TP Business Intelligence
M. Agier

Présentation de l'étude de cas

La société Orion

Cette société fictive, présente au niveau mondial, est spécialisée dans la commercialisation d'articles de sport et d'extérieur. Les données disponibles regroupent des informations sur :

- les employés
- les produits
- les clients
- les commandes
- les fournisseurs

Le siège social aux États-Unis, gère des filiales en Belgique (depuis 1999), Pays Bas, Allemagne, Royaume-Uni, Danemark, France, Italie, Espagne et Australie. Les produits sont vendus en magasin, par catalogue et par internet. Une carte de fidélité : 'Orion Star Club', propose beaucoup d'avantages. L'historique d'information va du 1er janvier 1998 au 31 décembre 2002.

Structure de l'organisation

Le siège social héberge la majeure partie des fonctions administratives, soit un nombre important d'employés, entre 600 et 800. Le siège social centralise aussi la gestion des stocks, la vente par catalogue, la vente par internet et l'import - export. Néanmoins, certains employés gèrent aussi ces fonctions depuis les différentes filiales.

Les employés sont enregistrés dans la base de données selon cinq niveaux :

- Pays
- Compagnie
- Département
- Section
- Groupe

Les informations complémentaires sur les employés sont notamment :

- Date d'entrée et de départ de l'employé
- Date de début et de fin de contrat (pour certain contrat)
- Adresse
- Sexe
- Salaire
- Responsable hiérarchique

L’offre

La société propose environ 5500 références. Certaines ne sont pas vendues dans tous les pays, d’autres, de part les volumes commercialisés, reflètent certaines particularités régionales, certains sports nationaux. Tous les noms sont fictifs.
Les produits sont organisés selon 4 niveaux :

- Ligne de produit
- Catégorie de produit
- Groupe de produit
- Produit

Chaque produit a un coût et un prix de vente. Le système informatique gère tous les prix en dollars. En utilisant les dates de début et de fin, ces prix varient en fonction du temps. Cet historique est sauvegardé. Le système gère aussi les remises pour certains produits, à certaines périodes. Les prix sont généralement uniques de part le monde.

Les clients

Les clients sont repartis à travers le monde, notamment dans les pays où se trouvent des filiales, mais pas uniquement. Les noms et adresses sont fictifs, même si les villes, régions/comtés et pays, sont réels. La base de données enregistre environ 90 000 clients, pas tous actifs.

L’adresse des clients comprend tout ou partie des informations suivantes :

- Rue
- Code postal
- Ville
- Région / département / conté
- Etat
- Pays
- Continent

Les clients sont classés dans des groupes en fonction de leur activité d’achat.

Les commandes

Chaque commande pointe vers le commercial qui a enregistré la vente. Environ 980 000 commandes sont enregistrées, commandes qui reflètent notamment les saisonnalités. Chaque commande comprend une ou plusieurs lignes, une ligne par produit.

Les fournisseurs

Chaque produit provient d'un fournisseur qui est basé dans un pays, mais toutes les commandes sont passées par le siège social. Il y a 64 fournisseurs, mais un seul fournisseur par produit.

Mise en place d'un système décisionnel

La société Orion souhaite améliorer sa performance à l'aide d'un système décisionnel.

