



Guide d'utilisation du service FG-iRODS

Version 1.2

Équipe FG-iRODS

11 mai 2021

1	Introduction	1
1.1	À propos de ce document	1
1.2	Le service FG-iRODS	1
2	Installation du client	3
2.1	Prérequis	3
2.2	Environnement	3
2.3	Installation des paquets	4
3	Utilisation du service FG-iRODS	5
3.1	Aide interactive	5
3.2	Répertoire de travail	5
3.3	Chargement des données	6
3.4	Espace de nom et chemin physique	6
3.5	Téléchargement de données	6
3.6	Structuration des données	7
3.7	Gestion des métadonnées	7
3.8	Contrôle d'accès	9

1.1 À propos de ce document

Ce document présente l'utilisation du client **iRODS** avec le **service iRODS (FG-iRODS)** proposé par France Grilles. Il est basé sur le contenu de la documentation en ligne du service FG-iRODS rédigée par Catherine Biscarat, Pierre Gay et Jérôme Pansanel.

La dernière version de ce document est disponible sur : <https://github.com/FranceGrilles/user-docs/tree/main/irods-fr>.

Copyright (c) 2021 CNRS, Université de Bordeaux et Université de Strasbourg.

Ce document est distribué sous la licence [Creative Commons Attribution 4.0 International license](https://creativecommons.org/licenses/by/4.0/).

1.2 Le service FG-iRODS

Le service FG-iRODS, proposé par l'infrastructure de recherche **France Grilles** a pour objectif de faciliter la gestion des données de recherche. Il repose sur :

- une infrastructure de stockage géographiquement distribuée de niveau *production* hautement disponible ;
- l'utilisation du logiciel **iRODS** ;
- un accompagnement personnalisé des utilisateurs ;
- un ensemble d'outils permettant d'en simplifier l'usage.

L'accès à ce service est nominatif et réalisé par une simple demande adressée à info@france-frilles.fr. Après étude de la demande et avant création des comptes, il est demandé aux utilisateurs :

- de signer les conditions d'accès au service FG-iRODS : http://www.france-grilles.fr/IMG/pdf/iRODS_Service_Policy.pdf ;
- de fournir un plan de gestion de données (PGD). Plusieurs exemples de PGD sont disponibles sur le site de DMP OPIDoR <https://dmp.opidor.fr/>.

Une fois les documents reçus, l'accès au service est ouvert, les identifiants et la documentation du service transmis aux utilisateurs.

2.1 Prérequis

L'installation du client iRODS nécessite de disposer d'un poste client fonctionnant avec le système CentOS 7 ou Ubuntu (16.04, 18.04), sur lequel il est possible d'installer de nouveaux logiciels.

2.2 Environnement

Le client en ligne de commande iRODS utilise un fichier pour stocker la configuration pour se connecter à l'infrastructure iRODS. Ce fichier est `~/irods/irods_environment.json` et doit être créé avec le contenu suivant :

```
{
  "irods_host": "ccirodsfg.in2p3.fr",
  "irods_port": 5555,
  "irods_zone_name": "FranceGrillesZone",
  "irods_user_name": "<username>",
  "irods_default_resource": "<default_resource>",
  "irods_client_server_negotiation": "request_server_negotiation",
  "irods_client_server_policy": "CS_NEG_REQUIRE",
  "irods_default_hash_scheme": "SHA256",
  "irods_default_number_of_transfer_threads": 4,
  "irods_encryption_algorithm": "AES-256-CBC",
  "irods_encryption_key_size": 32,
  "irods_encryption_num_hash_rounds": 16,
  "irods_encryption_salt_size": 8,
  "irods_match_hash_policy": "compatible",
  "irods_maximum_size_for_single_buffer_in_megabytes": 32,
  "irods_ssl_verify_server": "cert"
}
```

Les valeurs `<username>` et `<default_resource>` doivent être remplacées par les valeurs qui vous ont été transmises lors de l'enregistrement auprès du service.

