



Architecture Applicative de l'Avenir

PLUGIN DE GESTION DE
PROJET ACUBE J2EE
(génération de DAO et
configuration)



Version 1.0 du 01/02/08

Etat : draft



SUIVI DES MODIFICATIONS

Version	Rédaction	Description	Vérification	Date
1.0	T.Rigal	Initialisation		20/12/07



SOMMAIRE

1	PRE-REQUIS	4
2	INSTALLATION	5
3	FONCTIONNALITES PRESENTES.....	6
4	CONFIGURATION DES PREFERENCES	7
4.1	Préférences : JDBCWrapper	8
4.2	Préférences : Tables de conversion	9
4.3	Préférences : Templates	10
5	CONFIGURATION DES PROPRIETES DU PROJET.....	11
5.1	Configuration du projet.....	11
6	UTILISATION DU PLUGIN.....	14
6.1	Génération de la chaîne DAO	14
6.2	Ajout de méthode à la chaîne DAO	15
7	GESTION DES PROPRIETES D'UN PROJET ACUBE J2EE	16
8	ANNEXES.....	18
8.1	Synoptique de la gestion de la chaîne DAO.....	18
8.1.1	Synoptique de la création de la chaîne DAO	18
8.1.2	Synoptique de la gestion des delegates de la chaîne DAO	18
8.2	fichier plugin.properties	19
8.2.1	fichier plugin.properties.....	19

TABLEAUX

Erreur ! Aucune entrée de table d'illustration n'a été trouvée.

FIGURES

Figure 1	: préférences du plugin de gestion de projet ACube	7
Figure 2	: préférences du JDBCWrapper	8
Figure 3	: préférences SQL/Java.....	9
Figure 4	: préférences - templates.....	10
Figure 5	: propriétés d'un projet Acube utilisées par le plugin.....	11
Figure 6	: source de données du projet, ici la base de données COLIMATIC_DVLT_01.....	12
Figure 7	: ajout d'une base de données	13
Figure 8	: génération de la chaîne DAO	14
Figure 9	: sélection des tables.....	14
Figure 10	: ajout d'une nouvelle méthode au delegate.....	15
Figure 11	: ajout de la méthode listById au delegate	15
Figure 12	: assistant de gestion des propriétés.....	16
Figure 13	: gestion des propriétés - configuration de Log4j	16



1 PRE-REQUIS

Le plugin nécessite l'utilisation d'Eclipse 3.1 (ou versions supérieures) et d'une JRE 1.5.



2 INSTALLATION

Pour installer le plugin il suffit d'extraire le contenu du fichier *fr.gouv.diplomatie.acube.manager_2.5.0.zip* dans le répertoire d'installation d'Eclipse (*%ECLIPSE_HOME%*).

Une fois extrait le répertoire suivant doit être présent :

%ECLIPSE_HOME%/plugins/ fr.gouv.diplomatie.acube.manager_2.5.0 .



3 FONCTIONNALITES PRESENTES

Ce plugin intègre les fonctionnalités suivantes :

- l'assistant de génération des classes BusinessDelegate, DAO, DAOJdbc, DAOTest, et VO, des fichiers dao.properties et sql.properties, à partir du schéma de la base de donnée. Les Dao générés contiennent des méthodes de création, mise à jour, lecture, suppression par clé primaire, liste, liste par clé étrangère, test d'existence par clé primaire,
- l'assistant de personnalisation d'une classe Business (ajout de nouvelles méthodes, avec impact sur toute la chaîne),
- la personnalisation des templates de génération,
- le paramétrage du mapping type SQL -> type java,
- la configuration de la récupération des données avec le JdbcWrapper suivant les types java
- la gestion de la configuration des fichiers de propriétés du projet ACube J2EE

4 CONFIGURATION DES PREFERENCES

Après installation le plugin est immédiatement opérationnel en utilisant les « templates » et les « mappings » java/sql définis par défaut. Cependant la gestion des « préférences » Eclipse, accessible par le menu Window > Préférences > Projet ACube –v2.5 permet (facultativement) :

- L'édition et la modification des « templates » utilisés pour générer la chaîne DAO,
- la configuration du « mapping » entre les types SQL et les type Java.

