


Modélisation de l'entrepôt : Environnement

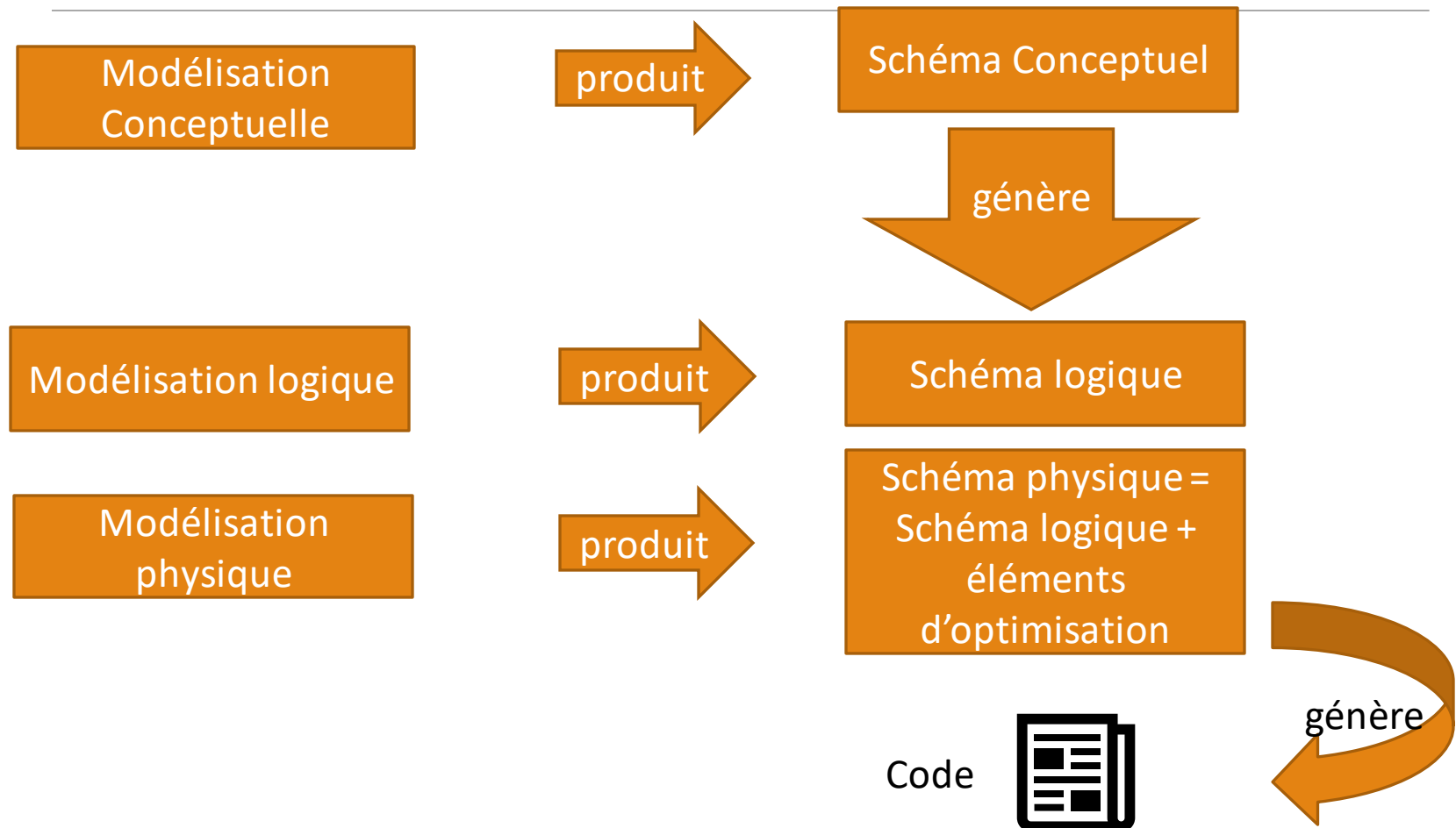
GIUSEPPE BERIO

Objectifs

 Apprendre la modélisation d'un entrepôt/datamart pour une mise en œuvre ROLAP

Apprendre la modélisation d'un entrepôt/datamart pour une mise en œuvre MOLAP

Rappel : niveaux de modélisation



Modèles de données (rappel)

Modèle conceptuel de données

- Éléments pour la représentation (précise, souvent dans un langage formalisé) de l'information sous-jacente les données, indépendants de toute mise en œuvre (par exemple ER, UML, Merise)

Modèle logique de données

- Éléments pour la représentation de données suivant une mise en œuvre informatique, mais indépendants de l'outil permettant cette mise en œuvre (par exemple, le modèle relationnel)

Modèle physique de données

- Un modèle de stockage de données représentées par le modèle logique correspondant typiquement à un modèle logique augmenté avec des paramètres propres au stockage (par exemple, des indexes)

Oracle SQL Developer (SQLD)

Outil gratuit : vous pouvez l'installer sur vos ordinateurs

Environnement de développement pour systèmes opérationnels et systèmes décisionnels

Accès à ORACLE, envoi de commandes SQL, DDL (Data Definition Language) et DML (Data Manipulation Language)

3-niveaux de modélisation (terminologie propre à Oracle)

Génération automatique/importation des schémas logiques et physiques (DDL) pour ORACLE (et autres SGBD comme DB2 et SQLServer)

Comparaison/Fusion de schémas

Importation/Exportation vers XMLA (Microsoft), OracleAW (Oracle) et CubeViews (IBM) pour le modèle multidimensionnel

Retro-ingénierie via connexion à une base relationnelle existante (ce qui probablement explique le nommage pour les 3 niveaux de modélisation)

SQLD : accès oracle

Oracle SQL Developer

Fichier Modifier Affichage Naviguer Exécuter Equipe Outils Fenêtre Aide

Connexions

- Connexions
- compteorade
- Connexions Oracle NoSQL
- Connexions au service de schéma de base de données

Rapports

- Tous les rapports
- Rapports Data Modeler
- Rapports de dictionnaire de données
- Rapports définis par l'utilisateur
- Rapports de vue analytique
- Rapports OLAP
- Rapports TimesTen

Page de début

Créer / Sélectionner une connexion de base de données

Nom de connexion	Détails de connexion
compteorade	BERIO_BDD@//...

Nom de connexion:

Nom utilisateur:

Mot de passe:

☐ Enregistrer le mot de passe ☒ Couleur de connexion

Oracle

Type de connexion: Rôle:

Nom d'hôte:

Port:

☒ SID:

☐ Nom de service:

☐ Authentification OS ☐ Authentification Kerberos

Statut:

Tous les tutoriels en ligne

Toutes les démonstrations en ligne

☒ Afficher au démarrage

Copyright © 2005, 2017, Oracle and/or its affiliates. All rights reserved.

Nom utilisateur : le vôtre

Mot de passe : le vôtre

Nom d'hôte : ora12c

SID : ORAETUD

SQLD : une fois connecté à Oracle

The screenshot displays the Oracle SQL Developer interface. The 'Connexions' (Connections) pane on the left shows a tree view of database objects under the 'moncompteoracle' connection. The main workspace shows a SQL query 'select * from client;' in the 'Feuille de calcul' (Worksheet) tab. The 'Fenêtre exécution d'instructions' (Execute Statement window) is open, showing the execution of the query. The 'Fenêtre montrant les résultats de l'exécution' (Execute Statement Results window) displays the results of the query in a table format.

Visualisation des objets dans la base de données

Connexions

- compteoracle
- moncompteoracle
 - Tables (filtré)
 - CATEGORIE
 - CLIENT
 - COMMANDE
 - CUBE_TEMPLATES
 - EMPLOYES
 - ENSEIGNANT
 - ETUDIANT
 - MATIERE
 - NOTE

Page de début

moncompteoracle

0,053 secondes

Feuille de calcul

Query Builder

select * from client;

