

# 6

## Conception du modèle de données avec le projet DTP (Data Tools Platform)

---

Ce chapitre présente l'outillage du projet Eclipse Data Tools Platform, centré sur la manipulation des structures de données, qui facilite la définition du modèle physique de la base de données.

DTP est un nouveau projet principal de la communauté Eclipse. Originellement proposé par Sybase en février 2005, il a entraîné dans son sillage une large et solide communauté et est actuellement géré par un comité comprenant Sybase, IBM et Actuate.

DTP est un framework orienté utilisateur centré sur les problématiques d'accès aux données (data centric), devenues incontournables dans le développement d'applications.

Le projet offre les avantages suivants :

- centralisation de la définition des connecteurs aux principales sources de données en un seul endroit, au niveau de l'instance de la plate-forme Eclipse ;
- centralisation de la définition des connexions et de leur création ;
- possibilité pour les outils d'accéder à un ensemble abstrait de connexions en masquant les détails de leur implémentation ;
- masquage de l'hétérogénéité des connexions et des outils pour chaque source de données ;
- possibilité de partager les définitions de connexions, facilitant l'intégration de la solution au niveau de la plate-forme Eclipse et l'utilisation.

## Architecture de DTP

Fondé sur le framework EMF, DTP consiste en un modèle de base, des connecteurs et des outils SQL pour l'interrogation et la manipulation des données. Il propose une architecture de conception orientée modèle (Model Driven Design) qui supporte les principales bases de données relationnelles du marché.

Les sections qui suivent détaillent les trois sous-projets clés qui constituent l'ossature du projet DTP. La version que vous utiliserez est la celle livrée en bundle avec Europa (DTP 1.5).

### *Sous-projets de la plate-forme DTP*

DTP (<http://www.eclipse.org/datatools/>) est le projet central, autour duquel gravite les sous-projets Model Base, CMF (Connection Management Framework) et SQL Development Tools.

#### **Model Base**

Ce projet fournit les fondations pour le développement de la couche d'accès aux données. Il utilise une approche orientée modèle avec UML, fondée sur le framework EMF (Eclipse Modeling Framework).

Il inclut les composants suivants :

- modèle de définition de donnée ;
- éditeur de modèle ;
- modèle SQL ;
- modèle d'interrogation SQL (spécifications SQL 99/03) ;
- modèle de requête SQL/XML.

Les principales caractéristiques de ce sous-projet sont les suivantes :

- gestionnaire de version, les modèles pouvant être aisément versionnés à travers le composant de gestion de versions intégré à Eclipse ;
- support d'un éditeur de modèle intégré visuel avec EMF ;
- support de javadoc, les modèles pouvant être publiés et documentés ;
- extensibilité, les modèles étant extensibles et n'étant pas dépendants d'un système de gestion de données propriétaire ;
- standardisation, les modèles étant compatibles avec les standards existants, comme SQL et JDBC.

#### **CMF (Connectivity Management Framework)**

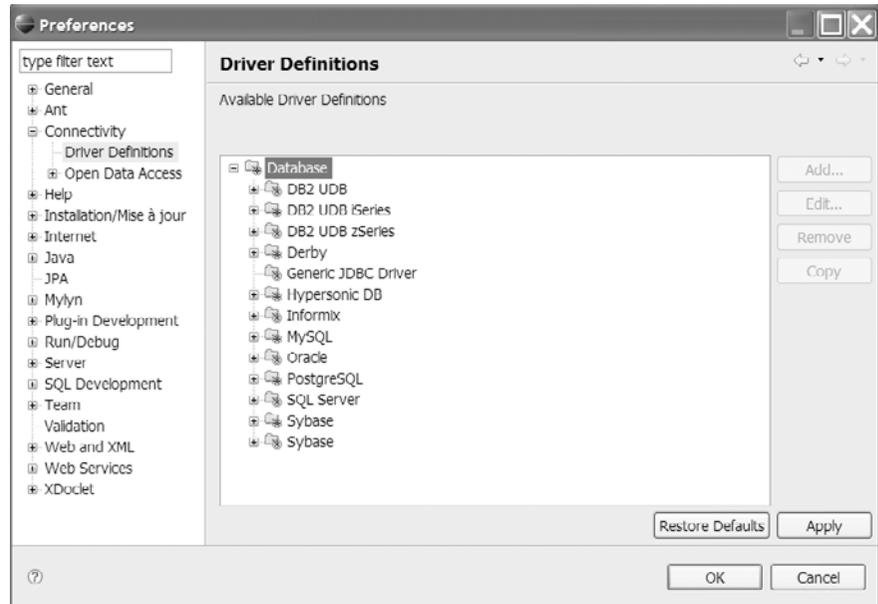
Le projet CMF fournit les ressources pour la gestion des connexions aux diverses sources de données JDBC ainsi que l'outillage de support approprié pour l'utilisation des connexions.

