

Intégration RECIA Grouper-Portail



Le GIP RECIA n'utilise plus ce type de groupe store et utilise uniquement le SmartLdapGroupStore

Note d'installation pour l'utilisation des groupes Grouper dans le portail esup 3.2.4 via le GroupStore ESCO du RECIA.

- [Description](#)
- [Intégration au portail Esup 3.2.4](#)
 - [La compilation](#)
 - [Prise en compte des groupes grouper par le portail](#)
 - [Configuration de Grouper](#)
- [Adaptation des sources uPortal](#)
 - [Patch du gestionnaire de groupes d'uPortal](#)
 - [Patch des sources du portail liés à l'intégration du groupStore ESCO](#)
- [Initialisation et déploiement](#)

conventions :

<esupSrc> : le répertoire du package du portail eSup

<updateSrc> : le répertoire update des sources de eSup

<customDir> : le répertoire des custom de eSup.

Description

Pour faciliter l'intégration, les sources ont été packagées sous forme de jar via un projet maven et sont disponibles sur le repository du RECIA, le nom de l'archive contient la version de grouper avec laquelle est est packagée (actuellement uniquement disponible avec grouper 1.6.3). Donc la récupération des jar sera automatique en suivant les étapes décrites après.

Les sources du projet peuvent être récupérées sur simple demande (actuellement disponible uniquement sur le svn du RECIA).

L'intégration choisie par le RECIA avec la version 2.6 du portail était l'utilisation d'un webservice qui interrogeait directement la base grouper via l'api grouper, ce web service avait été réalisé alors que le grouper-ws n'existait pas encore. Pour le passage à la version 3.2 du portail nous avons choisi d'intégrer directement les accès à l'API grouper dans le portail en ne passant plus par un système de webservices afin de gagner en performance. En fait les sources de l'ancien webservices sont intégrées directement dans le portail et il n'y a plus de déclaration webservices afin d'éviter les problèmes d'accès à des services extérieur.

Actuellement l'intégration avec grouper 1.6.3 fonctionne en production au RECIA et les résultats sont plutôt assez satisfaisant.

Pourquoi choisir ce groupStore plutôt que celui du JASIG :

- Intégration dynamique des groupes via un fichier de configuration qui lie un groupe uPortal et une branche définie d'un groupe Grouper, et on peut définir ainsi plusieurs mapping sans intervenir sur la base de données des groupes du Portail.

- Affichage, navigation dans l'arborescence des groupes grouper et listing de leurs membres possible via le groupManager, les groupes sont aussi visible dans le permissionManager et le channelManager.

Intégration au portail Esup 3.2.4

La compilation

- Ajout d'un repository

```
cp <esupSrc>/uPortal_rel-3.2.4/pom.xml <customDir>/uPortal/
```

Edition du fichier pom.xml pour ajouter les repository

```
vim <customDir>/uPortal/pom.xml
```

ajouter les lignes suivantes dans la liste des repositories défini, entre le repository jasig-3rd-party et sonatype-nexus-snapshots par exemple (ligne 66)

```
<repository>
  <id>Recia</id>
  <name>esco repo</name>
  <url>http://www.esco-portail.org/nexus/content/repositories/public/</url>
  <releases><enabled>true</enabled></releases>
  <snapshots><enabled>false</enabled></snapshots>
</repository>
```

- Ajout des dépendances

```
cp <esupSrc>/uPortal_rel-3.2.4/uportal-impl/pom.xml <customDir>/uPortal/uportal-impl/
```

Edition du fichier pom.xml pour ajouter les dépendances

```
vim <customDir>/uPortal/uportal-impl/pom.xml
```

ajouter les lignes suivantes dans la liste des dependencies, juste avant le commentaire ~~<!-- esup dependencies -->~~

```
<!-- esco dependencies -->
<dependency>
  <groupId>org.esco.grouper</groupId>
  <artifactId>esco-grouper-portal-groupManager</artifactId>
  <version>0.0.3-grouper-1.6.3</version>
  <exclusions>
    <exclusion>
      <groupId>org.springframework</groupId>
      <artifactId>spring</artifactId>
    </exclusion>
    <exclusion>
      <groupId>net.sf.ehcache</groupId>
      <artifactId>ehcache</artifactId>
    </exclusion>
  </exclusions>
</dependency>
<dependency>
  <groupId>edu.internet2.middleware.subject.provider</groupId>
  <artifactId>ldap-source-adapter</artifactId>
  <version>0.0.3-grouper-1.6.3</version>
  <exclusions>
    <exclusion>
      <groupId>org.springframework</groupId>
      <artifactId>spring</artifactId>
    </exclusion>
    <exclusion>
      <groupId>net.sf.ehcache</groupId>
      <artifactId>ehcache</artifactId>
    </exclusion>
  </exclusions>
</dependency>
```

des exclusions sont a définir afin d'éviter les incompatibilités avec les librairies uPortal.