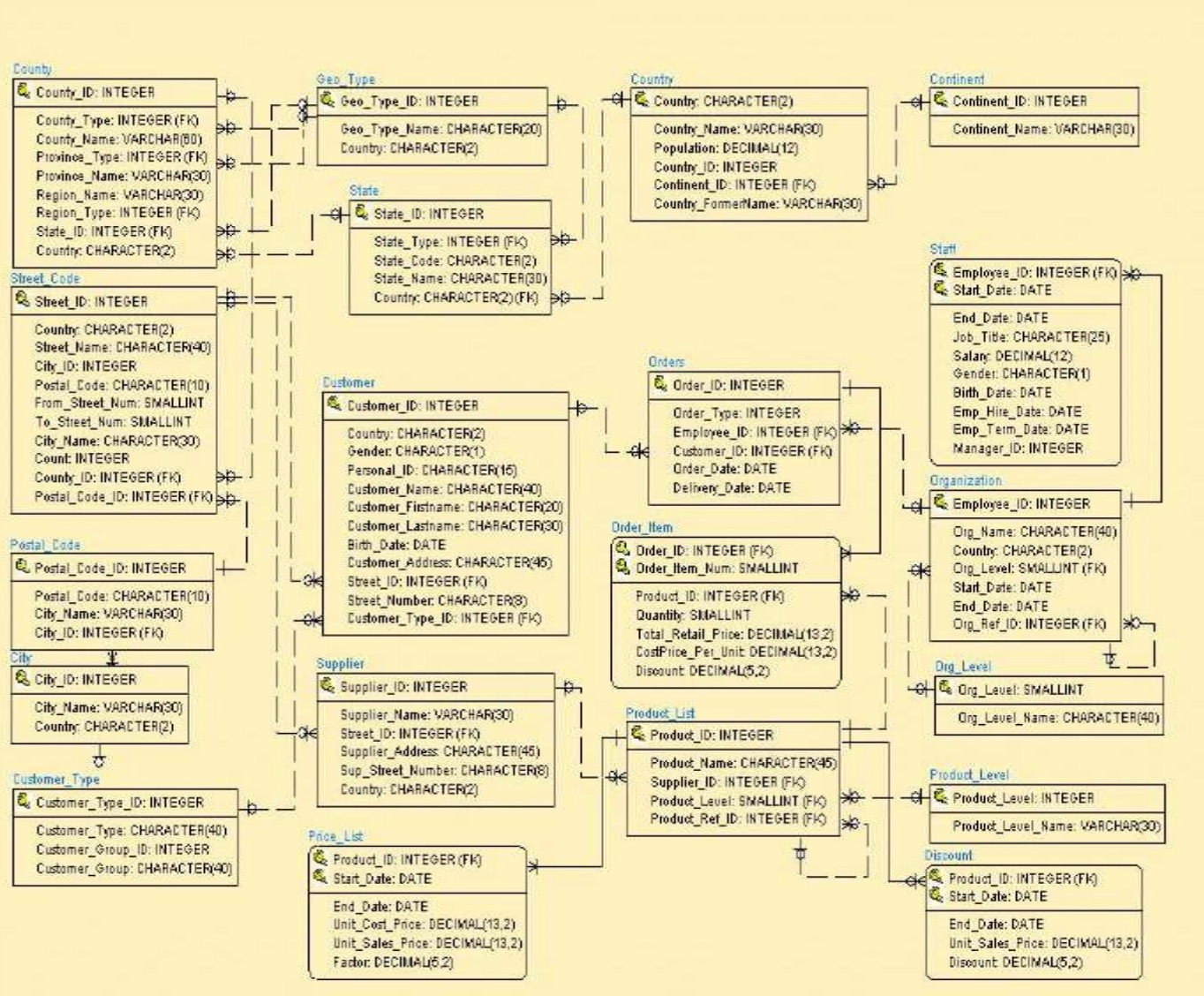
Voici quelques questions qui ont été recensées et auxquelles devrait répondre le système mis en place :

- Quels sont les produits qui se vendent le mieux ?
- Quels sont les produits en perte de vitesse ?
- Quels sont les produits qui contribuent très peu au chiffre d'affaire pour un pays et une année donnés ? Est-ce que ces produits peuvent être remisés ?
- Quelle est la marge générée par ce groupe de produit ?
- Est-ce que la marge dépend de la quantité vendue ?
- Est-ce que les remises font augmenter les ventes ?
- Est-ce que les remises font augmenter la marge ?
- Quels sont les commerciaux qui font le plus de ventes ?
- Quels sont les commerciaux qui performent le mieux par pays, sexe, âge, salaire ?
- Quels groupes de clients sont identifiés ?
- Quels sont les clients les plus rentables ?
- Quels fournisseurs proposent des produits rentables?
- Quelle est la moyenne et l'écart-type du chiffre d'affaire ?
- Quelles sont les variables qui expliquent le mieux l'importance du chiffre d'affaire ?
- Y-a-t'il une différence significative entre la moyenne de la somme du chiffre d'affaire géré par les commerciaux de sexe féminin et celle des commerciaux de sexe masculin ?

