

TP Conception de Datawarehouse

Initiation à ORACLE WAREHOUSE BUILDER

Cédric du Mouza

Ce TP s'appuie sur le tutorial de prise en main de Oracle Warehouse Builder, que vous pouvez trouver en anglais sur le site d'Oracle.

1 Se connecter

La procédure pour lancer oracle warehouse builder (OWB) est la suivante, à partir d'une station linux:

1. se connecter au serveur *kirov* soit en le sélectionnant dans la liste des serveur si elle apparaît sur le terminal, soit en tapant dans une console une fois connecté sur n'importe quel serveur:

```
ssh -X kirov
```

En cas d'échec vous pouvez aussi essayer depuis une console sous un autre serveur faire::

```
telnet kirov, puis une fois connecté,  
export DISPLAY=nom_de_votre_terminal:0.0
```

Cela permet d'exporter l'affichage et donc d'afficher ce que vous lancer sur le serveur *kirov* sur l'écran que vous utilisez.

2. une fois sur kirov, fixer le nom de la base que vous souhaitez "attaquer" en tapant:

```
export ORACLE_SID=NFE115
```
3. placez vous dans le repertoire de owb avec la commande

```
cd /opt/oracle/product/11.1.0/db_1/owb/bin/unix
```
4. créer un référentiel
 - (a) lancer `./reposinst.sh`, puis laisser les valeurs par défaut (localhost/1521/NFE115);
 - (b) à l'étape 2 choisir "gérer les espaces de travail Warehouse Builder";
 - (c) à l'étape 3, choisir "Créer un nouvel espace de travail Warehouse Builder", puis "Créer un espace de travail ayant pour propriétaire un nouvel utilisateur" (étape 4);
 - (d) me demander le mot de passe SYSTEM (ou SYS). Ensuite à l'écran suivant (étape 6) entrer votre login suivi d'un 1 et mot de passe. Choisir comme espace de travail l'espace "<monLogin>Espace" (voir figure 1);

Assistant Référentiel - Etape 6 sur 10 : Propriétaire d'espace de travail (nouveau)

Créez le nouvel utilisateur de base de données devant être le propriétaire de l'espace de travail. Entrez un nom utilisateur, un mot de passe utilisateur et un nom d'espace de travail.

Nom utilisateur du propriétaire d'espace de travail :

Mot de passe du propriétaire d'espace de travail :

Confirmation du mot de passe du propriétaire d'espace de travail :

Nom d'espace de travail :

Cliquez sur Suivant pour continuer.

Aide < Précédent Suivant > Fin Annuler

Figure 1: Création d'un espace de travail

Création d'un utilisateur de base de données

Saisissez les informations sur le nouvel utilisateur de base de données. Ce nouvel utilisateur de base de données peut être inscrit en tant qu'utilisateur d'espace de travail Warehouse Builder.

Nom utilisateur :

Mot de passe :

Ressaisir le mot de passe :

Nom utilisateur DBA :

Mot de passe d'utilisateur DBA :

Aide OK Annuler

Figure 2: Création d'un utilisateur de l'espace de travail

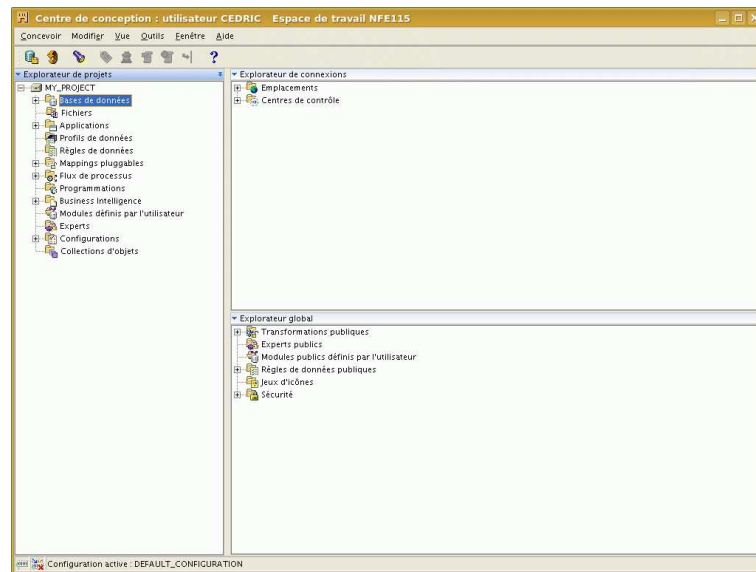


Figure 3: Accueil de OWB

- (e) Vous arrivez au choix des utilisateurs du référentiel. Faire alors “Créer un nouvel utilisateur” et créer un utilisateur avec le login “VotreLogin2” par exemple et pour mot de passe votre mot de passe (figure 2). Valider. La création peut prendre un certain temps.

5. une fois le repository créé, lancer l’outil à partir de la console linux avec la commande:

owbclient.sh

6. vous connecter avec votre login suivi de 1 et de votre password habituel lorsque la fenêtre apparaît sur le référentiel portant votre nom (ex monnom_c1/P123456C pour le login monnom_c)

Vous êtes maintenant avec une session de OWB ouverte comme celle de la figure 3 et le TP peut commencer.

2 Créer les modules

1. cliquer sur le menu **Concevoir** et sélectionner **Nouveau**
2. dans la fenêtre apparaissant, donner au projet le nom “votreLogin_REVIEW” et appuyer sur OK
3. dérouler le projet “votreLogin_REVIEW” et faire un clic-droit sur **Oracle** dans **Bases de Données**. Sélectionner **nouveau**.
4. Donner au module le nom “REVIEW_TGT” avec la description “Module Cible contenant les objets de 'review' ”. Il s’agira d’une cible de data warehouse. Faire **Suivant** en laissant les options par défaut.

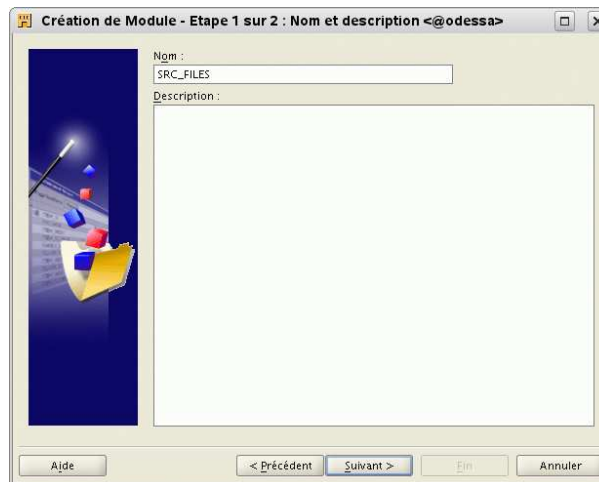


Figure 4: Créer le module SRC_FILES

3 Importer des sources

Nous allons ici considérer 2 types de sources:

- des fichiers “plats” de données (fichiers texte classiques)
- des bases de données, en l’occurrence ici une base Oracle

Comme les données et structures existent déjà, nous allons regarder comment les importer.

3.1 Importer des méta-données d’un fichier texte

La première étape, afin d’obtenir les métadonnées de ce fichier texte, est de créer un *module* dans le projet PRODUCT_REVIEW (celui par défaut) ou un nouveau que vous auriez créé grâce à l’onglet Créer>nouveau projet. Un module est un regroupement logique d’un ensemble d’objets. Le module contiendra les méta-datas spécifiques à cet ensemble d’objets et est par conséquent situé dans la section “Fichier” de OWB.

Pour créer un module pour un fichier “plat”:

1. dérouler le projet votreLogin_REVIEW
2. faire un click-droit sur le nœud Fichiers et sélectionner Nouveau
3. Utiliser l’aide pour spécifier le nom du module: SRC_FILES (par exemple, cf figure 4) puis passer à la fenêtre suivante
4. pour cette 2ème fenêtre, cliquer sur modifier l’emplacement. Vous arrivez à une nouvelle fenêtre. Préciser le chemin où se trouve le fichier texte de données:
/opt/oracle/product/11.1.0/db_1/owb/newowbdemo/sourcefiles (cf figure 5)
5. donner un nom à ce nouvel emplacement, par exemple LOC_FILES
6. cliquer sur ok

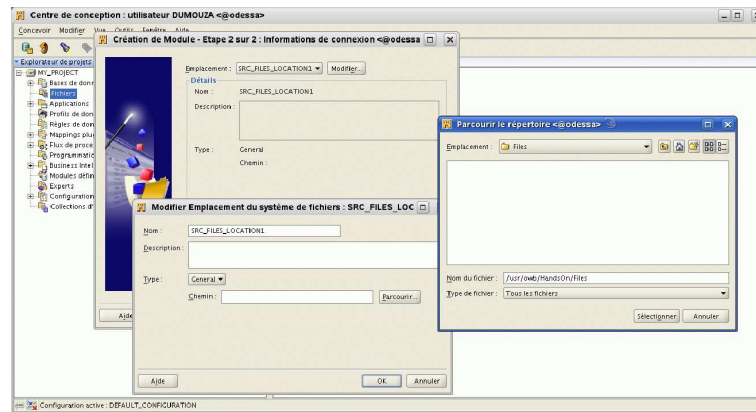


Figure 5: Créer une nouvelle localisation

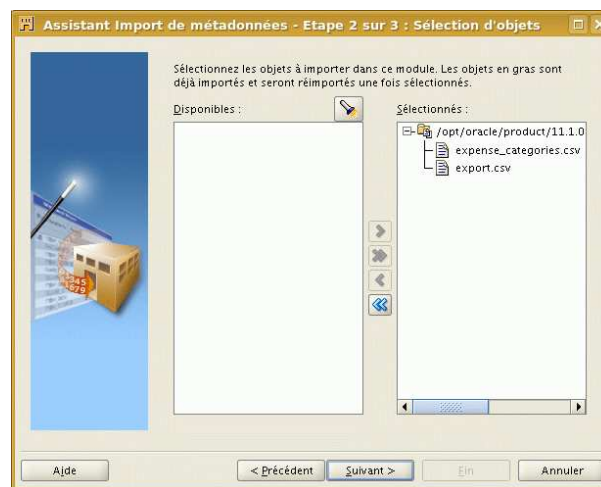


Figure 6: Sélectionner les fichiers à importer

7. cocher la case **importer une fois terminé** puis cliquer sur **suivant**, puis **fin**. A noter que si vous oubliez de cocher la case "importer une fois terminé" vous pouvez réaliser l'import ultérieurement en faisant un click droit sur le module **SRC_FILES** et en sélectionnant "import"
8. une fenêtre pour l'import des données s'ouvre ensuite. Cliquer sur **suivant** puis sélectionner tous les fichiers de données
9. dans la fenêtre de sélection des objets, faire passer les deux fichiers **expense_categories.csv** et **export.csv** dans la fenêtre de droite intitulée **Sélectionnés**. Vous devez être alors dans la situation de la figure 6.
10. il faut ensuite créer les métadonnées pour chacun de ces fichiers (la croix rouge indique qu'ils n'en ont pas). Pour cela surligner d'abord le fichier **expense_categories.csv** et appuyer sur le bouton **Modélisation**.
11. A la nouvelle fenêtre cliquer sur **suivant**. Regarder les informations dans la nouvelle fenêtre sans



Figure 7: Informations sur l'organisation du fichier

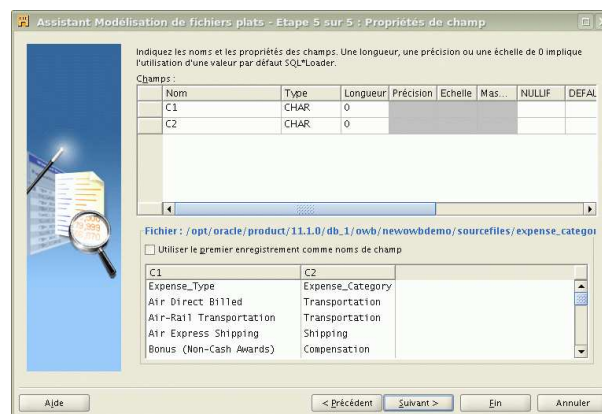


Figure 8: Affichage du fichier formaté, avec pseudo-noms pour les colonnes

modifier, puis cliquer sur suivant. Vous arrivez à la fenêtre de la figure 7.

12. Accepter toutes les valeurs par défaut en appuyant sur suivant jusqu'à ce que vous arriviez à l'écran de la figure 8
13. cocher la case "utiliser le premier enregistrement comme nom de champ" afin de donner les noms de colonnes d'après les données de la première ligne. Vous obtenez la figure 9.
14. valider et finir.
15. surligner ensuite le fichier `export.csv` et cliquer sur **Modéliser**
16. avancer sans changer les valeurs jusqu'à la fenêtre de la figure 10
17. changer le séparateur par défaut (";") pour le remplacer par ",", vous obtenez alors la figure 11. Revenir ensuite au bon séparateur avant de continuer;

