda \`  
`guid:76373236-b4c7-11d8-bb5e-eba42b5000d0 file:`pwd`/newcopy`

- Comment trouver les SEs disponibles localement ?

- `lcg-infosites --vo atlas se`

| Avail Space(Kb) | Used Space(Kb) | Type | SEs                  |
|-----------------|----------------|------|----------------------|
| 845990          | 1              | n.a  | sa3-se.egee.cesga.es |
| 102800000       | n.a            | n.a  | se201.grid.ucy.ac.cy |
| n.a             | n.a            | n.a  | dublin.desy.de       |
| n.a             | 22095          | n.a  | ctb04.gridctb.uoa.gr |
| 6200000         | 5642131        | n.a  | lxb1921.cern.ch      |

- Beaucoup d'autres informations peuvent être affichées

- le nom des serveurs LFC : utiliser l'argument *lfc* (au lieu de *se*)
  - toutes les ressources disponibles : argument *all*
  - autres arguments possibles : *ce*, *closeSE*, *wms*
  - attention : l'option '`--vo`' est **toujours** vitale
    - un SE peut être valide pour une VO et pas pour une autre



- **Sur un UI, si votre VO est 'gilda', le SE par défaut est :**
  - `$VO_GILDA_DEFAULT_SE`
- **Sur un WN, c'est le *closeSE***
  - utiliser la commande `lcg-infosite --vo gilda closeSE`
- **Dans les 2 cas, si l'on ne désire pas sélectionner à l'avance le SE (dans la commande `lcg-cr`, par ex.), on omet l'option `-d`, et le SE par défaut sera retenu**
  - par ex., dans un job, on ne sait pas sur quel site il sera exécuté, donc on ne sait pas quel SE il faudra spécifier
- **On pourrait ensuite retrouver ce SE à l'aide de `lcg-lr`**
  - ce n'est pas vraiment nécessaire, seul le GUID (ou le LFN) est important et devra être spécifié
    - par ex. dans une opération de copie, de relecture, ou d'effacement

- Dupliquer vos données vers un autre SE à partir du LFN (ou GUID ou SURL) du fichier
  - `lcg-rep --vo gilda \`  
`lfn:/grid/gilda/myname/macopie.test \`  
`-d lautreSE.cern.ch`
- Si vous listez maintenant ce LFN avec `lcg-lr`, vous obtiendrez cette liste :
  - `sfn://leSEchoisi.in2p3.fr/grid/gilda/tutorial/data/generated/2004-11-10/file7115df45-b4c7-11d8-bb5e-eba42b5000d0`
  - `srm://lautreSE.cern.ch/pnfs/cern.ch/data/gilda/generated/2005-11-28/file46034814-3021-47e0-a382-60cde5efbdf0`

- Pour effacer un replica sur un SE :
  - `lcg-del --vo gilda -s lautreSE.cern.ch \`  
`lfn:/grid/gilda/myname/macopie.test`
- Il faut spécifier le SE car s'il y a plusieurs replicas, la commande ne sait pas quel fichier effacer.
- Pour effacer tous les *replicas* d'un même fichier, il faut rajouter **l'option -a**
- Le catalogue LFC est mis à jour automatiquement

- Les principaux systèmes mis en jeu
- Les commandes de base
- **La gestion des données pendant un job sur la grille**
- Le catalogue LFC

- **Disposer des données produites dans le job (output)**
  - récupérer simplement les données sur votre machine de bureau
  - rendre ces données disponibles/accessibles sur la grille
- **Lire des données locales dans le job (input)**
  - spécifier des données au moment de la soumission du job
  - disposer vos données sur la grille avant un job
  - accéder à vos données grille depuis l'intérieur du job
- **Déplacer des données d'un point à un autre sur la grille**
- **Comment retrouver vos données après coup ?**