Il faut donc construire un entrepôt de données capable de répondre aux besoins de requête, de reporting, et d'analyses avancées.

Les données sources

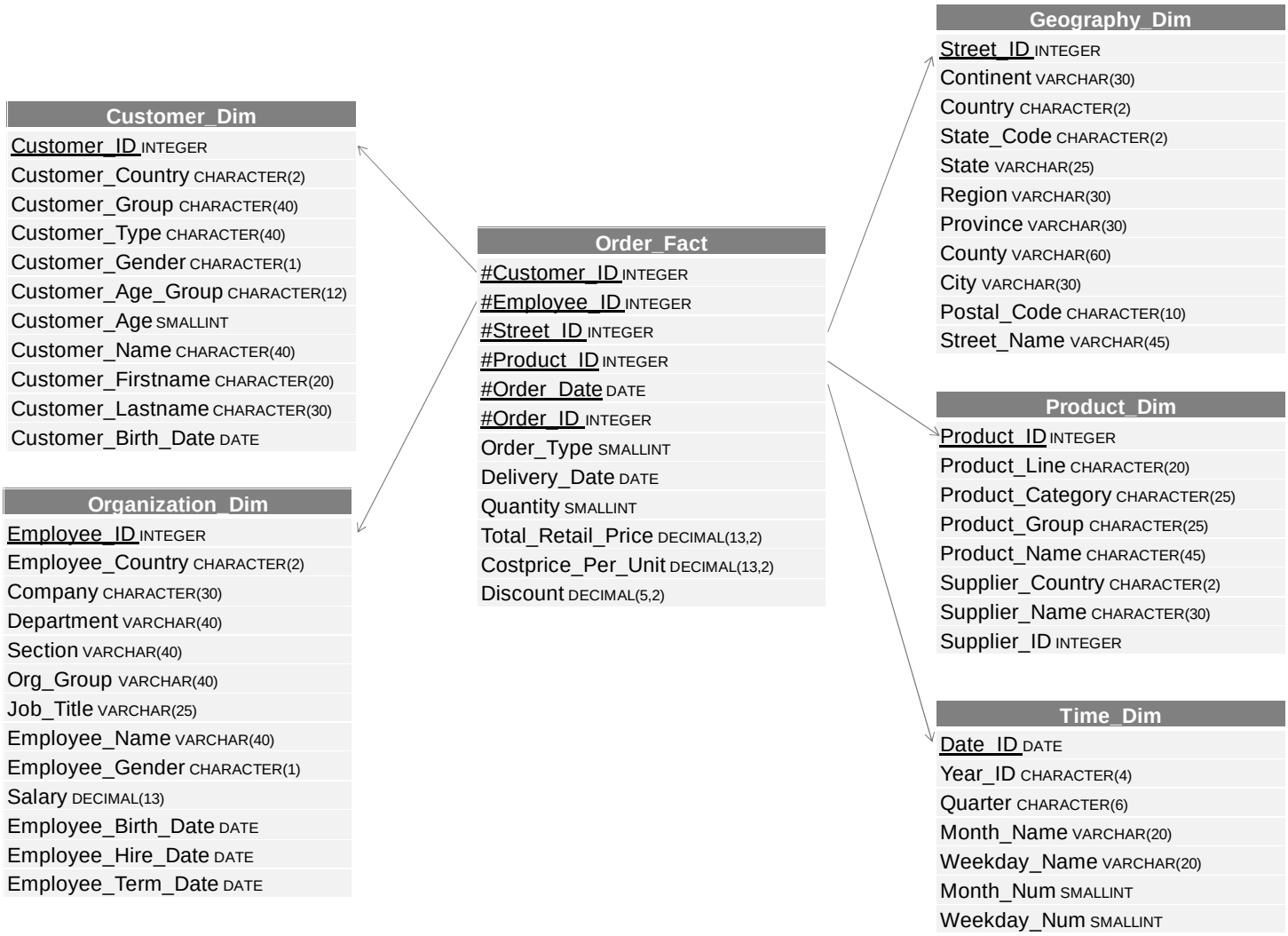
Voici le schéma relationnel de la base de données opérationnelle de l'entreprise d'où proviendront les données de l'entrepôt :