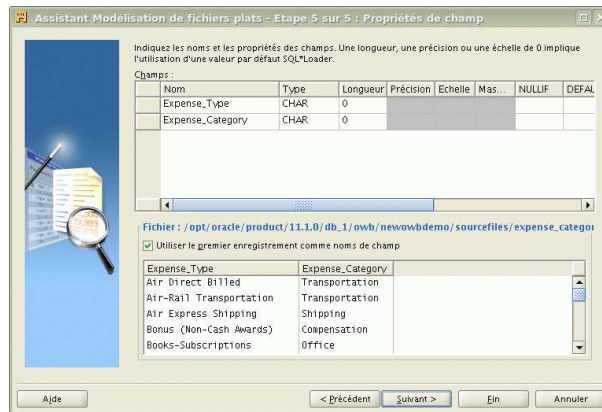


Figure 9: Affichage du fichier formaté, avec noms de colonne

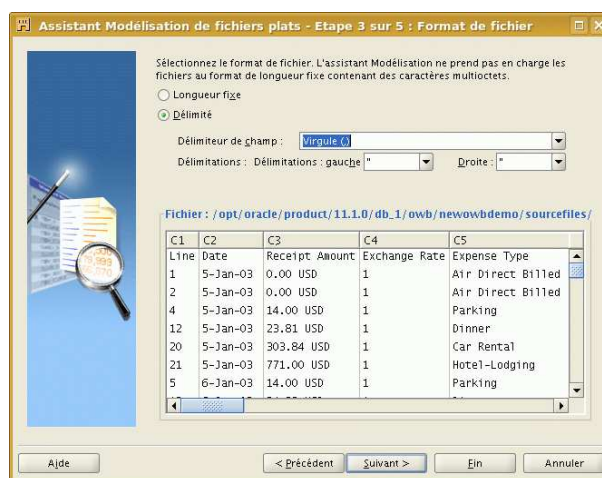


Figure 10: Informations sur l'organisation du fichier: bon séparateur

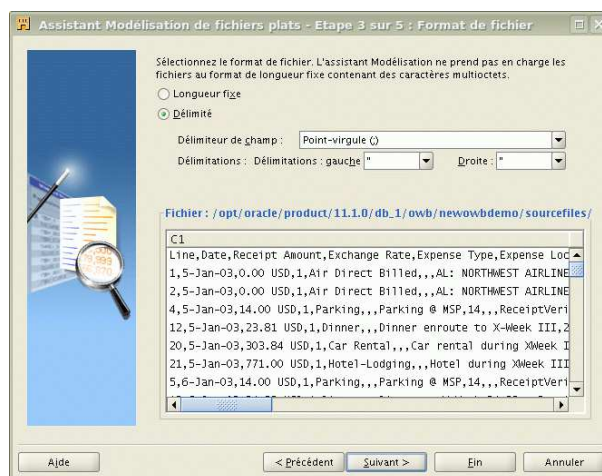


Figure 11: Informations sur l'organisation du fichier: mauvais séparateur

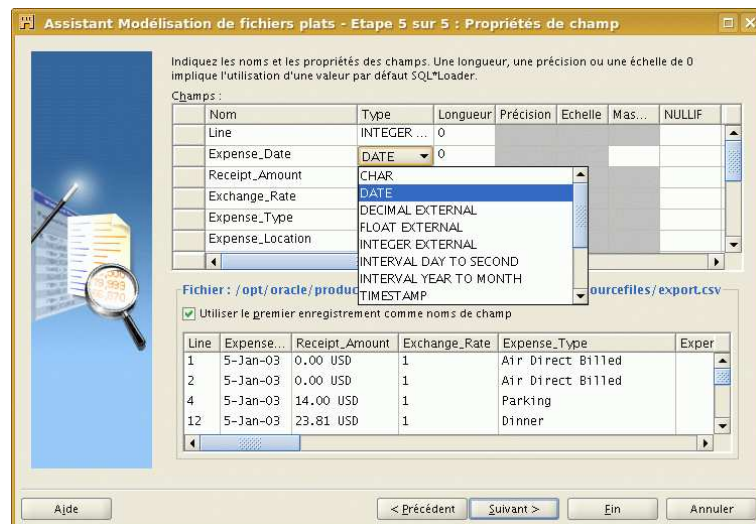


Figure 12: Modifier un attribut

18. à la fenêtre suivant, préciser qu'il s'agit d'un fichier contenant une seule sorte d'enregistrements (certains fichiers "plats" peuvent contenir de l'information pour plusieurs tables) puis cliquer sur suivant
19. là encore cocher la case "utiliser le premier enregistrement comme nom de champ" afin de donner les noms de colonnes d'après les données de la première ligne
20. DATE étant un mot réservé, vous ne pouvez le garder comme nom d'attribut. Le changer en Expense_Date. Changer le type de Expense_Date de CHAR en DATE (figure 12) avec un masque valant dd-mon-yy
21. les parenthèses étant un symbole réservé, renommer Reimbursable_Amount_(USD) en Reimbursable_A
- (figure 13; changer le type en DECIMAL_EXTERNAL
22. regarder en scrollant le tableau la colonne Type SQL qui montre les types SQL correspondant aux choix fait pour le type de chaque colonne
23. cliquer sur suivant, puis fin. L'import dans le repository s'effectue.
24. valider les changements (clic droit dans l'explorateur de projets sur SRC_FILES dans FICHIER, puis sélectionner "valider". Vous obtenez alors une fenêtre semblable à la figure 15.
25. aller dans le menu principal et sauvegarder en sélectionnant enregistrer tout

Avantages de cette méthode

Les bénéfices offerts par ce processus d'échantillonnage sont:

- Productivité: l'échantillonnage interactif des données vous permet de définir et capturer plus rapidement les métadonnées
- Exactitude: en étant guidé à travers les structures de données souvent longues et complexes, vous diminuez le risque d'erreurs et fournissez des métadonnées de grande qualité.

Assistant Modélisation de fichiers plats - Etape 5 sur 5 : Propriétés de champ

Indiquez les noms et les propriétés des champs. Une longueur, une précision ou une échelle de 0 implique l'utilisation d'une valeur par défaut SQL*Loader.

Champs :

Nom	Type	Longueur	Précision	Echelle	Masque	NU
Line	INTEGER ...	0				
Expense_Date	DATE	0			dd-mon-yy	
Receipt_Amount	CHAR	0				
Exchange_Rate	INTEGER ...	0				
Expense_Type	CHAR	0				
Expense_Location	CHAR	0				
Merchant_Name	CHAR	0				
Justification	CHAR	0				
Reimbursable_Amount_(USD)	CHAR	0				
Expense_Group	CHAR	0				

Fichier : /opt/oracle/product/11.1.0/db_1/owb/newowbdemo/sourcefiles/export.csv

☒ Utiliser le premier enregistrement comme noms de champ

Line	Expense...	Receipt_Amount	Exchange_Rate	Expense_Type	Expense_L
1	5-Jan-03	0.00 USD	1	Air Direct Billed	
2	5-Jan-03	0.00 USD	1	Air Direct Billed	
4	5-Jan-03	14.00 USD	1	Parking	
12	5-Jan-03	23.81 USD	1	Dinner	
20	5-Jan-03	303.84 USD	1	Car Rental	
21	5-Jan-03	771.00 USD	1	Hotel-Lodging	
5	6-Jan-03	14.00 USD	1	Parking	

Aide < Précédent Suivant > Fin Annuler

Figure 13: Modifier un attribut(2)

Assistant Modélisation de fichiers plats - Etape 5 sur 5 : Propriétés de champ

Indiquez les noms et les propriétés des champs. Une longueur, une précision ou une échelle de 0 implique l'utilisation d'une valeur par défaut SQL*Loader.

Champs :

Nom	Type SQL
Line	Valeur par défaut (NUMBER)
Expense_Date	Valeur par défaut (DATE)
Receipt_Amount	Valeur par défaut (VARCHAR2)
Exchange_Rate	Valeur par défaut (NUMBER)
Expense_Type	Valeur par défaut (VARCHAR2)
Expense_Location	Valeur par défaut (VARCHAR2)
Merchant_Name	Valeur par défaut (VARCHAR2)
Justification	Valeur par défaut (VARCHAR2)
Reimbursable_Amount	Valeur par défaut (NUMBER)
Expense_Group	Valeur par défaut (VARCHAR2)

Fichier : /opt/oracle/product/11.1.0/db_1/owb/newowbdemo/sourcefiles/export.csv

☒ Utiliser le premier enregistrement comme noms de champ

Line	Expense...	Receipt_Amount	Exchange_Rate	Expense_Type	Expense_L
1	5-Jan-03	0.00 USD	1	Air Direct Billed	
2	5-Jan-03	0.00 USD	1	Air Direct Billed	
4	5-Jan-03	14.00 USD	1	Parking	
12	5-Jan-03	23.81 USD	1	Dinner	
20	5-Jan-03	303.84 USD	1	Car Rental	
21	5-Jan-03	771.00 USD	1	Hotel-Lodging	
5	6-Jan-03	14.00 USD	1	Parking	

Aide < Précédent Suivant > Fin Annuler

Figure 14: Type SQL

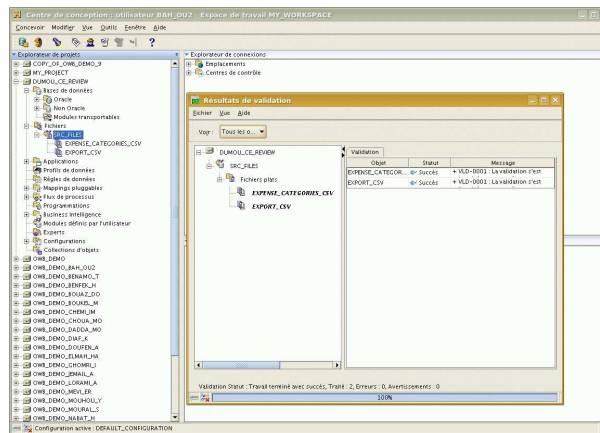


Figure 15: Fenêtre résultat de la validation

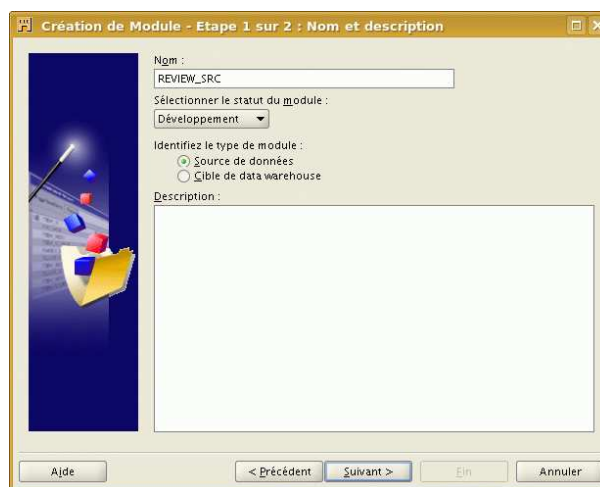


Figure 16: Créer le module REVIEW_SRC

3.2 Importer des métadonnées d'une base de données

De même que pour les fichiers texte, nous devons créer un module pour les métadonnées de la base de données source. Pour importer les métadonnées de la base de données nous procédons comme suit:

1. si ce n'est pas fait, ouvrir le projet `vo treLogin_REVIEW`, puis ouvrir le nœud `Base de données`
2. surligner le nœud `ORACLE` et faites un clic droit
3. sélectionner `Nouveau`
4. dans la fenêtre de création de module, nommer le module `REVIEW_SRC` et dire que le type du module est `source de données`, comme sur la figure 16
5. cliquer sur `Modifier` pour l'emplacement, et saisir les valeurs comme sur la figure 17 (le nom de service est `nfe115`, le login password est `exemple/exemple` et n'oubliez pas de dire que la base est une `11.1`. Le schéma utilisé s'appelle `EXEMPLE`).

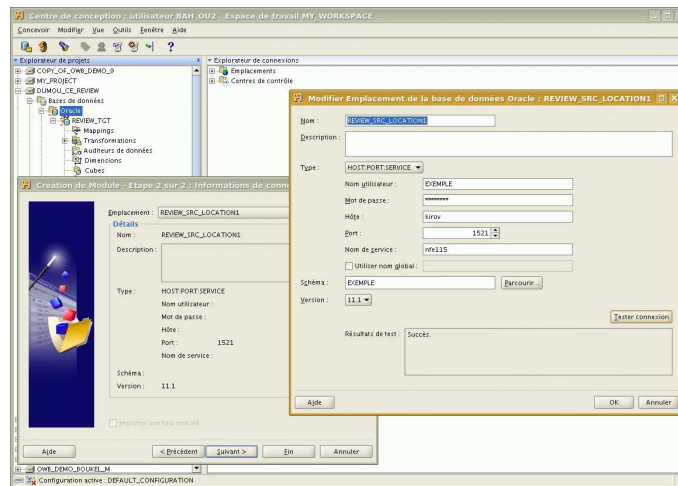


Figure 17: Créer un nouvel emplacement

6. valider, une fenêtre d'aide à l'import de métadonnées va alors s'ouvrir, semblable à celle de la figure 18
7. cliquer sur suivant
8. déplacer la table voulue (EMPLOYEE) dans la partie droite de la fenêtre pour sélectionner les métadonnées à importer (voir figure 19)
9. cliquer sur suivant, puis à la fenêtre suivante cliquer sur "options d'import avancées", vous obtenez alors la fenêtre de l'image 20
10. cliquer sur OK puis fin, vous obtenez la confirmation de la figure 21 (dérouler le champ table pour voir les attributs de la table comme sur la figure).
11. enfin en faisant clic droit sur REVIEW_SRC, cliquer sur valider.

Avantages de cette méthode

Importer des tables et autres objets en provenance d'une base de données est plus simple que de travailler sur des fichiers "plats" (texte). Cependant ce processus d'échantillonnage offre quand-même certains avantages:

- productivité: comme vous pouvez importer les définitions il n'y a pas besoin de faire les entrées des données à la main, et toutes les clés et contraintes sont maintenues. Dans des sections ultérieures nous verrons comment la création de flux de données bénéficiera de ceci.
- productivité (2): une fois que vous avez ces définitions et la connexion à la source dans vos métadonnées, vous pouvez rafraîchir ces définitions. Warehouse Builder vous montrera les différences et vous pourrez choisir de les implanter. Ceci réduit le temps et l'effort nécessaire afin de détecter et harmoniser les changements.
- exactitude: l'import laisse peu de place aux erreurs et garantit donc ainsi des métadonnées de plus grande qualité



Figure 18: Fenêtre d'aide à l'import



Figure 19: Sélection des métadonnées à importer

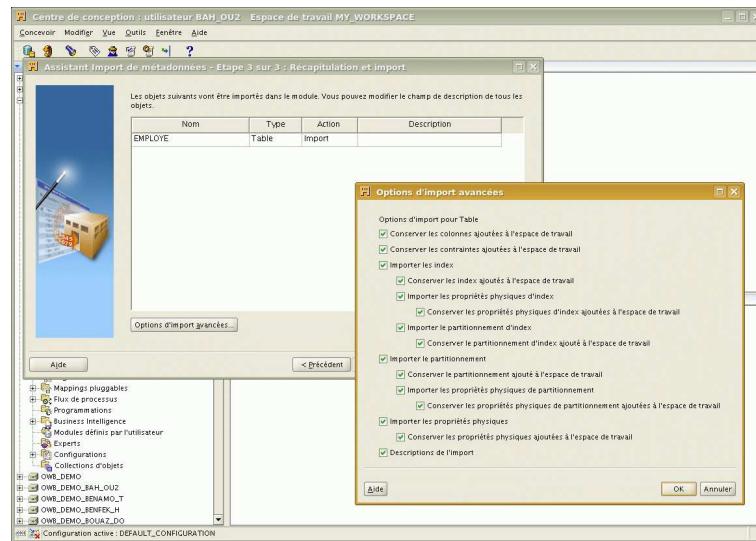


Figure 20: Options d'import disponibles

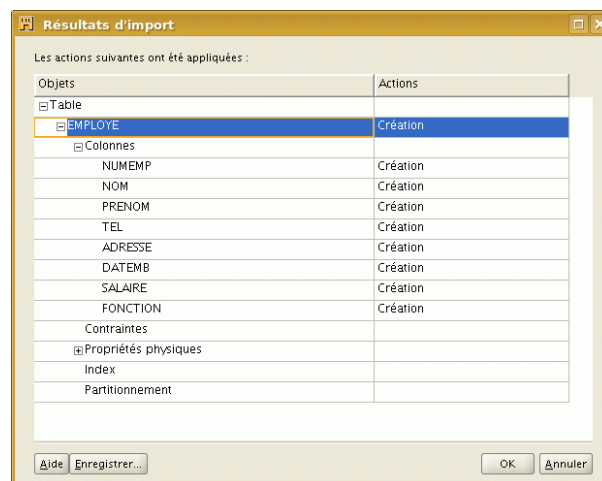


Figure 21: Résumé des métadonnées importées

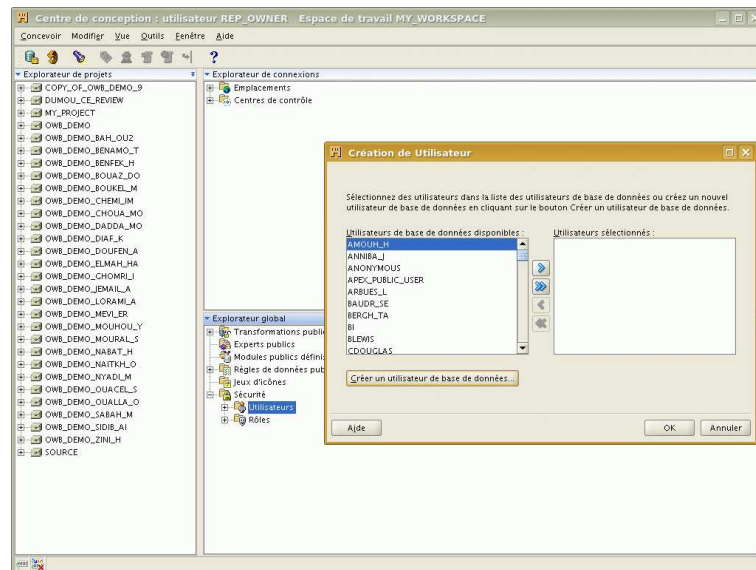


Figure 22: Création d'un utilisateur cible

4 Concevoir le système cible

OWB fournit des fonctionnalités avancées pour la modélisation relationnelle ou multi-dimensionnelle: il permet de définir des objets relationnels comme des tables, des vues, des vues matérialisées, des tables externes,..., et des objets cibles comme des dimensions et des cubes.