2.3 Installation des paquets

Les paquets pour le client iRODS sont disponibles pour les systèmes CentOS 7, Ubuntu 16.04 et Ubuntu 18.04.

Les instructions d'installation pour les systèmes de gestion de paquets *APT* et *YUM* sont détaillées sur la page <https://packages.irods.org>.

Une fois que le dépôt est correctement configuré, il faut installer le paquet `irods-icommands`.

La vérification de l'installation du client est réalisée avec :

```
$ iinit
```

La commande **iinit** permet d'ouvrir une session vers l'instance iRODS. Elle vous demande le mot de passe pour vous connecter. Une fois que vous avez terminé votre transfert de données, vous pouvez terminer la session avec la commande **iexit**.

La commande **ienv** affiche l'environnement iRODS :

```
irods_version - 4.2.8
irods_client_server_negotiation - request_server_negotiation
irods_encryption_key_size - 32
irods_environment_file - /home/<user>/.irods/irods_environment.json
irods_default_hash_scheme - SHA256
irods_default_number_of_transfer_threads - 4
irods_host - ccirodsfg.in2p3.fr
irods_client_server_policy - CS_NEG_REQUIRE
irods_session_environment_file - /home/<user>/.irods/irods_environment.json.15934
irods_default_resource - <default_resource>
irods_encryption_algorithm - AES-256-CBC
irods_encryption_num_hash_rounds - 16
irods_encryption_salt_size - 8
irods_match_hash_policy - compatible
irods_ssl_verify_server - cert
irods_maximum_size_for_single_buffer_in_megabytes - 32
irods_port - 5555
irods_user_name - <username>
irods_zone_name - FranceGrillesZone
```

Utilisation du service FG-iRODS

3.1 Aide interactive

ihelp permet d'afficher la liste des commandes iRODS, ainsi que l'aide sur une commande spécifique :

```
$ ihelp ils
Usage: ils [-ArlLv] dataObj|collection ...
Usage: ils --bundle [-r] dataObj|collection ...
Display data Objects and collections stored in irods.
Options are:
-A ACL (access control list) and inheritance format
-l long format
-L very long format
-r recursive - show subcollections
-t ticket - use a read (or write) ticket to access collection information
-v verbose
-V Very verbose
-h this help
--bundle - list the subfiles in the bundle file (usually stored in the
/myZone/bundle collection) created by iphybun command.

iRODS Version 4.2.8          ils
```

La liste complète des commandes disponibles est également disponible dans la [documentation officielle iRODS](#).

3.2 Répertoire de travail

La commande **ils** affiche le contenu du répertoire courant avec lequel vous travaillez sur le système FG-iRODS (par défaut, il s'agit de votre répertoire utilisateur) :

```
$ ils
/FranceGrillesZone/home/<username>:
```

- *FranceGrillesZone* : le nom de la zone iRODS
- */home/<username>* : votre répertoire personnel

Il est possible de modifier le répertoire sur lequel le client iRODS se connecte en ajoutant la lignes suivante au fichier de configuration iRODS :

```
"irods_cwd": "<chemin_repertoire>",  
"irods_home": "<chemin_repertoire>",
```

Il faut remplacer *<chemin_repertoire>* par le chemin souhaité par défaut.

3.3 Chargement des données

Dans cette section, des fichiers vont être chargés vers FG-iRODS. Le fichier utilisé pour ces exemples est `foo.bin`, il peut être remplacé par un autre fichier de votre choix. Si vous souhaitez travailler avec le fichier `foo.bin`, vous pouvez le créer avec la commande suivante :

```
$ dd if=/dev/urandom of=foo.bin count=65536
```

Le fichier est copié vers l'infrastructure iRODS avec la commande :

```
$ iput -K foo.bin
```

L'option `-K` permet de vérifier le *checksum* et de le stocker dans la base de données. Il est recommandé de l'utiliser systématiquement. Le fichier est maintenant disponible sur FG-iRODS :

```
$ ils  
/FranceGrillesZone/home/<username>:  
  foo.bin
```