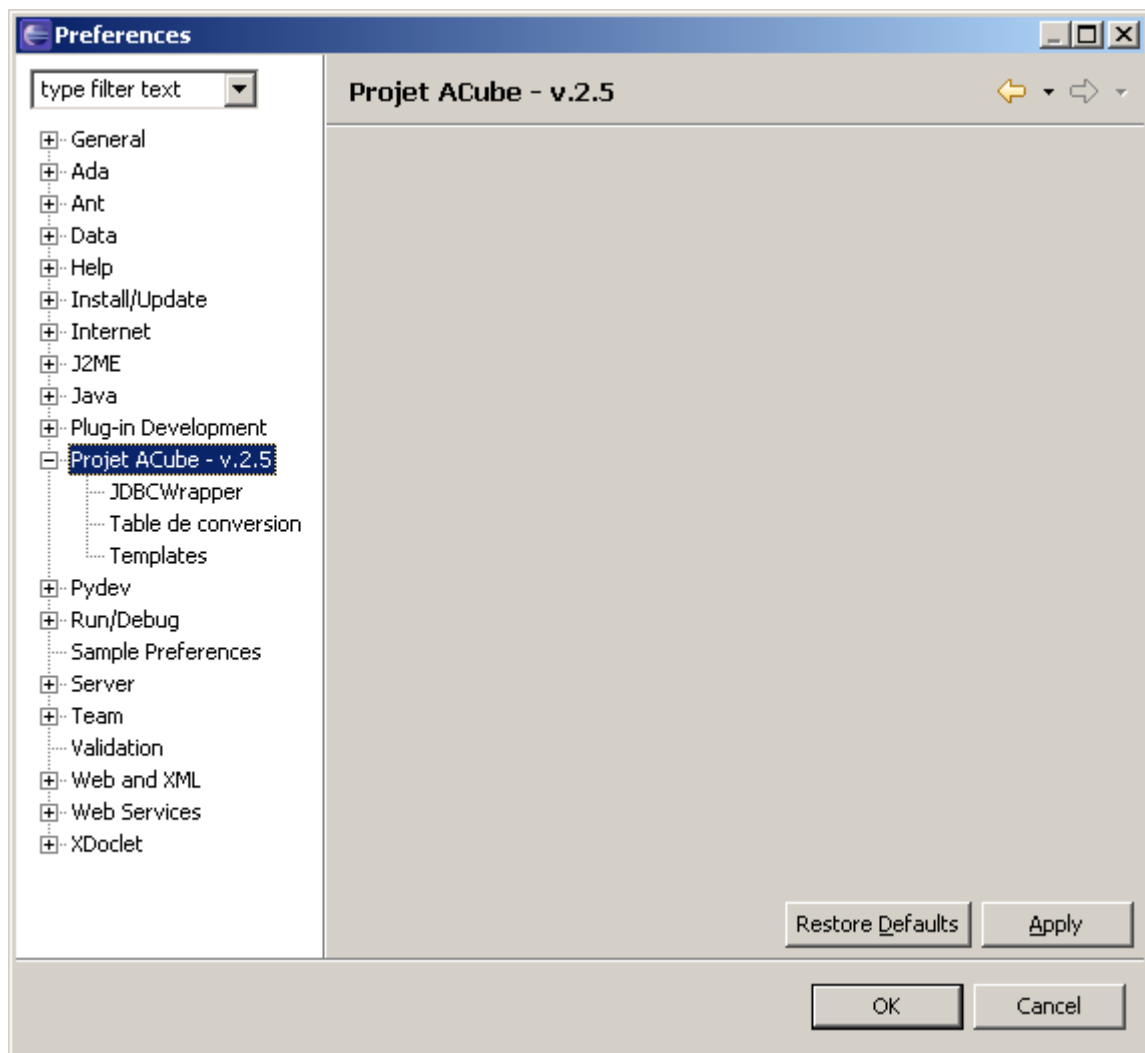


Figure 1 : préférences du plugin de gestion de projet ACube

4.1 PREFERENCES : JDBCWRAPPER

La gestion des préférences du JDBCWrapper fait apparaître un tableau à quatre colonnes :

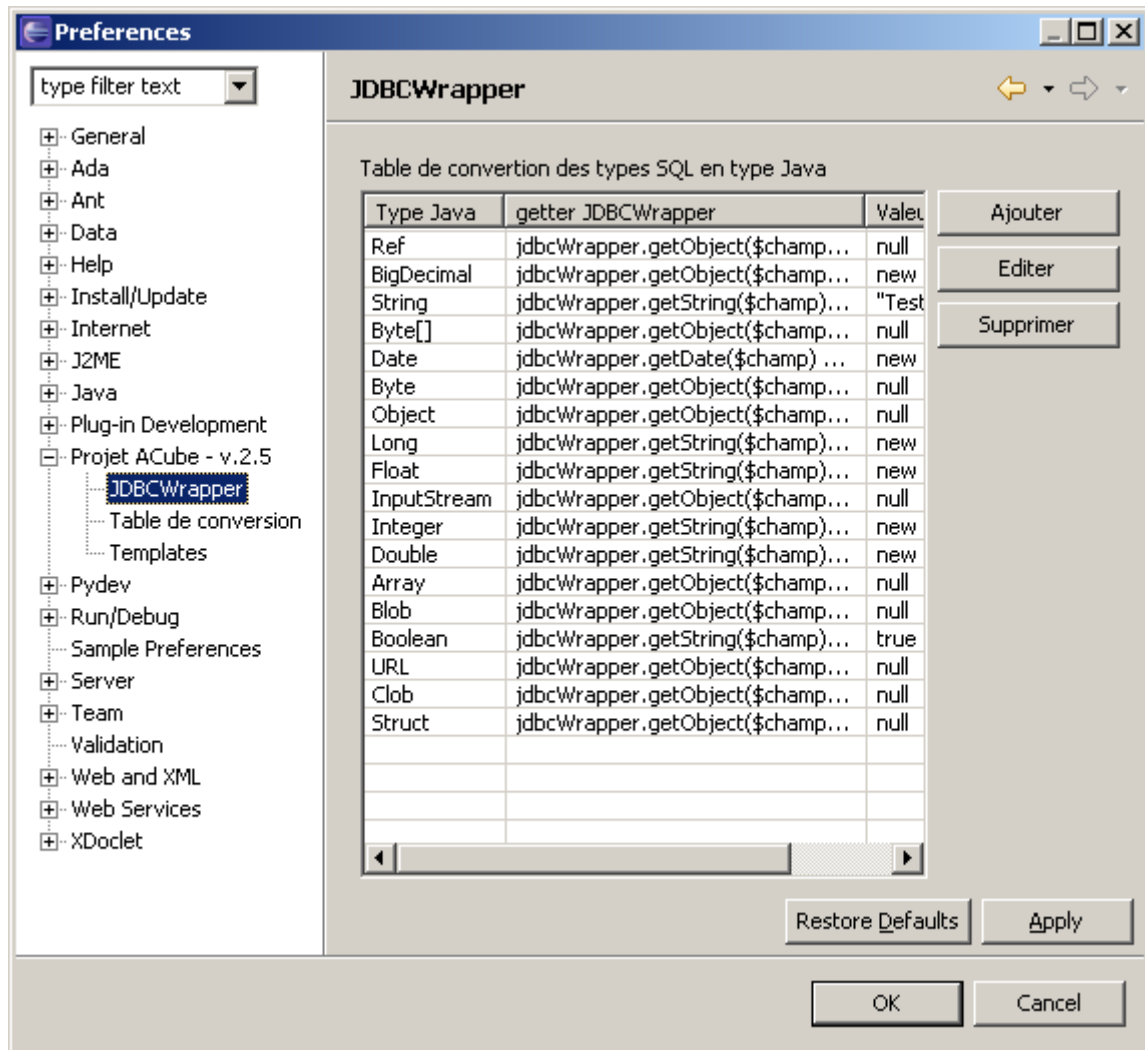


Figure 2 : préférences du JDBCWrapper

La gestion des préférences du JDBCWrapper permet de configurer le « mapping » entre les types SQL et les type Java.