OWB sépare explicitement la représentation dimensionnelle de son implantation. Ainsi la même conception dimensionnelle peut être implantée comme un entrepôt relationnel (ROLAP) ou multi-dimensionnel (MOLAP). Nous allons voir ici pour un entrepôt cible en ROLAP. En relationnel, OWB supporte la conception et l'implantation des schémas étoile ou flocon. Nous allons également modéliser un schéma en étoile simple consistant en un cube avec des clés étrangères référençant 2 dimensions.

4.1 Créer un utilisateur cible et une cible

La cible contient les métadonnées des objets cibles que l'on conçoit. Chaque cible doit être associée à un schéma utilisateur cible. En tant que propriétaire de l'espace de travail vous pouvez définir les stratégies de sécurité pour votre implantation.

1. dans la fenêtre en bas à droite (Explorateur Global) dérouler l'onglet Sécurité. Faire un clic droit sur Utilisateur et sélectionner nouveau.
2. choisir de créer un nouvel utilisateur appelé <votreLogin>_EXPENSE_WH de mot de passe votreMotDePasse. Me demander le mot de passe système. Vérifier que le compte créé apparait bien dans la liste. La création de cet utilisateur permet de créer également un schéma associé pour stocker physiquement les données cibles.
3. créer ensuite l'adresse (location en anglais) de la cible. Pour cela dans la fenêtre en haut à droite (Explorateur de connexions), dérouler Bases de données et faire clic droit sur Oracle,

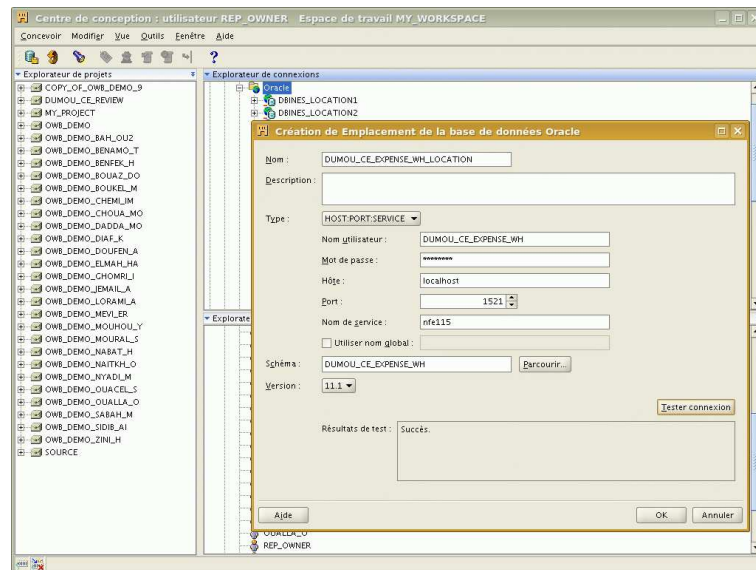


Figure 23: Création d'un lieu cible

choisir nouveau (figure 23). Les paramètres sont: pour le nom utilisateur, LOGIN_EXPENSE_WH, votre mot de passe, pour hôte localhost avec port 1521, nom de service nfe115 et schéma LOGIN_EXPENSE_WH.

4.2 Créer des tables externes

Les tables externes permettent de directement interroger les fichiers “plats” depuis la base de données oracle, ainsi que de réaliser des jointures, transformations, etc, de ces fichiers. L'utilisation de tables externes pour accéder aux sources de données non-relationnelles offre de multiples avantages, parmi lesquels:

- génération de SQL et jointure hétérogènes: avec les tables externes, vous pouvez interroger et transformer les données contenues dans les fichiers plats directement, le code SQL sera généré par OWB. De plus OWB supporte les jointures entre les fichiers plats et les tables relationnelles;
- performance: interroger les fichiers plats via OWB qui les considère comme des tables, permet de faire des accès parallèles à ces fichiers (normalement souvent impossible car verouillage). Par conséquent les performances grâce aux traitements parallèles sont améliorées;

Passons à la création de 2 tables externes:

1. tout d'abord importer des métadonnées pour la démo stockées dans un fichier “d'export” avec extension “.mdl”. Pour cela cliquer dans le menu sur “concevoir”, choisir “importer” puis “Métadonnées Warehouse Builder”. Les données sont dans le fichier OWB_DEMO.mdl situé dans le répertoire /opt/oracle/product/11.1.0/db_1/owb/newowbdemo. Garder les options par défaut. Normalement un projet OWB_DEMO doit apparaître.
2. dans le projet OWB_DEMO, dérouler Bases de données, cliquer sur Oracle avec le bouton droit et choisir nouveau.



Figure 24: Création d'un module cible

3. choisir pour nom de module `EXPENSE_WH`, garder l'option développement et vérifier que l'option "cible de datawarehouse" est sélectionnée (figure 24)
4. ensuite choisir la location créée précédemment comme sur la figure 25
5. dérouler le nouveau nœud `EXPENSE_WH` créé, et cliquer avec le bouton droit sur `Tables externes`, choisir `nouveau` et donner le nom `EXPENSE_CATEGORIES`. Sélectionner le fichier `EXPENSE_CATEGORIES_` (figure 26), pour l'emplacement choisir celui par défaut.
6. recommencer l'étape précédente pour créer la table externe de nom `EXPENSE_DATA` en choisissant cette fois le fichier `EXPORT_CSV`.
7. Vérifier que les deux tables externes sont bien créées sous le nœud `Tables externes`.
8. il faut ensuite configurer les détails du système de fichiers physique pour chacune des 2 tables externes. Pour cela faire un clic droit sur `EXPENSE_CATEGORIES` et choisir `Configure` (figure 27)
9. cliquer ensuite sur `Fichiers de données` et choisir `Créer` (figure 28).
10. garder le nom du nœud par défaut (`NEW_FICHIER_DE_DONNEES_1`), saisir le nom de fichier de données `expense_categories.csv` et choisir pour l'emplacement du fichier `SOURCE_LOCATION` (figure 29).
11. faire la même chose pour la table externe `EXPENSE_DATA`, en donnant le nom de fichier de données `export.csv`.
12. vous avez fini de créer 2 tables externes pointant sur les 2 fichiers plats. Sauvegarder.

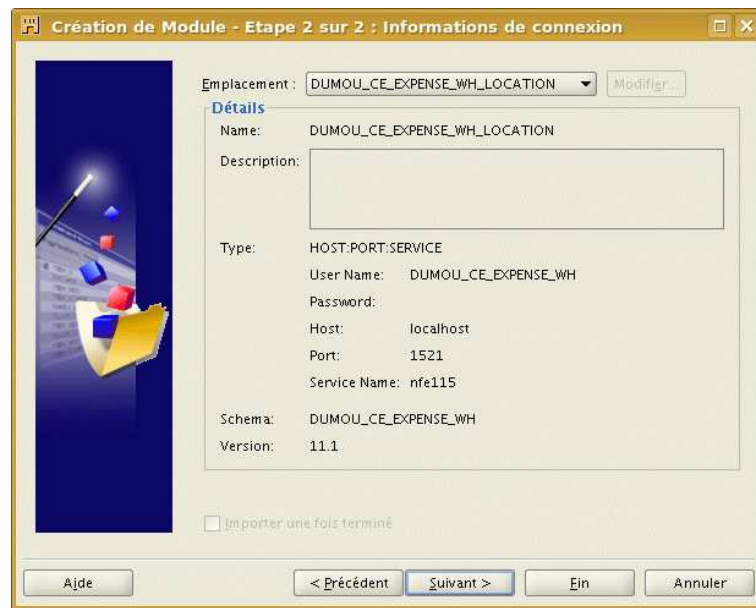


Figure 25: Création d'un module cible (2)

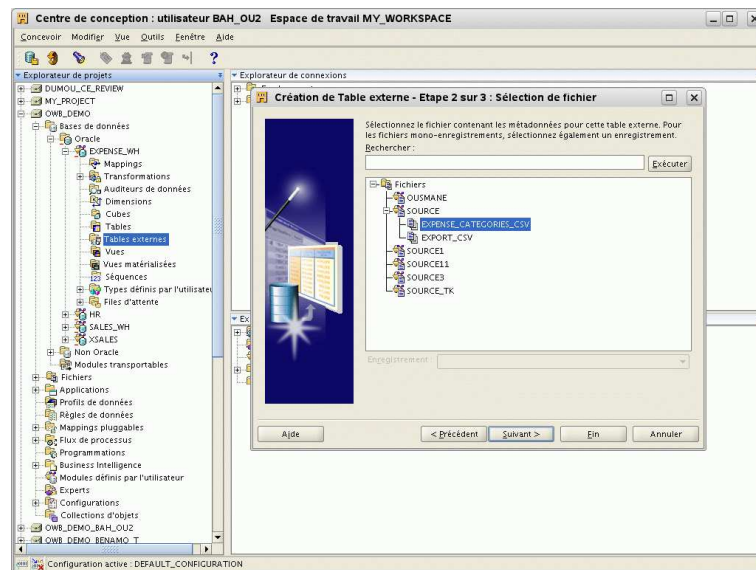


Figure 26: Création d'une table externe

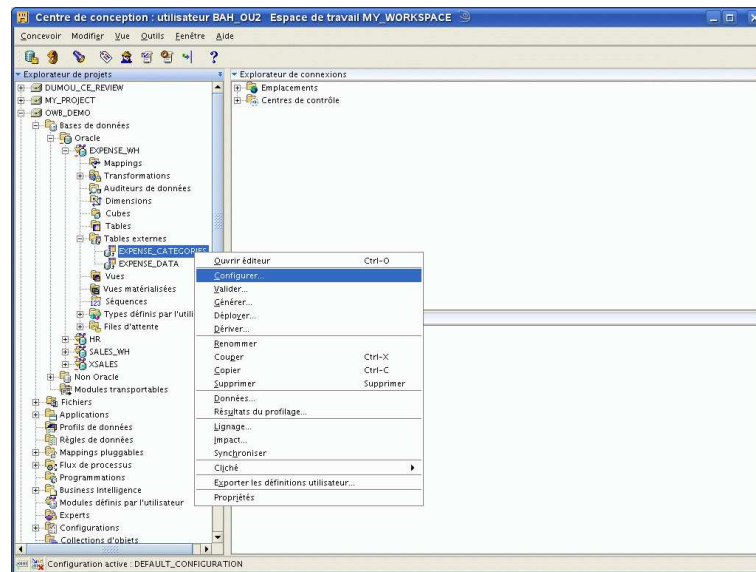


Figure 27: Configuration du système de fichiers

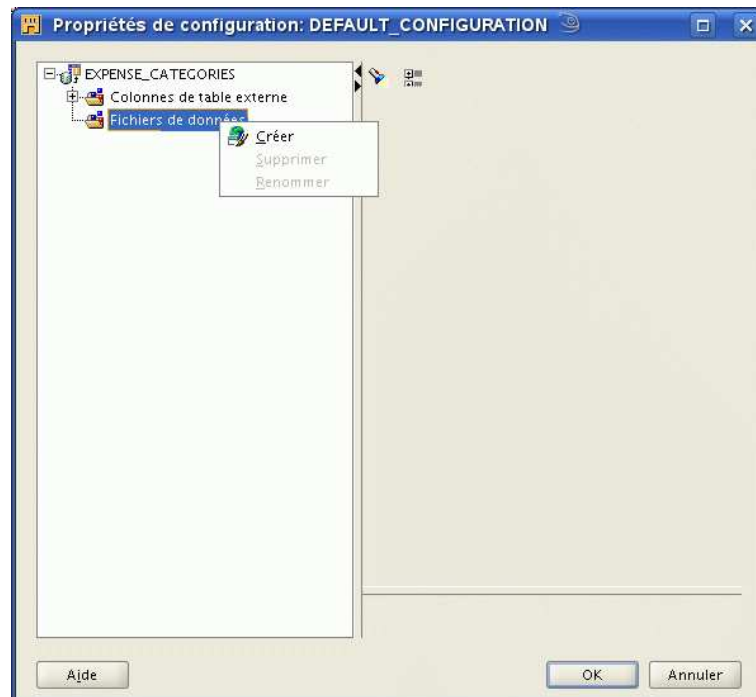


Figure 28: Configuration du système de fichiers (2)