Le framework pour la définition ou la création de drivers est fondé sur un certain nombre de modèle proposés par défaut (d'autres modèles peuvent être créés par le biais de points d'extension DMF).

Une fonctionnalité clé du sous-projet connecteur est la possibilité offerte aux utilisateurs de saisir une fois leur information et de la réutiliser à travers l'ensemble des outils d'Eclipse. Conçu sur le modèle de DMF, le framework CMF définit des connexions spécifiques à travers la configuration des données et fournit des instances de connexions.

La figure 6.1 illustre les définitions de drivers accessibles par le biais du menu Préférences d'Eclipse.

**Figure 6.1**  
*Définitions de drivers sous Eclipse*



Le projet CMF propose également dans la vue Eclipse un explorateur de sources de données sous la forme d'instances de profils de connexion, ou CP (Connection Profile), associé à un framework d'accès aux données, ou ODA (Open Data Access). Cela permet aux applications d'accéder aux données à partir de sources de données standards et personnalisées.

ODA permet la connectivité des données entre les consommateurs et fournisseurs de données à travers des interfaces d'exécution en fournissant une abstraction à travers le concept de donnée, de la même manière que CMF fournit une abstraction à travers la notion de connexion à ces mêmes données.

### SQL Development Tools

Ce projet fournit les frameworks et outils permettant de travailler avec SQL et d'étendre les fonctions des outils d'interrogation. L'éditeur de routines et le débogueur de routines (Routine Editor et Routine Debugger) fournissent un moyen d'interagir avec les procédures stockées et les fonctions.

Le parseur de requêtes SQL permet aux outils SQL d'utiliser un arbre d'analyse associé à la requête SQL. De son côté, le framework SQL Execution Plan permet une analyse fine et le tuning des requêtes SQL. Il est également possible d'exécuter des procédures stockées.

## Mise en œuvre de DTP

Après cette rapide présentation du projet DTP, vous allez mettre en œuvre l'outillage du projet DTP à travers l'étude de cas webstock.

Le script de création du modèle webstock figure sur la page Web dédiée à l'ouvrage pour les cibles Apache Derby et Hypersonic, SGBD que nous supposons installé et configuré (voir en annexe pour les détails d'installation et de configuration). L'outillage DTP (version DTP 1.5), qui n'est pas livré en standard dans la plate-forme Europa, doit être correctement installé et mis à jour par le biais du gestionnaire de mises à jour Eclipse.

La figure 6.2 illustre l'état de la configuration de l'installation après mise à jour de l'outillage DTP.

