

# ACSSI

DATA  
MANAGEMENT  
ERP

**talend**  
A Qlik Company

# OPEN STUDIO 8.0

CAS PRATIQUE 1

# CONTEXTE

### 1. Objet de la prestation

- La société Citroën souhaite se doter d'une solution décisionnelle avec l'outil Talend pour l'alimentation de son datawarehouse.

### 2. Contexte du projet

- La société Citroën dispose de nombreuses données de ventes et référentiel sous forme de fichiers plats mais ne dispose pas de solution d'alimentation.
- Dans le cadre de la recherche d'une performance et d'une industrialisation accrue dans l'intégration et la transformation de ses données, la société Citroën souhaite se doter d'un système décisionnel intégrant les données actuelles et futures contenues dans ses fichiers plats et ses bases de données. Celle-ci doit lui permettre d'avoir à disposition un datawarehouse stable lui permettant de réaliser les différentes activités de diagnostic, d'analyse et de bilan sur ses opérations.

### 3. Données

- Les données de Citroën sont principalement disponibles dans des fichiers plats de type excel.

# CAHIER DES CHARGES: RÈGLE DE GESTION

### HUB:

Notre HUB sera constitué d'un ensemble de tables correspondant à l'ensemble de nos fichiers : 1 fichier => 1 table. Cette insertion sera faite en mode Insert/Update.

Une fois notre HUB créé nous aurons donc en base de données autant de tables que nous avons de fichiers dans DATA\_TP1.

Les fichiers sont à charger en l'état, aucune modification ne leur sera apporté. Nous essaierons un maximum d'avoir en BDD le type correspondant à la donnée dans nos fichiers ( INT pour les entiers etc ... ).

Au besoin vous pouvez faire une première version de vos flux en mode « Supprimez la table si elle existe et la créer » / Insérer.

# CAHIER DES CHARGES: RÈGLE DE GESTION

### DWH:

Au moins 7 tables sont à créer. Vous trouverez le détail des règles de gestion ci-après. Chaque table est à alimenter en mode insert/update. A nouveau, au besoin nous pouvons passer par une première version en mode « Supprimez la table si elle existe et la créer » puis Insérer la donnée.

### *td\_ref\_options:*

Cette table est la concaténation des tables Couleur, Options, Puissance et Energie du HUB.

Pour différencier chaque ligne, une colonne contenant le type de l'option est à créer.

### *td\_ref\_financement:*

Cette table ressemble à celle du HUB. Cependant, il faut convertir l'entretien et les intérêts en pourcentage.

# CAHIER DES CHARGES: RÈGLE DE GESTION

### DWH(suite):

#### *td\_ref\_vehicule:*

Cette table ressemble à celle du HUB. Cependant, il faut calculer la marge HT correspondant à la différence entre le prix de base du véhicule et le coût de fabrication.

#### *td\_ref\_vendeur:*

Cette table ressemble à celle du HUB. Cependant, il faut calculer et créer un nouveau champ: emailVendeur.

Ce champ est construit de la façon suivante:

→ 1ère lettre du prénom.nom@citroen.com (ex: r.caron@citroen.com)

# CAHIER DES CHARGES: RÈGLE DE GESTION

### DWH(suite):

#### *tf\_cde\_devis:*

Cette table ressemble à celle du HUB. Cependant, il faut ajouter 2 champs supplémentaires :

- CommandeAssocie: flag indiquant si une commande est associée.
- DelaiCommande: Nombre de jours entre le devis et la date de la commande.

#### *tf\_cde\_commande\_detail:*

Cette table correspond au détail de chaque commande.

Il faut pouvoir obtenir une ligne par options dans la commande.

Pour éviter les erreurs de doublons, nous créerons le champ CodeCommande comme étant la concaténation du CodeCommande et de la date de la commande dans le hub.

Nous rajouterons un champ IdLigneCommande qui doit être incrémenté en fonction du nombre de lignes pour chaque commande.

Pour chacune des lignes, il faut ajouter le montant de celle ci correspondant au prix de l'option auquel on ajoute une TVA.

Attention cette TVA peut être amenée à changer. Il vous est donc demandé de la variabiliser.

## INTRODUCTION

# CAHIER DES CHARGES: RÈGLE DE GESTION

### DWH(suite):

#### tf\_cde\_commande\_entete:

Cette table correspond à l'entête de chaque commande.

Il va donc falloir obtenir une ligne par code commande.

Pour éviter les erreurs de doublons, le CodeCommande devra être la concaténation du CodeCommande et de la date dans le hub.

Pour chacune des lignes, il faut ajouter le montant total de la commande correspondant au calcul suivant:

- $\text{MontantLigne} (= \text{montant de chacune des options/type d'énergie etc..}) + \text{PrixDeBaseVehicule}$

Il faut aussi calculer la prime du vendeur dans le champs PrimeVendeur de cette même table. La prime du commercial étant sous la forme d'un pourcentage du prix total de la vente.

Chaque attribut de la commande correspond à un id de la table associée.

# CAHIER DES CHARGES: RÈGLE DE GESTION

Il serait apprécié de recevoir un rapport d'exécution par e-mail.

Le format de ce mail est libre. Nous souhaitons juste avoir dans ce rapport les anomalies s'il y en a, le nombre de ligne chargé par table et d'autres infos que vous jugerez utiles.

Les adresses emails des destinataires ainsi que la configuration du serveur smtp seront présent dans 2 tables distinctes : Paramétrage et Mails.

Une table de suivi des traitements devra être alimentée : elle contiendra la date de l'exécution, l'heure de début et de fin, la durée, le nombre de ligne insérée et le statut (OK, En cours, En erreur).

A BIENTÔT !

**MERCI POUR VOTRE  
PARTICIPATION**