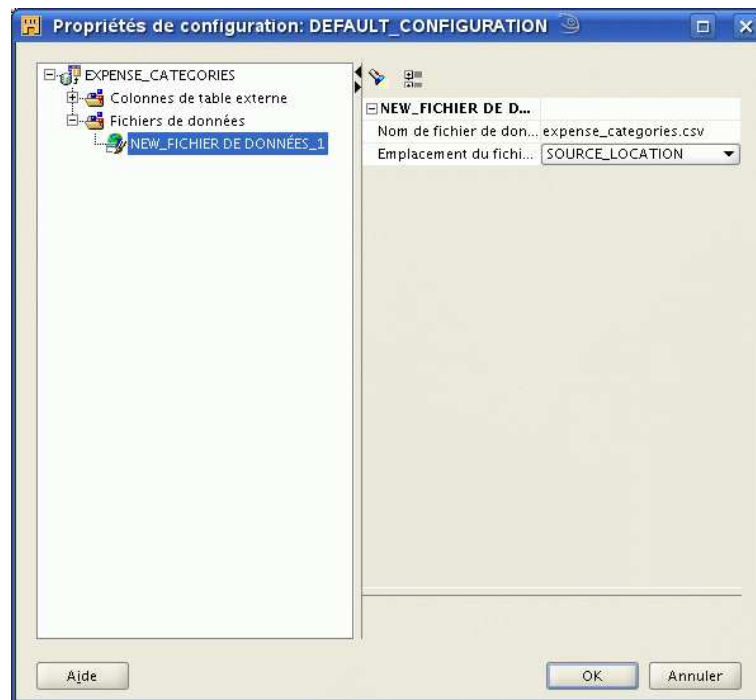


Figure 29: Configuration du système de fichiers (3)

4.3 Créer les dimensions

Une dimension se constitue d'un ensemble de niveaux et de hiérarchies définies en utilisant ces niveaux. Les utilisateurs utilisent ces hiérarchies pour faire du *roll-up* ou *drill-down* des résultats. Pour créer une dimension, il faut définir dans l'ordre:

- les attributs des dimensions
- les niveaux
- les attributs de niveaux
- les hiérarchies

4.3.1 Examiner la dimension **PRODUCT** existante

1. dans l'explorateur de projet se placer dans **OWB_DEMO > Bases de Données > Oracle > SALES_WH > Dimensions**. Faire un clic droit sur **PRODUCTS** et sélectionner **Ouvrir éditeur**.
2. vous obtenez une nouvelle fenêtre semblable à la figure 30. L'éditeur d'objets de données est l'interface permettant de facilement concevoir et gérer une grande variété d'objets de la base de données ou des dimensions. Il permet également de concevoir et modifier différents schémas d'entrepôts via l'utilisation de diagrammes et de feuilles de propriétés.
3. pour voir les attributs d'une dimension, cliquer sur l'onglet **Attributs** dans le panneau **Détails de dimension**. Observer (au-dessus du tableau) qu'une séquence **PROD_DIM_SEQ** a été assignée

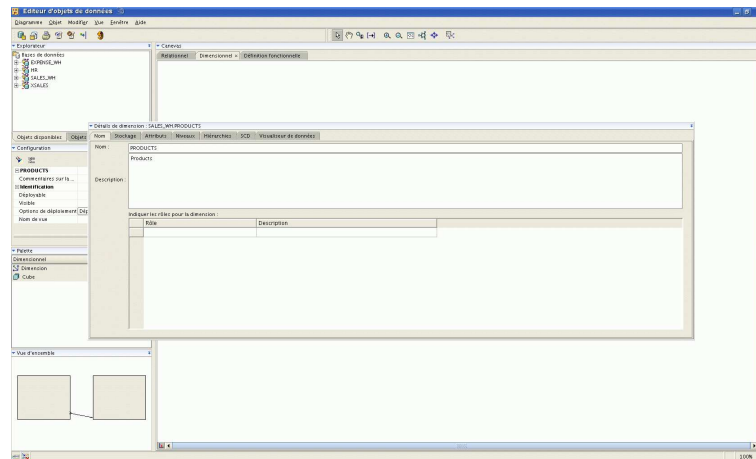


Figure 30: L'éditeur d'objets de données

à la dimension, afin de peupler une clé pour la dimension lorsqu'on chargera les données. Il va servir pour peupler ici l'attribut **ID** qui a été identifié comme identificateur **substituer**. Cela signifie que cela va être la clé de chaque niveau de la dimension, qui sera unique grâce à la séquence. **SOURCE_ID** est l'identificateur fonctionnel, ce qui signifie que c'est la clé identifiée dans la source de données.

4. Dans la colonne **description**, on peut avoir les valeurs par exemple **Description abrégée** et **Description longue**. Cela peut-être exploité par les outils de business intelligence qui utilisent tantôt des descriptions courtes, tantôt des descriptions longues dans leurs affichages.
5. pour voir les niveaux de la dimension et les attributs des niveaux, cliquer sur l'onglet **Niveaux**. Remarquer qu'il y a 4 niveaux et que chacun a un ensemble d'attributs possibles. Par exemple le niveau **PRODUCT** implémente tous les attributs de dimension comme des attributs pour ce niveau (cf. figure 31), alors que **CATEGORY** n'implémente qu'un sous-ensemble de ces attributs. Dans une implantation relationnelle, un attribut de niveau devient une colonne dans une table. Les attributs de niveau n'ont pas, normalement, obligatoirement le même nom que les attributs de dimension, mais OWB fait ce choix par défaut. Lors de la définition des niveaux vous n'avez pas besoin de vous souciez de l'ordre dans lequel vous les créez. Les niveaux sont uniquement organisés dans les hiérarchies.
6. Pour voir la hiérarchie de la dimension, cliquer sur l'onglet hiérarchie (fig. 32). Dans cet exemple une seule hiérarchie a été définie **PRO_STD** avec les niveaux triés suivant un certain ordre. Créer une autre hiérarchie de dimensions comme sur la figure 33 (mais laisser **PRO_STD** comme hiérarchie par défaut).
7. lorsque vous définissez une dimension (onglet **SCD**), vous pouvez aussi spécifier la stratégie de stockage lors de changement dans les valeurs de dimensions. Il existe 3 stratégies. Si vous souhaitez stocker l'historique complet des changements choisir le **Type 2**. Cliquer sur **Paramètres**. Vous pouvez alors spécifier pour chaque attribut de la dimension l'option:
 - **déclencher l'historique**: cette option conduit au stockage d'une nouvelle version des enregistrements si cet attribut de dimension change;
 - **Date de validité**: mettre cette option pour l'unique attribut qui stocke la date et l'heure où l'enregistrement est devenu effectif;

Détails de dimension : SALES_WH.PRODUCTS

Nom Stockage Attributs Niveaux Hiérarchies SCD Visualiseur de données

	Niveau	Description
1	PRODUCT	Product
2	SUBCATEGORY	Product Subcategory
3	CATEGORY	Product Category
4	TOTAL	Total

Attributs de niveau pour PRODUCT :

	Nom d'attribut de dim...	Applic...	Nom d'attribut de nive...	Description
1	ID	☒	ID	
2	NAME	☒	NAME	Product Name
3	DESCRIPTION	☒	DESCRIPTION	Product Description
4	SOURCE_ID	☒	SOURCE_ID	Product Source ID
5	PACK_SIZE	☒	PACK_SIZE	Product Pack Size
6	LIST_PRICE	☒	LIST_PRICE	Product List Price
7	VALID_FROM_DATE	☒	VALID_FROM_DATE	product date
8	VALID_TO_DATE	☒	VALID_TO_DATE	product expiry date

Figure 31: Visualiser les niveaux et leurs attributs

Détails de dimension : SALES_WH.PRODUCTS

Nom Stockage Attributs Niveaux Hiérarchies SCD Visualiseur de données

	Hiérarchie	Description	Valeur...
1	PROD_STD	Product Standard	

Niveaux pour PROD_STD :

	Niveau	Passer à un niveau
1	TOTAL	
2	CATEGORY	
3	SUBCATEGORY	
4	PRODUCT	

Figure 32: Exemple de hiérarchie

Détails de dimension : SALES_WH.PRODUCTS

Nom Stockage Attributs Niveaux Hiérarchies SCD Visualiseur de données

	Hiérarchie	Description	Valeur...
1	PROD_STD	Product Standard	
2	TEST		

Niveaux pour TEST :

	Niveau	Passer à un niveau
1	CATEGORY	
2	SUBCATEGORY	
3	PRODUCT	
4	TOTAL	

Figure 33: Créer une nouvelle hiérarchie



Figure 34: Stratégie de stockage pour changement dans les dimensions

- **Date d'expiration:** idem pour l'unique attribut stockant la date et heure ou l'attribut est devenu non-effectif ou invalide.

Ainsi dans notre exemple, on stocke l'historique des changements chaque fois que LIST_PRICE, NAME ou PACK_SIZE d'un produit change.

8. les objets dimensionnels conçus en utilisant OWB sont ensuite déployés dans la base soit sous la forme relationnelle soit multi-dimensionnelle. On peut choisir la forme grâce à l'onglet **stockage**. En fonction du choix, OWB générera le code correspondant. Pour une implémentation relationnelle on peut choisir le stockage en étoile, flocon ou manuel. L'éditeur d'objet de données fournit également une interface de visualisation des données permettant de faire du drill down pour une dimension. Fermer l'éditeur et revenir à l'interface principale. Sauvegarder.

4.3.2 Créer la dimension REL_TIME en utilisant l'outil d'aide pour les dimensions temporelles

Les dimensions temporelles sont très utilisées dans les entrepôts de données et OWB permet de créer facilement de telles dimensions mais également de les peupler. Lorsqu'on crée une dimension temporelle avec l'outil d'aide, OWB crée automatiquement les mappings pour peupler la dimension temporelle.

1. dérouler le module EXPENSE_WH. Faire un clic droit sur **Dimension** et sélectionner **Nouveau>Utilisation de l'assistant temporel**.
2. entrer comme nom de dimension **REL_TIME**, puis à l'écran suivant choisir le stockage **ROLLAP**.
3. A la page de génération de données, vous spécifiez l'intervalle de temps pour vos données. Cette information est utilisée pour générer un mapping qui peuple la dimension temporelle. Pour la date de



Figure 35: Créer une nouvelle dimension temporelle

début choisir 2003 et un nombre d'années de 3. Vérifier que l'option **calendrier** est bien cochée. Vous obtenez la figure 35. Si votre dimension temporelle contient des données fiscales, vous devez spécifier le mois et le jour de début pour l'année fiscale et le jour de début pour la semaine fiscale. Vous devez également spécifier les conventions pour les limites de mois et trimestre qui peuvent être 544 ou 445 (pour la signification de ces chiffres je n'ai aucune idée, je pense que ça veut dire que le trimestre à par exemple 3 mois, 1 de 5 semaines et 2 de 4, mais si quelqu'un à une meilleure explication je suis preneur ;-)).

4. à l'écran suivant sélectionner **hiérarchie normale** et cocher tous les niveaux comme sur la figure 36.
5. faire **suivant**, observer le détail des paramètres de création, faire **suivant** pour finir.
6. observer que OWB a créé 4 objets nécessaires pour une dimension temporelle: l'objet dimension **REL_TIME**, la séquence **REL_TIME_SEQ** qui peuple la clé de substitution **ID** de la dimension temporelle, la table **REL_TIME** pour supporter l'implantation relationnelle de la dimension temporelle en stockant physiquement les données temporelles, et le plus important le mapping **REL_TIME_MAP** pour peupler la dimension temporelle (voir image 37).
7. si vous êtes curieux de voir à quoi ressemble **REL_TIME_MAP**, double-cliquez sur le nœud de mapping pour lancer l'éditeur de mapping. En réorganisant un peu les différentes "tables" du mapping et en les agrandissant vous pouvez obtenir une représentation semblable à la figure 38. Vous pouvez évaluer la complexité du mapping et le temps économisé pour la création d'une dimension temporelle, qui, on le rappelle, est très fréquemment utilisé. Sauvegarder.

4.3.3 Créer la dimension **REL_CATEGORY** en utilisant l'outil d'aide

1. dans le module **EXPENSE_WH** faire un clic droit sur le nœud **dimension** et choisir de créer une nouvelle dimension en utilisant l'assistant, lui donner le nom **REL_CATEGORY**.
2. à la page concernant le mode de stockage, choisir **ROLAP** et cliquer sur **suivant**.



Figure 36: Créer une nouvelle dimension temporelle (2)

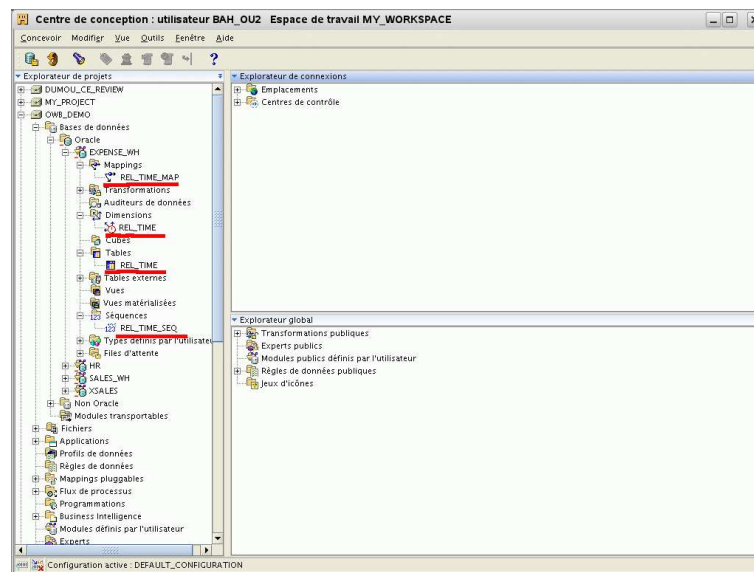


Figure 37: Créer une nouvelle dimension temporelle (3)

