

R5C06 Exploitation de bases de données S5

Compte-rendu TP1 – Construction d'un entrepôt de données avec Talend

Aminata OUMOU RASSOUL NGOM

Groupe 13
S5 ~ Parcours C
BUT Informatique

Tables des matières

Introduction

1. Installation et configuration de Talend et du JDK Java 11
2. Création du projet Talend
3. Connexions aux bases sources et récupération des métadonnées
 - 3.1 Vérification des bases via SQL Developer
 - 3.2 Connexion et récupération des métadonnées dans Talend
4. Recensement des attributs des tables sources
5. Crédit des tables de l'entrepôt
6. Crédit des clés étrangères
7. Mise à jour des métadonnées pour l'entrepôt

Conclusion

Introduction

Le présent TP avait pour objectif de construire un entrepôt de données à partir de plusieurs bases de production existantes, en utilisant l'outil **Talend Open Studio for Data Integration (version 8.0.1)**. Le projet consistait à :

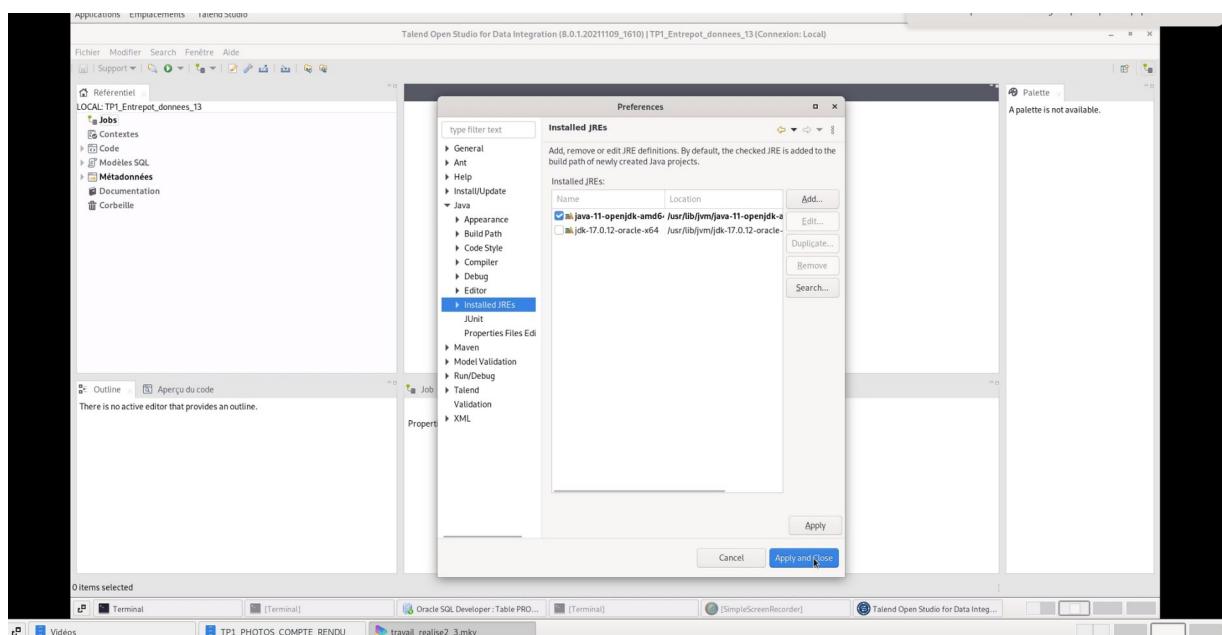
- Installer et configurer Talend et le JDK Java 11.
- Créer un projet Talend et définir les connexions aux bases sources.
- Récupérer les métadonnées des tables sources.
- Créer les tables de l'entrepôt et leurs clés étrangères.
- Mettre à jour les métadonnées dans Talend pour l'entrepôt.

L'objectif final était de préparer un entrepôt de données opérationnel et cohérent pour les étapes ultérieures d'alimentation et d'analyse.

1. Installation et configuration de Talend et du JDK Java 11

L'installation de Talend Open Studio a nécessité la mise en place préalable du **JDK Java 11**, requis pour la compatibilité avec Talend 8.0.1. Une difficulté a été la gestion de **deux versions de Java simultanément**, car SQL Developer nécessitait **Java 17** tandis que Talend fonctionnait correctement avec **Java 11**. L'environnement a été configuré de façon à utiliser la version appropriée selon le logiciel lancé.

