

Migration

Vous trouverez ici la documentation nécessaire pour faire passer vos projets, chartes graphiques et configurations de Chainedit 1 vers ChainEdit 2.

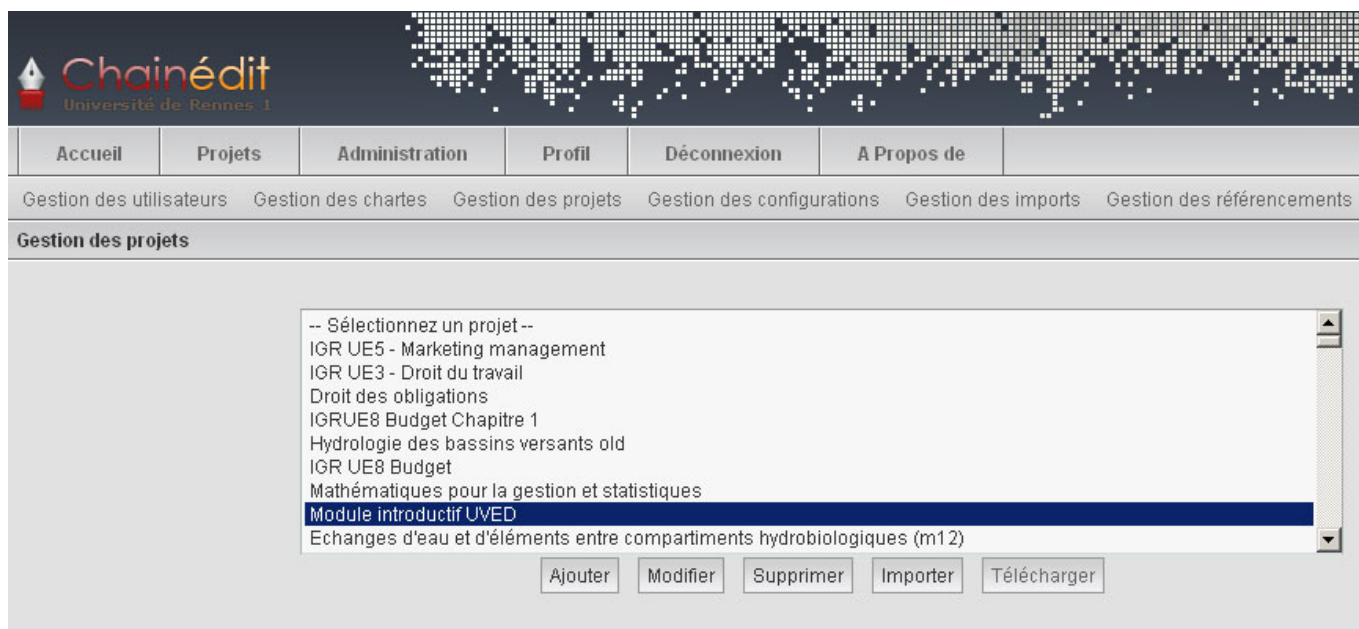
- [1. Précision](#)
- [Exécution de ChainEdit sous Eclipse](#)
- [Lien Utiles](#)
- [Installeur](#)

1. Précision

Certains outils de compression et décompression de fichiers peuvent poser problème lors de l'intégration dans Chainedit. Nous utilisons quant à nous 7-Zip. Les fichiers compressés via cet outil peuvent être intégrés parfaitement.

2. Migration d'un projet

Il n'y a pas d'intervention particulière à réaliser sur le projet. Pour la migration, il suffit, dans l'ancien ChainEdit, de télécharger le fichier compressé du projet (Onglet Administration / Gestion des Projets).