4.2 PREFERENCES : TABLES DE CONVERSION

La gestion des préférences des tables de conversion fait apparaître un tableau à quatre colonnes :

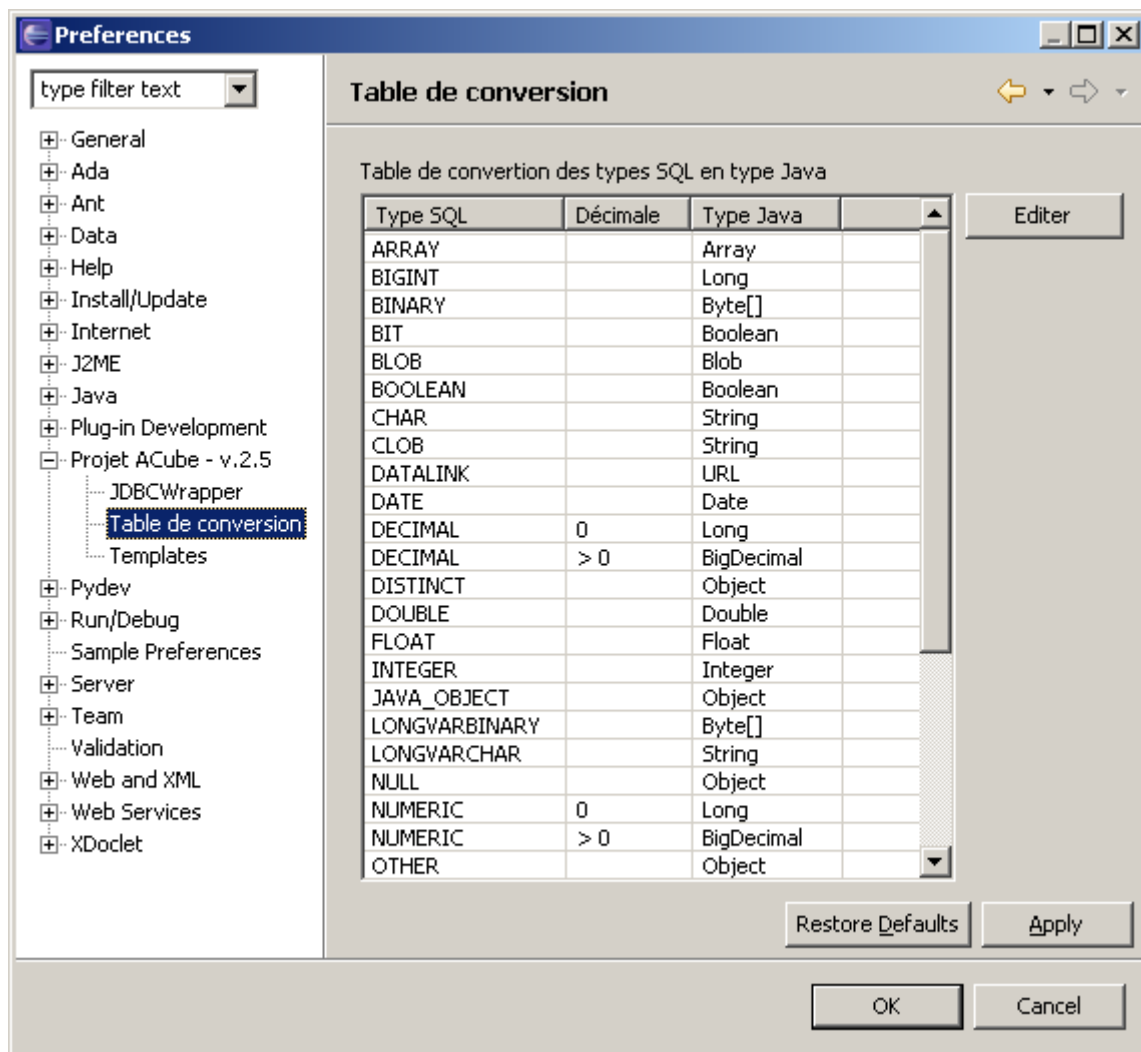


Figure 3 : préférences SQL/Java

4.3 PREFERENCES : TEMPLATES

Les templates sont utilisées par le moteur « Velocity » mis en œuvre par le plugin pour générer du code java ou des fichiers de configuration automatiquement.

La gestion des préférences des « templates » permet l'édition et la modification des « templates » utilisés pour générer ou modifier la chaîne DAO :