Icones d'exécution

Fenêtre exécution d'instructions

Sortie de script

Résultat de requête

Tâche terminée en 0,053 secondes

NOM	PRÉNOM	ID	VILLE_ID
roger	jean	1	1

Fenêtre montrant les résultats de l'exécution

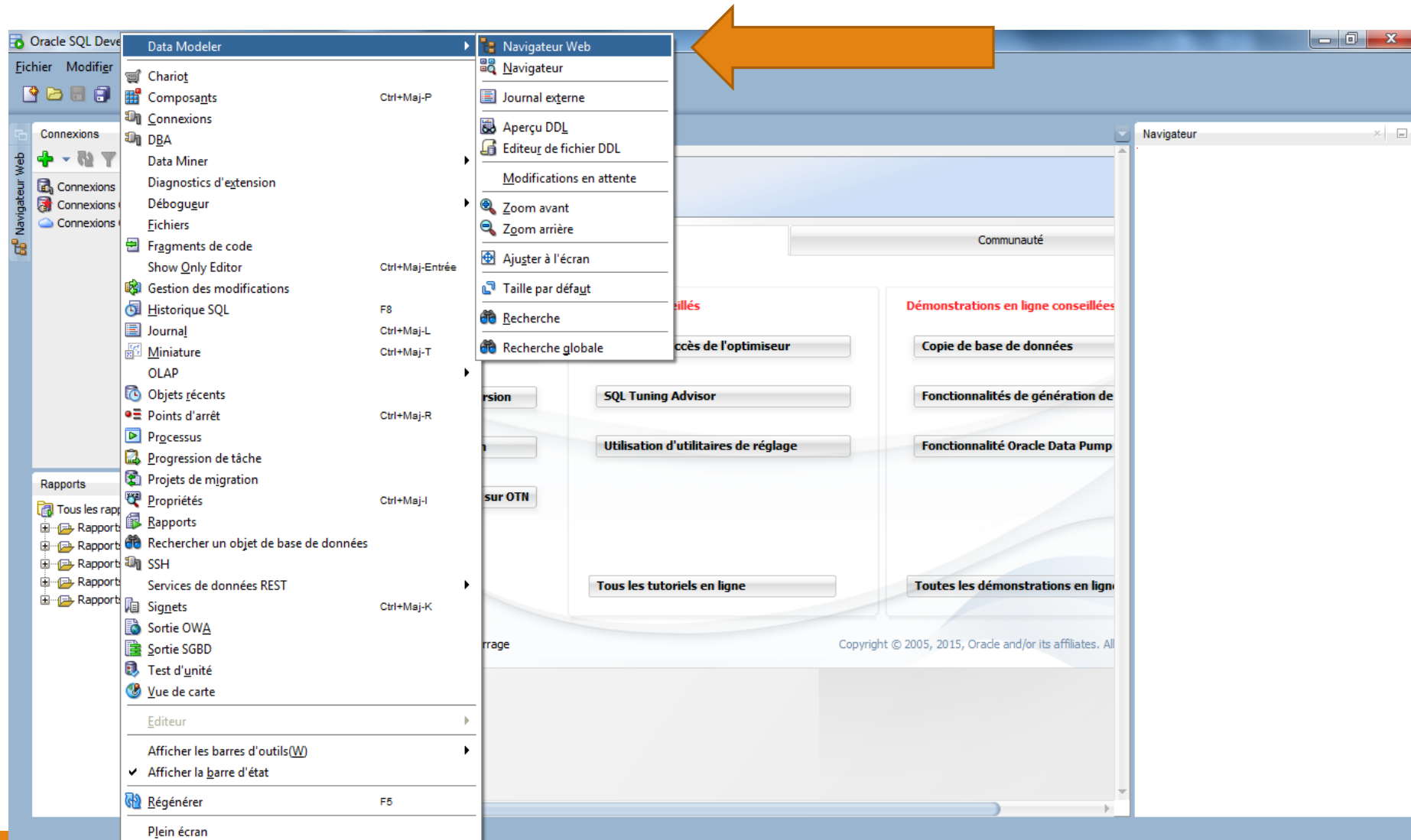
NOM	PRÉNOM	ID	VILLE_ID
roger	jean	1	1

Rapports

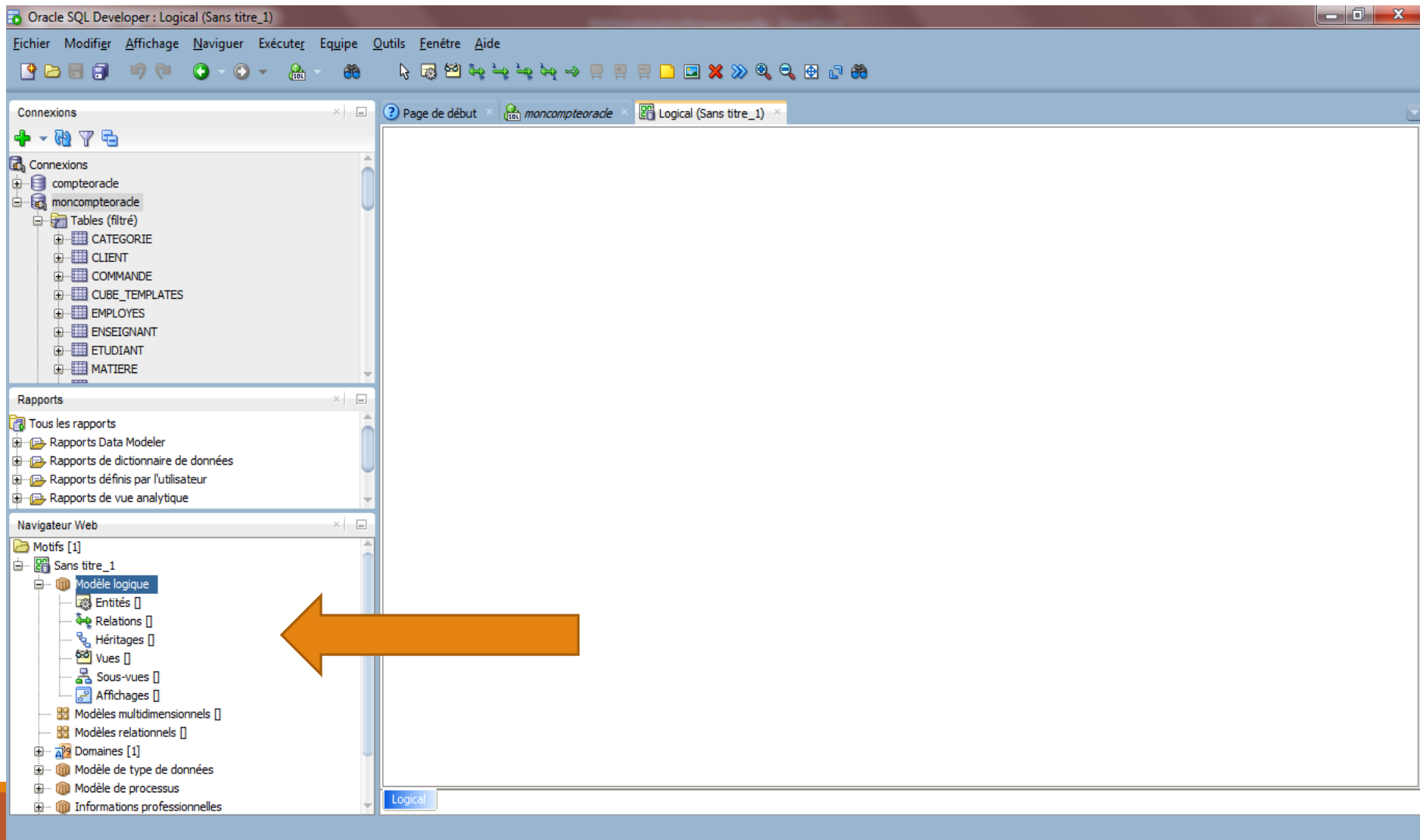
- Tous les rapports
- Rapports Data Modeler
- Rapports de dictionnaire de données
- Rapports définis par l'utilisateur
- Rapports de vue analytique
- Rapports OLAP
- Rapports TimesTen

Ligne 2 Colonne 1 | Insertion | Modifié | Windows : C

SQLD : ACCÈS INTERFACE DE MODÉLISATION



SQLD : une fois lancée l'interface de modélisation

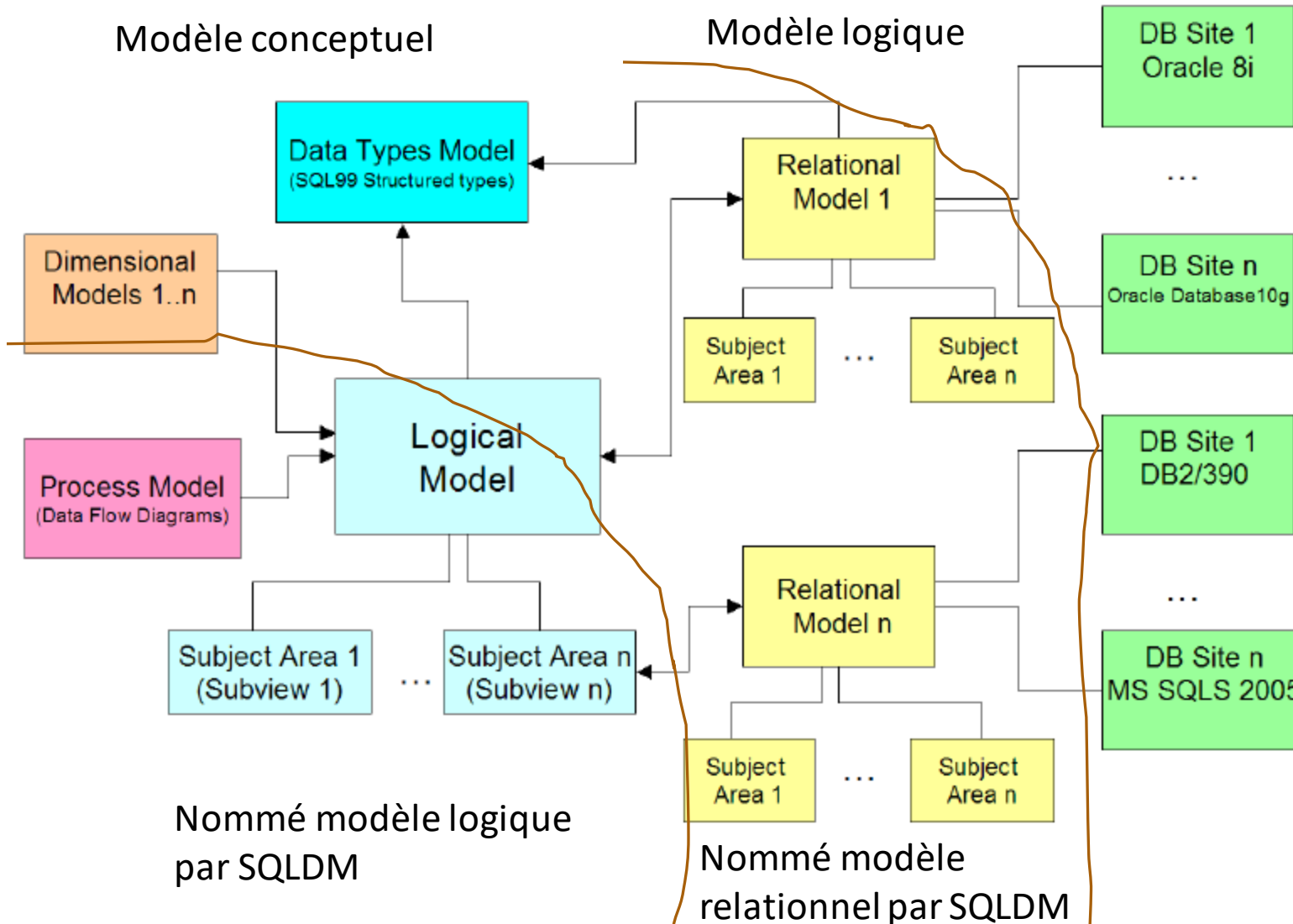


Modèles dans SQLD(M)