Ces tables sont stockées dans la base de données Microsoft Access nommée orion.mdb, hormis la table Staff stockée dans le fichier Microsoft Excel nommé staff.xls.

Schéma de l'entrepôt

Voici le schéma en étoile de l'entrepôt de données :



Création des tables de l'entrepôt

Une fois le schéma en étoile validé, il faut créer l'entrepôt sous Oracle.

1) Travail à réaliser :
   - Créer un utilisateur nommé orion_DW_user qui sera le propriétaire de l'entrepôt :
     - CREATE USER orion_DW_user IDENTIFIED BY orion_DW_user;
     - GRANT ALL PRIVILEGES TO orion_DW_user;
   - Implémenter la création des tables de l'entrepôt sous Oracle en spécifiant bien les clés primaires et les clés étrangères.

Maintenant que les tables de l'entrepôt sont créées, il faut réaliser les processus qui vont remplir ces tables à partir des données sources.

Découverte de Talend Open Studio

2) Travail à réaliser :

- Ouvrir Talend Open Studio.
- Créer un nouveau projet nommé orion_project avec l'option Java.
- Dans la fenêtre « Generation Engine Initialization in progress », cocher la case Always run in background puis cliquer sur Run in background.

Avant de commencer à travailler sous Talend, toujours attendre que la « Generation Engine Initialization in progress » (en bas à droite) soit terminée.

La fenêtre de Talend Open Studio est composée des vues suivantes :

- Barres d'outils et menus (en haut)
- Repository (en haut à gauche) : Ce référentiel contient tous les éléments techniques du projet
- Design Workspace (au centre) : Cet espace de modélisation permet de concevoir graphiquement les business model et les jobs.
- Palette (en haut à droite) : Cette palette graphique permet d'accéder aux différents composants
- Différentes vues (en bas au centre) :
  - Job : infos sur le job sélectionné
  - Component : configuration du composant sélectionné
  - Run job : exécution des jobs
  - Problems : erreurs
- Outline et Code Viewer (en bas à gauche) : Ces fenêtres fournissent un aperçu du code et du schéma du job ou du business model.

Business Models

Un business model permet de modéliser avec des composants graphiques, le processus à mettre en place.

Pour ce système décisionnel, voici le processus à mettre en place :

Des données sources, fichier Excel + base de données Access (composants Input et Database), vont être traitées par différents jobs ETL (composant Gear) pour remplir l'entrepôt (composant Database). Un datamart (composant Database) sera créé ensuite à partir de l'entrepôt. Les utilisateurs (composant Actor) accèderont à l'entrepôt ou au datamart par le biais de leur PC grâce aux outils de restitution (composant Terminal).

3) Travail à réaliser :

- Créer un nouveau Business model nommé orion_model.
- Créer le modèle en choisissant les différents composants graphiques situés dans la palette.

