

The logo for France Grilles features a stylized blue map of France with a white grid pattern overlaid on it. The text "France Grilles" is written in a bold, black, sans-serif font, with "France" in a smaller size than "Grilles". The logo is set against a dark background with a pattern of binary code (0s and 1s) and green lines radiating from the center.

France **Grilles**

# Gestion des données

C. Loomis (CNRS/LAL)

- **Les principaux composants**
- **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

- **Les services grilles gèrent *les fichiers* qui contient des données**
  - « Write once, read mostly » : les fichiers sont invariants sur la grille
  - Plusieurs réplicas : redondance, sauvegarde, accès optimisé
  - Plusieurs protocoles d'accès : gsiftp, http(s), POSIX
- **Autres formats ne sont pas supporter directement**
  - Base de données
  - Données stockées hors grille
  - Services qui fournir des données



# Les Principaux Composants

- **LFC : gestion de l'espace de noms logiques**
  - Garde la trace de la localisation physique des diverses répliques d'un fichier sur la grille
  - Chaque fichier a un GUID (Global Unique Identifier)
  - Chaque réplique est désigné par un SURL (Storage URL)
  - Arborescence et des commandes façon « système de fichiers »
- **SRM : gestion de l'espace de noms physiques**
  - Contient une base des données pour les SURLs
  - Systèmes de stockage (disques, bands, MSS, ...)
  - Services qui supportent les protocoles de transfert
  - Utilise le GUID (ou le SURL) et fournit le TURL (Transport URL)
- **Les protocoles de transfert de données**
  - Accès à la distance : rfiio, gsiftp, http(s)
  - Accès local (dans un site) : POSIX

- **Toutes les commandes liées à la gestion de données supposent d'avoir un proxy valide**
  - Détermination de la VO utilisée par l'utilisateur
  - Détermination des droits d'accès

# Commandes lcg\_utils...

- **Ensemble de commandes fournissant des fonctionnalités complètes**
  - Interaction LFC et SE/SRM
  - Documentation : ‘man lcg-\*’ ou ‘lcg-\*’
- **Commandes de transferts**
  - lcg-cp : copie simple, y compris de/à disque locale
  - lcg-cr : copie sur la grille; enregistrement dans LFC
  - lcg-del : supprimer un fichier
  - lcg-rep : copie (réplica) fichier, mise à jour LFC

## ➤ Informations sur un fichier ou replica

- lcg-gt get TURL
- lcg-lr list replicas
- lcg-la list alias
- lcg-lg list GUID

## ➤ Opérations spécifiques : à ne pas utiliser normalement...

- lcg-aa add alias
- lcg-ra remove alias
- lcg-sd set FileStatus to done (marquer le transfert terminé)
- lcg-rf register file in LFC
- lcg-uf unregister file from LFC

# Transférer sur la Grille...

- **Tous les données sur la grille existent dans un « Storage Element » : service avec interface SRM accessible par la grille pour le stockage de données**
  
- **Copier un fichier entre l'ordinateur local et la grille:**
  - `lcg-cr -d se.example.org \  
-l lfn:/grid/vo.u-psud.fr/myname/macopie.test \  
/home/myhome/toto`
  
- **Réponse :**
  - `guid:76373236-b4c7-11d8-bb5e-eba42b5000d0`



# ... Transférer sur la Grille

## ➤ GUID :

- Identifier unique et permanente

## ➤ Logical File Name (LFN)

- Nom d'un fichier sur la grille est indépendant de sa localisation physique
- Comporter des dossiers et sous-dossiers et l'utilisateur est libre de les organiser à sa guise
- Les dossiers non existants sont créés



# Récupérer les Données

- **Localisez les données en utilisant le LFN ou le GUID :**
  - `lcg-lr lfn:/grid/vo.u-psud.fr/macopie.test`
  - `lcg-lr guid:76373236-b4c7-11d8-bb5e-eba42b5000d0`
  
- **Ces commandes retournent le(s) SURL(s) : nom des réplicas**
  - 1 seul réplica par SE
  - `srm://se.example.com/grid/vo.u-psud.fr/tutorial/data/generated/2004-11-10/file7115df45-b4c7-11d8-bb5e-eba42b5000d0`
  
- **Obtenir une copie locale des données sur la grille : lcg-cp**
  - `lcg-cp guid:76373236-b4c7-11d8-bb5e-eba42b5000d0 newcopy`
  - `lcg-cp lfn:/grid/vo.u-psud.fr/macopie.test newcopy`
  - `lcg-cp srm://se.example.com/grid/vo.u-psud.fr/tutorial/data/generated/2004-11-10/file7115df45-b4c7-11d8-bb5e-eba42b5000d0 newcopy`

- **Sur un UI ou un WN, si votre VO est ‘vo.u-psud.fr’, le SE par défaut est désigné par la variable:**
  - \$VO\_VO\_U\_PSUD\_FR\_DEFAULT\_SE
  - C’est le SE sur lequel on écrit par défaut quand aucun n’est spécifié
- **Pour utiliser le SE par défaut, il suffit d’omettre l’option -d (lcg-cr et lcg-rep)**
  - Par ex., dans un job qui doit pouvoir s’exécuter sur plusieurs sites en utilisant le SE “local”
- **Après écriture du fichier, il est possible de retrouver le SE utilisé avec lcg-lr**
  - Rarement nécessaire car on peut utiliser le GUID ou LFN
  - Essentiellement si plusieurs réplicas