Le fichier peut être supprimé avec la commande suivante :

```
$ irm foo.bin
```

3.4 Espace de nom et chemin physique

iRODS fournit une abstraction de l'emplacement physique des fichiers. Par exemple, `/FranceGrillesZone/home/<username>/foo.bin` est le chemin logique utilisé par iRODS. Pour savoir où sont réellement stockées les données, il faut utiliser l'option `-L` avec la commande `ils` :

```
$ ils -L  
/FranceGrillesZone/home/<username>:  
  <username>          0 mcia;mcia-fgirods1      33554432 2020-11-20.09:30 & foo.bin  
    sha2:veVzp+ApMzyVRzZN0BZIkDyFuqUp/4tM4sLVACp00B8=   generic    /vault1/resc/  
↪home/<username>/foo.bin
```

Le résultat de cette commande nous indique que :

- le fichier `foo.bin` est enregistré par FG-iRODS comme : `/FranceGrillesZone/home/<username>/foo.bin`;
- son propriétaire est `<username>`;
- il a été chargé sur la ressource de stockage `mcia`;
- il n'y a qu'un seul réplica, dont l'identifiant est `0`;
- sa taille est de 33554432 octets;
- son *checksum* a été enregistré (`sha :veVzp+ApMzyVRzZN0BZIkDyFuqUp/4tM4sLVACp00B8=`).

3.5 Téléchargement de données

Le fichier stocké dans FG-iRODS peut être téléchargé avec :

```
$ iget -K foo.bin foo-restore.txt
```

Le fichier `foo.bin` a été téléchargé et nommé `foo-restore.txt`. Avec l'option **-K** option, le *checksum* du fichier local est comparé avec le *checksum* du fichier sur FG-iRODS.

3.6 Structuration des données

3.6.1 Création d'une collection

Sur votre ordinateur, les données sont organisées dans des répertoires. Avec iRODS, elles sont organisées de la même manière, sauf que ces dossiers sont appelés des *collections*.

Pour créer une collection iRODS :

```
$ mkdir mycollection
```

Le fichier `foo.bin` peut être déplacé dans la collection `mycollection` avec :

```
$ mv foo.bin mycollection
$ ls -L mycollection
/FranceGrillesZone/home/<username>/mycollection:
  <username>          0 mcia;mcia-fgirods1      33554432 2020-11-20.10:18 & foo.bin
  sha2:veVzp+ApMzyVRzZN0BZIkDyFuqUp/4tM4sLVACp00B8=  generic  /vault1/resc/
↪home/<username>/mycollection/foo.bin
```

Vous pouvez voir que le chemin logique de la collection `/FranceGrillesZone/home/<username>/mycollection` a un répertoire physique : `/vault1/resc/home/<username>/mycollection`. Ainsi, les données n'arrivent pas n'importe où sur un serveur iRODS, mais se placent dans cette structure.

Les données peuvent être chargées directement dans une collection :

```
$ iput -K -r bar.txt mycollection
$ ls /FranceGrillesZone/home/<username>/mycollection
/FranceGrillesZone/home/<username>/mycollection:
  bar.txt
  foo.bin
```

L'option **-r** permet un chargement récursif.

3.6.2 Naviguer à travers les collections

Le répertoire courant de travail correspond à l'emplacement sur lequel vous travaillez dans l'arborescence iRODS. Pour afficher votre répertoire courant sur iRODS, utilisez :

```
$ ipwd
/FranceGrillesZone/home/<username>
```

Si vous ne spécifiez pas le chemin complet, mais uniquement un chemin relatif tel que `mycollection/<file>`, iRODS utilise automatiquement le répertoire courant de travail comme préfixe. Vous pouvez vous déplacer dans l'arborescence et modifier ce répertoire courant de travail avec la commande **icd** :

```
$ icd mycollection
```

3.7 Gestion des métadonnées

iRODS est un logiciel disposant de nombreuses fonctionnalités reposant sur l'utilisation des métadonnées.