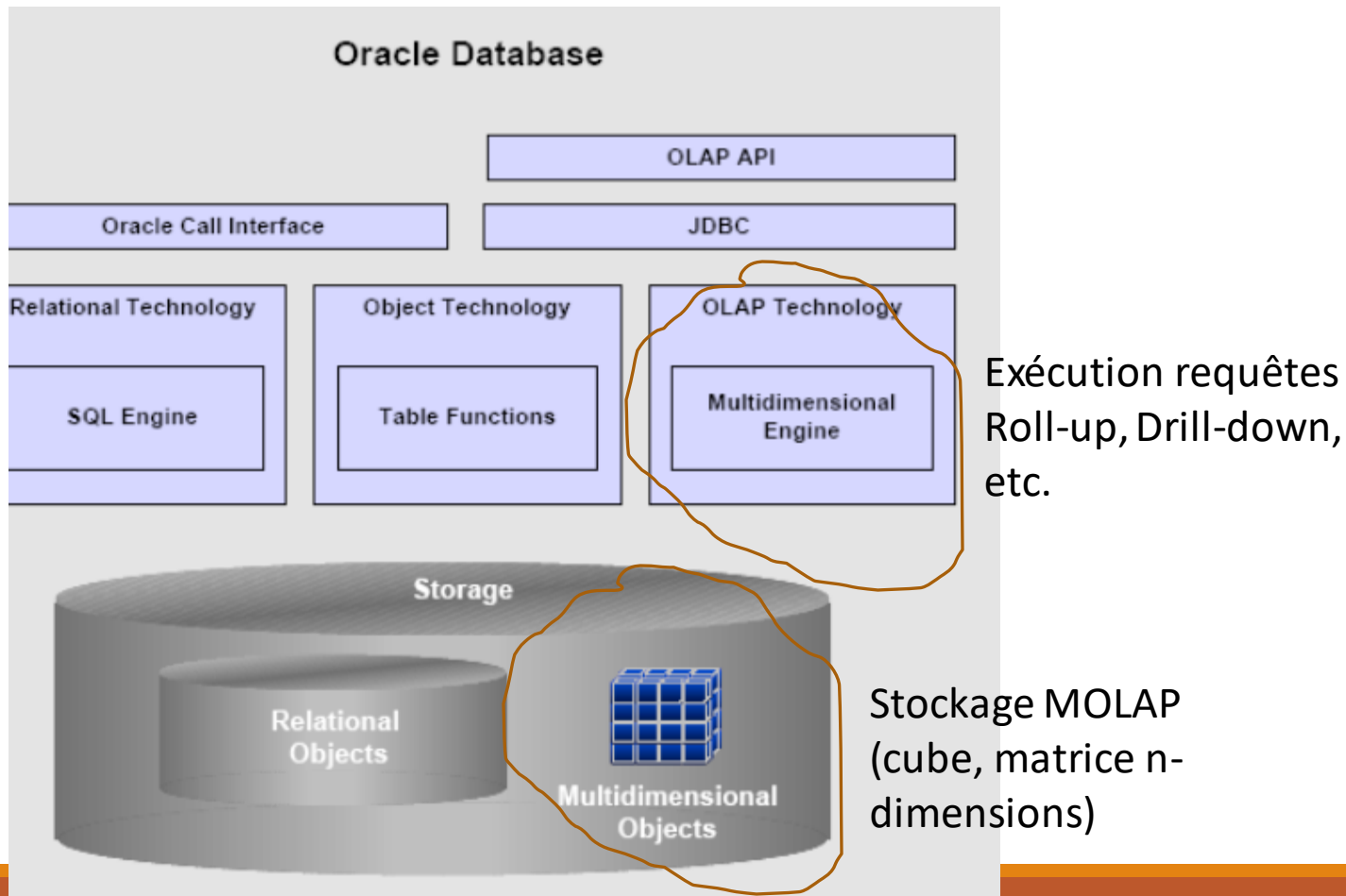
Modèle physique

Modèle conceptuel

Modèle logique



Oracle Analytical workspace



Objectifs du TD

Compréhension de la modélisation multi-dimensionnelle par les pratiques de modélisation avec SLQD dans un contexte ROLAP (exercices)

- Modélisation des sources/données traitées par l'ETL
- Modélisation (conceptuelle) dimensionnelle
- Génération du schéma logique
- Compréhensions du schéma physique
- Génération du code

SQLD, modélisation des sources et pour les données traitées par un ETL

Création d'un schéma conceptuel (notations interchangeables Bachman/Barker (alternatives à MERISE et autres notations)

- Organisation en *sous-vues* ou *affichages*, avec possibilité d'ajouter des vues
- Support pour *l'objet-relationnel*

Génération automatique d'un schéma relationnel à partir du schéma conceptuel

- Cela **vous évite** de le faire manuellement (gain de temps et limitation di risque d'erreur)
- L'organisation est la même que le schéma conceptuel

Support pour la définition d'un schéma physique (à partir du schéma relationnel)

Génération automatique du DDL (déploiement vers le SGBD à utiliser)

- Cela **vous évite** de le faire manuellement (gain de temps et limitation du risque d'erreur)

Utilisation de la retro-ingénierie pour reconstituer un schéma d'une base (relationnelle) existante

- Essentiel pour la ré-conception d'une source ou pour sa compréhension et utilisation dans le but de développer un schéma dimensionnel

Rappel : fonctionnement ROLAP

Données brutes (sources)

ClientID	Quantité	Produit	Date
1	100	54	01/01/2013
1	200	76	02/02/2013

1, 100a, 54, 01/02/2013; 2, 200, 56, 02/01/2013

Composant ETL (intégration et fiabilisation de données) – données disponibles

ClientID	Quantité	Produit	Date
1	100	54	01/01/2013
1	200	76	02/02/2013
3	100	54	01/02/2013
2	200	56	02/01/2013

Mesure
précalculée

ClientID	Σ Quantité
1	300
3	100
2	200

Mesure

ClientID	Σ Quantité	Produit	Date
1	100	54	01/01/2013
1	200	76	02/02/2013
3	100	54	01/02/2013
2	200	56	02/01/2013

Composant ETL (alimentation de la table de faits/agrégation)

Σ Quantité	Produit
200	54
200	76
200	56

Une ligne=Un fait, à savoir une *observation*, représentée par une mesure, sur le passé → table de faits

Composant
d'entreposage

Σ Quantité	Mois
300	01/2013
300	02/2013

Modèle logique
dimensionnel ROLAP

Σ Quantité	Année
600	2013

Dimension et hiérarchie

Démo fonctionnement 2 (ROLAP)

Composant ETL (extraction
de données) – Staging

Non typé

correction/typage

Typé

1, 100a, 54, 01/02/2013; 2,
200, 56, 02/01/2013

1, 100, 54, 01/02/2013; 2,
200, 56, 02/01/2013

produit inexistant

ClientID	Quantité	Produit	Date
1	100	54	01/01/2013
1	200	78VGVS	02/02/2013

suppression

ClientID	Quantité	Produit	Date
1	100	54	01/01/2013

normalisation

ClientID	Quantité	Produit	Date	ville
1	100	54	01/01/2013	Nantes
1	200	78	02/02/2013	Nantes

ClientID	Quantité	Produit	Date
1	100	54	01/01/2013
1	200	78	02/02/2013

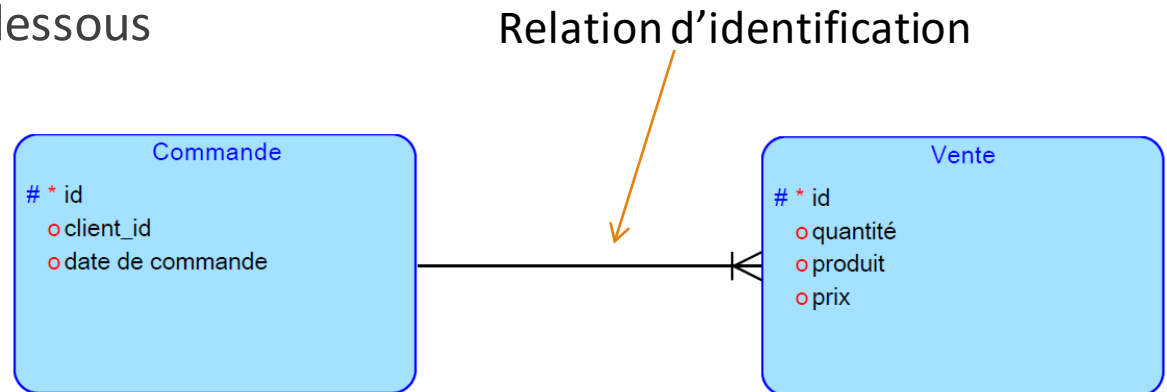
ClientID	ville
1	Nantes

ClientID	Quantité	Produit	Date	Ville d'achat
1	100	54	01/01/2013	Nantes
1	200	78	02/02/2013	Nantes