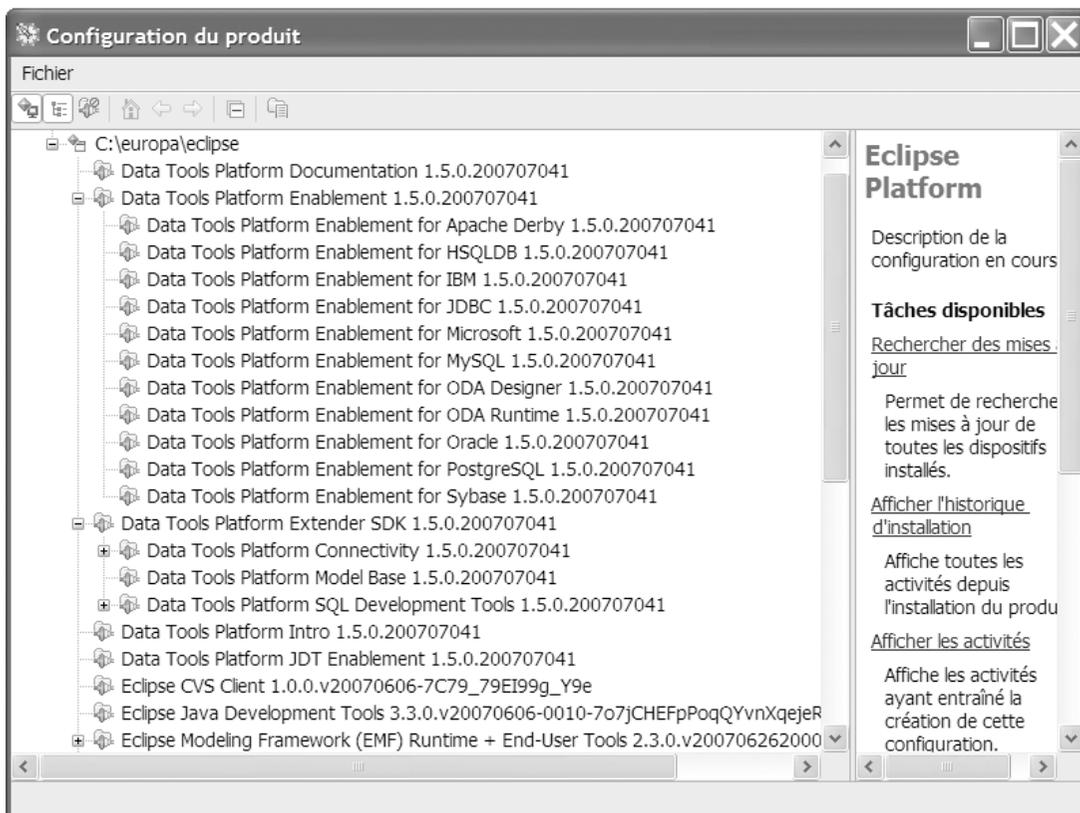


Figure 6.2

Configuration de l'installation après mise à jour de l'outillage DTP

### Connexion à la base Apache Derby

La première chose à effectuer pour la mise en route de DTP est la configuration de la connexion à la base cible. Nous nous fondons ici sur la base Derby, une base de données relationnelle Java qui offre la particularité de pouvoir être embarquée dans des applications Java.

Derby est le nom de l'implémentation Open Source du SGBD dérivée d'IBM Cloudscape, la version commerciale de la base d'IBM.

Vous utiliserez essentiellement cette base pour vos tests et votre développement, étant entendu que, pour des déploiements finaux en production, il est recommandé d'utiliser des bases telles que MySQL 5 ou équivalentes.

L'installation et la prise en main de Derby ne présentent pas de réelles difficultés non plus que sa configuration. Nous supposons que le répertoire d'installation de la base Derby (désigné par la variable d'environnement DERBY\_HOME) se trouve sous C:\Tools\db-derby-10.2.2.0-bin.

1. Positionnez dans la variable classpath de votre installation les bibliothèques derby.jar et derbytools.jar sous le répertoire lib de DERBY\_HOME.
2. Lancez la commande ij pour créer la base de test webstockdb (voir figure 6.3).

```
C:\Tools\db-derby-10.2.2.0-bin>java -Dij.protocol=jdbc:derby:org.apache.derby.tools.ij
ools.ij
version ij 10.2
ij> connect 'webstockDB;create=true';
ij>
```

Figure 6.3

*Commande interactive ij sous DOS*

3. La commande ij étant une application Java, vous devez lancer une JVM et indiquer le nom complet de la classe principale que vous souhaitez exécuter (ici l'outil ij). En cas d'exception, vérifiez que le fichier jar derbytools.jar est présent. La commande connect permet la connexion à la base de données, en l'occurrence ici la chaîne, ou URL, de connexion `jdbc:derby:webstockDB;create=true`. La propriété `create=true`, passée au driver JDBC Derby, spécifie la création de la base si celle-ci n'existe pas.
4. Quittez la session de connexion à la base Derby en entrant exit.
5. Lancez la commande dir sous Windows ou ls sous UNIX. Vous devez constater qu'un nouveau répertoire webstockdb a été créé, ainsi que les sous-répertoires webstockdb/log et webstockdb/seg0, qui contiennent les données de la base webstockdb.