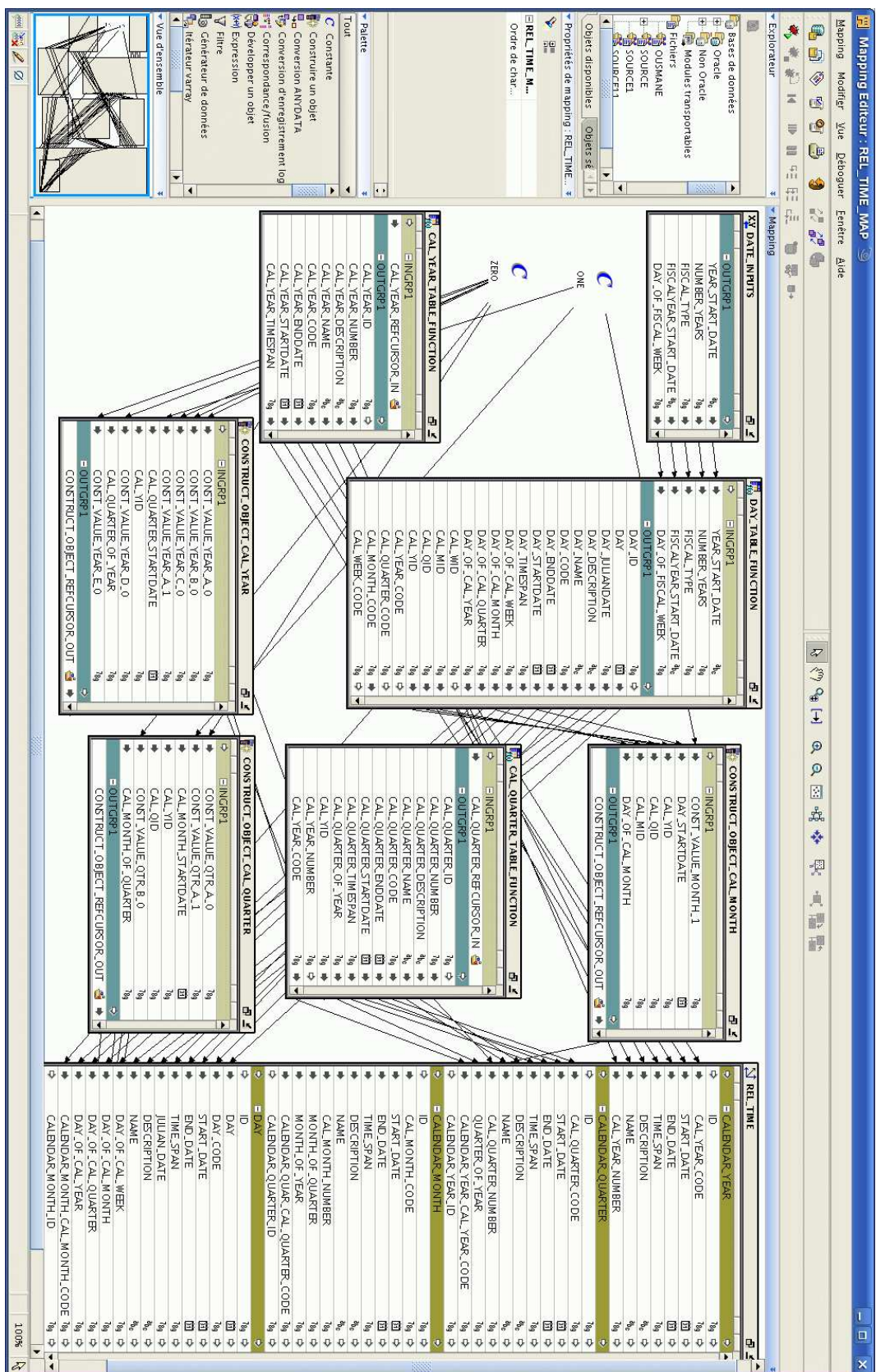


Figure 38: Le mapping de la dimension temporelle

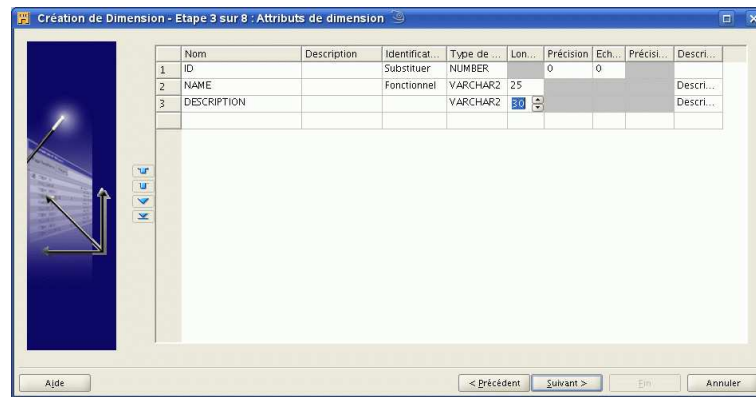


Figure 39: Attributs de dimension par défaut

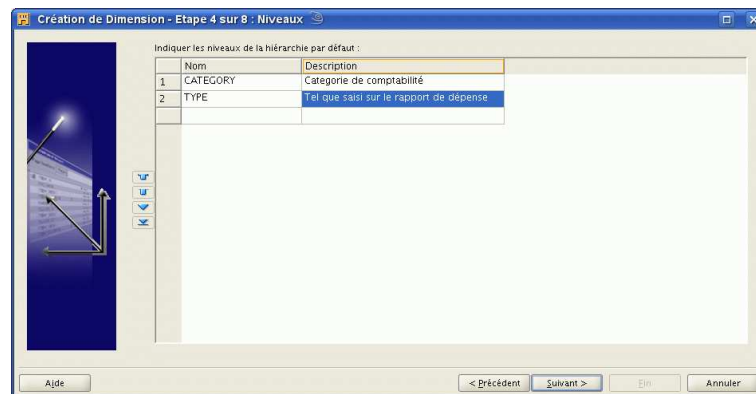


Figure 40: Niveaux de la hiérarchie

3. à la page sur les attributs de dimensions, vous trouvez les 3 attributs par défaut: ID, NAME et DESCRIPTION. Noter que pour l'attribut ID l'identificateur est du type substitution et que NAME est une clé fonctionnelle. La clé de substitution doit impérativement être un nombre. Pour l'attribut NAME changer la longueur à 30 (figure 39).
4. sur la page des niveaux spécifier les niveaux comme sur la figure 40 pour la hiérarchie par défaut de la dimension.
5. sur la page des attributs de niveau, pour chacun des deux niveaux définis précédemment sélectionner tous les attributs (figure 41).
6. choisir à l'écran suivant le Type 1 "Ne pas conserver d'historique". Valider tout. Vous pouvez vérifier qu'une table et qu'une séquence ont été ajoutées en plus de la dimension. Sauvegarder.

4.4 Créer les cubes

Les cubes contiennent des mesures et sont liés à une ou plusieurs dimensions. Les axes d'un cube contiennent des valeurs de dimension et le "corps" du cube contient les valeurs mesurées. La plupart des valeurs sont numériques et additives. Par exemple les données de vente peuvent être organisées dans un cube dont

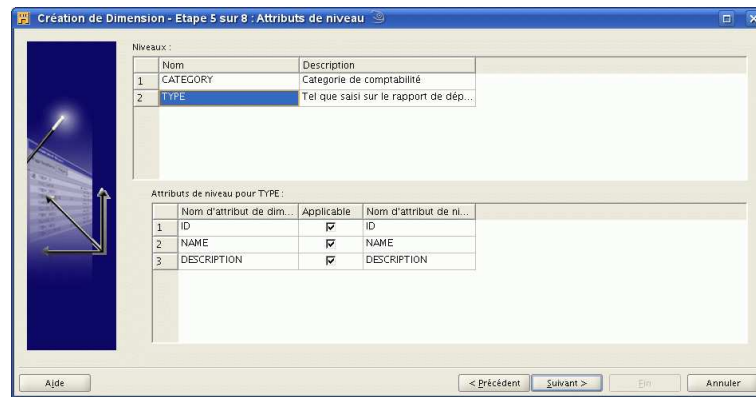


Figure 41: Attributs pour chaque niveau

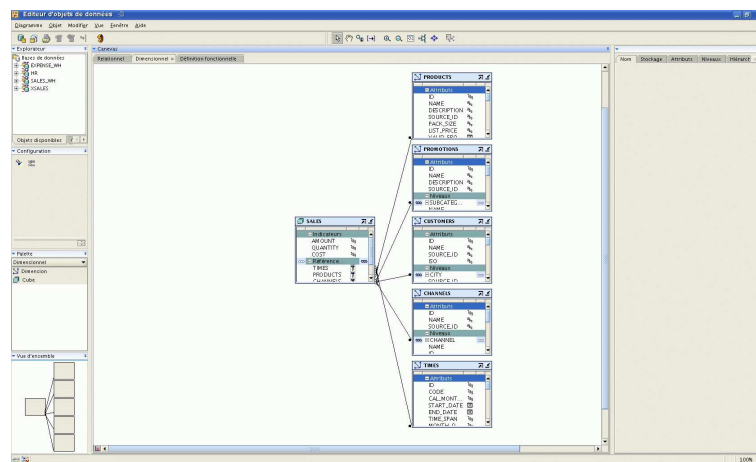
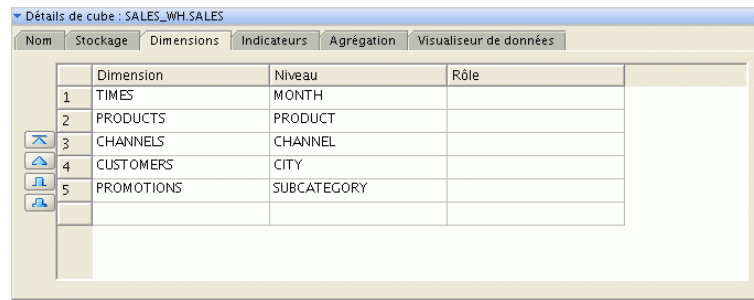


Figure 42: Schéma de conception du cube

les axes contiennent des valeurs pour les dimensions **TIME**, **PRODUCT** et **CUSTOMER** et dont le corps contient les valeurs pour les mesures de “valeurs des ventes”. Dans une implantation relationnelle, un cube est lié aux tables dimension par des clés étrangères et est constitué d’un ensemble de mesures. Pour créer un cube il faut d’abord définir les mesures du cube et les dimensions du cube.

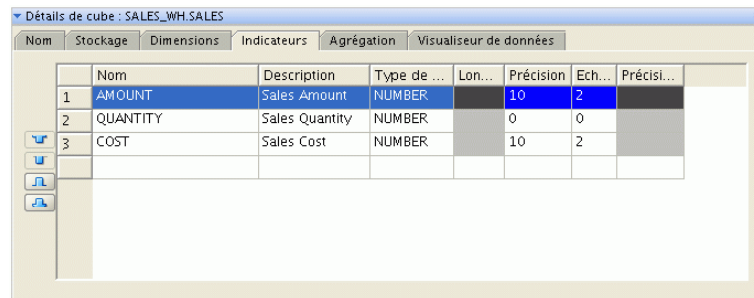
4.4.1 Examiner le cube **SALES** prédéfini

1. dérouler **OWB_DEMO>Bases de données>Oracle>SALES_WH**. Dans le nœud Cubes faire un clic droit sur **SALES** et sélectionner **Ouvrir éditeur**. Vous obtenez une fenêtre comme sur la figure 42 qui représente le schéma de conception du cube. Remarquer que le cube **SALES** est relié à 5 dimensions: **TIMES**, **PRODUCTS**, **CHANNELS**, **PROMOTION** et **CUSTOMERS** (on reconnaît les boîtes de dimension ou de cube au petit logo dans le coin en haut à gauche de chaque boîte).
2. pour examiner les dimensions du cube **SALES**, cliquer sur l’onglet **Dimensions** dans la fenêtre **Détails du cube**. Observer figure 43 que le cube est lié à des niveaux appartenant aux 5 dimensions. Par exemple le cube fait référence au niveau **PRODUCT** de la dimension **PRODUCTS**. La colonne **Role** affiche les rôles des dimensions (si celles-ci ont été précisées ultérieurement).



	Dimension	Niveau	Rôle
1	TIMES	MONTH	
2	PRODUCTS	PRODUCT	
3	CHANNELS	CHANNEL	
4	CUSTOMERS	CITY	
5	PROMOTIONS	SUBCATEGORY	

Figure 43: Les dimensions du cube



	Nom	Description	Type de ...	Lon...	Précision	Ech...	Précisi...
1	AMOUNT	Sales Amount	NUMBER		10	2	
2	QUANTITY	Sales Quantity	NUMBER		0	0	
3	COST	Sales Cost	NUMBER		10	2	

Figure 44: Les indicateurs du cube

- pour observer les mesures du cube, cliquer sur l'onglet **indicateurs**. Noter qu'il y a 3 mesures et que tous sont numériques. Ces mesures peuvent être agrégées pour déterminer le coût total, la quantité pour une période, client ou produit, des ventes (figure 44).
- cliquer sur l'onglet **Agrégations** pour voir les agrégations qui doivent être réalisées pour chaque dimension que le cube référence. Regarder les différentes fonctions d'agrégation possibles ainsi que les méthodes de régénération de récapitulatif (rafraîchissement des valeurs agrégées), ainsi que la stratégie de récapitulatif pour le cube qui permet de définir les niveaux pour lesquels les données doivent être précalculées pour chaque dimension.
- On peut implanter un cube dans une forme relationnelle ou multi-dimensionnelle. Pour l'implantation relationnelle une table relationnelle stocke les données du cube. Lorsque le cube est implanté dans l'environnement multi-dimensionnel, les données du cube sont stockées dans un espace de travail analytique. Pour choisir le stockage, cliquer sur l'onglet **Stockage** (figure 45). Pour l'implantation relationnelle, vous pouvez choisir de créer des index bitmap afin d'améliorer les performance lors de l'évaluation de requêtes.
- comme pour les dimensions, on peut visualiser les données du cube également (onglet **Visualiseur de données**. Fermer la fenêtre.

4.4.2 Créer le cube REL_EXPENSE en utilisant l'assistant

On va créer un cube simple avec 2 dimensions REL_TIME et REL_CATEGORY.



Figure 45: Choix du stockage du cube

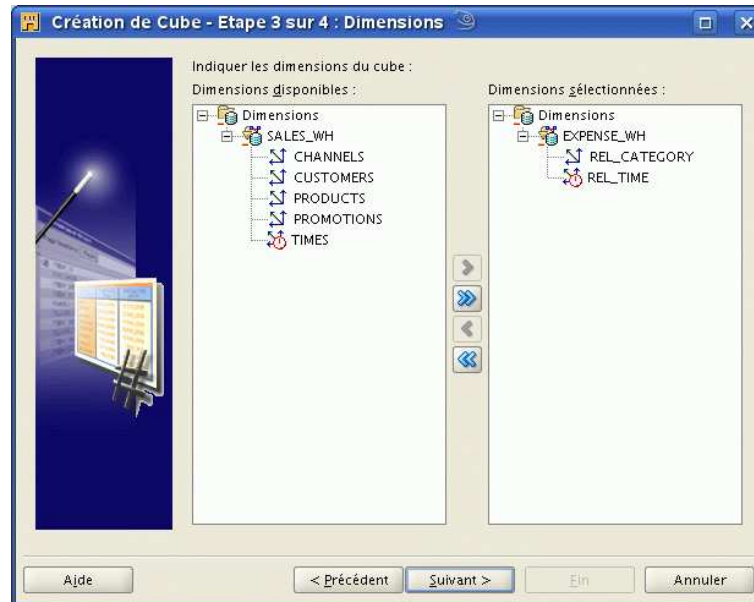


Figure 46: Choix des dimensions

1. Aller dans le nœud EXPENSE_WH et faire un clic droit sur Cube, choisir nouveau et Utilisation de l'assistant. Lui donner le nom REL_EXPENSE.
2. choisir le stockage ROLAP. sur la page des dimensions, sélectionner les dimensions de EXPENSE_WH, à savoir REL_TIME et REL_CATEGORY, comme dimensions pour notre cube (figure 46).
3. à la page des mesures (indicateurs), créer l'indicateur EXPENSE et laisser les valeurs par défaut (figure 47). Achèver la création.
4. ouvrir l'éditeur du cube (double-clic sur le cube, ou clic droit en choisissant d'ouvrir avec l'éditeur. Examiner le cube comme le cube précédent, puis fermer l'éditeur (figure 48).