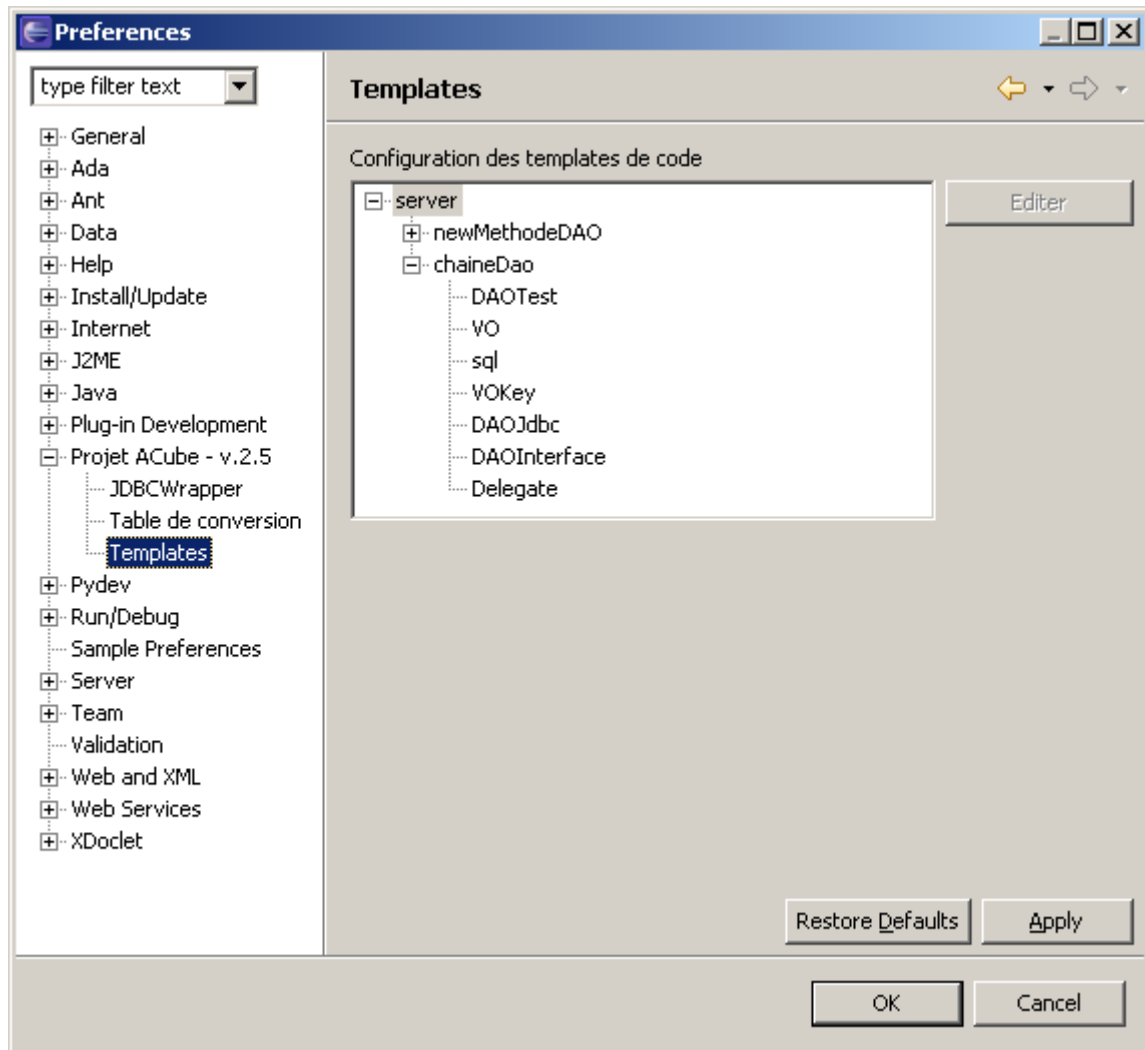


Figure 4 : préférences - templates

Les fichiers « template » utilisés par le moteur « Velocity » se trouvent dans le répertoire du plugin (%ECLIPSE_HOME%\plugins\fr.gouv.diplomatie.acube.manager_2.5.0) dans les répertoires \templates\chaineDao (templates de génération du DAO et \templates\newMethodeDAO (templates de modification des classes Business et Integration du DAO et de modification du fichier sql.properties).

5 CONFIGURATION DES PROPRIETES DU PROJET

5.1 CONFIGURATION DU PROJET

Ce chapitre explique rapidement comment utiliser les principales fonctionnalités du plugin. Avant toute utilisation il est nécessaire de configurer le plugin pour le projet ACube en cours. En vue Java dans la fenêtre d'exploration des projets :

- cliquer droit sur le projet ACube pour faire apparaître le menu contextuel du projet
- dans le menu contextuel sélectionner « Properties ».
- sélectionner la propriété « projet ACube », saisir les différents paramètres du projet. Par défaut, les données correspondent au gabarit initial d'un projet ACube J2EE.