La deuxième dépendance est facultative, nous l'utilisons afin d'éviter des connexions multiples au ldap, cela agit comme la définition d'un pool ldap. Il s'agit en fait de l'intégration faite par l'université de washington disponible ici <http://staff.washington.edu/fox/grouper/>, d'autres fonctions peuvent aussi être utilisées mais le mieux est de se référer à la documentation fournie dans les sources du projet.

Pour des raisons de praticité ces sources ont été intégrées sous forme de module au projet maven du groupStore ESCO.

Prise en compte des groupes grouper par le portail

- Fichier à adapter

```
cp <esupSrc>/uPortal_rel-3.2.4/uportal-impl/src/main/resources/properties/groups/compositeGroupServices.xml
<customDir>/uPortal/uportal-impl/src/main/resources/properties/groups/
```

- Ajouter le service pour le groupStore ESCO en ajoutant cette définition :

```
<!-- ESCO-Grouper service to use Gouper groups -->
<service>
  <name>ESCO-GROUPER</name>
  <service_factory>org.esco.portal.groups.grouper.ESCOREferenceIndividualGroupServiceFactory</service_factory>
  <entity_store_factory>org.esco.portal.groups.grouper.EntityStoreFactory</entity_store_factory>
  <group_store_factory>org.esco.portal.groups.grouper.EntityGroupStoreFactory</group_store_factory>
  <entity_searcher_factory>org.esco.portal.groups.grouper.EntitySearcherFactory</entity_searcher_factory>
  <internally_managed>false</internally_managed>
  < caching_enabled>true</ caching_enabled>
</service>
```

- Modifier le service du groupStore "local" ainsi

```
<service>
  <name>local</name>
  <service_factory>org.esco.portal.groups.grouper.ESCOREferenceIndividualGroupServiceFactory</service_factory>
  <entity_store_factory>org.jasig.portal.groups.ReferenceEntityStoreFactory</entity_store_factory>
  <group_store_factory>org.esco.portal.groups.grouper.ESCOREferenceEntityGroupStoreFactory</group_store_factory>
  <entity_searcher_factory>org.jasig.portal.groups.ReferenceEntitySearcherFactory</entity_searcher_factory>
  <internally_managed>true</internally_managed>
  < caching_enabled>true</ caching_enabled>
</service>
```

Cette définition est nécessaire afin de pouvoir retourner les groupes grouper liés à un groupes uPortal, cela est effectué sous forme de decorator.

- Définir le mapping entre les groupes Grouper et locaux

définir le fichier <customDir>/uPortal/uportal-impl/src/main/resources/properties/groups/esco-GroupLoad.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE properties SYSTEM "http://java.sun.com/dtd/properties.dtd">
<properties>

  <comment>
    Properties used to define how the the Grouper groups should be loaded
    in the native uPortal groups.
  </comment>

  <!-- Duration validity, in seconds, for the Grouper requests. -->
  <entry key="grouper.requests.cache.duration">25</entry>

  <!--
    Defines the grouper groups to load in uPortal groups.
    keys denote a uPortal group name and values are semicolon separated
    lists of grouper stems. The group loaded are the root groups
    in the stems.