Spécification des données sources

4) Travail à réaliser :

- Placer les différents fichiers Excel et Access dans un répertoire C:/orion.
- Etablir une connexion orion_BD à la base access orion.mdb :
  - Dans le Repository, clic droit sur Metadata / Db Connections / Create connection
- Récupérer les schémas des tables
  - Clic droit sur la connexion orion_BD : Retrieve Schema
  - Cliquer sur Next.
  - Sélectionner les tables nécessaires au projet.
  - Cliquer sur Next.
  - Pour chaque table, il y a le type de chaque colonne dans la base de données sources (DB Type) et sa traduction dans Talend (Type). Pour simplifier, seulement 3 types seront utilisés ici : Double, String et Date. Modifier alors pour chaque table, la traduction des DATETIME en Date (au lieu de String).
  - Cliquer sur Finish.
- Récupérer le schéma du fichier staff.xls
  - Dans le Repository, clic droit sur Metadata / File Excel / Create file Excel
  - Name : staff
  - Cliquer sur Next
  - File : C:/orion/staff.xls
  - Sélectionner la feuille _t1 du fichier staff.xls (Sheet)
  - Cliquer sur Next
  - Cocher l’option Set heading row as column names puis Refresh Preview
  - Cliquer sur Next
  - Name : staff
  - Modifier les types des colonnes de façon à n’avoir que des Double, des String ou des Date.
  - Cliquer sur Finish

Spécification des données cibles

5) Travail à réaliser :
   - Etablir une connexion orion_DW à l'entrepôt de données :
   - Récupérer les schémas des tables

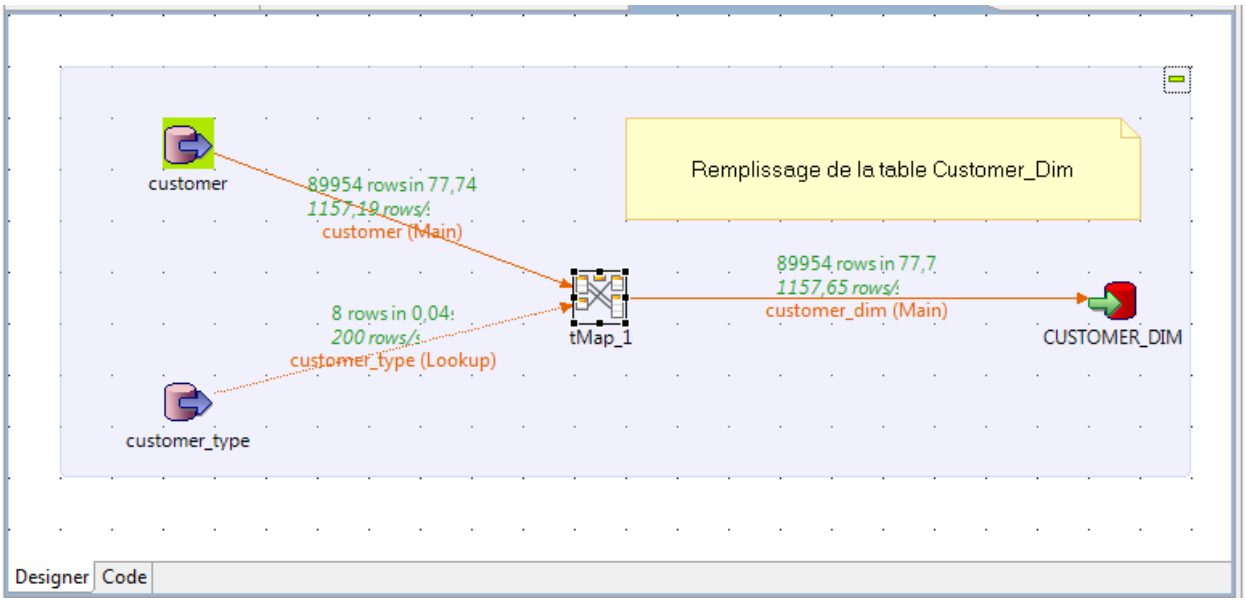
Les données sources et cibles sont maintenant disponibles dans le Repository. Il faut alors construire les différents jobs pour remplir les tables de l'entrepôt.