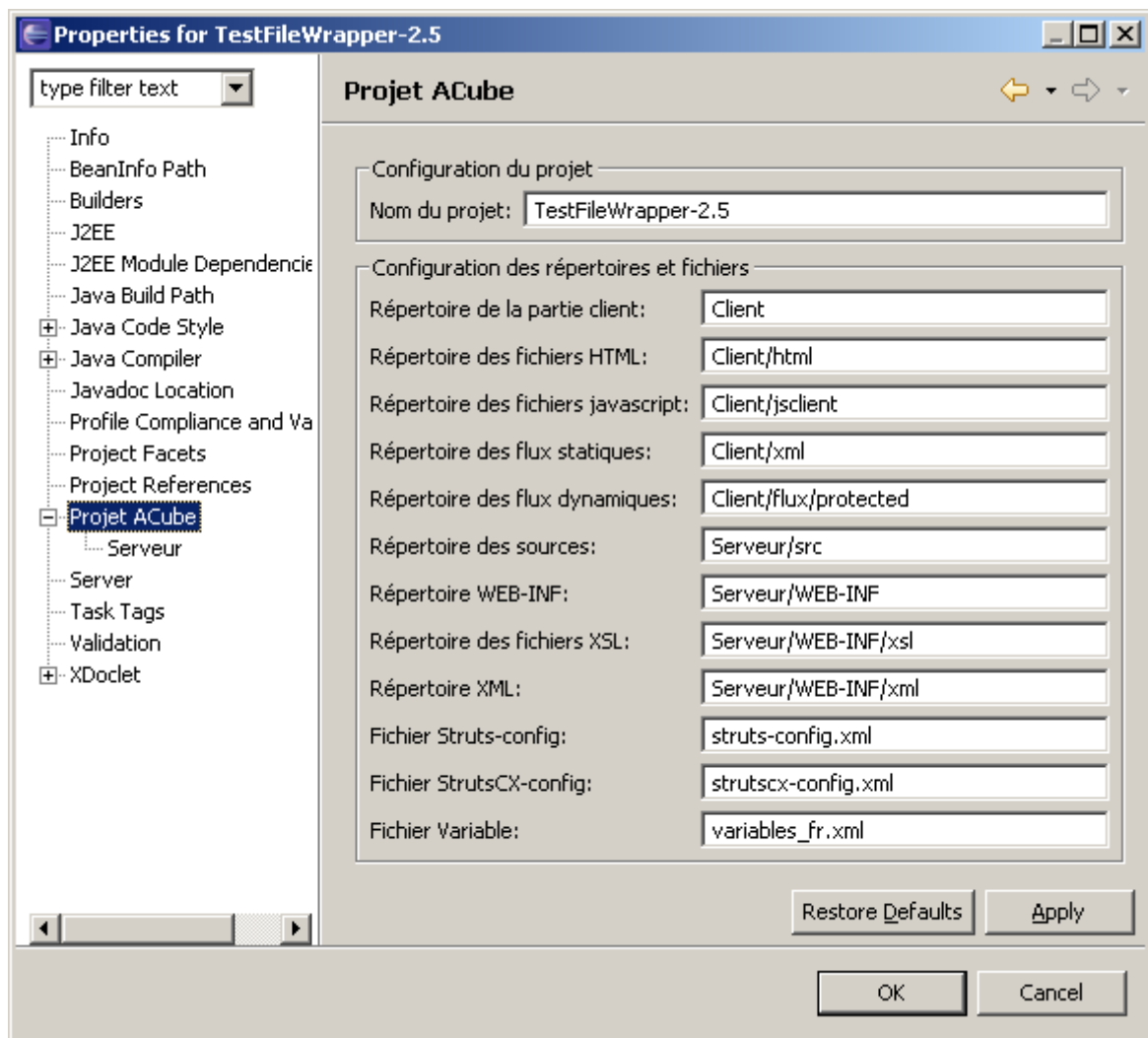


Figure 5 : propriétés d'un projet Acube utilisées par le plugin

Il faut ensuite ajouter une source de données au projet pour pouvoir récupérer le schéma de la base de données :

- cliquer sur le nœud Projet ACube et sélectionner « Serveur » sous la propriété Projet ACube.