3.7.1 Création de métadonnées

Il est possible d'ajouter à chaque fichier une ou plusieurs métadonnées représentées sous forme de triplet *Attribute, Value, Unit* (AVU). Ces triplets sont ajoutés dans la base iCAT d'iRODS et peuvent être recherchés. Les métadonnées sont ajoutées avec la commande :

```
$ imeta add -d foo.bin 'length' '20' 'words'
```

Le champ *Unit* peut être vide :

```
$ imeta add -d foo.bin 'project' 'example'
```

Les métadonnées peuvent également être ajoutées à une collection :

```
$ imeta add -C mycollection 'author' 'John Smith'
```

3.7.2 Affichage des métadonnées

Pour afficher les métadonnées d'un objet de données (fichier), il faut entrer :

```
$ imeta ls -d foo.bin
AVUs defined for dataObj /FranceGrillesZone/home/<username>/mycollection/foo.bin:
attribute: length
value: 20
units: words
```

et pour une collection, la commande suivante :

```
$ imeta ls -C mycollection
AVUs defined for collection /FranceGrillesZone/home/<username>/mycollection:
attribute: author
value: John Smith
units:
```

3.7.3 Recherche avec les métadonnées

La recherche de fichiers ou de collections à l'aide des métadonnées est effectuée avec la commande suivante :

```
$ imeta qu -d 'length' = '20'
collection: /FranceGrillesZone/home/<username>/mycollection
dataObj: foo.bin
```

3.7.4 Recherche avancée

Afin d'effectuer une recherche plus fine de fichiers ou de collections, il est possible d'interroger directement le catalogue iCAT avec la commande **iquest** :

```
$ iquest "select COLL_NAME, META_COLL_ATTR_VALUE where META_COLL_ATTR_NAME like
↔'author'"
COLL_NAME = /FranceGrillesZone/home/<username>/mycollection
META_COLL_ATTR_VALUE = John Smith
-----
```

Les résultats peuvent être filtrés à l'aide d'un ou plusieurs attributs :

```
$ iquest "select COLL_NAME, META_COLL_ATTR_VALUE where META_COLL_ATTR_NAME like
↳ 'author' \
and META_COLL_ATTR_VALUE like 'John%'"
COLL_NAME = /FranceGrillesZone/home/<username>/mycollection
META_COLL_ATTR_VALUE = John Smith
-----
```

NOTE : le caractère “%” est un caractère générique (*wildcard*).

Si vous recherchez un objet de données plutôt qu'une collection, il faut remplacer *META_COLL_ATTR_NAME* par *META_DATA_ATTR_NAME*. De nombreux attributs peuvent être utilisés pour les recherches. Pour les afficher, utilisez :

```
$ iquest attrs
```

3.8 Contrôle d'accès

iRODS propose un mécanisme de droits d'accès similaire au système disponible sur les systèmes UNIX (ACL). Il permet de contrôler les droits de lecture, d'écriture et de propriété. Pour afficher les droits d'accès à la collection actuelle :

```
$ ils -r -A
/FranceGrillesZone/home/<username>/mycollection:
    ACL - <username>#FranceGrillesZone:own
    Inheritance - Disabled
bar.txt
    ACL - <username>#FranceGrillesZone:own
foo.bin
    ACL - <username>#FranceGrillesZone:own
```

Les droits d'accès à un fichier sont spécifiés après le mot-clé *ACL*. Dans cet exemple, *<username>* est propriétaire de tous les fichiers affichés. Aucune autre personne ne peut y accéder.

Les collections ont un attribut *Inheritance*. Lorsque la valeur de cet attribut est égale à *Enabled*, l'ensemble du contenu de la collection hérite des droits d'accès de la collection. Cet héritage ne s'applique qu'aux nouveaux fichiers copiés dans la collection.

La modification des droits d'accès pour autoriser un collègue à accéder à ses données se fait avec :

```
$ ichmod read <colleague> foo.bin
```

L'utilisateur *<colleague>* peut maintenant accéder en lecture au fichier *foo.bin*.