Remplissage de la table Customer_Dim

6) Travail à réaliser :
   - Pour chaque colonne de la table Customer_Dim, spécifier de quelle(s) donnée(s) source elle dépend.

Table cible	Colonne cible	Table(s) source	Colonne(s) source	Remarques

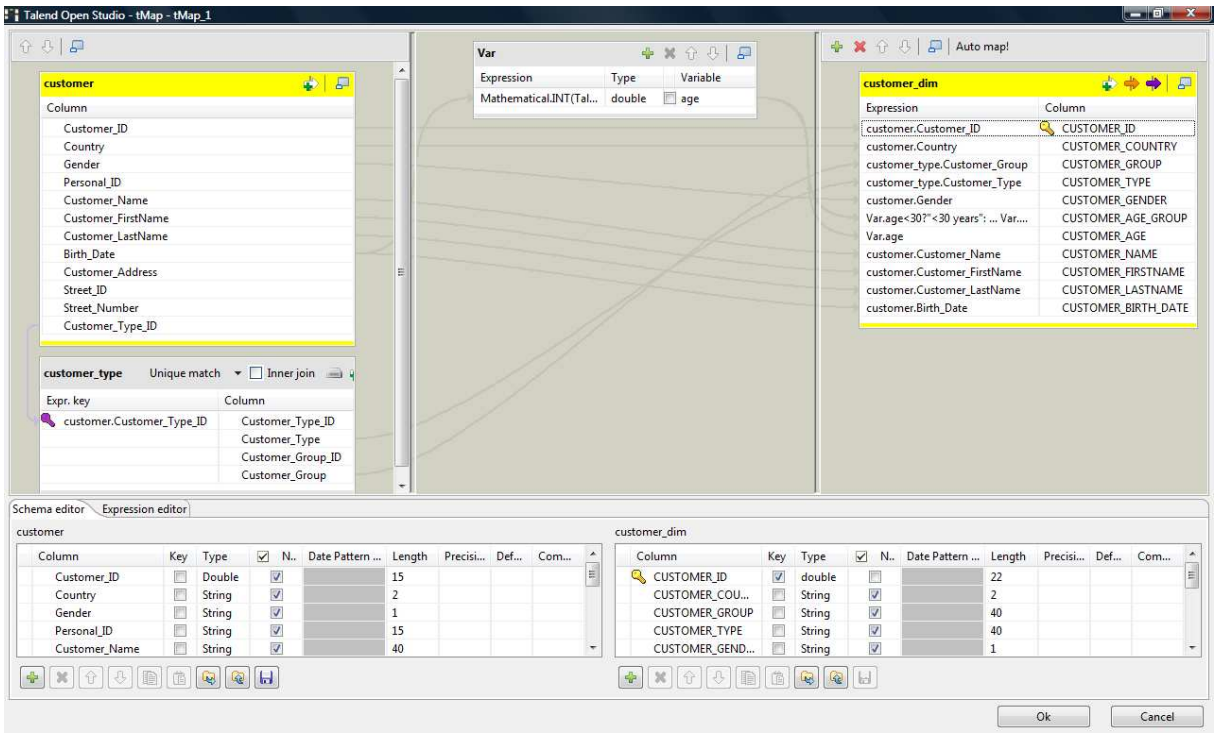
7) Travail à réaliser :
   - Créer un job nommé Job01_Customer_Dim.



   - Choisir les tables sources (Customer puis Customer_Type) et les importer dans le Design Workspace avec l'option tAccessInput.
   - Choisir la table cible (Customer_Dim) et l'importer dans le Design Workspace avec l'option tOracleOutput.