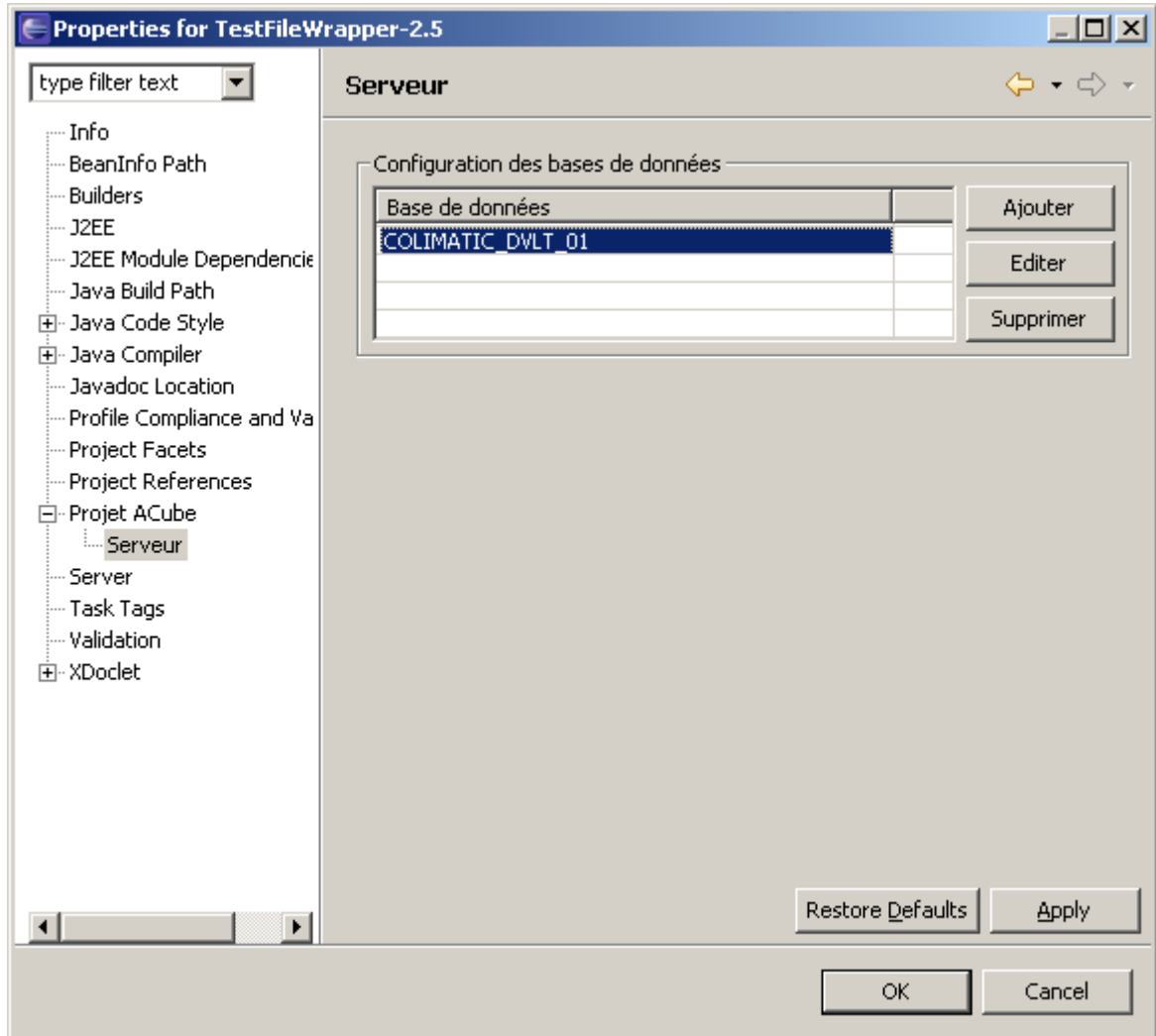


Figure 6 : source de données du projet, ici la base de données COLIMATIC_DVLT_01

- cliquer sur le bouton Ajouter,

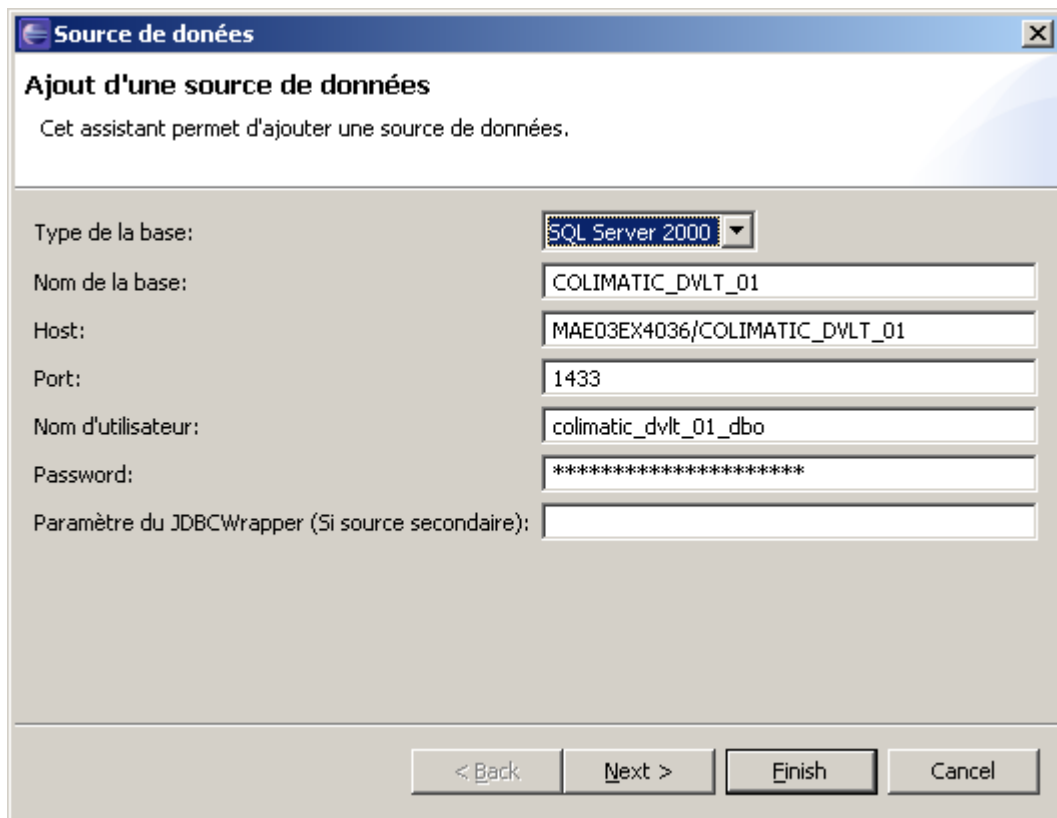


Figure 7 : ajout d'une base de données

- dans la fenêtre « Ajout d'une source de données » saisir les propriétés suivantes :

- **type de la base de données** : Type de serveur de base de données (SQL Server, Oracle, MySQL, Sybase)
- **nom de la base** : nom du schéma de la base de données
- **port** : Port d'accès à la base
- **nom d'utilisateur** : utilisateur pour se connecter à la base (**Attention** : l'utilisateur doit avoir des droit propriétaires pour pouvoir lire les méta données de la base)
- **password** : mot de passe de l'utilisateur
- **paramètre du JDBCWrapper** : paramètre nécessaire lorsque l'accès à la base de données est configuré dans une ressource JNDI et non pas par le fichier server.properties.

- cliquez sur « Finish ».

Le projet est maintenant configuré pour utiliser le plugin.

6 UTILISATION DU PLUGIN

6.1 GENERATION DE LA CHAÎNE DAO

Avant de pouvoir générer la chaîne DAO, les propriétés du projet doivent être renseignées (Voir §4.1)

- cliquer droit sur le projet, puis sur *Projet ACube > Générer la chaîne DAO*, Le plugin va alors se connecter à la base pour récupérer la liste des tables :



Figure 8 : génération de la chaîne DAO

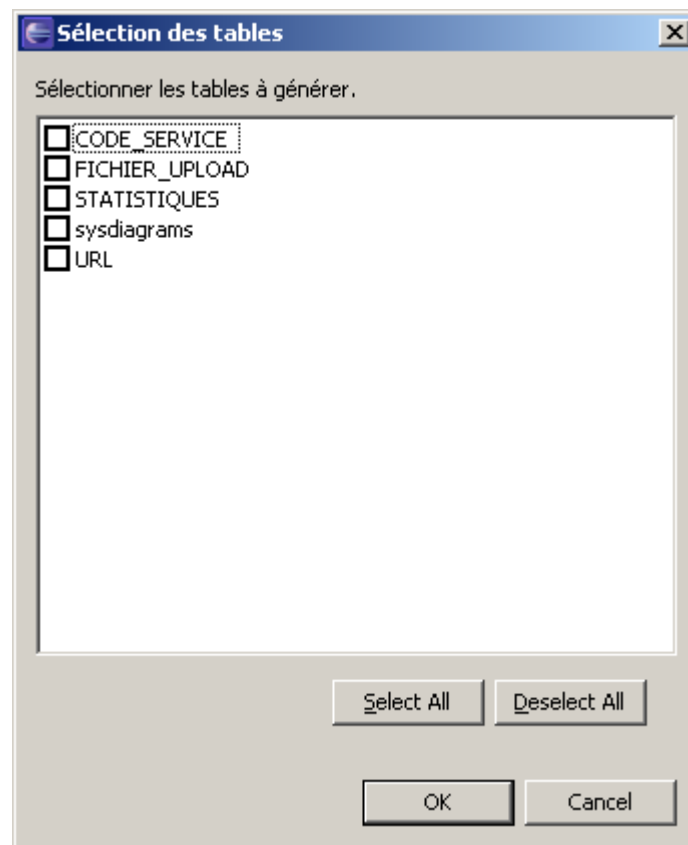


Figure 9 : sélection des tables

- sélectionner les tables dont vous voulez générer les DAO puis cliquez sur OK. Le plugin va récupérer les informations des tables sélectionnées et construire la chaîne DAO à partir de ces informations.