5 Concevoir les mappings ETL pour les flux de données

Le but de cette section est de décrire comment concevoir et réaliser les flux, et les utiliser pour définir les mouvements et transformations de données dans OWB. Il existe un large ensemble d'opérateurs de mapping disponibles dans l'éditeur de mapping de OWB. Dans cette section nous allons apprendre à créer des mappings pour extraire des données depuis des sources, les transformer et les charger dans les cibles.

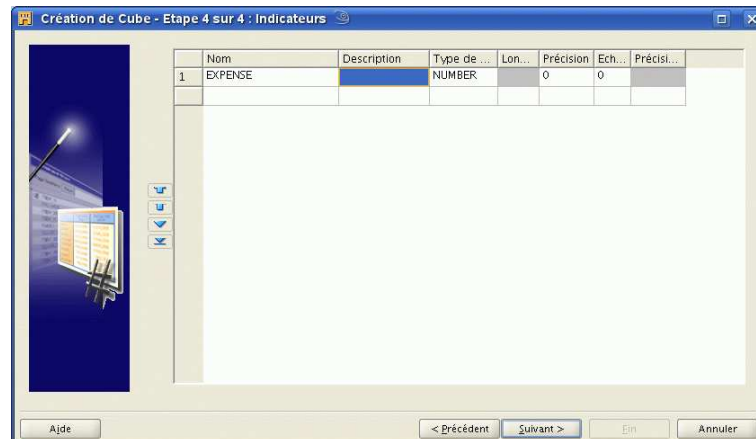


Figure 47: Création d'un indicateur

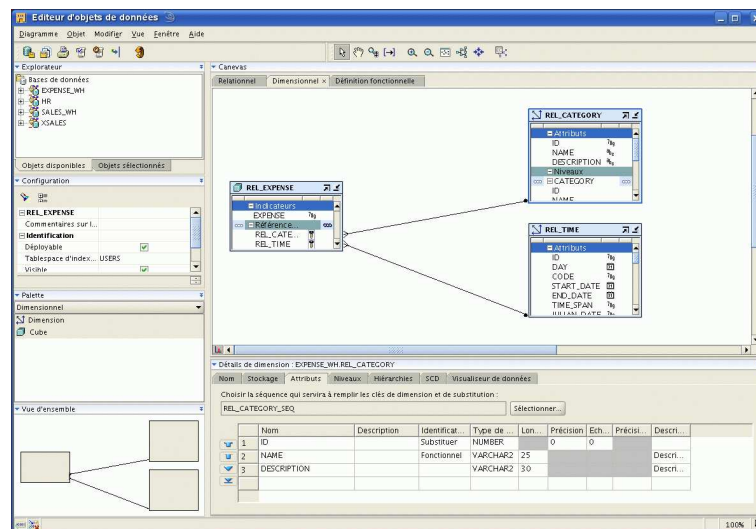


Figure 48: Visualiser le cube créé

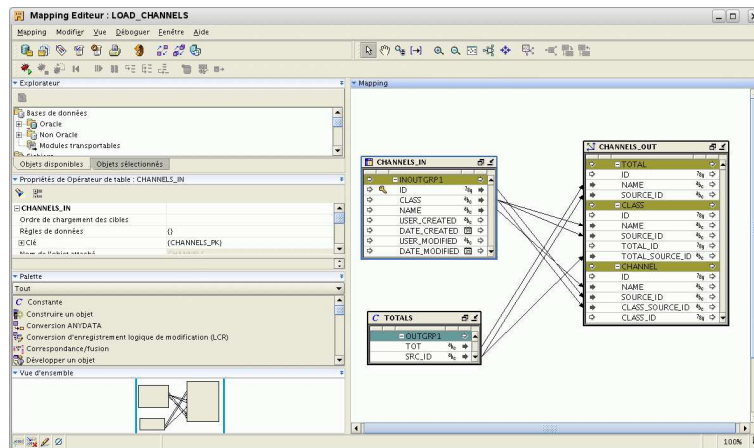


Figure 49: Exemple de mapping

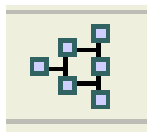


Figure 50: Icône de auto-positionnement

5.1 Concevoir les mappings

On va définir ici les mappings qui extraient les données des sources, les transforment and les chargent dans le module cible EXPENSE_WH.

5.1.1 Examiner un mapping pré-défini entre une table relationnelle et une dimension

Vous allez examiner le mapping pré-défini LOAD_CHANNEL entre la table relationnelle CHANNELS et la dimension CHANNELS.

1. dérouler le module OWB_DEMO>Databases>Oracle>SALES_WH>Mappings. Double-cliquer sur le nœud LOAD_CHANNELS. Vous obtenez la fenêtre de la figure 49. Vous pouvez cliquer sur icône du menu de la figure 50 pour voir tous les objets à leur taille originelle disposés au mieux dans la fenêtre, et le bouton figure 51 pour mettre une fenêtre d'opérateur à sa taille maximale.
2. sur la figure 49 vous voyez les opérateurs suivants:
 - un opérateur de *Table* appelé CHANNELS_IN



Figure 51: Bouton d'agrandissement

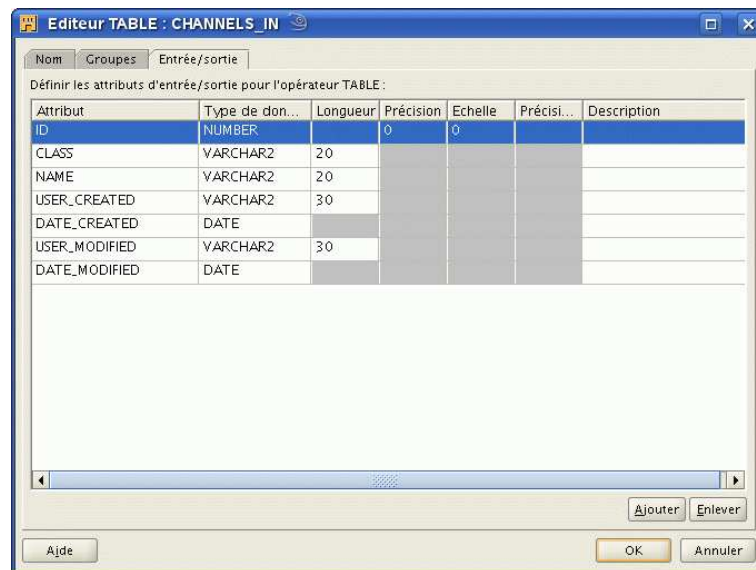


Figure 52: Editeur de l'opérateur Table

- un opérateur de *Constante* appelé TOTALS
 - un opérateur de *Dimension* appelé CHANNELS_OUT
 - des lignes de connexions entre les attributs
3. cliquer sur le cadre de CHANNELS_IN pour en faire l'objet sélectionné. Faire un clic droit sur le cadre et sélectionner Ouvrir détails pour lancer l'éditeur de Table. Vous obtenez la fenêtre de la figure 52. Regarder l'information disponible et modifiable en regardant chacun des onglet. Fermer ensuite cette fenêtre.
 4. Pour vérifier à quelle table est rattaché cet opérateur, cliquer sur le cadre de l'opérateur et regarder les information alors affiché dans la colonne propriétés de l'opérateur de table. Vous voyez que l'opérateur est lié à la table CHANNELS du schéma XSALES.
 5. dans le mapping LOAD_CHANNELS l'opérateur de dimension est nommé CHANNELS_OUT est est lié à CHANNELS dans le schéma cible SALES_WH.
 6. toujours dans ce mapping, l'opérateur constante nous permet de définir des valeurs constantes qui peuvent être utilisées n'importe quand lors du mapping. L'opérateur de constantes produit un seul groupe de sorties qui peut contenir une ou plusieurs attributs pour les constantes. Pour rajouter un opérateur de constante dans la fenêtre, vous pouvez faire un drag and drop de l'opérateur à partir de la palette. Un exemple d'utilisation d'un opérateur de constante est de charger la valeur de la date courante du système dans l'opérateur de table de sortie. Pour ajouter un attribut à l'objet de constante TOTALS, cliquer avec le bouton droit sur le cadre de l'opérateur de constante TOTALS et sélectionner Ouvrir détails. Cliquer sur l'onglet Attributs de sortie. en cliquant sur le bouton Ajouter vous pouvez ajouter des attributs à l'objet de constante TOTALS. Comme vous voyez (figure 53), 2 attributs ont déjà été ajoutés. Cliquer sur Annuler pour refermer la fenêtre.

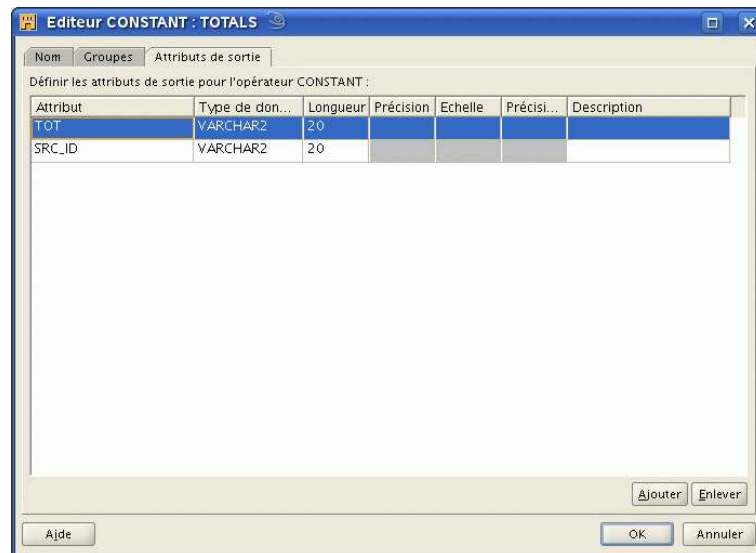


Figure 53: Editeur de la constante TOTAL

7. sélectionner l'attribut TOT de l'opérateur de constante TOTALS. A gauche, dans le panneau **Propriétés d'attribut** regarder en particulier le champ **Expression**. Cliquer sur ce champ, puis sur l'icone avec ... pour afficher le constructeur d'expressions comme sur la figure 54. La valeur de l'attribut TOT est la chaîne de caractères 'Channels Total'. Fermer l'éditeur d'expressions.
8. dans la fenêtre montrant le mapping sont représentées les lignes de connexion des:
 - attributs de l'opérateur de table CHANNELS_IN aux attributs de l'opérateur de dimension CHANNELS_OUT
 - attributs de l'opérateur de constante TOTALS aux attributs de l'opérateur de dimension CHANNELS_OUT

Dans cet exemple les opérateurs sont connectés en reliant individuellement les attributs des opérateurs entre-eux. Les lignes de connexion représentent graphiquement comment les données seront extraites depuis une source vers une cible, via des opérateurs. Fermer l'éditeur de mapping.

5.1.2 Concevoir un mapping entre une table externe et une dimension

1. Dans l'explorateur de projet, dérouler OWB_DEMO>Base de données>Oracle>EXPENSE_WH. Faire un clic droit sur Mappings et choisir Nouveau. Donner le nom pour le mapping de REL_CATEGORY_MAP. Dans l'explorateur de projet une entrée pour le mapping REL_CATEGORY_MAP a été ajoutée. L'éditeur de mapping se lance.
2. De la palette à gauche faire un drag and drop pour déposer un opérateur de dimension dans la fenêtre de représentation du mapping. La fenêtre d'ajout d'un opérateur de dimension apparaît comme sur la figure 55.
3. sélectionner dans la fenêtre d'ajout la dimension REL_CATEGORY, l'objet dimension est alors ajouté dans l'espace de représentation.