- Ajouter ensuite, le composant Processing / tMap, pour faire le lien entre les données sources et les données cibles.
- Ajouter les liens entre les différents composants :
    - A partir des composants tAccessInput, clic droit, Row, Main
    - Renommer les liens avec customer et customer_type.
    - A partir du composant tMap, clic droit, Row, *New output* (Main)
    - Donner un nom au lien : customer_dim.
    - Une fenêtre s'ouvre répondre yes, pour prendre en compte le schéma de la table cible.
- Dans le Design workspace, vous afficherez un petit commentaire pour décrire le job à l'aide du composant Misc / Note (à faire pour tous les jobs).

- Dans le composant tOracleOutput :
    - Dans la vue Component, spécifier dans Action on table : Clear table, cela supprime les données de la table avant d'en insérer de nouvelles (à faire pour tous les jobs).
- Dans le composant tMap :



    - Double-clic sur le composant tMap.
    - Faire une jointure entre les deux tables sources.
    - Relier les colonnes sources aux colonnes cibles.
    - Créer une nouvelle variable avec l'âge des clients :
        - Expression :
          Mathematical.INT(TalendDate.formatDate("yyyy",TalendDate.
          getCurrentDate()))-

Mathematical.INT(TalendDate.formatDate("yyyy",customer.Birth_Date))

- Type : double
- Variable : age

- La colonne cible CUSTOMER_AGE est égale à cette variable age.
- La colonne cible CUSTOMER_AGE_GROUP est définie de la façon suivante :
  - Var.age<30?"<30 years":
    Var.age<46?"30-45 years":
    Var.age<61?"46-60 years":
    Var.age<76?"61-75 years":
    ">75 years"
- Cliquer sur OK

- Dans la fenêtre Run, cocher la case Statistics, puis lancer le job en cliquant sur Run (ou F6).
- Sous Oracle, vérifier le résultat du job en lançant les requêtes suivantes :

```
SELECT COUNT(*)
FROM Customer_Dim;

SELECT *
FROM Customer_Dim
WHERE ROWNUM<10;
```

Remplissage de la table Product_Dim

8) Travail à réaliser :

- Pour chaque colonne de la table Product_Dim, spécifier de quelle(s) donnée(s) source elle dépend.
- Créer un job nommé Job02_Product_Dim.
- Choisir les tables sources et la table cible.
- Ajouter le composant Processing / tMap puis ajouter les liens entre les différents composants (renommer les liens avec des noms pertinents).
- Dans le premier composant tAccessInput (product_list par exemple) :
  - Modifier la requête pour ne prendre en compte que les produits : Dans la vue Component, dans Query, modifier la requête.
  - Procéder de la même façon pour les autres composants.
- Dans le composant tMap :
  - Faire les jointures entre les différentes tables sources.
  - Relier les colonnes sources aux colonnes cibles.
- Lancer le job avec les statistiques.
- Vérifier le résultat du job sous Oracle.

Remplissage de la table Organization_Dim

9) Travail à réaliser :
   - Pour chaque colonne de la table Organization_Dim, spécifier de quelle(s) donnée(s) source elle dépend.
   - Créer le job Job03_Organization_Dim.
   - Lancer le job et vérifier le résultat sous Oracle.

Remplissage de la table Time_Dim

Dans cette table, il faut rentrer toutes les dates du 01/01/1998 au 31/12/2002.

10) Travail à réaliser :
   - En vous aidant de l'exemple ci-dessous, créer un programme en PL/SQL, qui remplit cette table.

```
DECLARE
 …
 vQuarter CHARACTER(6);
 vMonth_Name VARCHAR(20);
 …
BEGIN
 …
 WHILE …
 LOOP
  …
  vQuarter := TO_CHAR(vDate_ID,'YYYY')||'Q'||TO_CHAR(vDate_ID,'Q');
  vMonth_Num := TO_NUMBER(TO_CHAR(vDate_ID,'MM'));
  …
  INSERT INTO Time_Dim VALUES (…);
  …
 END LOOP;
END;
/
```

   - Exécuter le programme sous Oracle et vérifier le résultat.