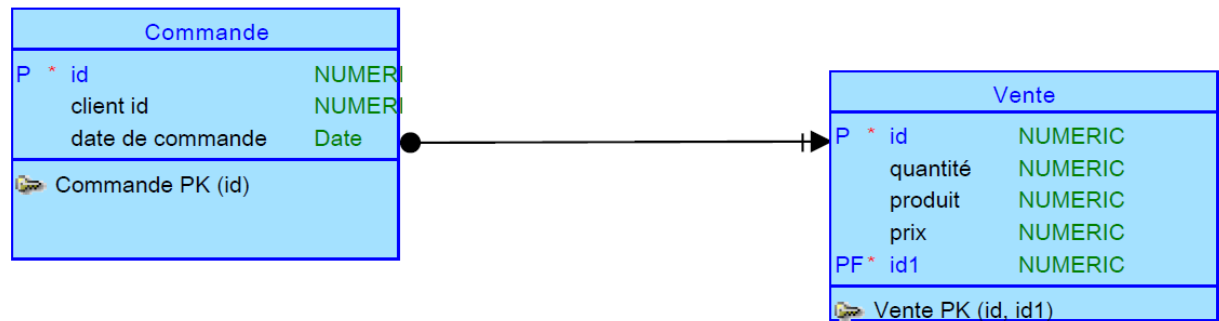
Exercice 1 (représentation de données disponibles : sources ou staging ETL)

Créer le *schéma conceptuel* (on pourrait faire cela avec UML et quelques adaptations) ci-dessous dans un *schéma local*

Barker



Bachman
(Affiche les types et
les clés étrangères)



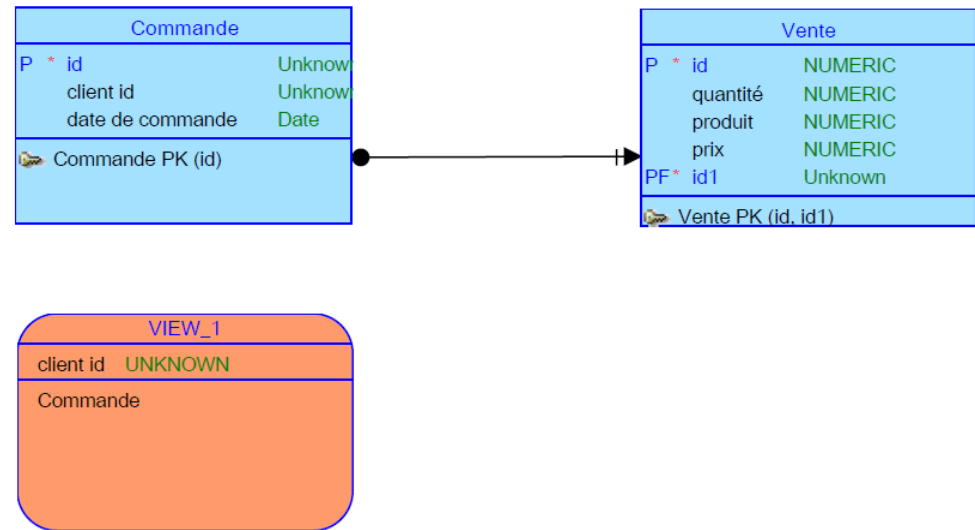
Exercice 1 bis

Parfois il est utile de spécifier une vue (externe) à savoir une requête nommée

Rajoutez **une vue (externe)** au schéma conceptuel de l'exercice 1

Vue : tous les mois où il y a au moins une vente

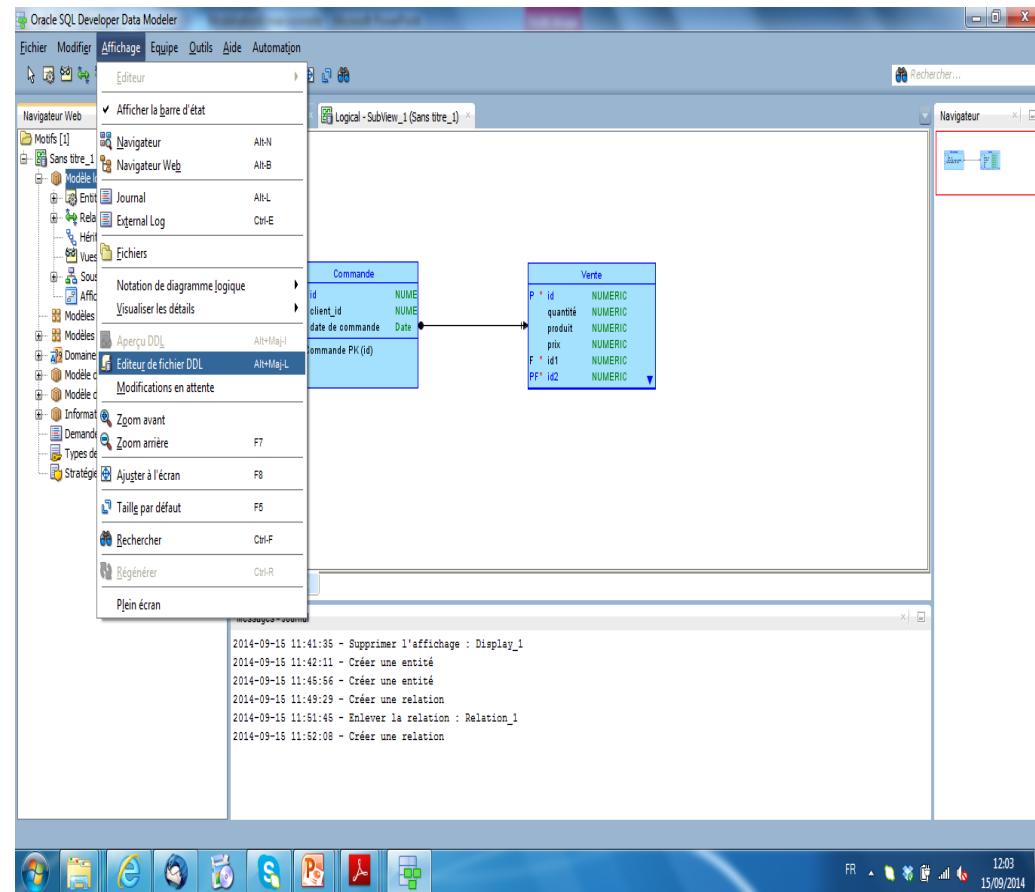
Affichez la vue dans le schéma local



Exercice 2 (réalisation des bases de données extraites, nettoyées, intégrées)

Générez automatiquement un ***schéma relationnel*** à partir du schéma de l'exercice 1

Ensuite, générez automatiquement le ***DDL (Data Definition Language script)*** à partir du schéma relationnel (code SQL)



Exercice 2 bis

Modifiez le ***schéma relationnel*** résultat de l'exercice 2 (par exemple, modifier un type, rajouter une colonne, modifier la vue, etc.)

Régénérez le schéma conceptuel

Refaites les 2 actions ci-dessus plusieurs fois et analysez l'impact sur le schéma conceptuel

Exercice 3

Utilisez le DDL généré (EXERCICE 2) pour créer les tables et les autres éléments dans votre espace ORACLE 12c :

- Connectez vous à ORACLE 12c (si pas encore connecté ou connexion perdue)
- Copiez-collez le script DDL généré dans la fenêtre d'exécution d'instructions
- Exécutez ce script utilisant une des icones appropriées
- Vérifiez que l'exécution a bien fonctionné (aucune erreur n'est affichée dans la fenêtre de résultats)