Le lancement de Talend a été testé afin de vérifier l'absence d'erreurs, étape cruciale pour assurer le bon déroulement des tâches suivantes.



Vérification de la bonne version java dans Talend

2. Création du projet Talend

Un projet Talend a été créé pour centraliser l'ensemble des jobs liés au TP. Chaque job correspond à une tâche spécifique de création ou d'intégration de données dans l'entrepôt. La création du projet a consisté à définir le **workspace**, nommer le projet et préparer l'environnement pour accueillir les tâches futures.

3. Connexions aux bases sources et récupération des métadonnées

3.1 Vérification des bases via SQL Developer

Les informations de connexion aux bases sources ont été fournies :

- **Source 1 :**

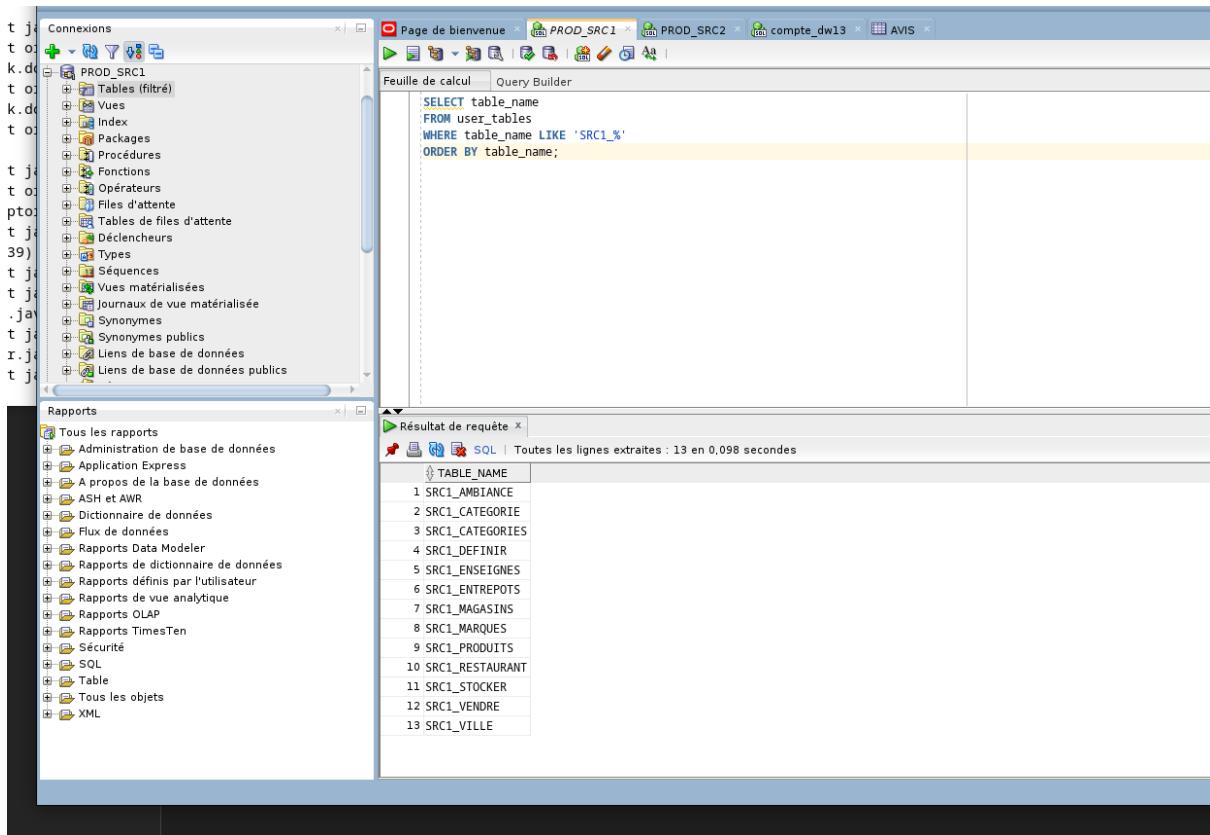
- User : PROD_SRC1
- Password : src1
- Host : telline.univ-tlse3.fr
- Port : 1521
- SID : etupre

- **Source 2 :**

- User : PROD_SRC2
- Password : src2
- Host : brehat.univ-tlse3.fr
- Port : 1521
- SID : etusec

Avant toute action dans Talend, chaque connexion a été testée et approuvée dans SQL Developer, afin de vérifier que l'accès aux bases était fonctionnel et que les tables nécessaires étaient présentes. Cette vérification a été effectuée par pur mesure de sécurité, pour éviter d'éventuelles erreurs lors de la récupération des métadonnées dans Talend.