- Le job génère des données dans le répertoire de travail du WN, et vous souhaitez simplement les récupérer sur votre machine locale (UI)
- A la fin du job, les fichiers nouveaux sont placés dans un espace de stockage temporaire (sur le WMS)
- Vous les récupérerez via “glite-wms-job-output”
- **Éléments clés :**
  - vous devrez spécifier les noms des fichiers que vous souhaitez récupérer dans le JDL du job par la clause :
    - **OutputSandbox = {"higgs.root", "graviton.HDF"};**
  - méthode fortement déconseillée pour de gros fichiers (> 100MB)  
→ espace de stockage temporaire limité sur le WMS

- **Pour vos données « locales » :**
  - c'est facile : utiliser le champ *InputSandbox* du JDL
    - **InputSandbox = {"input-ntuple.root", "job.sh"};**
  - attention : on ne peut pas utiliser ce champ pour de trop gros transferts (qqes MB seulement)
- **Pour vos données « grille » :**
  - spécifiez les données grille à l'aide du champ *InputData* :
    - **InputData = {"lfn:/grid/gilda/myfile.dat",  
"guid:f62344d9-ca25-458a-adf7-9d8150031dcd"};**

- Le WMS place les infos concernant la correspondance avec les fichiers réels dans le fichier *.BrokerInfo* sur le nœud d'exécution distant (WN)
- Dans le script d'exécution de votre job, utilisez la commande *glite-brokerinfo* & les commandes *lcg-\** pour obtenir une copie proche des fichiers **uniquement si c'est nécessaire**. En effet :
  - si elles sont déjà accessibles localement, il est inutile de faire une copie supplémentaire
  - si elles ne le sont pas, le CE n'a pas pu être sélectionné par le RB pour exécuter votre job
- Il faudra toutefois faire une copie locale sur les disques de votre WN (*lcg-cp*), dans votre zone de travail
  - sauf si vous utilisez GFAL (voir plus loin)



- Les principaux systèmes mis en jeu
- Les commandes de base
- La gestion des données pendant un job sur la grille
- **Le catalogue LFC**

- **Le catalogue LFC (LCG File Catalog) présente une arborescence et des commandes façon POSIX :**
  - il est possible de lister les références, de créer une nouvelle arborescence, de faire des liens symboliques...
  
- **Par défaut, le catalogue utilisé est le catalogue central de la VO.**
  - **`lcg-infosites -vo gilda lfc`**  
retourne le nom du LFC central
  - **`lcg-infosites -vo gilda lfcLocal`**  
retourne la liste des catalogues locaux de la VO
  
- **Pour utiliser un catalogue local :**
  - `export LFC_HOST=<hostname du LFC choisi>`

- Elles permettent la gestion des LFNs
- Racine de l'arborescence d'un LFC :
  - /grid/<nom de la VO>
  - on peut stocker cette racine dans la variable LFC\_HOME
    - export LFC\_HOME=/grid/gilda
  - si on définit cette variable, l'option '**-l lfn:mylfn**' est en fait interprétée comme '**-l lfn:\$LFC\_HOME/mylfn**'
- Lister les références :
  - lfc-ls /grid/gilda/...
  - lfc-ls \$LFC\_HOME/...
- Créer un nouveau répertoire :
  - lfc-mkdir /grid/gilda/.../<mon rep>
- Autres ex. de commandes :
  - lfc-ln, lfc-rm, lfc-rename, lfc-getacl, lfc-setacl
  - lfc-chmod, lfc-chown, lfc-setcomment, lfc-delcomment
- Aide en ligne : commande '**man lfc-xx**' ou simplement '**lfc-xx**'

- **Jusqu'ici, les fichiers accédés par le job sont toujours locaux**
  - en lecture comme en écriture
  - c'est un handicap quand on veut lire/écrire de gros fichiers
    - il faut les transférer d'abord
    - surtout s'ils sont plus gros que l'espace disque disponible sur le WN
- **Pour accéder directement à des fichiers distants : GFAL ou RFIO**
  - GFAL : pour 'Grid File Access Library'
    - cette librairie est une API C, à la norme POSIX
    - elle permet, directement depuis le programme en C/C++, d'accéder aux fichiers disponibles sur un SE (même lointain)
  - RFIO : pour 'Remote File Input Output'
    - même fonctionnalité que GFAL
  - la syntaxe des noms de fichiers est la syntaxe SURL
  - moyen économique, il évite des transferts inutiles ou volumineux