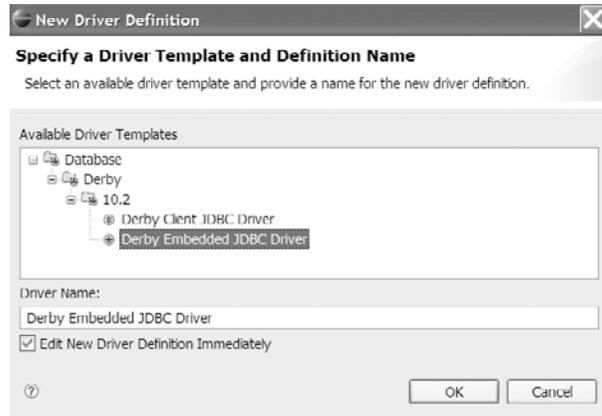
#### **Copie de la base de données**

Pour créer une copie de votre base de données Derby, vous devez simplement stopper l'instance Derby, effectuer une copie du répertoire et des sous-répertoires (ici webstockdb) et relancer le serveur. La restauration de la base est aussi simple : stoppez le serveur, puis remplacez la structure de la base avec la copie restaurée, et relancez le serveur.

6. Revenez sur l'IDE Eclipse, et cliquez sur le bouton Add après avoir sélectionné dans le menu Preferences d'Eclipse la définition du driver Derby 10.2.
7. Sélectionnez l'option Derby Embedded Driver (voir figure 6.4).

Figure 6.4

Définition du driver  
Derby avec DTP

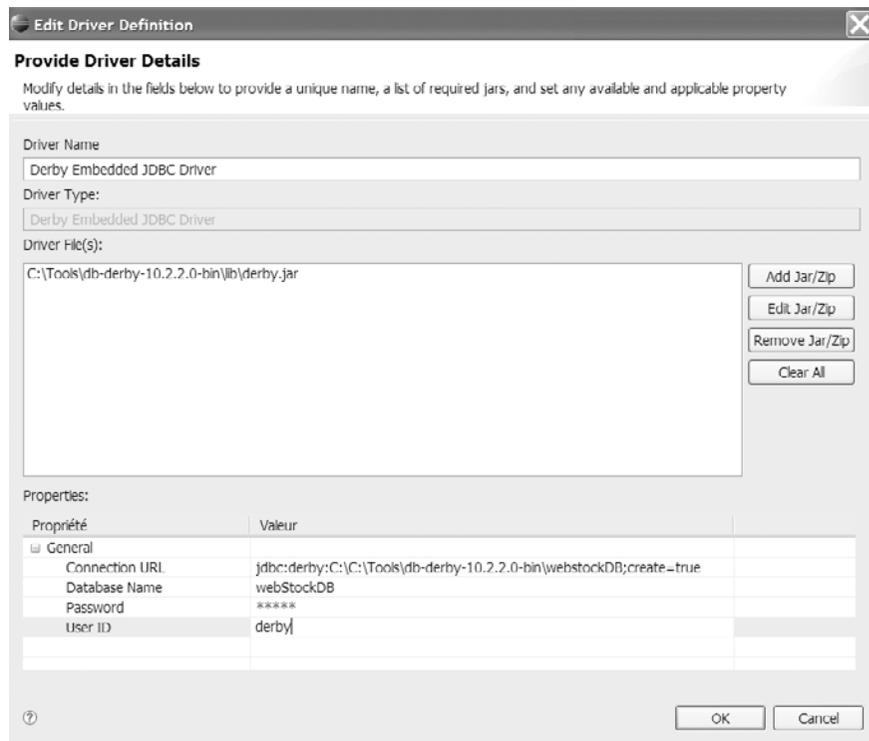


8. Supprimez le fichier driver derby.jar, et ajoutez la définition des drivers suivants à partir de l'emplacement `DERBY_HOME\lib\` (voir figure 6.5) :

- Connection URL
- Database Name : webstockdb
- Password : admin
- User Id : derby

Figure 6.5

Configuration  
du driver Derby  
avec DTP



9. Cliquez sur OK et fermez la boîte de dialogue Preferences d'Eclipse.

## Création d'un profil de connexion

Une fois effectuée la définition du driver Apache Derby, vous pouvez utiliser cette définition comme base pour la création d'un profil de connexion.

1. Ouvrez la perspective Database Development (via Window, Open Perspective et Other). Le bureau Eclipse doit ressembler à celui illustré à la figure 6.6.