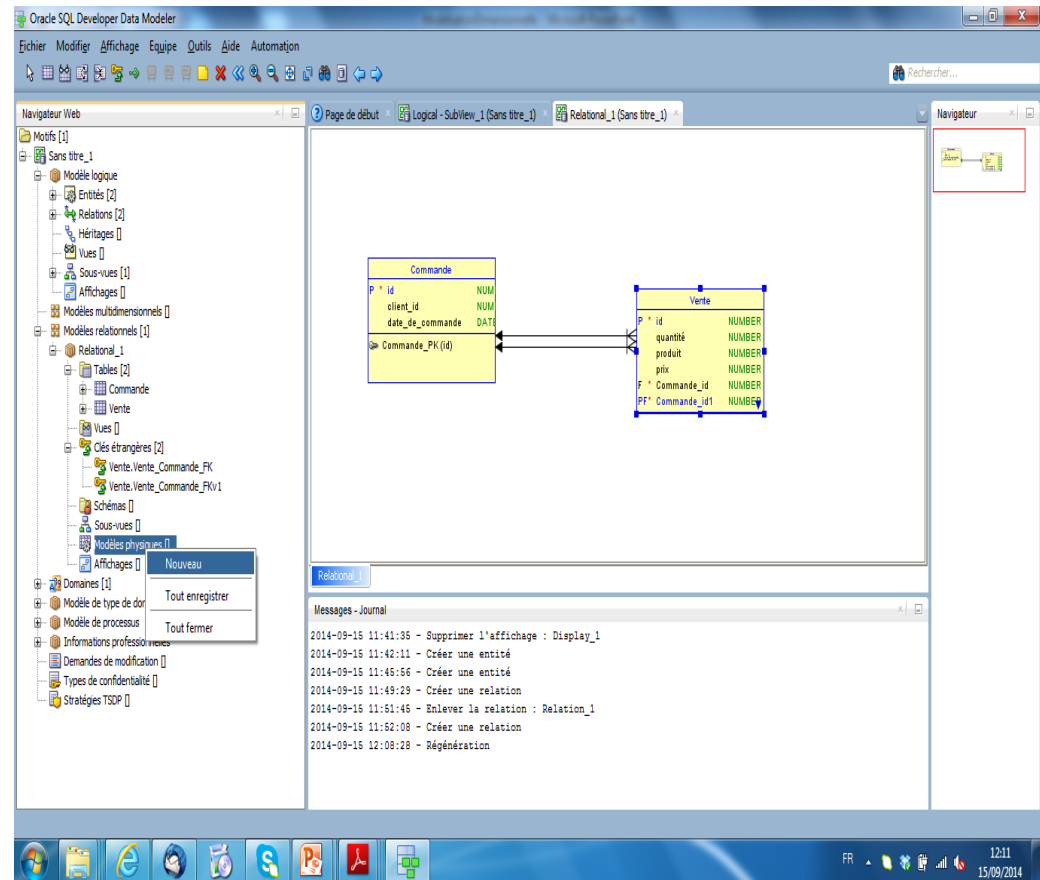
Créez quelques données dans les tables (insert into)

Exercice 4 (rajout d'un modèle physique/optimisation)

Créez un *schéma physique (vide)*

Ajoutez/repérez les informations traitées (via les tables ou directement)

Régénérez le **DDL** (si vous y avez rajouté des informations)



SQLD pour la modélisation dimensionnelle

Création d'un schéma dimensionnel conceptuel (avec une notation propre à l'outil)

Rattachement manuel d'un schéma conceptuel dimensionnel à un schéma conceptuel (fait/mesure, dimension, niveau /identifiant/informations)

- Mais aucun contrôle n'est fait sur la cohérence

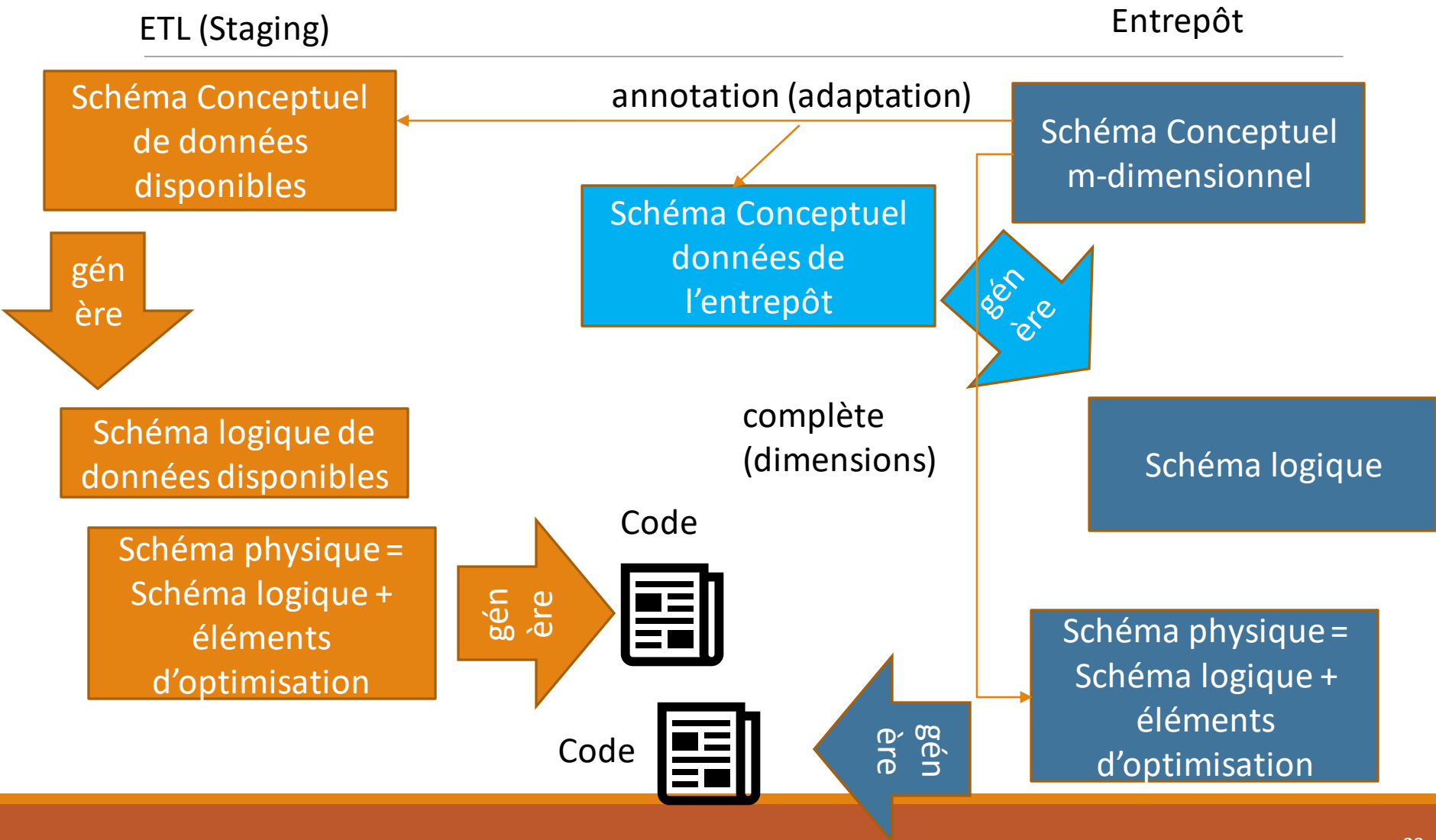
Génération/mise à jour d'un modèle physique contenant les dimensions et contrôle de cohérence avec le modèle logique

Génération automatique du DDL

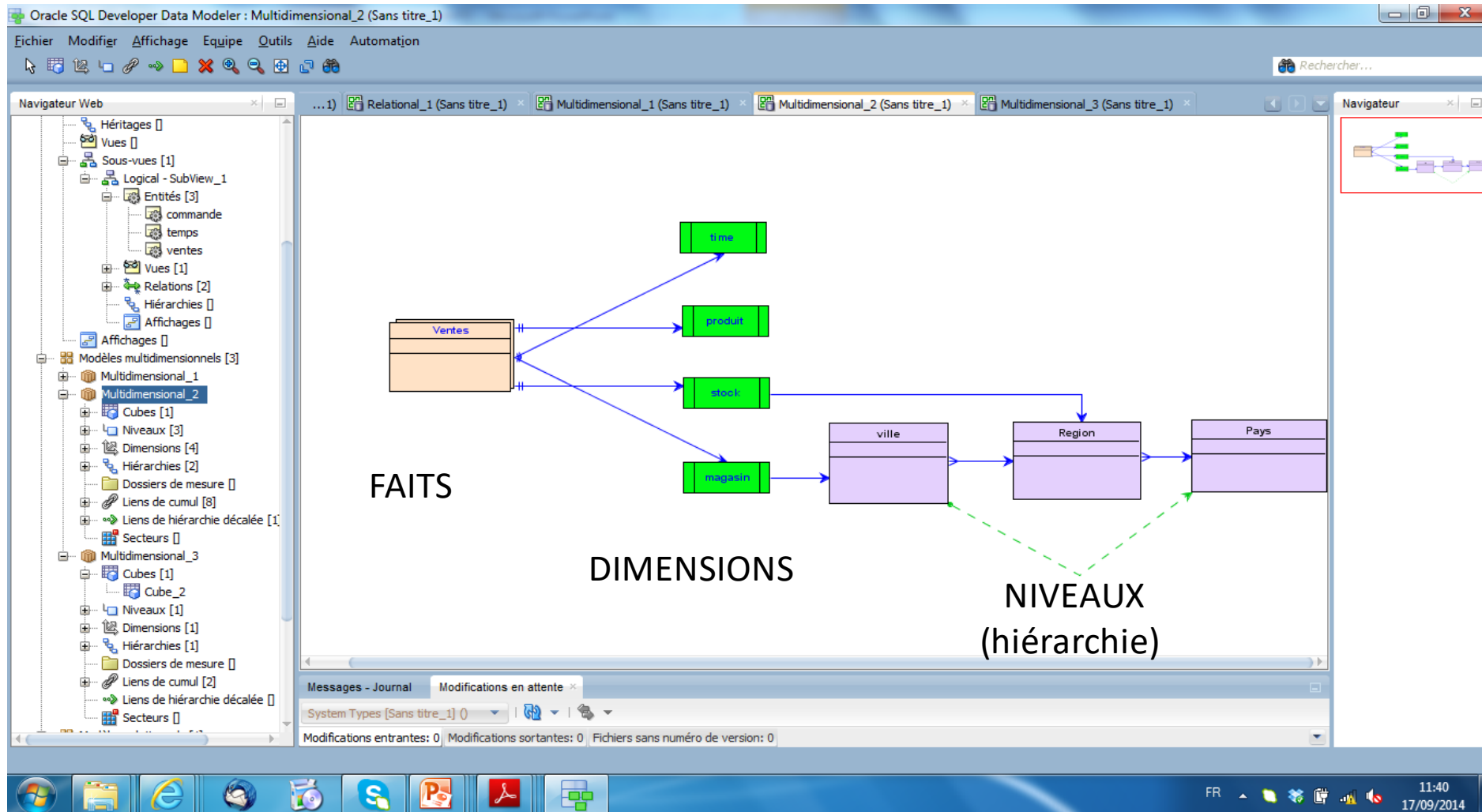
- Cela vous évite de le faire manuellement (gain de temps et limitation di risque d'erreur