Chainédit
Université de Rennes 1

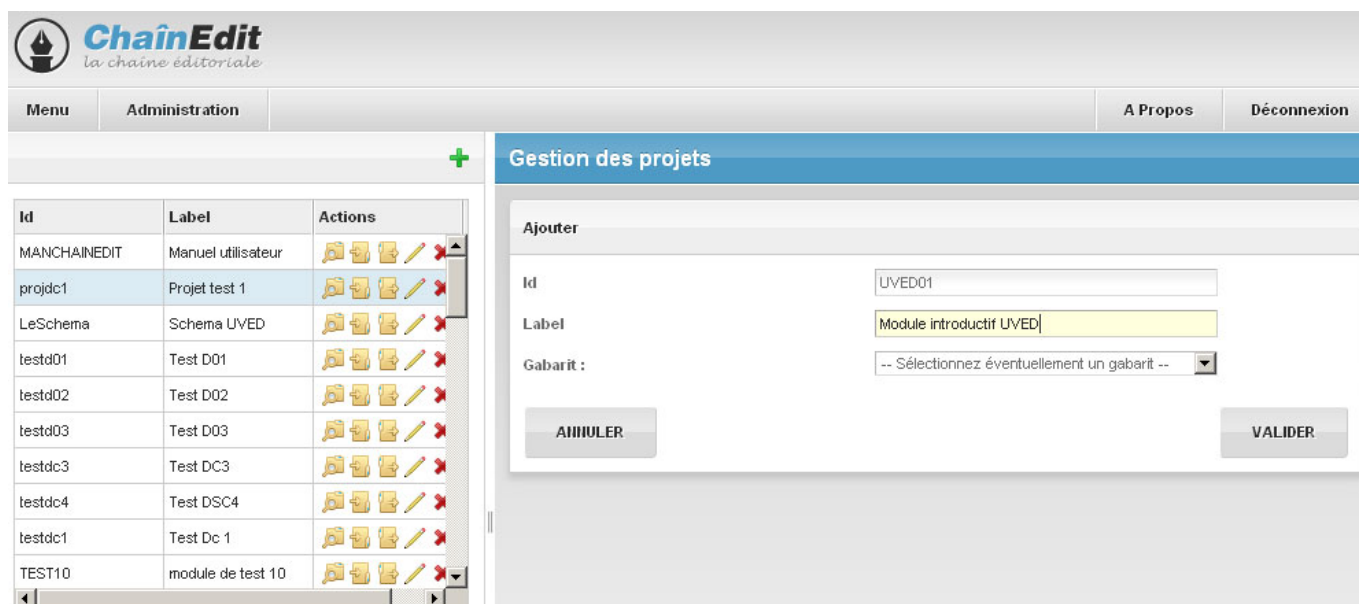
Accueil Projets Administration Profil

Résultat de la compression du projet : Module introductif UVED

Résultat de la génération de l'archive : Télécharger l'archive Zip

Retour





















Dans le nouveau ChainEdit, dans la partie Administration / Gestion des Projets, on ajoute un nouveau projet :



ChainEdit
la chaîne éditoriale

Menu Administration A Propos Déconnexion

Gestion des projets

Id	Label	Actions
MANCHAINEDIT	Manuel utilisateur	 
projdc1	Projet test 1	 
LeSchema	Schema UVED	 
testd01	Test D01	 
testd02	Test D02	 
testd03	Test D03	 
testdc3	Test DC3	 
testdc4	Test DSC4	 
testdc1	Test Dc 1	 
TEST10	module de test 10	 

Ajouter

Id:

Label:

Gabarit :

ANNULER VALIDER

Il est impératif de donner le même identifiant (Id) au projet que dans l'ancienne version de ChainEdit. On saisit également le label (qui peut être différent) et on clique sur Valider.

Gestion des projets

Ajouter

Id	<input type="text" value="UVED01"/>
Label	<input type="text" value="Module introductif UVED"/>
Schéma :	<input type="text" value="UVED\UvedConfig.xml"/> ▼
Catégorie	<input type="text" value="Catégorie UVED"/> ▼
Répertoire de publication	<input type="text"/>
Url d'accès à la publication :	<input type="text"/>
Référencement :	<input type="text" value="-- Sélectionnez un référencement --"/> ▼