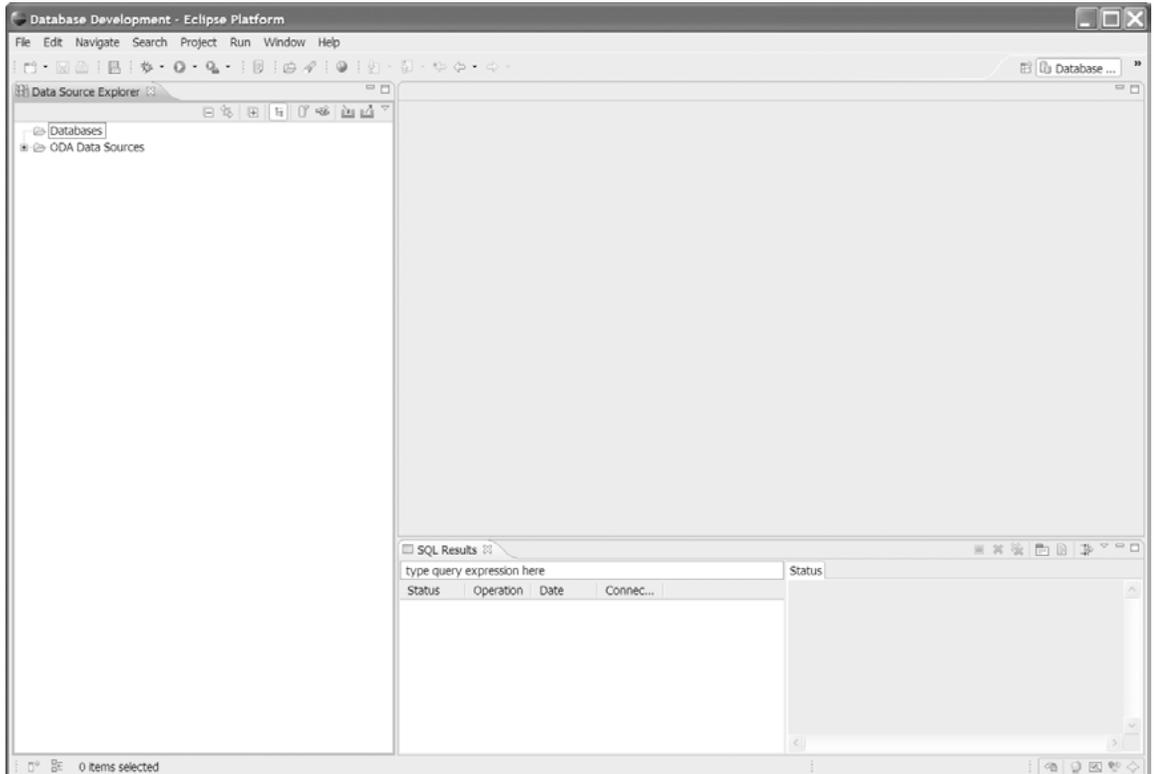


Figure 6.6

Création d'un profil de connexion avec DTP (1/3)

2. L'explorateur de sources de données Eclipse est accessible en sélectionnant Databases puis New.

Figure 6.7

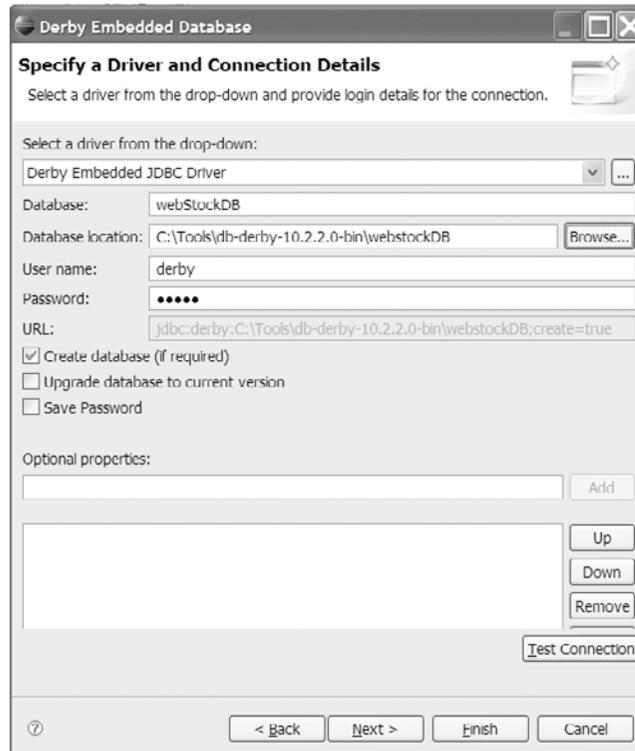
Création d'un profil de connexion avec DTP (2/3)