    For instance, all groups in the stem esco:admin that are not member of another group
    are loaded in the Portal Administrator group:
  <entry key="Portal Administrators">esco:admin</entry>
  -->
  <entry key="Administrateurs Portail">esco:admin:central</entry>
  <entry key="Tout le monde">esco:Etablissements;esco:Applications</entry>
  <entry key="Propriétaires de fragment">esco:admin:esco-lo</entry>
</properties>
```

Configuration de Grouper

il est nécessaire de placer les fichiers de configuration de grouper, pour faire cela il vous faudra récupérer les fichiers suivants dans le répertoire conf de l'api-grouper (normalement sur la machine où grouper est installé):

- grouper.properties
- grouper.hibernate.properties
- ehcache.xml
- grouper.ehcache.xml
- morphString.properties
- sources.xml

et les placer dans le répertoire <customDir>/uPortal/uportal-impl/src/main/resources/

Et adapter selon les besoins les fichiers suivants :

- grouper.properties
- grouper.hibernate.properties (ajouter la ligne suivante dans ce fichier afin d'éviter un problème de compatibilité entre hibernate et antlr **hibernate.query.factory_class=org.hibernate.hql.classic.ClassicQueryTranslatorFactory**)
- sources.xml

*Si vous avez décidé d'installer le module ldap-source-adapter il vous faudra configurer le fichier sources.xml selon cet exemple :

```
<?xml version="1.0" encoding="utf-8"?>

<!--
  LdapSourceAdapter configuration
-->

<sources>

  ...

  <source adapterClass="edu.internet2.middleware.subject.provider.LdapSourceAdapter">
    <id>uwpds</id>
    <name>UW_PDS</name>
    <type>person</type>

    <!-- Note that most of the ldap configuration is in the properties file
         The filename can be a file in your classpath or an absolute pathname
    -->

    <init-param>
      <param-name>ldapProperties_file</param-name>
      <param-value>ldap.properties</param-value>
    </init-param>

    <init-param>
      <param-name>SubjectID_AttributeType</param-name>
      <param-value>uwRegID</param-value>
    </init-param>
    <init-param>
      <param-name>Name_AttributeType</param-name>
      <param-value>uwNetID</param-value>
    </init-param>
    <init-param>
      <param-name>Description_AttributeType</param-name>
      <param-value>displayName</param-value>
    </init-param>

    <search>
      <searchType>searchSubject</searchType>
      <param>
        <param-name>filter</param-name>
        <param-value>
          (& (uwRegID=%TERM%) (objectclass=uwPerson))
        </param-value>
      </param>
      <param>
        <param-name>scope</param-name>

```

```

        <param-value>
            SUBTREE_SCOPE
        </param-value>
    </param>
    <param>
        <param-name>base</param-name>
        <param-value>
            </param-value>
        </param>
    </search>
</search>
<search>
    <searchType>searchSubjectByIdentifier</searchType>
    <param>
        <param-name>filter</param-name>
        <param-value>
            (&(|(uwNetID=%TERM%)(uwStudentID=%TERM%))(objectclass=uwPerson))
        </param-value>
    </param>
    <param>
        <param-name>scope</param-name>
        <param-value>
            SUBTREE_SCOPE
        </param-value>
    </param>
    <param>
        <param-name>base</param-name>
        <param-value>
            </param-value>
        </param>
    </search>

<!-- use the firstlastfilter to allow: last, first lookup -->
<search>
    <searchType>search</searchType>
    <param>
        <param-name>filter</param-name>
        <param-value>
            (&(|(uwNetID=%TERM%)(sn=%TERM%)(uwNetId=*)(objectclass=uwPerson)))
        </param-value>
    </param>
    <param>
        <param-name>firstlastfilter</param-name>
        <param-value>
            (&(|(uwPersonRegisteredSurname=%LAST%)(uwPersonRegisteredFirstMiddle=%FIRST%*)(uwNetid=*)
(objectClass=uwPerson))
        </param-value>
    </param>
    <param>
        <param-name>scope</param-name>
        <param-value>
            SUBTREE_SCOPE
        </param-value>
    </param>
    <param>
        <param-name>base</param-name>
        <param-value>
            </param-value>
        </param>
    </search>
    <!-- Attributes you would like to display when doing a search -->
    <attribute>uwNetID</attribute>
    <attribute>uwEWPName</attribute>
    <attribute>uwSWPName</attribute>
    <attribute>uwEWPDept1</attribute>

</source>

</sources>

```

côté RECIA nous avons juste modifié ces paramètres par rapport au fichier sources.xml original et personnalisé de grouper :

```
....
<source adapterClass="edu.internet2.middleware.subject.provider.LdapSourceAdapter">
  <id>esco:ldap</id>
  <name>JNDI Source Adapter esco:ldap</name>
  <type>person</type>

  <!-- Note that most of the ldap configuration is in the properties file
        The filename can be a file in your classpath or an absolute pathname
  -->

  <init-param>
    <param-name>ldapProperties_file</param-name>
    <param-value>ldap.properties</param-value>
  </init-param>