Sélectionnez les chartes à associer à ce projet

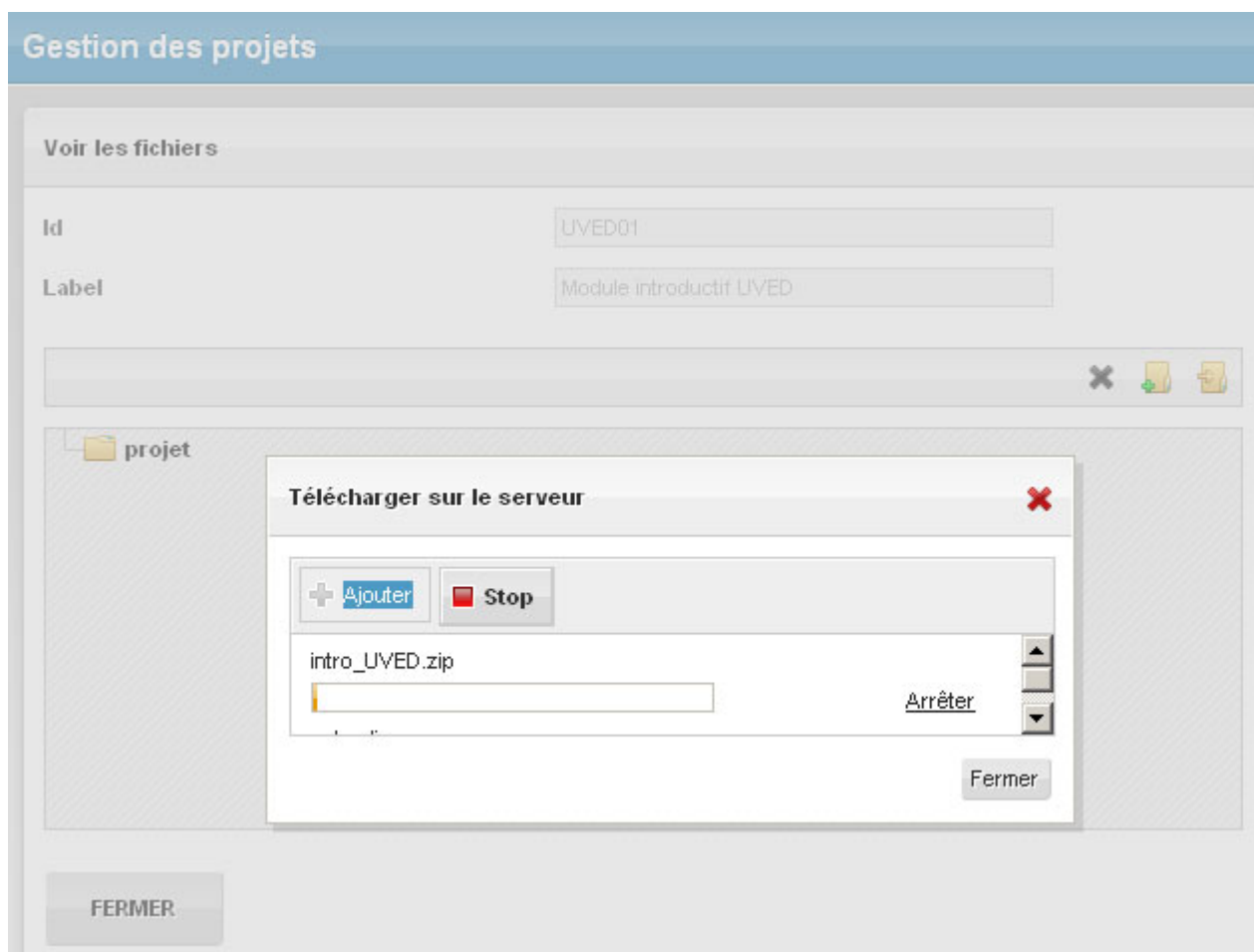
Charte UVED 2

Charte UVED

➡

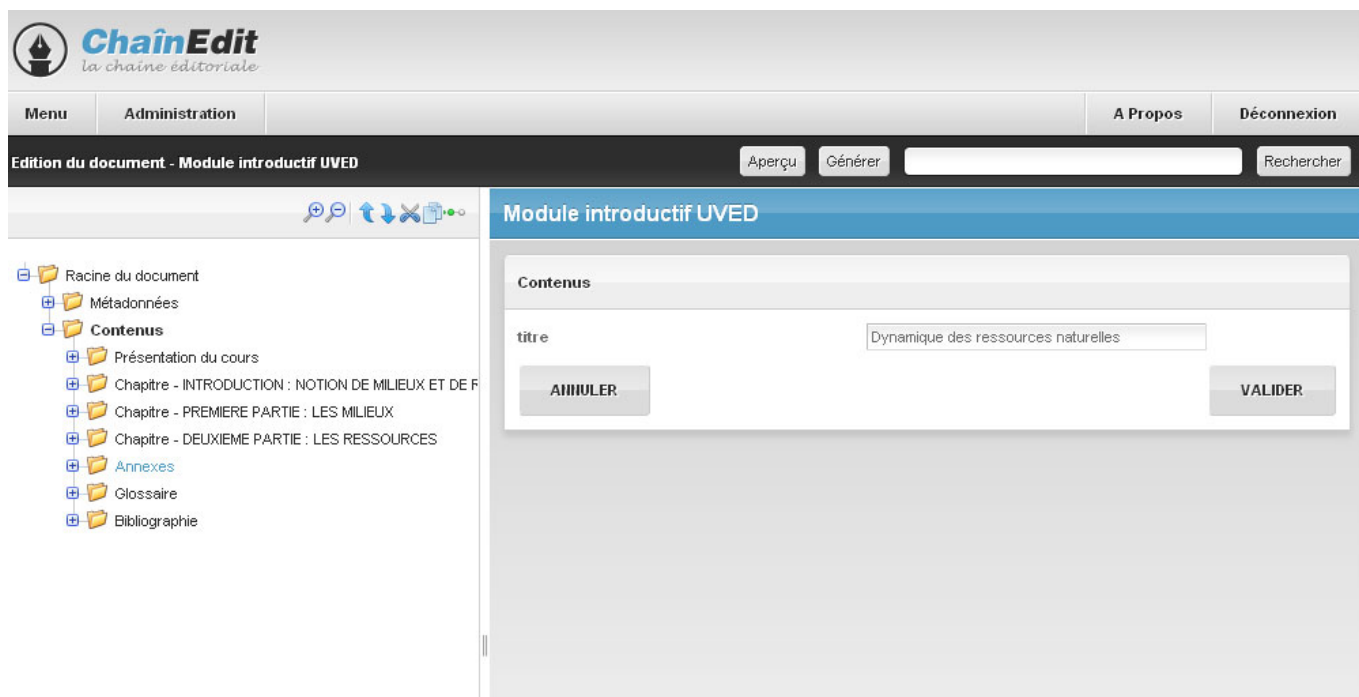
Après enregistrement du projet, il faut déposer le fichier compressé (en cliquant au préalable sur l'icône 📁). Sur le dossier projet, on sélectionne l'option « Télécharger sur le serveur ».

On renseigne ensuite quel est le schéma du projet, le cas échéant la catégorie à laquelle il appartient, les chartes qui lui sont associées...



Puis, une fois ce fichier téléchargé, il faut le décompresser.

Il est possible de vérifier que la migration s'est bien déroulée dans l'édition du projet. Si le projet s'ouvre sans problème, s'il est possible de le générer, la migration du projet s'est faite correctement.



3. Migration d'une configuration

En ce qui concerne les configurations, si vous possédez une configuration différente de celles d'ores et déjà présentes dans ChainEdit, vous devez procéder aux modifications suivantes avant l'intégration :

Dans le fichier XXXConfig.xml (XXX étant votre configuration), il faut remplacer

```
<?xml version="1.0" encoding="ISO-8859-1"?>
```

par

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

Ensuite, il faut modifier les balises ayant un attribut `type="typefck"`. La valeur de cet attribut vient "`typeEditor`" et les paramètres changent également.

Par exemple :


```
<BALISE nom="PARAGRAPHE" titre="Paragraphe" type="typefck"
groupe="paragraphe">
  <PARAMETRE nom="configFck" valeur="Fck1"/>
  <PARAMETRE nom="classFck"
valeur="org.esupportail.chainedit.domain.beans.FckTransformImpl"/>
  <ATTRIBUT nom="titre" titre="titre" />
</BALISE>
```