# Dupliquer les données

- **Dupliquer des données vers un autre SE à partir du LFN (ou GUID ou SURL) du fichier**
  - `lcg-rep \`  
`lfn:/grid/vo.u-psud.fr/myname/macopie.test \`  
`-d autre-se.example.org`
- **lcg-lr liste la l'ensemble des réplicas :**
  - `srm://leSEchoisi.in2p3.fr/grid/vo.u-psud.fr/tutorial/`  
`data/generated/2004-11-10/file7115df45-b4c7-11d8-`  
`bb5e-eba42b5000d0`
  - `srm://lautreSE.cern.ch/pnfs/cern.ch/data/vo.u-`  
`psud.fr/generated/2005-11-28/`  
`file46034814-3021-47e0-a382-60cde5efbdf0`

## ➤ Pour effacer un seul réplica :

- `lcg-del -s autre-se.example.org \`  
`lfn:/grid/vo.u-psud.fr/myname/macopie.test`
- Il faut spécifier le SE car s'il y a plusieurs réplicas, la commande ne sait pas quel fichier effacer

## ➤ Pour effacer tous les réplicas :

- `lcg-del -a \`  
`lfn:/grid/vo.u-psud.fr/myname/macopie.test`

## ➤ Le catalogue LFC est mis à jour automatiquement

# Job : Données Sortantes

## ➤ Utilisation du « OutputSandbox »

- Les fichiers spécifiés sont stockés dans le WMS
- Récupération via la commande `glite-wms-job-output`
- Méthode fortement déconseillée pour de gros fichiers (> 10Mo)

## ➤ Stocker sur la grille

- `lcg-cr \`
  - `lfn:/grid/vo.u-psud.fr/myname/mesdonnees.txt \`  
`monFichier.txt`
- Méthode préférée pour les gros fichiers

# Job : Données Entrantes

## ➤ Pour les petit fichiers : « InputSandbox »

- InputSandbox = {"input-ntuple.root", "job.sh"};
- Ne pas utiliser pour les gros fichiers (> 1 Mo)

## ➤ Pour les données « grilles » :

- Pour influencer la sélection du site utilisez « InputData »
- Ça ne garantit pas que le fichier est vraiment locale
- L'accès aux données est de la responsabilité du job
- La commande glite-brokerinfo permet d'obtenir la liste des fichiers indiquée dans « InputData »

- **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**
- **Pour utiliser un catalogue local (déconseillé) :**
  - Trouver un : `lcg-infosites -vo vo.u-psud.fr lfcLocal`
  - `export LFC_HOST=<hostname du LFC choisi>`
  - Utiliser les commandes `lfc-*` normalement



- **Elles permettent la gestion des noms logiques (LFN)**
  - Rarement utilisées directement :  
plutôt via les commandes lcg-\*
  - Aucun impact sur les données elle-mêmes (réplicas)
  - Lister les références : lfc-ls /grid/vo.u-psud.fr/...
- **Racine de l'arborescence d'un LFC :**
  - /grid/<nom de la VO>
  - On peut stocker cette racine dans LFC\_HOME
    - export LFC\_HOME=/grid/vo.u-psud.fr
  - Si on définit cette variable
    - '-l lfn:mylfn' → '-l lfn:\$LFC\_HOME/mylfn '

# ... Commandes LFC

- **Créer un nouveau répertoire :**
  - `lfc-mkdir /grid/vo.u-psud.fr/.../<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 : ‘man lfc-\***’ ou ‘**lfc-\***’

- **La copie sur le WN des fichiers complète n'est pas toujours souhaitable, en particulier si on veut lire/écrire de gros fichiers**
  
- **GFAL : 'Grid File Access Library'**
  - Pour accéder directement à des fichiers distants
  - Une API pour C/C++, Python, Perl
  - Les appels possibles sont « POSIX-like » :
    - `gfal_open`, `gfal_read`, `gfal_write`, `gfal_close`, ...
  - La syntaxe des noms de fichiers est la syntaxe SURL
  - Documentation : `'man gfal'`, `'man gfal_*`', ...
  
- **Pour les écritures, si possible, préférer l'écriture locale et un transfert après**



# Commande glite-brokerinfo

- **Récupérer des informations sur un job soumis via un WMS**
  - Ensemble des options possibles : `glite-brokerinfo -help`
- **Principales options**
  - `glite-brokerinfo getCE`
  - `glite-brokerinfo getDataAccessProtocol`
  - `glite-brokerinfo getInputData`
  - `glite-brokerinfo getSEs`
  - `glite-brokerinfo getCloseSEs`
  - `glite-brokerinfo getSEMOUNTPoint <SE>`
  - `glite-brokerinfo getSEFreeSpace <SE>`
  - `glite-brokerinfo getLFN2SFN <LFN>`
  - `glite-brokerinfo getSEProtocols <SE>`
  - `glite-brokerinfo getSEPort <SE> <Protocol>`