- **Projet LCG**
  - <http://lcg.web.cern.ch/LCG/>
- **Projet EGEE (tutorial en ligne)**
  - <http://www.eu-egee.org/try-the-grid>
- **LCG user**
  - <http://lcg.web.cern.ch/LCG/users/users.html>
- **LCG User Guide**
  - <https://edms.cern.ch/file/454439//LCG-2-UserGuide.html>
- **LCG FAQ (pas à jour mais ça aide quand même)**
  - <https://edms.cern.ch/file/495216/1/LCG-Faq.html>
- **Doc diverses sur la grille**
  - <https://gus.fzk.de/pages/docu.php>
- **User Support**
  - <http://gus.fzk.de>  
pour soumettre un 'bug report' (incidents sur la VO, pour un site ...)



- **> edg-brokerinfo --help**
- edg-brokerinfo getCE
- edg-brokerinfo getDataAccessProtocol
- edg-brokerinfo getInputData
- edg-brokerinfo getSEs
- edg-brokerinfo getCloseSEs
- edg-brokerinfo getSEMOUNTPOINT <SE>
- edg-brokerinfo getSEFreeSpace <SE>
- edg-brokerinfo getLFN2SFN <LFN>
- edg-brokerinfo getSEProtocols <SE>
- edg-brokerinfo getSEPort <SE> <Protocol>
- edg-brokerinfo getVirtualOrganization

- **Gestion des réplicas. Pour l'aide en ligne : 'man lcg-xx '**
  - ou plus simplement pour une aide (très) courte : '**lcg-xx** '
- **lcg-cr copyAndRegister**
- **lcg-del deleteFile**
- **lcg-gt getTurl**
- **lcg-rep replicateFile**
- **lcg-lr listReplicas**
- **lcg-cp copyToLocal**
- **lcg-la listAlias**
- **lcg-lg listGuid**
- **lcg-aa addAlias**
- **lcg-ra removeAlias**
- **lcg-sd setFileStatusToDone**
- **lcg-rf registerFile**
- **lcg-uf unregisterFile**



- **> lcg-infosites --help**
- `lcg-infosites --vo voname [--is BDII] arg(s)`
- **--vo**: VO name (mandatory).
- **--is**: BDII to query (default to `$LCG_GFAL_INFOSYS`)
- **se**: The names of the SEs supporting user's VO
- **ce**: The names of the CEs where user's VO can run
- **closeSE**: The names of the SEs associated with each CE
- **wms**: It publishes the names of the WMSs available for each VO
- **lfc (lfcLocal)**: Name of the LFC (local) servers
- **tag**: The names of the tags relative to the software installed
- **all**: It displays together the informations provided by 'se, ce'

- **Data Transfer Functions**

- srmPrepareToGet
  - *srmStatusOfGetRequest*
- srmPrepareToPut
  - *srmStatusOfPutRequest*
- srmCopy
  - *srmStatusOfCopyRequest*
- srmBringOnline
  - *srmStatusOfBringOnlineRequest*
- srmReleaseFiles
- srmPutDone
- srmAbortRequest
- srmAbortFiles
- srmSuspendRequest
- srmResumeRequest
- srmGetRequestSummary
- srmExtendFileLifeTime
- srmGetRequestTokens

- **Space Management Functions**

- srmReserveSpace
  - *srmStatusOfReserveSpaceRequest*
- srmReleaseSpace
- srmUpdateSpace
- srmGetSpaceMetaData
- srmChangeSpaceForFiles
  - *srmStatusOfChangeSpaceForFilesRequest*
- srmExtendFileLifeTimeInSpace
- srmPurgeFromSpace
- srmGetSpaceTokens

- **Permission Functions**
  - srmSetPermission
  - srmCheckPermission
  - srmGetPermission
- **Directory Functions**
  - srmMkdir
  - srmRmdir
  - srmRm
  - srmLs
    - *srmStatusOfLsRequest*
  - srmMv
- **Discovery Functions**
  - srmGetTransferProtocols
  - srmPing