devient :

```
<BALISE nom="PARAGRAPHE" titre="Paragraphe" type="typeEditor"
groupe="paragraphe">
  <PARAMETRE nom="configEditorPlugins"
valeur="safari,spellchecker,pagebreak,style,layer,table,save,advhr,advimage,advl
ink,emotions,iespell,inlinepopups,insertdatetime,preview,media,searchreplace,pri
nt,paste,directionality,fullscreen,noneditable,visualchars,nonbreaking,xhtmlxtra
s,template,glossaire,biblio,webo,alt,equajaxe,equatex"/>
  <PARAMETRE nom="configEditorToolbar1"
valeur="fullscreen,code,print,|,undo,redo,|,cut,copy,paste,pastetext,pasteword,|
,search,replace"/>
  <PARAMETRE nom="configEditorToolbar2"
valeur="bold,italic,underline,striketrough,sub,sup,cite,|,charmap,insertdate,in
serttime"/>
  <PARAMETRE nom="configEditorToolbar3"
valeur="glossaire,biblio,webo,alt,equajaxe,equatex"/>
  <PARAMETRE nom="classEditor"
valeur="org.esupportail.chainedit.external.interfaces.defaultImpl.EditorTransfor
mImpl"/>
  <ATTRIBUT nom="titre" titre="titre" />
</BALISE>
```

Les paramètres "`configEditorToolbar`" ne seront précisés dans ce fichier XML, que si les outils qu'ils concernent sont utilisés dans l'élément défini par la balise. Ainsi, dans cet exemple, on définit l'élément PARAGRAPHE. On lui met les 3 paramètres correspondant aux barres d'outils nécessaires. La 3ème barre d'outils qui permet l'utilisation des fonctionnalités telles que la bibliographie, le glossaire, les formules mathématiques, etc. ("`configEditorToolbar3`") sera utilisée moins

systématiquement que les 2 autres.

Pour intégrer la configuration dans le nouveau ChainEdit, il faut aller dans la partie Administration / Gestion des configurations et cliquer sur l'icône  pour modifier la configuration. En se positionnant sur le dossier à la racine, portant le nom de la configuration, il est possible

- soit de télécharger les fichiers un par un.
- soit de télécharger un fichier compressé contenant tous les fichiers de la configuration, puis de décompresser ce dernier.

4. Migration d'une charte

4.1. Encodage

Dans chaque feuille XSL, il faut remplacer

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes"?>
```

par

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
```

4.2. Traitements

Auparavant, il y avait par exemple une charte pour la génération en odt ou pdf, une autre charte pour une génération en html... Par exemple, le fichier charte.xml d'une 1ère charte (génération en Open Office) était construit comme ci-dessous :

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes"?>
<CHARTe label="Charte ODF NEUTRE" id="ODFNEUTRE">
  <TRAITEMENTS type="openOffice" nature="openOffice">
    <FICHIERXSL ordre="1" chemin="\xsl\CIRM_LANCEUR.xsl"/>
    <REPertoireCHARTe chemin="\META-INF"/>
    <REPertoireCHARTe chemin="\Pictures"/>
    <REPertoireCHARTe chemin="\mimetype"/>
    <REPertoireCHARTe chemin="\settings.xml"/>
    <REPertoireCHARTe chemin="\styles.xml"/>
    <REPertoireCIBLE ressource="non" default="oui" chemin="\oo"
charte="oui"/>
  </TRAITEMENTS>
</CHARTe>
```

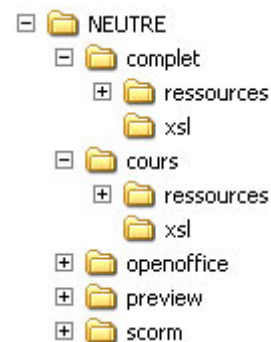
Une 2ème charte (génération en html) avait un fichier charte.xml ressemblant à ce qui suit :


