

6. Grouper Shell

Shell Grouper pour effectuer des opérations en ligne de commande

A partir du répertoire de grouper-api :

Mode interactif :

```
./bin/gsh.sh
```

Mode commandes :

```
./bin/gsh.sh /chemin_fichier
```

Exemples de commandes :

```
addRootStem("groupes", "Groupes")
```

Ajoute le dossier racine qui a pour identifiant "groupes" et pour nom "Groupes". Un dossier racine est nécessaire.

```
addStem("groupes", "admin", "Administrateurs")
```

Ajoute le dossier (Stem) identifiant *admin*, nom *Administrateurs* dans le dossier qui a pour identifiant *groupes* (le dossier racine dans l'exemple ci-dessus). Le dossier créé aura pour identifiant complet *groupes:admin*

```
addGroup("groupes:admin", "grouper", "Administrateurs Grouper")
```

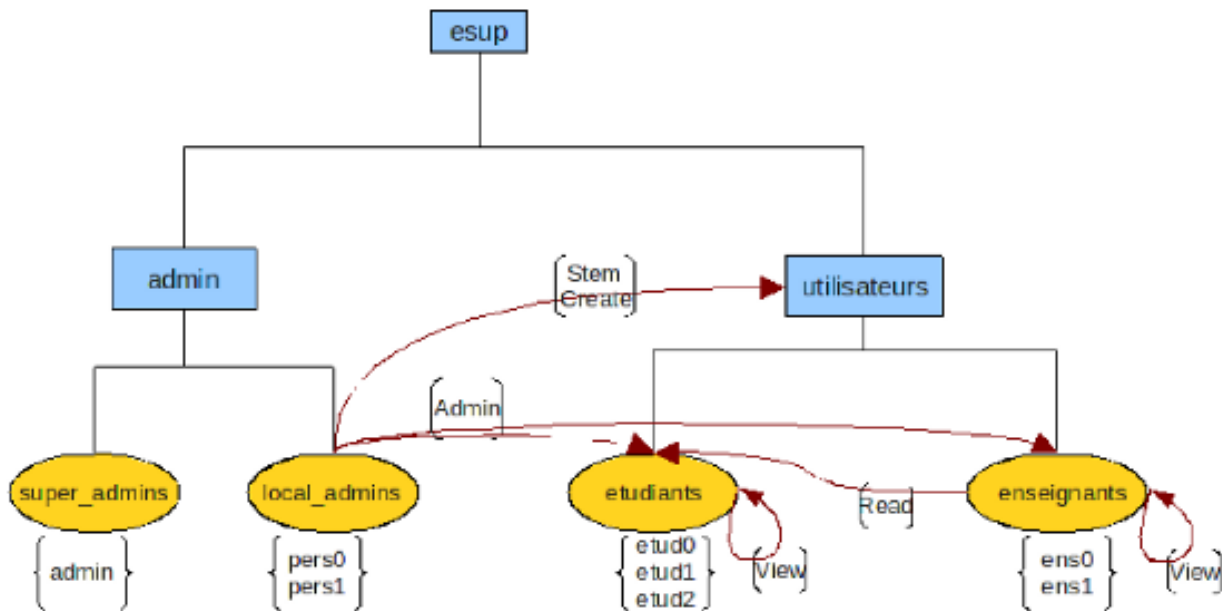
Ajoute le groupe qui a pour identifiant *grouper* et pour nom *Administrateurs Grouper* dans le dossier qui a pour identifiant complet *groupes:admin*. Le groupe créé aura pour identifiant complet *groupes:admin:grouper*

```
addMember("groupes:admin:grouper", "toto")
```

Ajoute le compte "toto" comme membre du groupe *groupes:admin:grouper*. Ce compte doit exister dans la ou les sources de données de Grouper (voir configuration grouper API, sources de données)

Exemple de script

Cet exemple de script permet de générer hiérarchie de groupes suivante :



```

// Initialisation de la session.
grouperSession = GrouperSession.startRootSession();

// Ajout des descriptions pour les dossiers et groupes
// crees lors de l'initialisation de Grouper.
rootStem = StemFinder.findByName(grouperSession, "esup");
rootStem.setDescription("Racine ESUP");
rootStem.setDisplayExtension("ESUP");
rootStem.store();

adminStem = StemFinder.findByName(grouperSession, "esup:admin");
adminStem.setDescription("Racine pour les groupes d'administration");
adminStem.setDisplayExtension("Admin");
adminStem.store();

sAdmin = GroupFinder.findByName(grouperSession, "esup:admin:super_admins");
sAdmin.setDescription("Groupe des super administrateurs");
sAdmin.setDisplayExtension("Super_Admins");
sAdmin.store();

// Ajout de l'utilisateur admin dans le groupe de confiance.
addMember("esup:admin:super_admins", "admin");

// Création du dossier utilisateur.
stem = addStem("esup", "utilisateurs", "Utilisateurs");
stem.setDescription("Racine pour les groupes d'utilisateurs");
stem.store();

// Création et peuplement du groupe d'administration local.
lAdmin = addGroup("esup:admin", "local_admins", "Local_Admins");
lAdmin.setDescription("Groupe des administrateurs locaux");
lAdmin.store();
String[] targetSubjects = {"pers0", "pers1"};
for (String targetSubject : targetSubjects) { addMember("esup:admin:local_admins", targetSubject); }

// Création et peuplement du groupe d'enseignants.
teachers = addGroup("esup:utilisateurs", "enseignants", "Enseignants");
teachers.setDescription("Un groupe d'enseignants");
teachers.store();
targetSubjects = new String[]{"ens0", "ens1"};
for (String targetSubject : targetSubjects) { addMember("esup:utilisateurs:enseignants", targetSubject); }
  
```

```
// Création et peuplement du groupe d'étudiants.
teachers = addGroup("esup:utilisateurs","etudiants", "Etudiants");
teachers.setDescription("Un groupe d'étudiants");
teachers.store();
targetSubjects = new String[] {"etud0", "etud1", "etud2"};
for (String targetSubject : targetSubjects) { addMember("esup:utilisateurs:etudiants", targetSubject); }

// Positionnement des privilèges
// Par défaut les groupes sont visibles de tous, on supprime ce droit.
revokePriv("esup:admin:super_admins", "GrouperAll", AccessPrivilege.VIEW);
revokePriv("esup:admin:local_admins", "GrouperAll", AccessPrivilege.VIEW);
revokePriv("esup:utilisateurs:etudiants", "GrouperAll", AccessPrivilege.VIEW);
revokePriv("esup:utilisateurs:enseignants", "GrouperAll", AccessPrivilege.VIEW);

// local_admin
grantPriv("esup:utilisateurs", "esup:admin:local_admins", NamingPrivilege.STEM);
grantPriv("esup:utilisateurs", "esup:admin:local_admins", NamingPrivilege.CREATE);
grantPriv("esup:utilisateurs:enseignants", "esup:admin:local_admins", AccessPrivilege.ADMIN);
grantPriv("esup:utilisateurs:etudiants", "esup:admin:local_admins", AccessPrivilege.ADMIN);

// etudiants
grantPriv("esup:utilisateurs:etudiants", "esup:utilisateurs:etudiants", AccessPrivilege.VIEW);

// enseignants
grantPriv("esup:utilisateurs:enseignants", "esup:utilisateurs:enseignants", AccessPrivilege.VIEW);
grantPriv("esup:utilisateurs:etudiants", "esup:utilisateurs:enseignants", AccessPrivilege.READ);
```