Utilisation de la retro-ingénierie pour obtenir un schéma dimensionnel

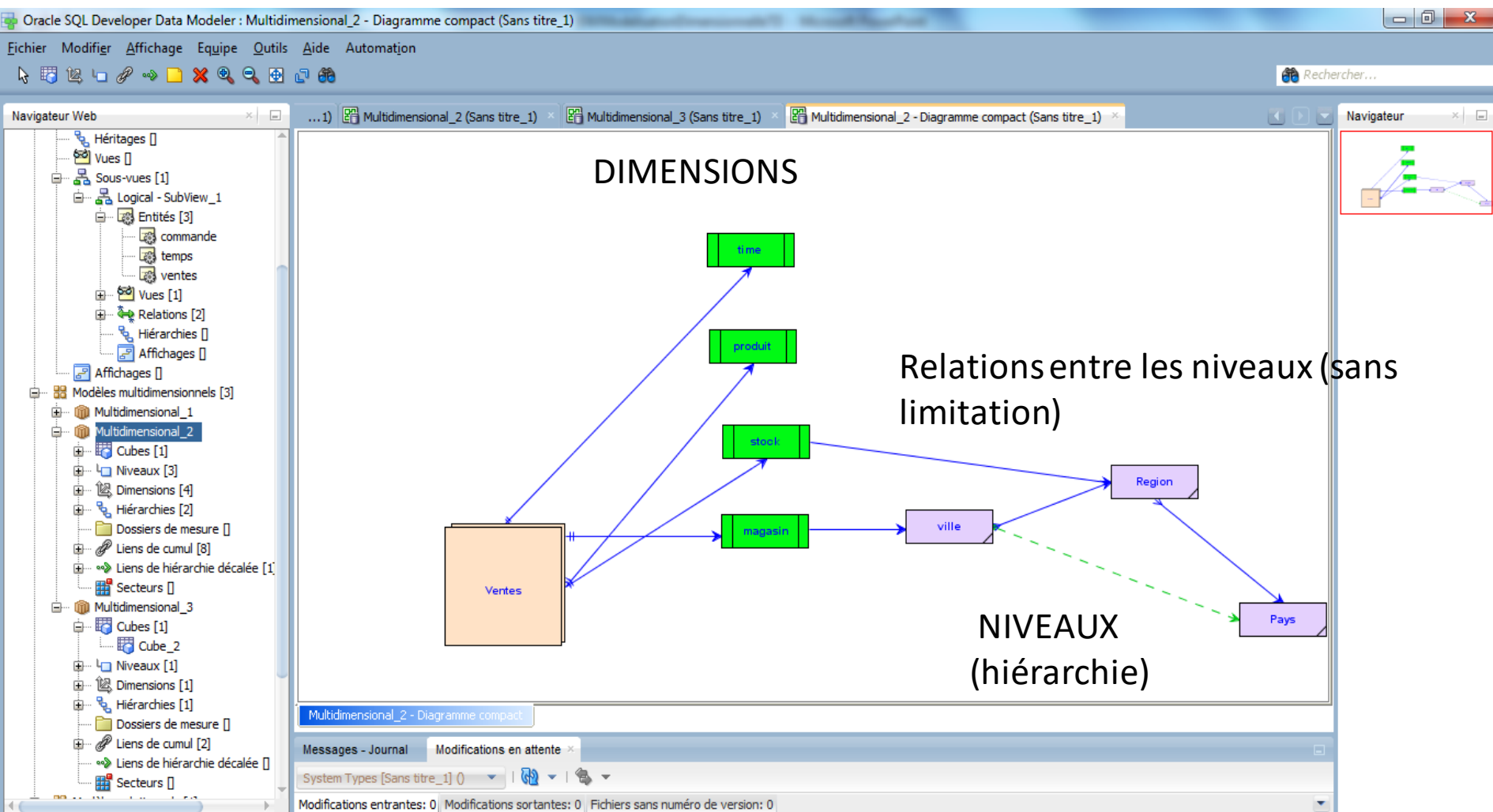
Organisation (de base)



Modèle dimensionnel (Conceptuel)



modèle compact (structure sans attributs, clés et autres informations)



Signification des relations

Entre cube et dimension

Entre niveaux

Entre dimension et niveau

- Réutilisation des niveaux à partir d'un niveau
- Nouvelle hiérarchie

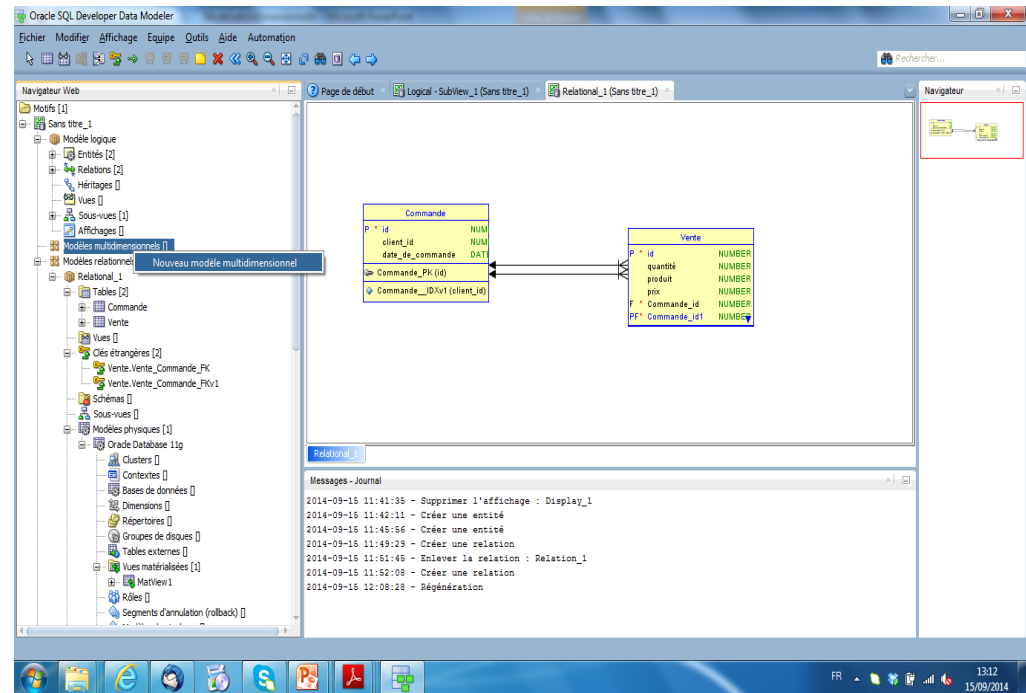
Exercice 5

Utilisez le ***schéma conceptuel*** résultat de l'exercice 1 pour définir un ***schéma dimensionnel***

Faits : ventes-quantité (l'entité Vente fournit la quantité comme donnée de base) ; granularité par produit simple

Dimensions : produit, 1 niveau

Mesure : somme (quantité)



Exercice 5 Bis

Utilisez le ***schéma conceptuel*** résultat de l'exercice 1 pour définir un ***schéma dimensionnel***

Faits : ventes-quantité

Dimensions : produit, (*client*), *temps*

Mesure : somme (quantité)

Exercice 6

Générez un ***schéma relationnel (et physique – appelé « modèle Oracle » en SQLD)*** à partir du schéma dimensionnel résultat de l'exercice 5 (ou 5 bis)



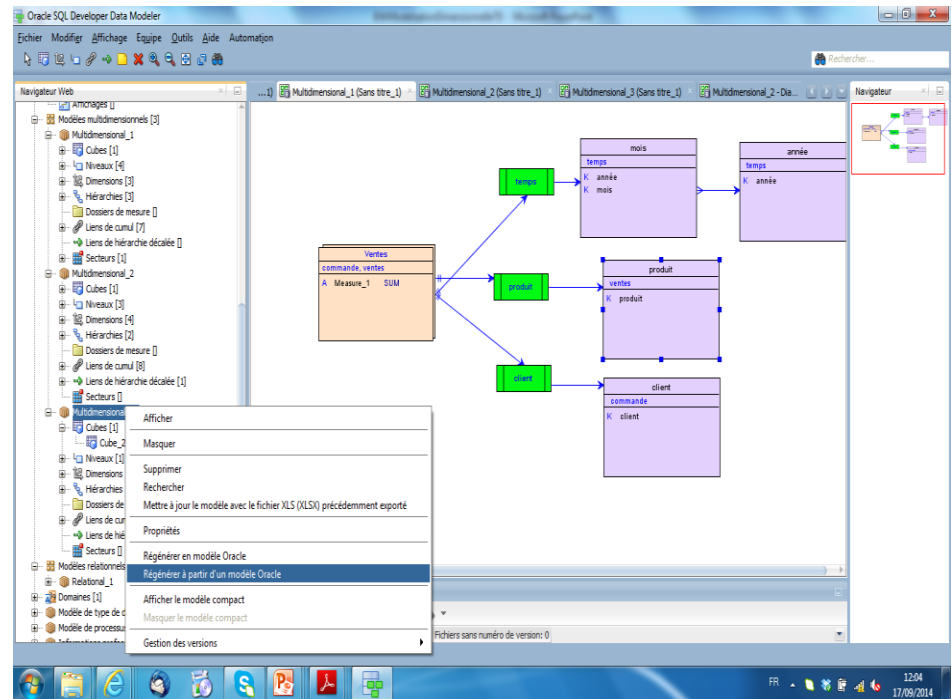
Appréciez, ***à travers les différentes couleurs***, la structure du schéma qui doit être flocon, étoile ou constellation