Une fois connecté à chaque base, le contenu des tables a été examiné pour s'assurer que seules les tables utiles au projet seraient extraites et utilisées.

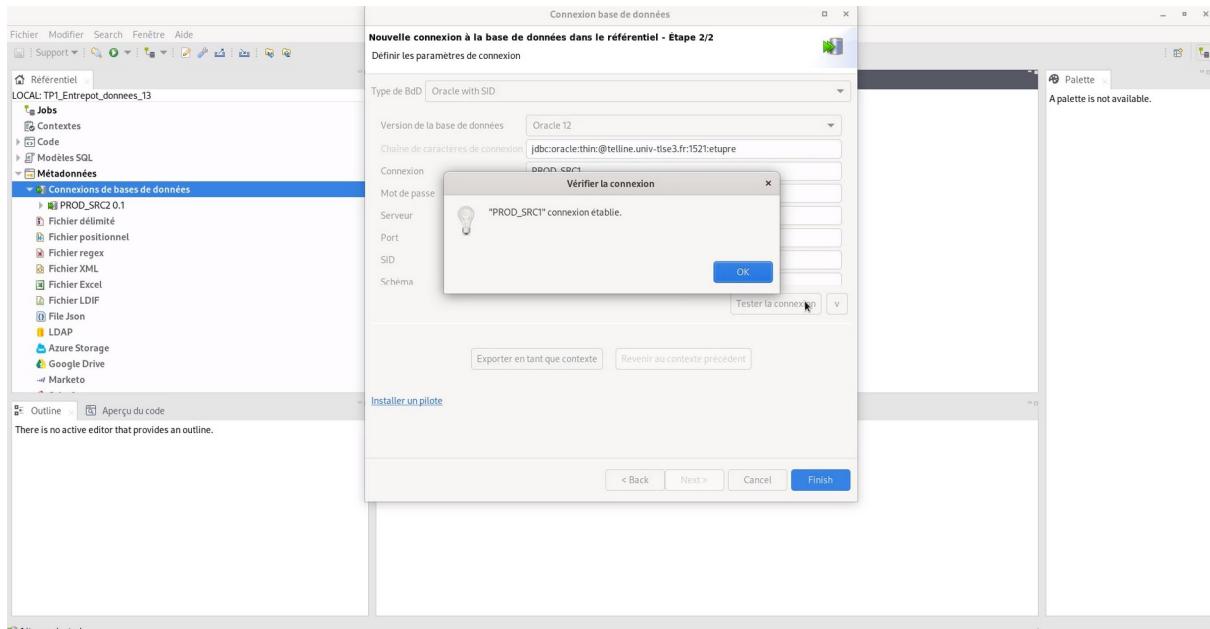


Vérification des tables pour PROD_SRC1 et PROD_SRC2 dans sqlDeveloper

3.2 Connexion et récupération des métadonnées dans Talend

Après validation des connexions dans SQL Developer, les bases ont été connectées dans Talend via le référentiel. Pour chaque base source (prod_SRC1 et prod_SRC2), les informations de connexion fournies dans le TP ont été saisies : utilisateur, mot de passe, host, port et SID. Chaque connexion a été testée pour s'assurer de la validité des informations et de l'accès correct aux bases avant de valider leur création dans Talend.

Seules les tables nécessaires au projet ont été sélectionnées pour l'extraction, conformément aux instructions du TP. À ce stade, aucune adaptation des métadonnées n'a été effectuée.



Test de la connexion pour PROD_SRC1

4. Recensement des attributs des tables sources

Avant la création des tables dans l'entrepôt, il a été nécessaire de recenser les définitions des attributs (type de données, longueur, précision, échelle, nullabilité) pour chaque table des sources PROD_SRC1 et PROD_SRC2.