  <init-param>
    <param-name>Multiple_Results</param-name>
    <param-value>false</param-value>
  </init-param>
....
```

c'est à dire la classe de l'adapterClass qui change et l'ajout des deux init-param à la place des param ldap (INITIAL_CONTEXT_FACTORY, PROVIDER_URL, SECURITY_AUTHENTICATION, SECURITY_PRINCIPAL, SECURITY_CREDENTIALS)

et ajouter un fichier ldap.properties avec cet exemple de conf :

```
# ldap source adapter properties
#
# for a complete list of properties see:
#   http://code.google.com/p/vt-middleware/wiki/vtldap#Configuration_Properties
#
#

edu.vt.middleware.ldap.ldapUrl=ldap://pds.example.edu:389
edu.vt.middleware.ldap.base=dc=example,dc=edu
edu.vt.middleware.ldap.searchScope=SUBTREE

# authn if simple
edu.vt.middleware.ldap.serviceUser=cn=admin,dc=example,dc=edu
edu.vt.middleware.ldap.serviceCredential=admin_password

# authn for sasl external (certificates)
#edu.vt.middleware.ldap.authType=EXTERNAL
#edu.vt.middleware.ldap.tls=true
#edu.vt.middleware.ldap.serviceUser=cn=admin.example.edu
# these to use PEM format cert and key
#pemCaFile=/path/to/ca.pem
#pemCertFile=/path/to/cert.pem
#pemKeyFile=/path/to/key.pem

# pooling options
edu.vt.middleware.ldap.pool.minPoolSize = 2
edu.vt.middleware.ldap.pool.maxPoolSize = 5
```

Adaptation des sources uPortal

Patch du gestionnaire de groupes d'uPortal

L'utilisation de "." comme séparateur dans le nommage des groupes dans Grouper pose un problème au gestionnaire groupes d' uPortal. Ce problème a déjà été reporté dans le [Jira de uPortal|<https://issues.jasig.org/browse/UP-2994>] Bill Brown a déjà proposé un patch [<https://issues.jasig.org/secure/attachment/11917/rel-3-2-patches.patch>].

Pour ce faire, j'ai tout simplement modifié les fichiers source de uPortal avec les différences décrites dans le fichier patch.

Patch des sources du portail liés à l'intégration du groupStore ESCO

Afin de pouvoir intégrer correctement les groupes grouper à la façon RECIA il est nécessaire de modifier quelques classes du portail, d'une part pour éviter des problèmes de blocage lors de la remontées des groupes et de leur membres quand il commence à y en avoir beaucoup (cas du contexte RECIA, rencontré à partir de 6000-7000 groupes).

Recopier les fichiers à éditer

```
mkdir -p <customDir>/uportal-impl/src/main/java/org/jasig/portal/channels/groupsmanager/  
mkdir <customDir>/uportal-impl/src/main/java/org/jasig/portal/channels/groupsmanager/wrappers/  
cp <esupSrc>/uPortal_rel-3.2.4/uportal-impl/src/main/java/org/jasig/portal/channels/groupsmanager/  
/CGroupsManager.java <customDir>/uPortal/uportal-impl/src/main/java/org/jasig/portal/channels/groupsmanager/  
cp <esupSrc>/uPortal_rel-3.2.4/uportal-impl/src/main/java/org/jasig/portal/channels/groupsmanager/  
/GroupsManagerXML.java <customDir>/uPortal/uportal-impl/src/main/java/org/jasig/portal/channels/groupsmanager/  
cp <esupSrc>/uPortal_rel-3.2.4/uportal-impl/src/main/java/org/jasig/portal/channels/groupsmanager/wrappers/  
/GroupWrapper.java <customDir>/uPortal/uportal-impl/src/main/java/org/jasig/portal/channels/groupsmanager/  
/wrappers/
```

et adapter en fonction de ces diff (les fichiers sont en pièce jointe dans cette documentation) :

diff <esupSrc>/uPortal_rel-3.2.4/uportal-impl/src/main/java/org/jasig/portal/channels/groupsmanager/CGroupsManager.java <customDir>
/uPortal/uportal-impl/src/main/java/org/jasig/portal/channels/groupsmanager/CGroupsManager.java