6.2 AJOUT DE METHODE A LA CHAINE DAO

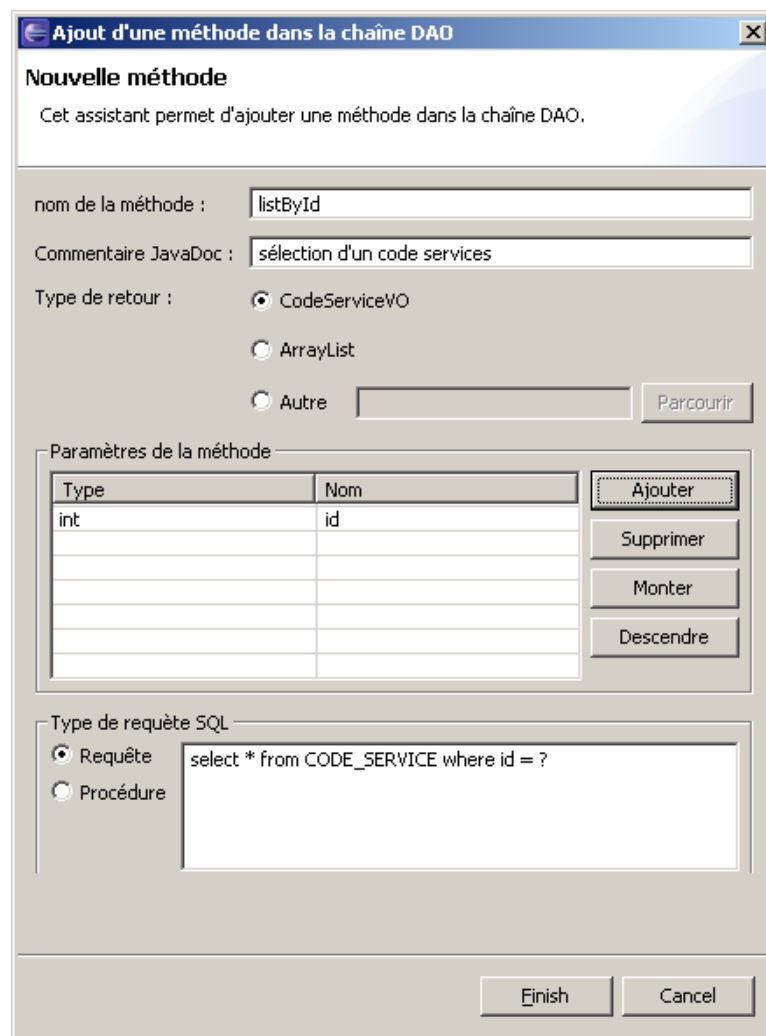
Avant de pouvoir ajouter une méthode à une classe *Business*, les propriétés du projet doivent avoir été renseignées (Voir §4.1) et les DAO doivent avoir été générés grâce au plugin (Voir §4.2) :

- cliquer droit sur *la classe Business* où vous souhaitez ajouter une méthode.
- cliquez sur *Projet ACube > Ajouter une méthode au delegate*. Saisir les différents champ puis cliquez sur OK. Le plugin va ajouter la méthode dans le fichier Delegate, DAO, DAOJdbv, DAOTest et la requête dans le fichier sql.properties :



Figure 10 : ajout d'une nouvelle méthode au delegate

Par exemple :



Ajout d'une méthode dans la chaîne DAO

Nouvelle méthode

Cet assistant permet d'ajouter une méthode dans la chaîne DAO.

nom de la méthode :

Commentaire JavaDoc :

Type de retour : CodeServiceVO
 ArrayList
 Autre

Paramètres de la méthode

Type	Nom
int	id

Type de requête SQL

Requête
 Procédure

Figure 11 : ajout de la méthode *listById* au delegate

7 GESTION DES PROPRIETES D'UN PROJET ACUBE J2EE

Les propriétés d'un projet ACube J2EE sont définies à travers un ensemble de fichiers génériques présents dans %SOURCES%\properties et livrés dans \WEB-INF\classes.

Les valeurs des propriétés génériques de ces fichiers de propriétés sont présentes dans le fichier build.project du projet ACube J2EE.

Le plugin permet de définir ces valeurs via un assistant de configuration :

- cliquer droit sur le projet ACube,
- sélectionner ProjetACube > Assistant de gestion des propriétés,



Figure 12 : assistant de gestion des propriétés

L'assistant de gestion des propriétés fait apparaître pour chaque fichier de propriétés des fenêtres de saisie :