Cette étape a été réalisée via SQL Developer avec une requête de type :

```
-- Exemple pour la table CATEGORIES
SELECT column_name, data_type, data_length, data_precision, data_scale, nullable
FROM all_tab_columns
WHERE owner = 'PROD_SRC1' AND table_name = 'CATEGORIES';
```

Cette requête a été répétée pour toutes les tables nécessaires, afin de créer un grand tableau de référence répertoriant toutes les colonnes et leurs caractéristiques. Ce tableau a servi de base pour définir les colonnes et leurs types dans l'entrepôt de données, garantissant la cohérence entre les sources et l'entrepôt.

POUR PROD1-REQUÊTE 1					Table Cible	Attribut Cible	Type	Longueur	Table Source	Attribut(s) Source(s)	Formule d'extraction / Remarques
CODEC	NUMBER	22	0	N							
<u>DESIGC</u>	VARCHAR2	30		Y							
<u>CODESUPC</u>	NUMBER	22	0	Y	<u>DW13_MARQUES</u>	Code_Mar	NUMBER	22	<u>SRC1_MARQUE</u>	CODEM	Même valeur que source
REQUÊTE2											
CODES	NUMBER	22	0	N							
<u>DESIGS</u>	VARCHAR2	30		Y	<u>DW13_CATEGORIES</u>	Code_Cat	NUMBER	22	<u>SRC1_CATEGORIE</u>	CODEC	Même valeur que source
<u>SIEGE</u>	VARCHAR2	30		Y		Desig_Cat	VARC	30	<u>SRC1_CATEGORIE</u>	DESIGC	Même valeur que source
CA	NUMBER	22		Y			HAR2				
WEB	VARCHAR2	30		Y							
REQUÊTE3											
<u>RAISONSOC</u>	VARCHAR2	30		N	<u>DW13_PRODUCTS</u>	Code_Prod	NUMBER	22	<u>SRC1_PRODUI</u>	CODEP	Même valeur que source
TEL	VARCHAR2	10		Y		Desig_Prod	VARC	50	<u>SRC1_PRODUI</u>	DESIGP	Même valeur que source
<u>ADRPCL</u>	NUMBER	22	5	0			HAR2				
<u>ADRVILLE</u>	VARCHAR2	30		Y		Prix_Upl	FLOAT	22	<u>SRC1_PRODUI</u>	PRIXU	Même valeur que source
CODES	NUMBER	22	38	0			TS				

Exemples d'informations obtenues des tables des sources PROD_SRC1 et PROD_SRC2

5. Création des tables de l'entrepôt

La création des tables dans Talend s'est révélée longue et minutieuse, chaque colonne devant être définie individuellement avec son type, sa longueur et ses contraintes.

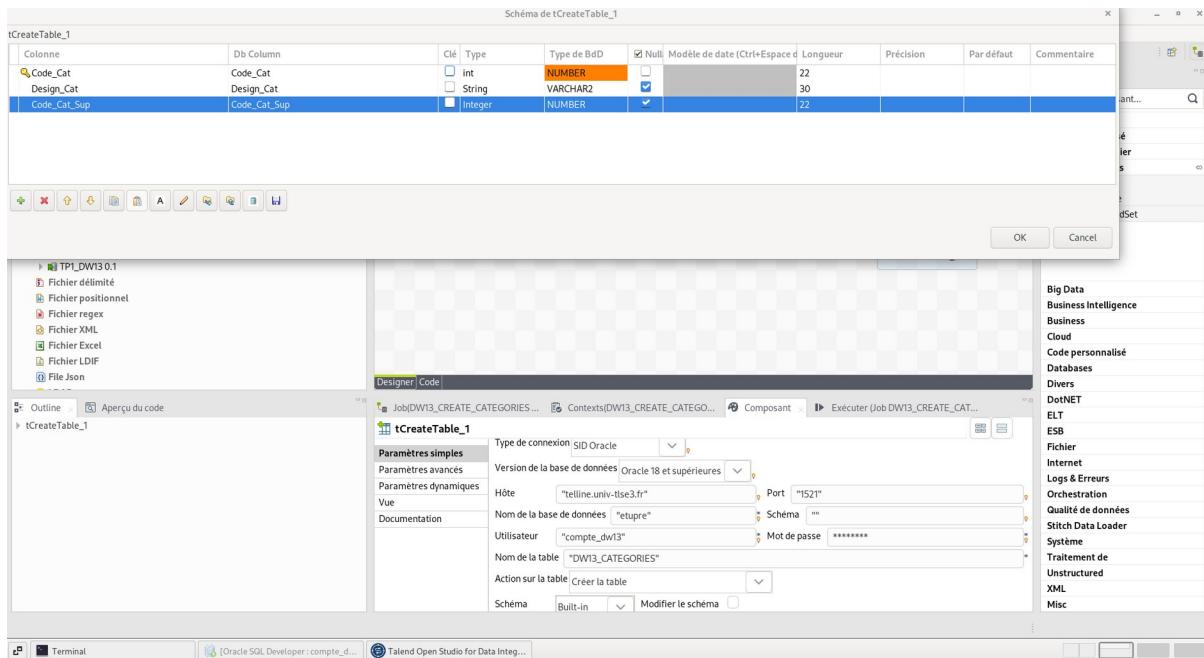
Pour cette tâche, les composants **tCreateTable** et **tDBRow** ont été utilisés :