```
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes"?>
<CHARTe label="Charte NEUTRE" id="NEUTRE">
  <TRAITEMENTS type="autre" nature="htm">
    <FICHIERXSL ordre="1" chemin="\xsl\CIRM_LANCEUR.xsl"/>
    <REPERTOIRECHARTe chemin="\ressources"/>
    <REPERTOIRECIBLE ressource="non" default="non" chemin="/site"
charte="oui"/>
    <REPERTOIRECIBLE ressource="oui" default="oui" chemin="/site/html"
charte="non"/>
  </TRAITEMENTS>
</CHARTe>
```

Dorénavant, une charte peut correspondre à plusieurs traitements (pdf, html...). Et le fichier charte.xml contient tous les traitements, comme vous pouvez le voir dans l'exemple ci-après :

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<CHARTRE id="NEUTRE" label="Charte NEUTRE">
  <CHARTRECONFIG id="CIRM\CIRMConfig.xml"/>
  <TRAITEMENTS nature="complet" type="autre">
    <FICHIERXSL ordre="0" chemin="\complet\xsl\CIRM_LANCEUR.xsl"/>
    <REPERTOIRECHARTRE chemin="\complet\ressources"/>
    <REPERTOIRECIBLE chemin="/site" default="non" charte="oui"
ressource="non"/>
    <REPERTOIRECIBLE chemin="/site/html" default="oui" charte="non"
ressource="oui"/>
  </TRAITEMENTS>
  <TRAITEMENTS nature="cours" type="autre">
    <FICHIERXSL ordre="0" chemin="\cours\xsl\CIRM_LANCEUR.xsl"/>
    <REPERTOIRECHARTRE chemin="\cours\ressources"/>
    <REPERTOIRECIBLE chemin="/site" default="non" charte="oui"
ressource="non"/>
    <REPERTOIRECIBLE chemin="/site/html" default="oui" charte="non"
ressource="oui"/>
  </TRAITEMENTS>
  <TRAITEMENTS nature="scorm" type="autre">
    <FICHIERXSL ordre="0" chemin="\scorm\xsl\CIRM_LANCEUR.xsl"/>
    <REPERTOIRECHARTRE chemin="\scorm\ressources"/>
    <REPERTOIRECIBLE chemin="/site" default="non" charte="oui"
ressource="non"/>
    <REPERTOIRECIBLE chemin="/site/html" default="oui" charte="non"
ressource="oui"/>
  </TRAITEMENTS>
  <TRAITEMENTS nature="openoffice" type="openOffice">
    <FICHIERXSL ordre="0" chemin="\openoffice\xsl\CIRM_LANCEUR.xsl"/>
    <REPERTOIRECHARTRE chemin="\openoffice\META-INF"/>
    <REPERTOIRECHARTRE chemin="\openoffice\Pictures"/>
    <REPERTOIRECHARTRE chemin="\openoffice\mimetype"/>
    <REPERTOIRECHARTRE chemin="\openoffice\settings.xml"/>
    <REPERTOIRECHARTRE chemin="\openoffice\styles.xml"/>
    <REPERTOIRECIBLE chemin="\oo" default="oui" charte="oui"
ressource="non"/>
  </TRAITEMENTS>
  <TRAITEMENTS nature="preview" type="preview">
    <FICHIERXSL ordre="0" chemin="\preview\xsl\CIRM_LANCEUR.xsl"/>
    <REPERTOIRECHARTRE chemin="\preview\ressources"/>
    <REPERTOIRECIBLE chemin="\site" default="non" charte="oui"
ressource="non"/>
    <REPERTOIRECIBLE chemin="\site\html" default="oui" charte="non"
ressource="oui"/>
  </TRAITEMENTS>
</CHARTRE>
```

De ce fait, l'arborescence est modifiée. On a toujours un répertoire portant le nom de la charte, mais ce qui est généré n'est plus directement sous ce répertoire mais dans un sous-répertoire portant le



nom du traitement. L'arborescence d'une charte se présente comme ceci :

Il faut corriger la déclaration de la variable \$cheminSite dans les fichiers XSL concernés. Ainsi, dans les fichiers où la variable \$cheminSite est définie, il faut remplacer :


```
<xsl:variable name="cheminSite">
    <xsl:value-of select="concat($chainEditPath, '../site/html/')" />
</xsl:variable>
```

par :

```
<xsl:variable name="cheminSite">
    <xsl:value-of select="concat($chainEditPath, 'site/html/')" />
</xsl:variable>
```

4.3 Traitement Preview

4.4 Intégration

Pour intégrer la charte dans le nouveau ChainEdit, il faut aller dans la partie Administration / Gestion des configurations et cliquer sur l'icône  pour gérer les fichiers. En se positionnant sur le dossier à la racine, portant le nom de la charte, il est possible de télécharger le fichier compressé (attention à l'outil utilisé pour la compression) contenant tous les fichiers de la charte, puis de décompresser ce dernier.

From:
<https://wiki.univ-rennes1.fr/chainedit/> - chainedit wiki

Permanent link:
<https://wiki.univ-rennes1.fr/chainedit/doku.php?id=administrateurs:migration&rev=1297069622>

Last update: 2017/03/30 12:53