```
65,74c65
<      // -----
<      // -- Modif ESCO - A. Deman 10/2008
<      // -----
<      // Pour eviter de charger systematiquement
<      // tous les groupes.
<      // L'appell à la méthode init a été
<      // ajoutée dans les accesseurs de cette
<      // classe qui ne l'avaient pa déjà.
<
<      //init();
---
>      init();
102,109c93
<      // -----
<      // -- Modif ESCO - J. Gribonvald 07/2011
<      // -----
<      // Pour eviter de charger systematiquement
<      // tous les groupes et leurs membre
<      // sur le groupe everyone
<      // à l'accès des canaux comme le permissionManager.
<      /*Iterator allgroups = everyone.getAllMembers();
---
>      Iterator allgroups = everyone.getAllMembers();
122,126c106
<      */
<      // -----
<      // -- Modif ESCO - J. Gribonvald 07/2011
<      // -----
<      // --- Fin modif.
---
>      }
375c366
<      init();
---
>      init();
393,400c384
<      // -----
<      // Modif ESCO - A. Deman - 10/2008
<      // -----
<      // Ajout de l'appel à init()
<      // supprimé dans le constructeur.
<      init();
<      // --- Fin modif.
<      return (String)activities.get(token);
---
>      return (String)activities.get(token);
418,425d401
<      // -----
<      // Modif ESCO - A. Deman - 10/2008
<      // -----
<      // Ajout de l'appel à init()
<      // supprimé dans le constructeur.
<      init();
<      // --- Fin modif.
<
```

```
diff <esupSrc>/uPortal_rel-3.2.4/uportal-impl/src/main/java/org/jasig/portal/channels/groupsmanager/GroupsManagerXML.java <customDir>
/uPortal/uportal-impl/src/main/java/org/jasig/portal/channels/groupsmanager/GroupsManagerXML.java
```

```
239,242c239
< //      ===== Modification ESCO =====
< //      Used to handle Grouper groups.
< //      if (hasMembers) {
< //      =====
---
>      if (hasMembers) {
251c248
< //      } // Modification ESCO
---
>      }
```

```
diff <esupSrc>/uPortal_rel-3.2.4/uportal-impl/src/main/java/org/jasig/portal/channels/groupsmanager/wrappers/GroupWrapper.java
<customDir>/uPortal/uportal-impl/src/main/java/org/jasig/portal/channels/groupsmanager/wrappers/GroupWrapper.java
```

```
239,242c239
< //      ===== Modification ESCO =====
< //      Used to handle Grouper groups.
< //      if (hasMembers) {
< //      =====
---
>      if (hasMembers) {
251c248
< //      } // Modification ESCO
---
>      }
```

- Si utilisation du WS esup des groupes

```
cp <updateSrc>/uPortal/uportal-impl/src/main/java/org/esupportail/portal/ws/groups/PortalGroups.java <customDir>
/uPortal/uportal-impl/src/main/java/org/esupportail/portal/ws/groups/PortalGroups.java
```

éditer le fichier selon le diff suivant :

diff <updateSrc>uPortal/uportal-impl/src/main/java/org/esupportail/portal/ws/groups/PortalGroups.java <customDir>uPortal/uportal-impl/src/main/java/org/esupportail/portal/ws/groups/PortalGroups.java

```
35a36
> import org.esco.portal.groups.grouper.ESCOEntityGroupImpl;
284,285c285,290
<                                     // on remplit ce vecteur en remontant la hierarchie recursivement
<                                     getRecurContainingGroups(egi, groups, v);
---
>                                     // on appelle une méthode spécifique optimal pour les groupes grouper
>                                     if (egi instanceof ESCOEntityGroupImpl) {
>                                         getESCORecurContainingGroups(egi,v);
>                                     } else {
>                                         getRecurContainingGroups(egi, groups, v);
>                                     }
364,365c369,383
<
<
---
>
>     /**
>     * Même méthode que la précédente mais spécifique aux groupes esco. Méthode optimale pour obtenir les
groupes parents !
>     */
>     private void getESCORecurContainingGroups(IEntityGroup egi, Vector v) {
>         String[] list = egi.getKey().split(":");
>         StringBuffer group = new StringBuffer();
>         for (String current : list) {
>             if (group.length()>0) {
>                 group.append(":");
>             }
>             group.append(current);
>             v.add(group.toString());
>         }
>     }
```

Initialisation et déploiement

Et pour finir : ant init deploy

Bien entendu, il faut redémarrer le portail.