- **tCreateTable** permettait de générer la structure des tables directement dans la base cible à partir des métadonnées définies.
- **tDBRow** a été utilisé pour exécuter des requêtes SQL supplémentaires lorsque certaines personnalisations étaient nécessaires.

Au cours du processus, plusieurs erreurs ont été rencontrées, liées notamment à des conflits de syntaxe ou des incohérences de type/longueur. La réussite finale a été validée par l'exécution complète des jobs Talend, par exemple :

Job DW13_CREATE_TABLES terminé à 15:30 17/10/2025. [Code de sortie = 0]

Une vérification parallèle dans **SQL Developer** a également été effectuée pour s'assurer que les tables existaient effectivement dans l'entrepôt.



Spécification de chaque colonne de la table

6. Création des clés étrangères

La définition des clés étrangères a constitué la partie la plus complexe de la mise en place de l'entrepôt. Plusieurs problèmes ont été rencontrés :

- Certaines colonnes des clés étrangères avaient des longueurs ou types incompatibles avec les clés primaires de référence.
- Talend ne permettait pas la création de certaines clés étrangères si les lignes correspondantes n'étaient pas correctement définies dans les tables.
- Des erreurs Java apparaissaient parfois après l'exécution des jobs, notamment concernant les guillemets dans les requêtes SQL.

Pour résoudre ces problèmes, les actions suivantes ont été nécessaires :

- Ajuster les longueurs et types des colonnes des clés étrangères pour correspondre exactement aux clés primaires.
- Exécuter des commandes ALTER TABLE directement via SQL Developer pour certaines tables avant que Talend n'accepte la création des clés étrangères.
- Utiliser les composants tAlterTable et tDBRow dans Talend pour appliquer les modifications et créer les clés étrangères.

Après avoir résolu toutes les erreurs et incohérences, la réussite finale a été validée par l'exécution complète des jobs Talend, par exemple :

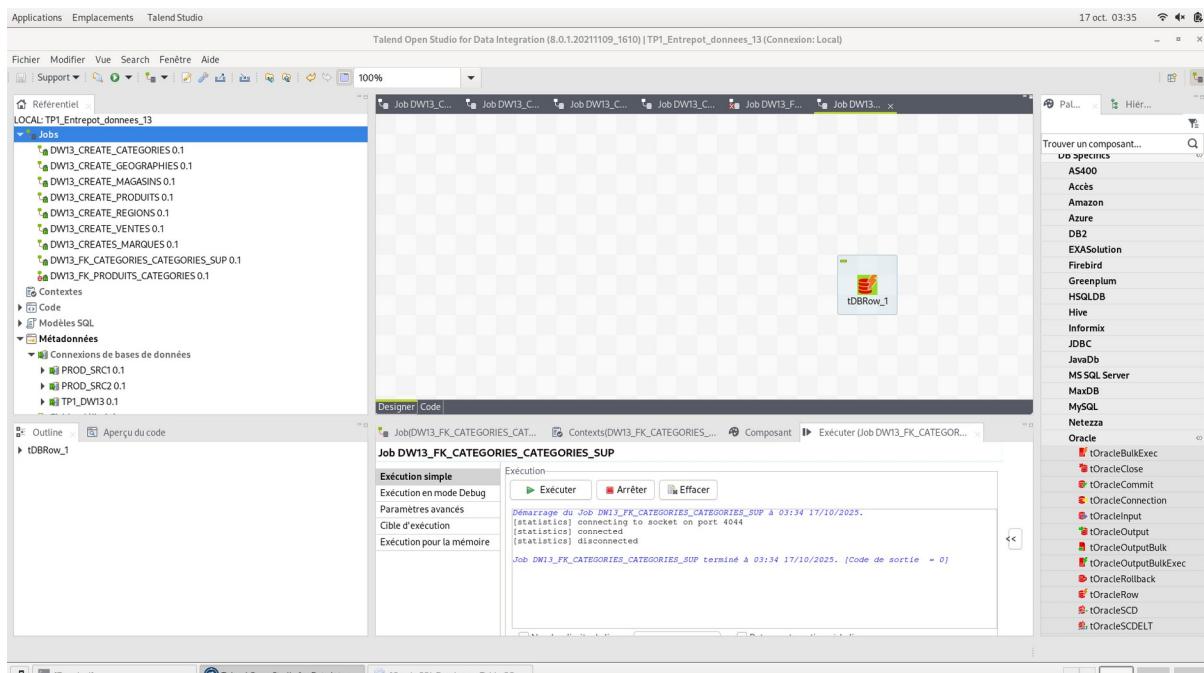
Job DW13_FK_VENTES_PRODUITS terminé à 15:35 17/10/2025. [Code de sortie = 0]