3. Sélectionnez Derby Embedded Database, puis cliquez sur Next.
4. Donnez un nom au profil de connexion (par exemple MyDerby) et optionnellement une description, puis cliquez sur Next (voir figure 6.8).

**Figure 6.8**

*Création d'un profil de connexion avec DTP (3/3)*



5. Cliquez sur Finish pour terminer la création du profil de connexion.

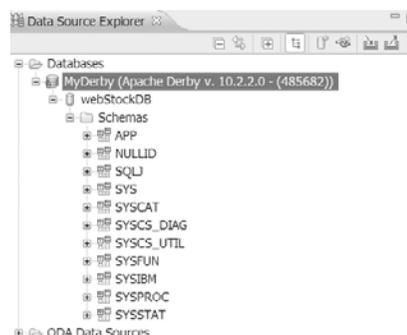
Vous êtes prêt à vous connecter à l'instance Apache Derby.

6. Développez le nœud Databases dans la vue Data Source Explorer, puis sélectionnez le profil de connexion que vous venez de créer par clic droit, et sélectionnez Connect.

Après quelques secondes, vous devriez voir apparaître le profil de connexion développé avec le contenu de la base Derby webstockdb vide (voir figure 6.9).

**Figure 6.9**

*La vue explorateur de source de données DTP*



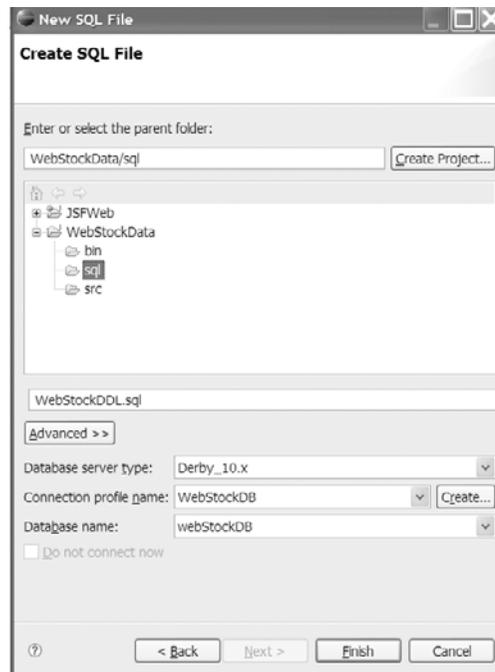
## Script SQL de création du schéma de base de données

À ce stade, vous allez créer un script SQL et l'exécuter sur l'instance webstockdb que vous venez de créer. Pour ce faire, vous allez créer un projet pour stocker le fichier SQL et l'associer au profil SQL.

1. Sélectionnez File, New, Project et Java Project à partir de la liste des projets disponibles, puis cliquez sur Next.
2. Saisissez webstockdata comme nom de projet, et cliquez sur Finish. Votre projet doit apparaître dans la vue Navigateur de l'IDE Eclipse.
3. Créez un dossier (*via* les options du menu contextuel Eclipse New et Folder) SQL pour contenir le fichier de script SQL webstockddl.sql.
4. Faites un clic droit sur le projet créé, et sélectionnez New, Other, SQL Development et SQL File, puis cliquez sur Next.
5. Remplissez les champs suivants (voir figure 6.10) :
  - Nom du fichier SQL : webStockDDL
  - database server type : Derby\_10.x
  - Connection profile name : WebStockDB
  - Database name : webStockDB

Figure 6.10

Création du fichier SQL avec l'outillage DTP



6. Cliquez sur Finish.

7. Faites un copier-coller du contenu du fichier script de création du schéma de la base webStock dans le fichier webStockDDL.sql *via* l'éditeur de script associé.

8. Cliquez sur Execute All pour lancer l'exécution du script. Si tout se passe bien, vous devez voir s'afficher dans l'explorateur de source de données les tables illustrées à la figure 6.11.