Éditez et analysez le ***schéma physique*** pour voir quelles informations ont y été rajoutées

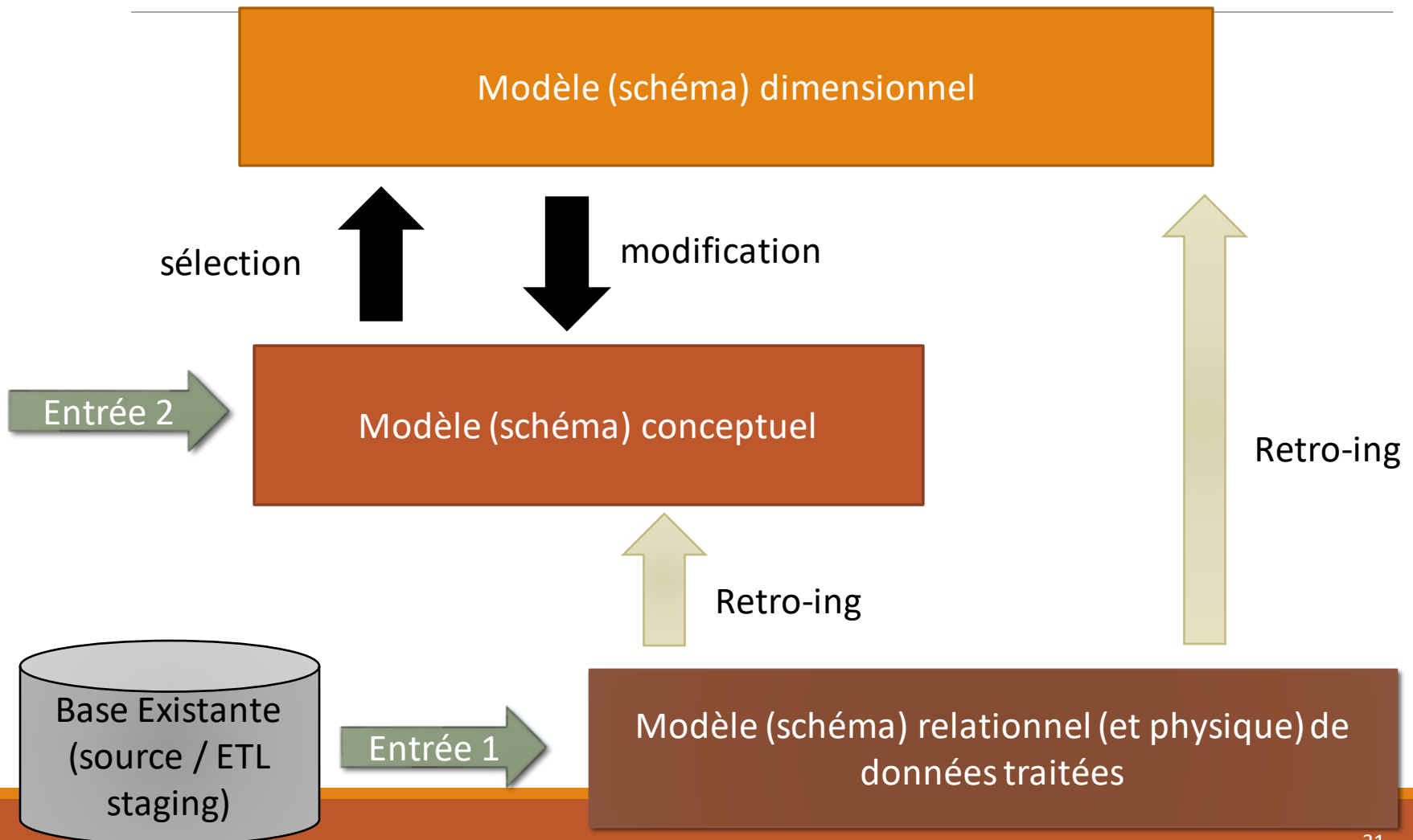
Générez le ***DDL***

Exercice 7

Utilisez le *schéma relationnel* résultat de l'exercice 2, pour définir un *schéma dimensionnel*