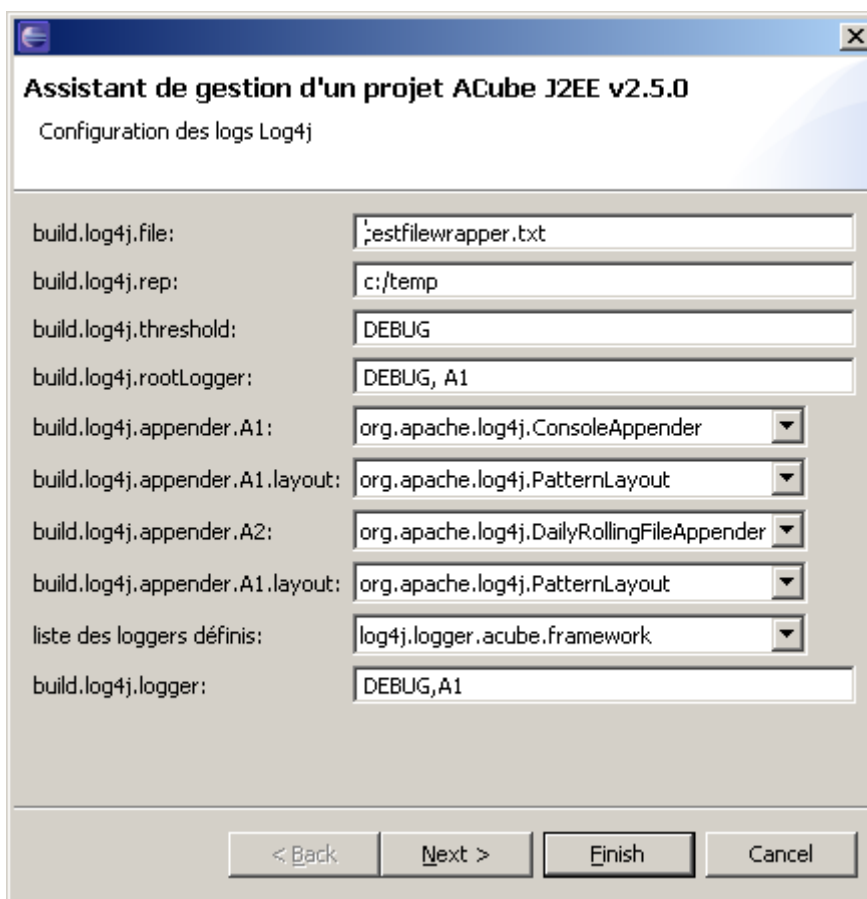


Figure 13 : gestion des propriétés - configuration de Log4j

Les fenêtres de l'assistant de gestion des propriétés concernent dans l'ordre d'affichage les composants suivants :



- configuration de log4j
- configuration du contexte (docBase et JDBCRealm)
- configuration de l'accès JDBC au serveur de base de données
- configuration du composant mail
- configuration des URLs d'authentification et déconnexion
- configuration du projet
- configuration du type de dao
- configuration du répertoire de Tomcat
- configuration des bibliothèques

N'est pas pris en compte : la configuration du mode de fonctionnement (cf build.project).

8 ANNEXES

8.1 SYNOPTIQUE DE LA GESTION DE LA CHAÎNE DAO

8.1.1 SYNOPTIQUE DE LA CREATION DE LA CHAÎNE DAO

Suite à l'action du menu « Générer la chaîne DAO » le code suivant est exécuté :

Dans la classe *fr.gouv.diplomatie.server.action.DAOGeneratorAction* :

- exécution de la méthode run
- rafraichissement du projet
- récupération des bases de données
- sélection de la base
- récupération de la liste des tables à générer
- génération de la chaîne DAO
- Instanciation d'un objet *fr.gouv.diplomatie.acube.server.dao.DAOGenerator*

dans la classe *fr.gouv.diplomatie.acube.server.dao.DAOGenerator*:

- Instanciation d'un contexte Velocity
- Création automatique des classes des packages Business, Integration et VO par un objet Velocity

8.1.2 SYNOPTIQUE DE LA GESTION DES DELEGATES DE LA CHAÎNE DAO

Suite à l'action du menu « Ajouter une méthode au Delegate » le code suivant est exécuté :

Dans la classe *fr.gouv.diplomatie.server.action.NewMethodAction* :

- exécution de la méthode run
- Instanciation d'un objet *fr.gouv.diplomatie.acube.server.dao.NewMethodWizard*

dans la classe *fr.gouv.diplomatie.server.dao.NewMethodWizard* :

- Instanciation d'un objet *fr.gouv.diplomatie.acube.server.dao.NewMethodGenerator*

dans la classe *fr.gouv.diplomatie.acube.server.dao.NewMethodGenerator*:

- Instanciation d'un contexte Velocity
- Modification automatique des classes des packages Business, Integration et du fichier sql.properties par un objet Velocity



8.2 FICHER PLUGIN.PROPERTIES

Ce fichier définit les valeurs par défaut des propriétés d'un projet ACube (telles que définies par le gabarit initial d'un project ACube J2EE).

8.2.1 FICHER PLUGIN.PROPERTIES

```
### Valeur par défaut des propriétés d'un projet A3 ###

a3.propriete.rep.client=Client
a3.propriete.rep.html=Client/html
a3.propriete.rep.js=Client/jsclient
a3.propriete.rep.fluxStat=Client/xml
a3.propriete.rep.fluxDyna=Client/flux/protected
a3.propriete.rep.src=Serveur/src
a3.propriete.rep.web.inf=Serveur/WEB-INF
a3.propriete.rep.xsl=Serveur/WEB-INF/xsl
a3.propriete.rep.xml=Serveur/WEB-INF/xml
a3.propriete.fichier.struts=struts-config.xml
a3.propriete.fichier.strutsCX=strutscx-config.xml
a3.propriete.fichier.variable=variables_fr.xml
a3.propriete.package.delegate=acube.projet.business
a3.propriete.package.dao=acube.projet.integration
a3.propriete.package.vo=acube.projet.vo
a3.propriete.class.logger=acube.framework.technical.log.BuildLogger
a3.propriete.fichier.daoProperties=dao.properties
a3.propriete.fichier.sqlProperties=sql.properties

#### Liste des templates du plugin ####

# Chaîne DAO #
##Delegate##
template.server.chaineDao.Delegate=templates/ChaineDAO/Delegate.java

##DAOInterface##
template.server.chaineDao.DAOInterface=templates/ChaineDAO/DAOInterface.java

##DAOJdbc##
```



```
template.server.chaineDao.DAOJdbc=templates/ChaineDAO/DAOJdbc.java

##DAOTest##
template.server.chaineDao.DAOTest=templates/ChaineDAO/DAOTest.java

##VO##
template.server.chaineDao.VO=templates/ChaineDAO/VO.java
template.server.chaineDao.VOKey=templates/ChaineDAO/VOKey.java

##SQL##
template.server.chaineDao.sql=templates/ChaineDAO/sql.sql

##Nouvelle méthode##
template.server.newMethodeDAO.sql=templates/newMethodeDAO/requete.sql
template.server.newMethodeDAO.DAO=templates/newMethodeDAO/DAO.java
template.server.newMethodeDAO.DAOJdbc=templates/newMethodeDAO/DAOJdbc.java
template.server.newMethodeDAO.DAOTest=templates/newMethodeDAO/DAOTest.java
template.server.newMethodeDAO.Delegate=templates/newMethodeDAO/Delegate.java
```