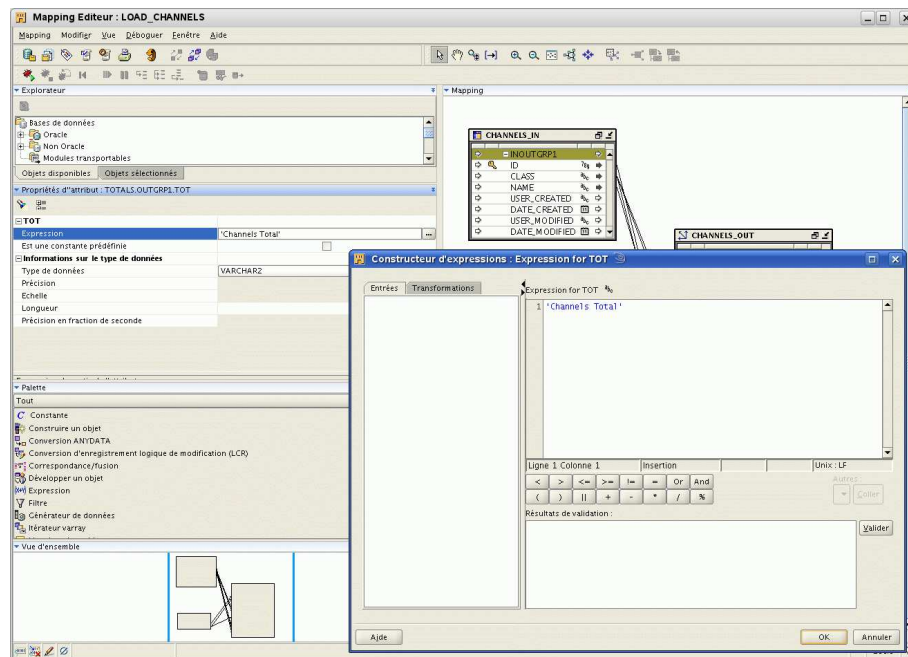


Figure 54: Constructeur d'expression

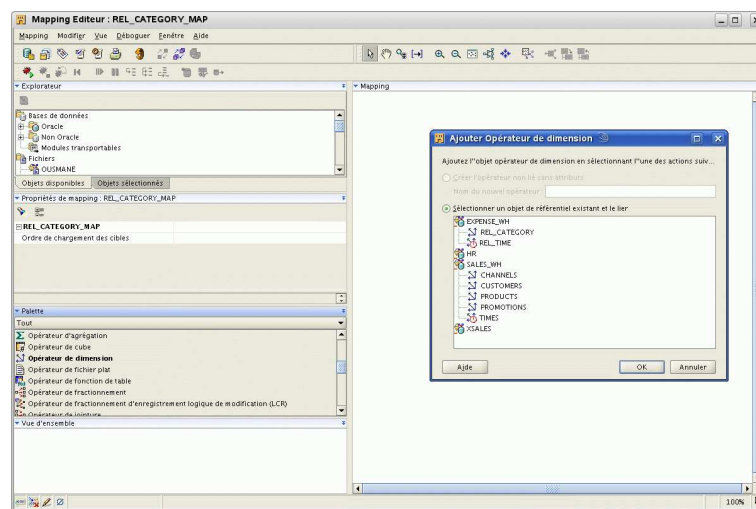


Figure 55: Fenêtre d'ajout d'opérateur

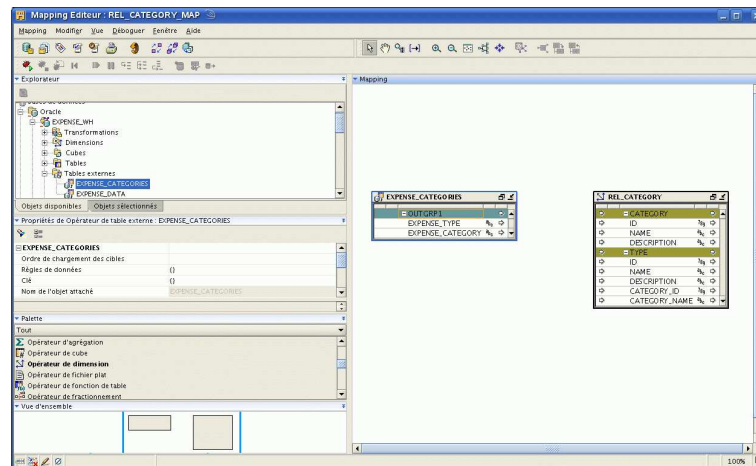


Figure 56: Ajouter une table externe

4. il est également possible d'ajouter des sources et cibles existantes à un mapping via le panneau d'exploration situé à gauche. Vérifier que l'onglet à gauche **Objets disponibles** est cliqué, puis trouver dans l'explorateur **Oracle>EXPENSE_WH>Tables externes**. Faire un drag and drop de **EXPENSE_CATEGORIES** dans la zone de représentation pour obtenir comme sur la figure 56.
5. connecter la source **EXPENSE_CATEGORIES** (table externe) à la cible **REL_CATEGORY** (la dimension) comme suit:
 - Drag and drop une ligne de connexion de **EXPENSE_CATEGORIES.OUTGRP1.EXPENSE_TYPE** vers **REL_CATEGORY.TYPE.NAME**
 - Drag and drop une ligne de connexion de **EXPENSE_CATEGORIES.OUTGRP1.EXPENSE_TYPE** vers **REL_CATEGORY.TYPE.DESCRPTION**
 - Drag and drop une ligne de connexion de **EXPENSE_CATEGORIES.OUTGRP1.EXPENSE_CATEGORY** vers **REL_CATEGORY.CATEGORY.NAME**
 - Drag and drop une ligne de connexion de **EXPENSE_CATEGORIES.OUTGRP1.EXPENSE_CATEGORY** vers **REL_CATEGORY.CATEGORY.DESCRPTION**
 - Drag and drop une ligne de connexion de **EXPENSE_CATEGORIES.OUTGRP1.EXPENSE_CATEGORY** vers **REL_CATEGORY.TYPE.CATEGORY_NAME**.

Vous obtenez la figure 57. Attention d'avoir bien vérifié lors des connexions que vous avez bien lié vers les attributs du niveau demandé car les 2 niveaux (CATEGORY et TYPE ont les mêmes attributs.

6. vous avez fini de concevoir le mapping. Maintenant générer le code du mapping en sélectionnant dans le menu **Mapping** l'entrée **Générer**. OWB génère alors le code permettant d'extraire les données et peupler la dimension (image 58). Fermer ensuite l'éditeur de mapping. Sauver.

5.1.3 Concevoir un mapping entre une table externe et un cube

1. se placer dans **OWB_DEMO>Bases de données>Oracle>EXPENSE_WH**. Cliquer sur **Mappings** et choisir **New**. Choisir le nom **REL_EXPENSE_MAP** pour le mapping.

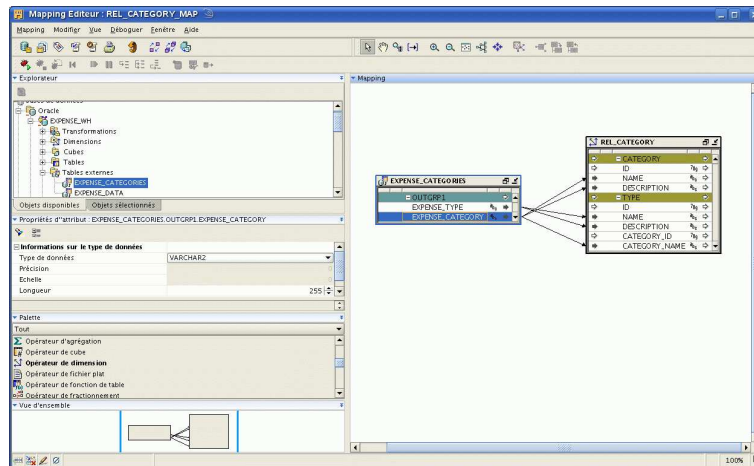


Figure 57: Création des connexions

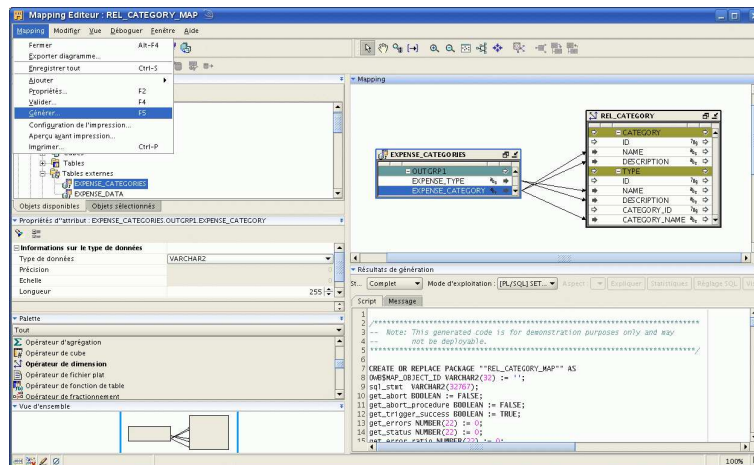


Figure 58: Générer le code

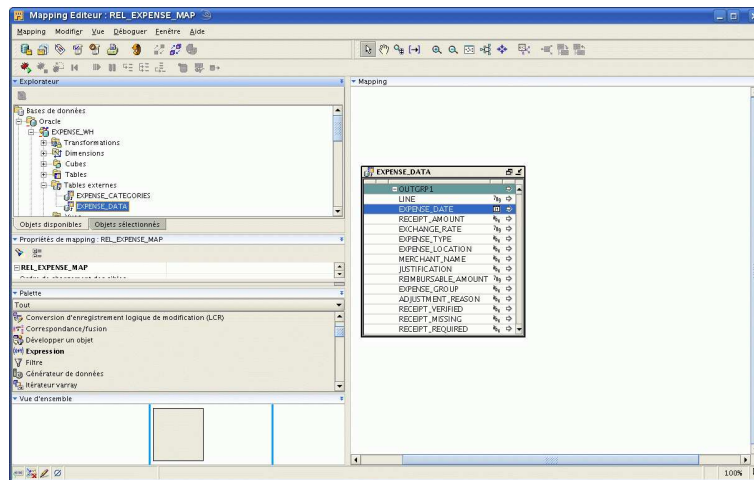


Figure 59: Ajouter une table externe au mapping

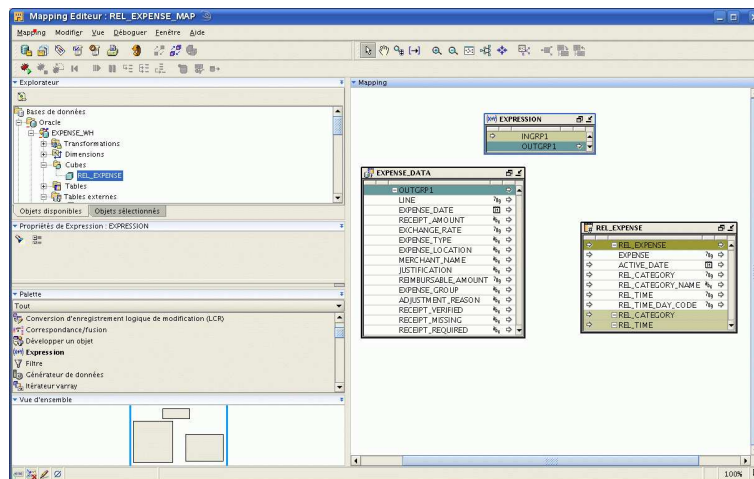


Figure 60: Ajouter opérateur d'expression

2. à gauche, dans le panneau d'exploration, vérifier que l'onglet Objets disponibles est sélectionné. Dérouler Oracle>EXPENSE_WH>Tables externes. Faire un drag and drop de EXPENSE_DATA dans la fenêtre de présentation (image 59).
3. faire de même pour déposer le cube REL_EXPENSE qui se trouve dans Oracle>EXPENSE_WH>Cubes dans la zone de présentation à droite du composant EXPENSE_DATA
4. utiliser l'éditeur de mapping pour faire un drag and drop d'un opérateur d'expression dans la fenêtre de présentation comme sur la figure 60. Utiliser les icônes des images 50 et 51 pour agrandir et mieux disposer les fenêtres des opérateurs.
5. tracer une connexion depuis l'attribut EXPENSE_DATA.OUTGRP1.EXPENSE_DATE vers l'attribut EXPRESSION.INGRP1 (voir figure 60)

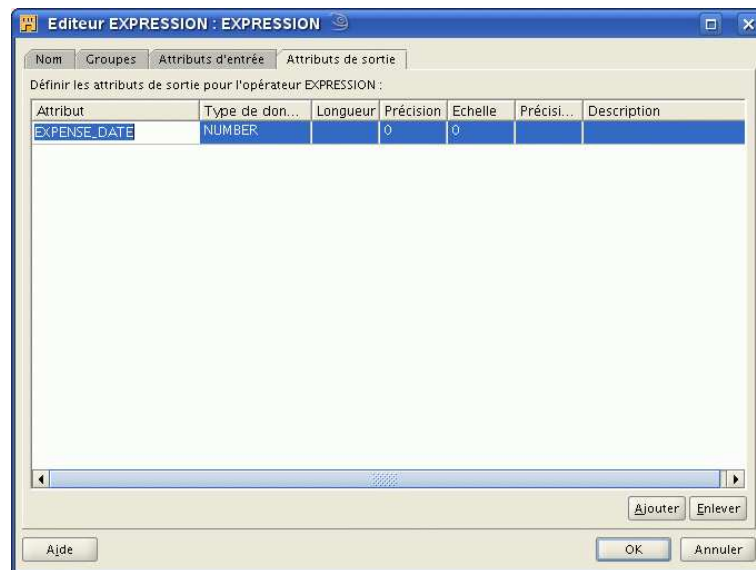


Figure 61: Ajouter un attribut de sortie à l'opérateur d'expression

6. faire un clic droit sur le cadre de l'opérateur d'expression et ouvrir les détails. Dans l'éditeur d'expression cliquer sur l'onglet **Attributs de sortie**.
7. cliquer sur **Ajouter**. Renommer OUTPUT1 en EXPENSE_DATE et vérifier que le type est bien NUMBER (figure 61). Cliquer OK.
8. cliquer sur le cadre de l'opérateur d'expression, puis sur l'attribut OUTGRP1.EXPENSE_DATE. A gauche vous avez une fenêtre d'éditeur d'attribut, cliquer sur le champ Expression puis sur . . .
9. dans l'éditeur d'expression taper `TO_NUMBER( TO_CHAR( INGRP1.EXPENSE_DATE , 'YYYYMMDD' ), '99999999' )`. (voir image 62). Cliquer sur Valider et si ok cliquer sur Ok.
10. Connecter l'opérateur source à l'opérateur cible comme suit:
 - tracer une ligne de connexion depuis `EXPRESSION.OUTGRP1.EXPENSE_DATE` vers `REL_EXPENSE.REL_EXPENSE`
 - tracer une ligne de connexion depuis `EXPENSE_DATA.OUTGRP1.EXPENSE_TYPE` vers `REL_EXPENSE.REL_EXPENSE`
 - tracer une ligne de connexion depuis `EXPENSE_DATA.OUTGRP1.REIMBURSABLE_AMOUNT` vers `REL_EXPENSE.REL_EXPENSE.EXPENSE`

Vous obtenez la figure 63.

11. la création du mapping est terminée. On peut générer le code correspondant (cliquer sur **Générer** dans le menu Mapping. Sauver.

5.2 Concevoir les process flows (flux de processus)

Après avoir créé les mappings qui définissent les opérations pour déplacer les données des sources vers les cibles, vous pouvez créer et définir des process flows. Les process flows mettent en relation les mappings et les activités externes à OWB, par exemple des emails. Dans OWB, un process flow est utilisé pour

Figure 64: Créer un compte utilisateur de workflow

gérer l'exécution de mappings et les activités liées, telles que le téléchargement de fichier sur le serveur de l'entrepôt en utilisant FTP ou en notifiant l'administrateur d'une erreur dans le processus de chargement.

Le scénario que nous allons regarder est classique. La dimension `REL_TIME` et `REL_CATEGORY` doivent être chargées après le cube `REL_EXPENSE` parce que le cube référence les valeurs des dimensions. Le chargement du cube débutera seulement après que les différentes dimensions soient chargées.

L'éditeur de process flow inclut une grande variété de d'activités que vous pouvez ajouter et relier ensuite avec des transitions pour concevoir un flow (flux). Une activité représente des unités de travail dans un process flow. Ces unités de travail peuvent impliquer des composants internes ou externes de OWB. Les transitions indiquent la séquence et les conditions à satisfaire pour pouvoir lancer les activités.

5.2.1 Enregistrer l'utilisateur de workflow oracle

Vous devez donner des rôles spécifiques à l'utilisateur de workflow oracle, le rôle `owf_mgr` de oracle, afin qu'il ait les privilèges pour exécuter un process flow dans OWB.

1. pour créer un utilisateur et un compte pour le workflow, lancer dans la console l'assistant avec la commande

```
../../../../wf/install/wfinstall.csh
```
2. saisir les paramètre de la figure 64: pour le compte mettre `owb_mge_votreLogin`, votre mot de passe, me demander le mot de passe système, pour le TNS mettre `localhost:1521:nfe115`. Valider. Attendre que la fenêtre de la console n'affiche plus de messages et affiche une fenêtre de succès.