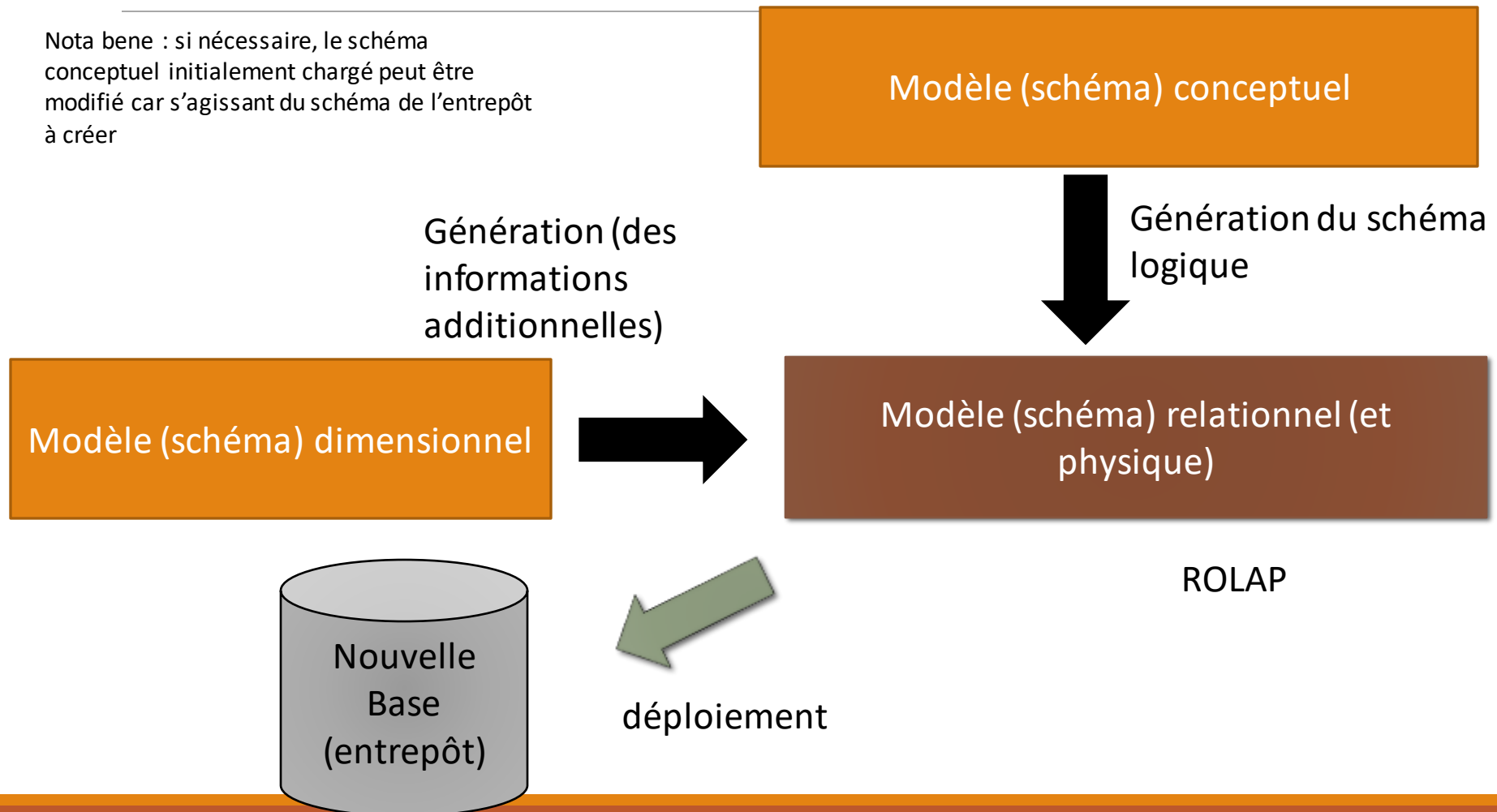


Synthèse modélisation : schéma dimensionnel

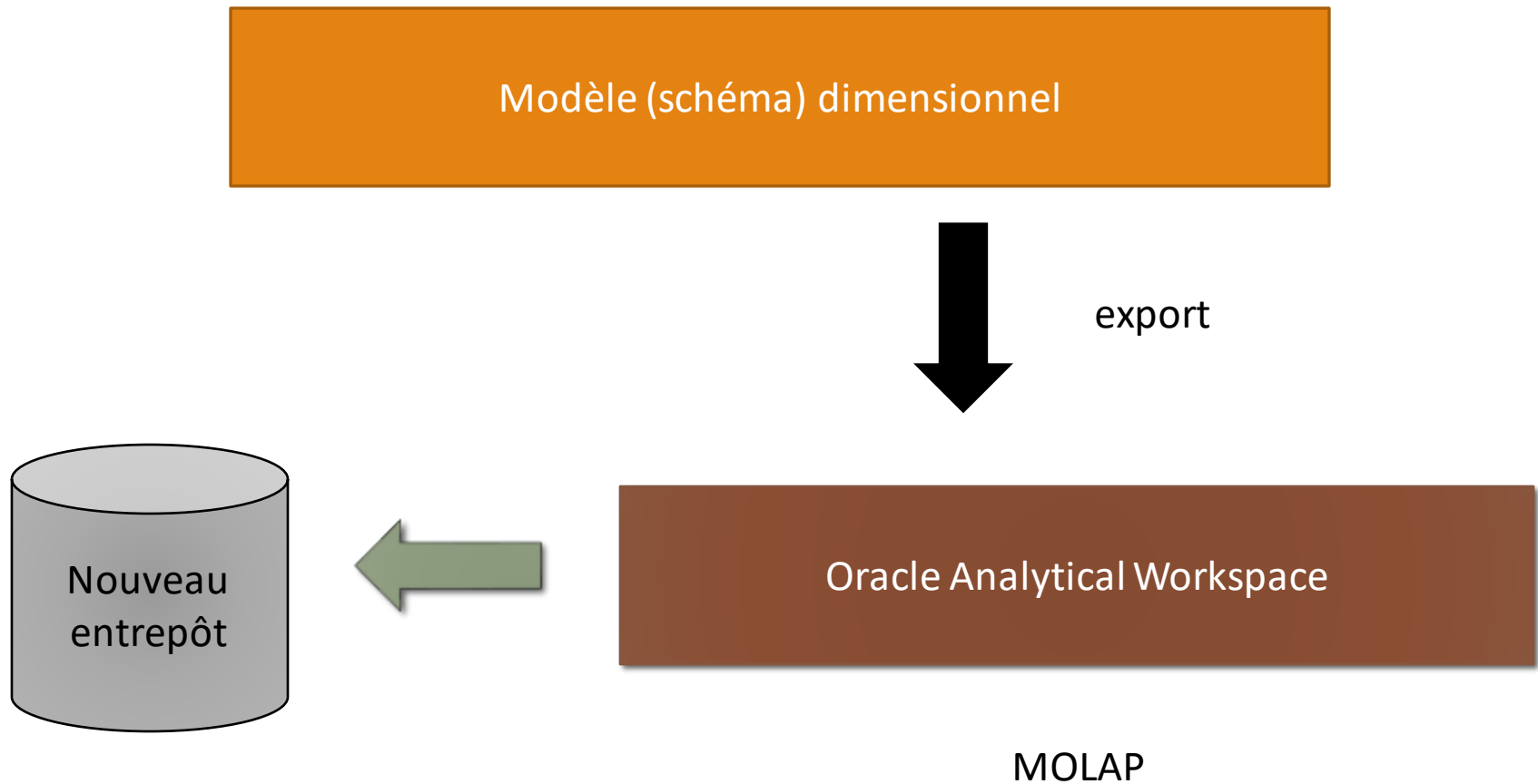


Génération/déploiement ROLAP

Nota bene : si nécessaire, le schéma conceptuel initialement chargé peut être modifié car s'agissant du schéma de l'entrepôt à créer



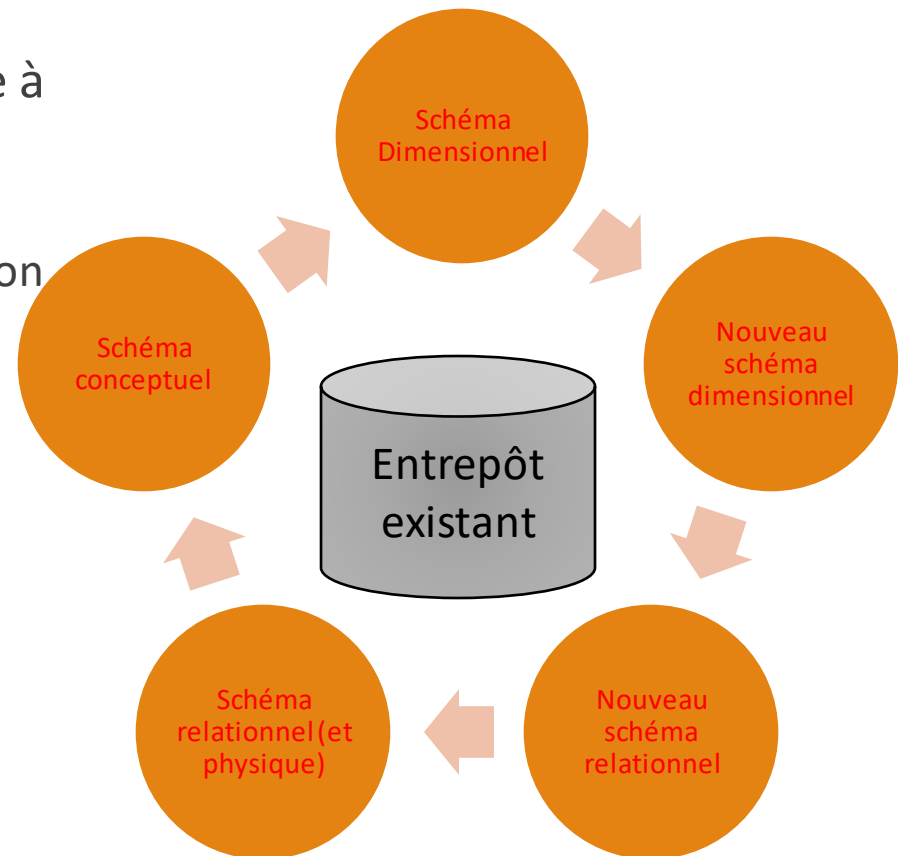
Export MOLAP



Synchronisation

SQLD(M) permet aussi de faire de la mise à jour/synchronisation entre

- Les différents niveaux de modélisation
- Un entrepôt existant et la mise à jour de son schéma suivie par un redéploiement



Principes méthodologiques et SQLD

SQLD fournit un vrai support aux méthodes basées sur des principes (méthodologiques) bien spécifiques

Plus précisément :

- SQLD nécessite d'un (ou plusieurs) schéma de sources, (typiquement 1 seul qui est le résultat de l'intégration des plusieurs schémas source se trouvant dans la zone « staging » de l'ETL), pour utiliser concrètement la modélisation dimensionnelle
- Ces schémas peuvent être **conceptuels ou relationnels** ; les schémas conceptuels peuvent avoir été obtenus par **retro-ingénierie** s'ils ne sont pas directement disponibles

On peut donc indiquer que même si potentiellement SQLD permet de développer **top-down** un schéma dimensionnel, l'outil semble conçu pour fournir un vrai support au développement **bottom-up**

Principes méthodologiques et SQLD

TOP-DOWN :

1. Développement d'un schéma dimensionnel souhaité
2. Définition et affectation à aux données disponibles (espace ETL – données sources, données intégrées)
3. Modélisation logique
4. Modélisation physique
5. Codage

BOTTOM-UP :

1. Schématisation de données disponibles (espace ETL – données sources, données intégrées)
2. Modélisation du schéma dimensionnel systématiquement annoté par la schématisation de données disponibles ; modification, si nécessaire, de la schématisation de données disponibles
3. Modélisation logique
4. Modélisation physique
5. Codage

Organisation bottom-up

ETL staging

Entrepôt

Schéma Conceptuel
de données
disponibles

gén
ère

Schéma logique de
données disponibles

Schéma physique =
Schéma logique +
éléments
d'optimisation

gén
ère

Code



Code



annotation (adaptation)

Schéma Conceptuel
données de
l'entrepôt

complète
(dimensions)

gén
ère

Schéma Conceptuel
m-dimensionnel

Schéma logique de
tables de dimension

Schéma logique TdF

Schéma physique =
Schéma logique +
éléments
d'optimisation

Exercice 1

Compréhension de la signification de dimension :

Introduire niveau semestre pour la dimension TEMPS

Comprendre qu'il est possible que les données stockées dans la table correspondante puissent « contredire » la relation 1 à n représentée par la hiérarchie dimensionnelle

→ impact, réponses erronées pour requêtes réécrites

Exercice 2

Introduction des relations entre « cube » et « dimension » (alternative à l'utilisation d'une entité partagée entre « cube » et « dimension » pour s'assurer de la « joignabilité » :

Refaire le schéma dimensionnel, spécifiant la « jointure » entre dimension « client » et « cube »

Continuer pour les autres dimensions, si nécessaire

Exercice 3

Représentation de la Table de Faits :

Rajouter quelques mesures au sein du schéma dimensionnel

Réutiliser les schémas des tables propres au « cube » pour obtenir la structure de base de la table de fait (merge)

Adapter les colonnes de la table produite par « merge » :

Mesures

Clé primaire (granularité de faits)

Exercice 4

Bases de la modélisation physique (hors dimensions) :

Choix (compromis fréquence de mise à jour/utilité (utilité : estimation taille, réécriture possible*, fréquence requêtes potentielles) et représentation des vues matérialisées

Choix et représentation de partitions de la table de faits (estimation du stockage table de faits/partitions)

Choix et représentation des index bitmap de jointure entre table de faits et dimensions (rapidité de mise à jour des vues matérialisées)

Stockage par colonnes de la tables de faits (rapidité de mise à jour des vues matérialisées vs estimation du stockage)

*à savoir la(les) mesure(s) doit(doivent) être agregable(s) ou additive(s) ou calculable(s) par mesures additives ou agregables