Figure 6.11

*Vue explorateur après création du schéma de la base webstock*



ARTICLEID [VARCHAR(5)]	NOMARTICLE [VARCHAR(30)]	ARTICLECATEGORIEID [VARCHAR(5)]	FOURNISSEURID [VARCHAR(5)]	DESCRIPTION [VARCHAR(50)]	POIDS [DECIMAL(10, 2)]	IMAGE_URL [VARCHAR(50)]
AR001	ABIT AG8 (Intel®I5P)	CAT03	FOU01	CARTE MERE ABIT SOCKET ...	125,00	abitag8.gf
AR002	ASUSTEK A7N8X	CAT03	FOU01	CARTE MERE ASUSTEK DE T...	110,00	asusteka7n8x.gf
AR003	HITACHI 120 GO 7200 RPM	CAT04	FOU02	DISQUE DUR HITACHI 120 G...	150,00	hitachi120RPM2004.gf

Figure 6.12

*Vue SQL Results après exécution d'une requête SQL*

### Vue SQL Results

Utilisez la vue SQL Results de l'outillage DTP pour afficher l'état des requêtes lors de leur exécution. Cette vue (qui s'obtient via Window, Show View, Other puis SQL Development SQL Results du menu Eclipse) affiche le résultat de l'instruction SQL courante sous forme tabulaire (statut de la requête, opération, date, profil de connexion).

9. Refaite les même étapes avec le script de peuplement de la base webstockdata disponible sur la page Web associée à l'ouvrage.
10. Vous pouvez afficher le contenu et extraire les données provenant des tables de la base par le biais de la vue Data Source Explorer.
11. Faites un clic droit, et sélectionnez les options contextuelles Data et Sample Contents pour l'affichage ou Data et Extract pour une extraction des données Vous pouvez également éditer les données affichés par le biais des options Data et Edit du même menu contextuel.

La figure 6.13 illustre l'affichage du contenu de la table Article.

ARTICLEID [VARCHAR(5)]	NOMARTICLE [VARCHAR(30)]	ARTICLECATEGORIEID [VARCHAR(5)]	FOURNISSEURID [VARCHAR(5)]	DESCRIPTION [VARCHAR(50)]	POIDS [DECIMAL(10, 2)]	IMAGE_URL [VARCHAR(50)]
AR001	ABIT AG8 (Intel @15P)	CAT03	FOU01	CARTE MERE ABIT SOCKET ...	125,00	abitag8.gf
AR002	ASUSTEK A7N8X	CAT03	FOU01	CARTE MERE ASUSTEK DE T...	110,00	asusteka7n8x.gf
AR003	HITACHI 120 GO 7200 RPM	CAT04	FOU02	DISQUE DUR HITACHI 120 G...	150,00	hitachi120RPM2004.gf

**Figure 6.13**

*Affichage du contenu de la table Article*

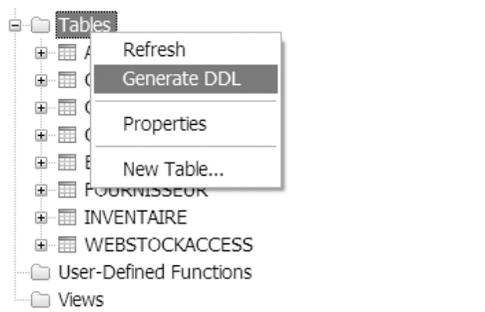
Pour insérer une ligne dans la table, il suffit de sauvegarder l'éditeur en sélectionnant File puis Save (ou Ctrl+S). La vue SQL Results affiche le résultat de l'insertion.

## Génération du DDL

Vous pouvez générer le DDL (Data Definition Language) à partir du modèle déjà généré en sélectionnant le dossier Tables du schéma webstock dans la vue explorateur de données et en cliquant sur Generate DDL dans le menu contextuel (voir figure 6.14).

**Figure 6.14**

*Génération du script DDL à partir du schéma de la base*



## En résumé

Vous en avez terminé avec les étapes de création de la base webstock qui sert de cadre à l'ouvrage.

Vous pouvez à présent vous concentrer sur le développement de la partie Web de l'étude de cas en utilisant les assistants et outils du projet Web Tools.