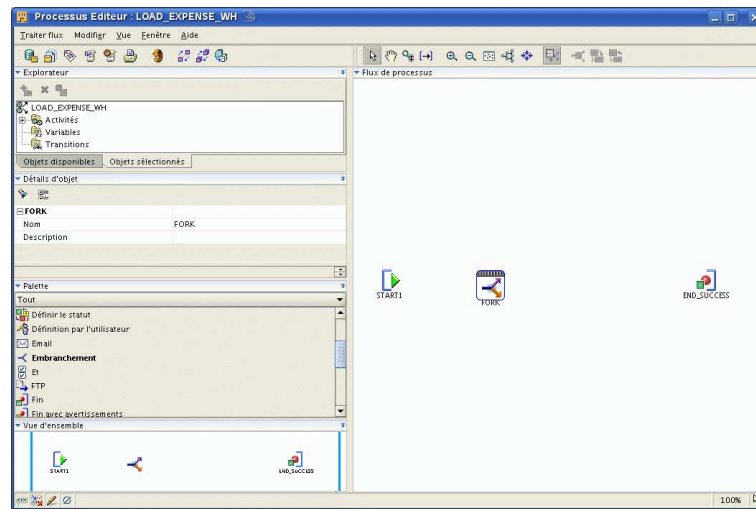


Figure 65: Créer un workflow (1)

3. se logger comme propriétaire du référentiel, se placer dans le nœud Sécurité, puis Utilisateur. Cliquer et choisir Nouveau.
4. vous pouvez dans la fenêtre de dialogue de création d'utilisateur sélectionner un utilisateur existant ou en créer un nouvel utilisateur. Choisir owb_mge_votreLogin.

5.2.2 Concevoir un process flow

1. avant de pouvoir commencer la conception du process flow, vous devez créer un module de process flow qui inclut un package de process flow. Créer un module de process flow appelé MY_PF_MODULE en faisant un clic droit dans OWB_DEMO > Flux de processus > module de flux, puis Nouveau.
2. créer un nouvel emplacement pour le module. Cet emplacement spécifie où vous voulez déployer vos process flows. Remarquer que OWB créer un emplacement avec un nom par défaut MY_PF_MODULE_LOCATION.
3. cliquer Editer pour spécifier les détails de l'emplacement du workflow. Changer le nom de l'emplacement en WH_OWF_LOCATION. Préciser les valeurs suivantes: (Password: le votre, Host: localhost, Port: 1521, Service: nfe115, Schema: owf_mgr_dumou_ce, Version: 2.6.4). Cliquer sur Fin.
4. dans la nouvelle fenêtre Création de Package de flux de processus saisir le nom PK
5. dans la nouvelle fenêtre intitulée Création de flux de processus donner le nom LOAD_EXPENSE_WH
6. OWB lance la fenêtre de l'éditeur de processus de la figure 65. Par drag and drop ajouter l'activité Embranchement dans le fenêtre. Cette activité permet de lancer plusieurs activités concurrentes après qu'une activité se soit achever. Vous obtenez alors la figure 65.
7. dans l'éditeur de processus, dans le panneau d'exploration, cliquer sur l'onglet Objets disponibles. Dans le module cible EXPENSE_WH prendre REL_CATEGORY_MAP et le déposer dans la fenêtre comme sur la figure 66.

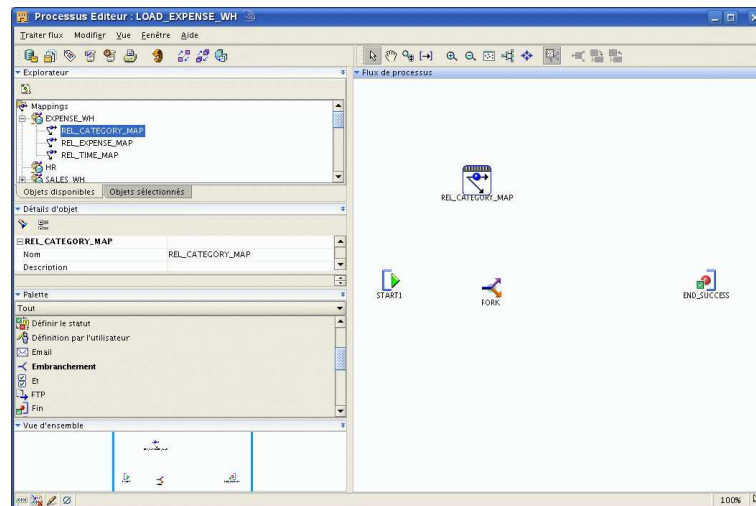


Figure 66: Créer un workflow (2)

8. dans l'éditeur de processus, dans le panneau d'exploration, cliquer sur l'onglet **Objets disponibles**. Dans le module cible **EXPENSE_WH** prendre **REL_TIME_MAP** et le déposer dans la fenêtre. Réarranger pour avoir une disposition comme sur la figure 67.
9. de la palette, faire glisser une activité **Et** à droite de **REL_CATEGORY_MAP** et **REL_TIME_MAP**.
10. dans le panneau de l'explorateur, cliquer sur l'onglet **Objets disponibles**. Dans le module cible **EXPENSE_WH** prendre **REL_EXPENSE_MAP** et le déposer dans la fenêtre à droite de l'activité **Et** voir image 68.
11. du panneau de la palette, faire glisser une activité **Fin avec erreurs** et **Fin avec avertissement** comme sur la figure 69.
12. connecter ensuite les différentes activités. Pour cela placer la souris sur l'activité **start** et vérifier que la souris se change en $\rightarrow$. Tracer une ligne depuis l'activité **Start1** jusqu'à **Fork**. Vous devez tracer en tout les connexions suivantes:
 - de **START1** à **FORK**
 - de **FORK** à **REL_CATEGORY_MAP**
 - de **FORK** à **REL_TIME_MAP**
 - de **REL_CATEGORY_MAP** à **AND1**
 - de **REL_TIME_MAP** à **AND1**
 - de **AND1** à **REL_EXPENSE_MAP**
 - de **REL_EXPENSE_MAP** à **END_ERROR**
 - de **REL_EXPENSE_MAP** à **END_SUCCESS**
 - de **REL_EXPENSE_MAP** à **END_WARNING**

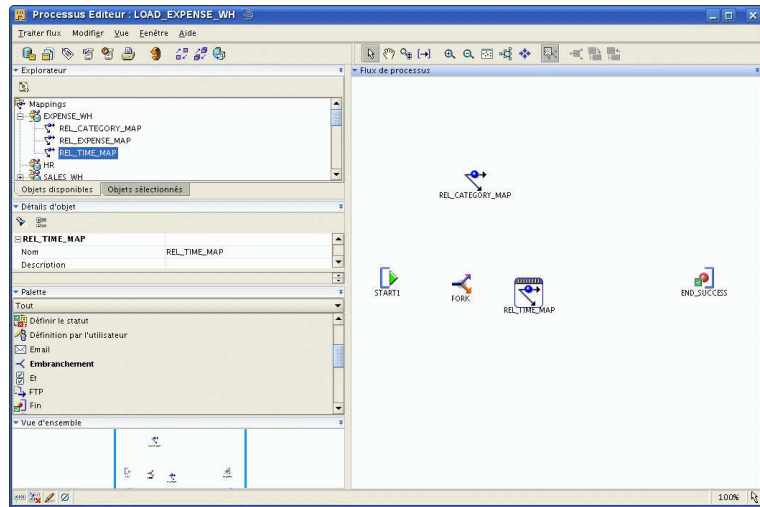


Figure 67: Créer un workflow (3)

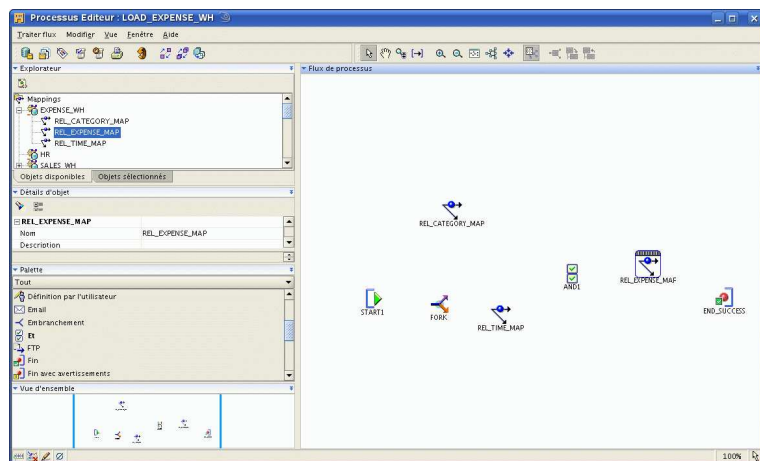


Figure 68: Créer un workflow (4)

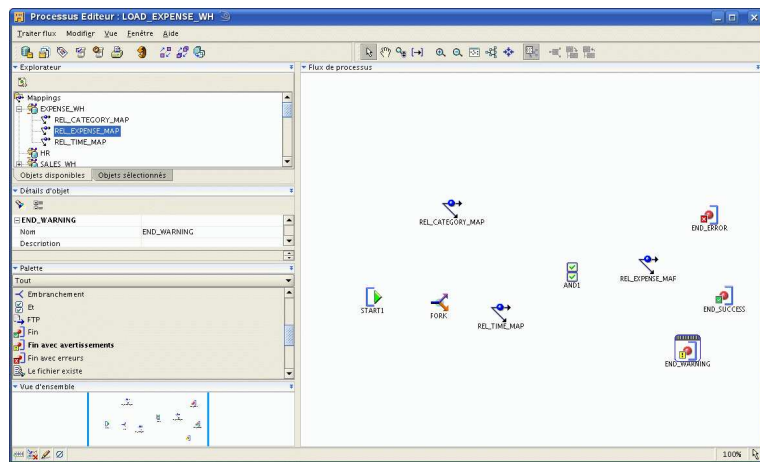


Figure 69: Créer un workflow (5)

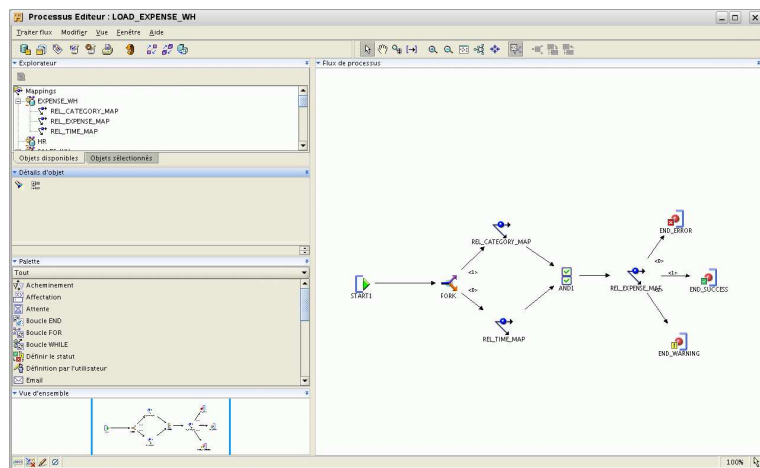


Figure 70: Créer un workflow (6)

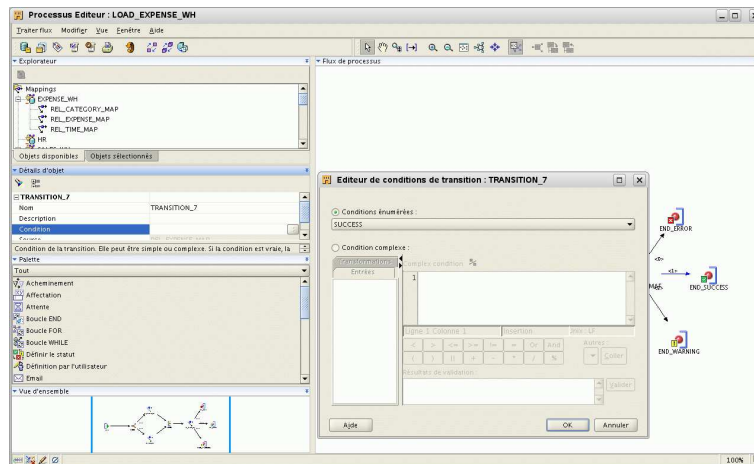


Figure 71: Créer un workflow (7)

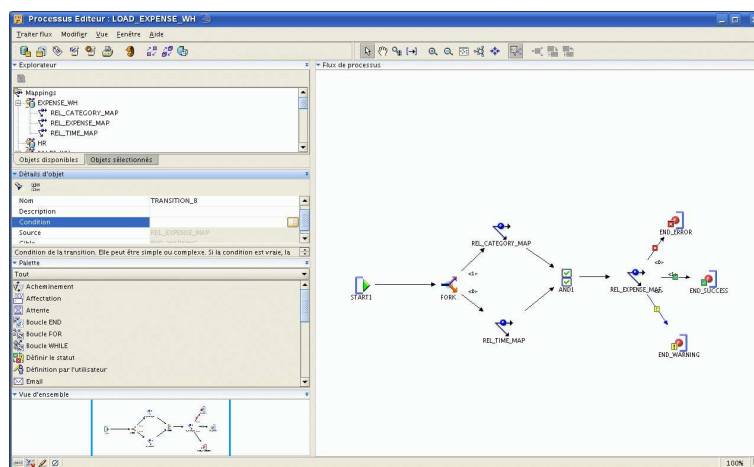


Figure 72: Créer un workflow (8)

Vous obtenez normalement la figure 70 Les transitions que vous avez ajouté sont des transitions sans condition. Par défaut c'est toujours le cas. Le process flow continue après que l'activité précédente s'achève quel que soit l'état final de l'activité précédente.

13. vous pouvez utiliser une transition pour exécuter une activité s'appuyant sur l'état final d'une activité précédente en créant une condition. Pour cela sélectionner la transition de REL_EXPENSE_MAP à END_SUCCESS. Dans le panneau de détail de l'objet à gauche, cliquer sur : dans la ligne condition. Choisir dans le menu déroulant la condition SUCCES (voir figure 71).
14. faire de même avec la transition de REL_EXPENSE_MAP à END_ERROR en choisissant cette fois ERROR comme condition. Cela indique que le process flow continue par cette transition si l'activité précédente termine avec une erreur.
15. faire de même avec la transition de REL_EXPENSE_MAP à END_WARNING. Vous obtenez la figure 72

16. générer le code de `LOAD_EXPENSE_WH` en sélectionnant `Generate` dans le menu du process flow. Le code généré est dans un langage standard de l'industrie appelé XML Process Definition Language (XPDL).
17. fermer la fenêtre en sauvant tout.