Remplissage de la table Geography_Dim

11) Travail à réaliser :
   - Pour chaque colonne de la table Geography_Dim, spécifier de quelle(s) donnée(s) source elle dépend.

- Créer le job Job04_Geography_Dim.
- Lancer le job et vérifier le résultat sous Oracle.

Remplissage de la table Order_Fact

12) Travail à réaliser :
- Pour chaque colonne de la table Order_Fact, spécifier de quelle(s) donnée(s) source elle dépend.
- Créer le job Job05_Order_Fact.
- Lancer le job et vérifier le résultat sous Oracle.

Lancement des jobs

Dans une étude réelle, les données sources évoluent en permanence. Les jobs doivent donc être planifiés régulièrement. Le lancement des jobs pourra se faire par exemple toutes les nuits pour prendre en compte les données modifiées pendant la journée. La planification des jobs peut se faire grâce au planificateur de tâches du système d'exploitation.

Remarque : Dans la solution (payante) Talend Integration Suite, la planification peut se faire directement sous Talend. De plus, il est possible de prendre en compte uniquement les données qui ont été modifiées pour accélérer le temps de chargement de l'entrepôt.

13) Travail à réaliser :
- Exporter chaque job dans le répertoire C:/orion.
  - Clic droit sur le job / Export Job Scripts
  - Export type : Autonomous job
  - Cocher l'option Extract the zip file
  - Options par défaut
- Relancer le script de suppression puis création des tables de l'entrepôt.
- Relancer le script de remplissage de la table Time_Dim.
- Relancer les jobs en cliquant sur les fichiers .bat.
- Vérifier que tout a bien fonctionné.

Les jobs vont maintenant être planifiés grâce au planificateur de tâches de Windows.

14) Travail à réaliser :
- Ouvrir le planificateur de tâches de Windows
  - Accessoires / Outils système / Planificateur de tâches
- Cliquer sur Créer une tâche… et donner lui un nom : orion_jobsETL
- Créer un déclencheur (dans 15 min par exemple)

- Créer une action :
  - Action : Démarrer un programme
  - Programme : C:\orion\Job01_Customer_Dim\Job01_Customer_Dim\Job01_Customer_Dim_run.bat
  - Commencer dans : C:\orion\Job01_Customer_Dim\Job01_Customer_Dim
- Ajouter de la même façon une action pour chaque job.
- Relancer le script de suppression puis création des tables de l'entrepôt.
- Relancer le script de remplissage de la table Time_Dim.
- Attendre que la tache se lance.
- Vérifier que tout a bien fonctionné.

Création du datamart

Pour la suite de l'étude, un datamart sera construit avec uniquement les clients membres du club Orion Gold et ayant acheté des vêtements ou des chaussures (pour accélérer les requêtes).

Il existe 3 solutions pour créer un datamart :
- Solution 1 : Vues logiques de l'entrepôt
- Solution 2 : Vues matérialisées de l'entrepôt
- Solution 3 : Base de données indépendante + jobs de remplissage

15) Travail à réaliser : (Implémentation de la solution 3)
- Créer un nouvel utilisateur orion_DM_user.
- Créer le script de création du datamart puis les jobs permettant de remplir les différentes tables.
  - Conditions à spécifier :
  - WHERE Customer_Group = 'Orion Club Gold members';
  - WHERE Product_Line LIKE 'Clothes%';
  - Cocher la case Inner join dans les jointures.
- Vérifier que tout a bien fonctionné.

16) Travail à réaliser : (Implémentation de la solution 1)
- Créer un nouvel utilisateur orion_DM_V_user.
- Créer le script de création des vues logiques.
- Vérifier que tout a bien fonctionné.

17) Travail à réaliser : (Implémentation de la solution 2)
- Créer un nouvel utilisateur orion_DM_MV_user.
- Créer le script de création des vues matérialisées.
- Vérifier que tout a bien fonctionné.