Toutes les clés étrangères ont été ainsi correctement créées et vérifiées, assurant la cohérence des relations entre les tables de l'entrepôt.

The screenshot shows the Talend Designer interface with the following details:

- Left Panel (Toolbox):** A tree view under the heading "Oracle" containing various components: tOracleBulkExec, tOracleClose, tOracleCommit, tOracleConnection, tOracleInput, tOracleOutput, tOracleOutputBulk, tOracleOutputBulkExec, tOracleRollback, tOracleRow, tOracleSCD, tOracleSCDELT, tOracleSP, and tOracleTableList.
- Central Area:**
 - Job Header:** Job(DW13_FK_PRODUITS_CATEGORIE...)
 - Contexts:** Contexts(DW13_FK_PRODUITS_CATEGORIE...)
 - Composant:** tDBRow_1(Oracle)
 - Description:** <-- Utilisation de tOracleRow (tDBRow)
- Component Configuration (tDBRow_1(Oracle)):**
 - Paramètres simples:**
 - Base de données: "etupre"
 - Utilisateur: "compte_dw13"
 - Schéma: Built-in
 - Mot de passe: *****
 - Nom de la table: "DW13_CATEGORIES"
 - Type de requête: Built-in (Guess Query)
 - Requête: ADD CONSTRAINT FK_PRODUITS_CATEGORIES
FOREIGN KEY (Code_Cat)
REFERENCES DW13_CATEGORIES(Code_Cat);
 - Nombre de lignes: AUCUNE
 - Source de données: (empty)
 - Paramètres avancés:** (not visible in the screenshot)
 - Paramètres dynamiques:** (not visible in the screenshot)
 - Vue:** (not visible in the screenshot)
 - Documentation:** (not visible in the screenshot)

Caption: Paramétrage du composant tDBRow



Réussite de la création de clé étrangère

7. Mise à jour des métadonnées pour l'entrepôt

Une fois les tables et les clés étrangères créées, les métadonnées de l'entrepôt ont été mises à jour dans Talend. Cette étape a consisté principalement à adapter les longueurs des clés étrangères pour correspondre aux clés primaires de référence et assurer la cohérence globale des tables.

Conclusion

Ce TP a permis de :

- Découvrir l'installation et la configuration de Talend Open Studio avec la gestion de différentes versions de Java.
- Créer un projet Talend et gérer les connexions aux bases de données sources.
- Vérifier les tables sources dans SQL Developer et recenser tous les attributs nécessaires pour créer un entrepôt cohérent.
- Créer minutieusement les tables et définir les clés étrangères tout en gérant les erreurs rencontrées.
- Vérifier l'intégrité des tables et des relations via SQL Developer.
- Se familiariser avec des outils comme Talend et SQL Developer qui, à première vue, ne sont pas faciles à utiliser. Les tâches répétitives, telles que la création de tables ou de clés étrangères nécessitant de répéter plusieurs fois les mêmes actions ou de faire face aux mêmes erreurs, ont permis de consolider les compétences sur ces

logiciels.

L'expérience acquise met en évidence l'importance de la cohérence des métadonnées et de la rigueur dans la gestion des relations pour la mise en place d'un entrepôt de données opérationnel.