

# Ateliers N°1

## Installation et Exploration des bases de données AdventureWorks2012 et AdventureWorksDW2012

---

### Objectif :

La base de données exemple **AdventureWorks2012** est un bon exemple d'une base de données **transactionnelle**. Elle possède tous les éléments nécessaires pour vous permettre de voir des exemples de schémas transactionnels.

La base de données exemple **AdventureWorksDW2012** est un bon exemple d'un **datawarehouse**. Elle possède tous les éléments nécessaires pour vous permettre de voir des exemples de divers types de modélisation dimensionnelle.

Après cet atelier, vous serez en mesure de:

- Comprendre pourquoi un schéma normalisé provoque des problèmes pour l'édition des rapports ;
- Comprendre le schéma en étoile/flocon ;
- Définir les types de colonnes de dimension et utilisez des hiérarchies naturelles ;
- Comprendre et résoudre le problème des dimensions changeant lentement ;
- Définir les types de colonnes de la table de faits.

### Installation du jeu de données

Pour les ateliers vous utiliserez les données transactionnelles que vous importez à partir de la base **AdventureWorks2012**.

**AdventureWorksDW2012** - Il s'agit d'un entrepôt de données relationnel qui s'exécute sur une instance du Database Engine.

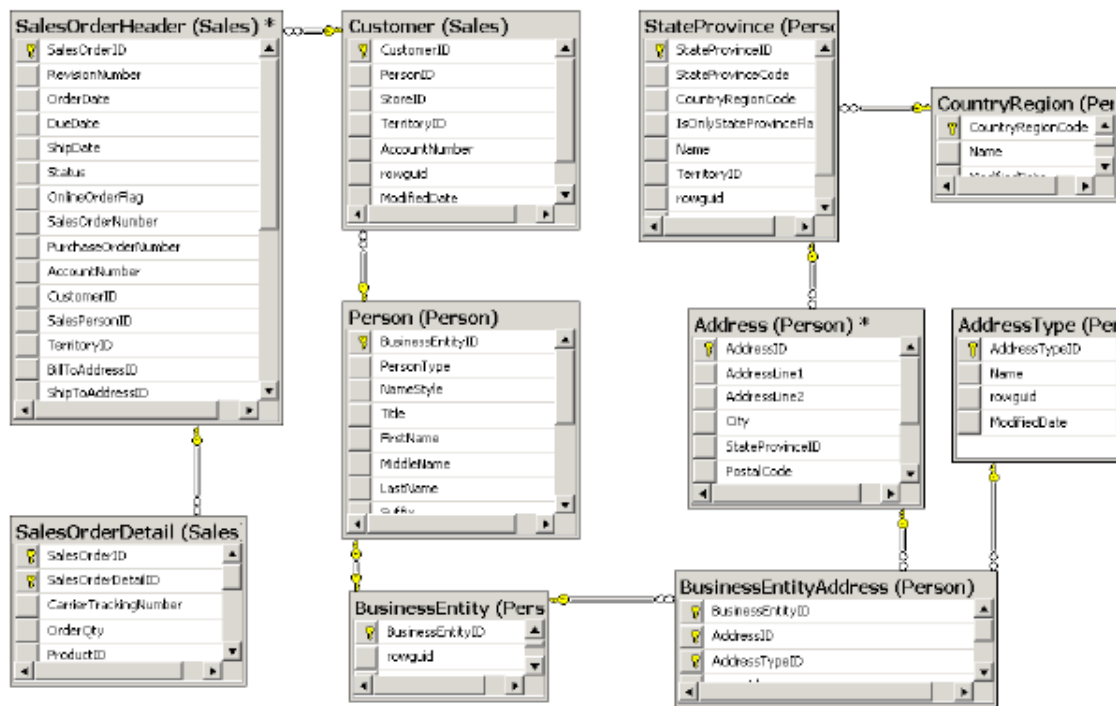
Pour installer cette base de données, procéder comme suit :

1. Télécharger la base de données **AdventureWorkDW2012** partir de la page d'échantillons de produits sur CodePlex.
2. Copier le fichier **AdventureWorksDW2012\_Data.mdf** dans le répertoire de données de l'instance locale du Database Engine de SQL Server. Par défaut, il se trouve dans C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\Data.
3. Démarrer **SQL Server Management Studio (SSMS)** et se connecter à l'instance du Database Engine.
4. Cliquer-droit sur Bases de données, cliquer sur **Joindre**.
5. Cliquer sur **Ajouter**.
6. Sélectionner le fichier de base de données **AdventureWorksDW2012\_Data.mdf** et cliquer sur OK. Si le fichier n'est pas répertorié, consulter le dossier C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\Data pour être sûr que le fichier est là.
7. Dans les **détails de la base de données**, supprimer l'entrée du fichier journal. Le programme d'installation suppose que vous avez un fichier journal, mais il n'y a aucun fichier journal dans l'échantillon. Un nouveau fichier journal sera créé automatiquement lorsque vous attachez la base de données. Sélectionner le fichier journal, cliquez sur **Supprimer** et puis cliquer sur **OK**.

## Explorer le schéma de la base AdventureWorks2012

Le schéma de base de données AdventureWorks2012 est fortement normalisé ; Il est conçu comme un schéma d'exemple pour soutenir des applications métier. Bien qu'un tel schéma fonctionne très bien pour les applications métier, elle peut causer des problèmes lorsqu'il est utilisé comme source pour les rapports.

Dans cet atelier, vous commencez par explorer un schéma relationnel normalisé. Supposons que vous deviez créer un rapport d'affaires du schéma relationnel AdventureWorks2012. Le rapport devrait inclure le montant des ventes accomplies à partir d'Internet dans différents pays sur plusieurs années. La tâche consiste à trouver les tables et les colonnes qui permettront de créer votre rapport. Vous commencez en enquêtant sur les tables qui stockent les données dont vous avez besoin, comme illustré dans la Figure 1.

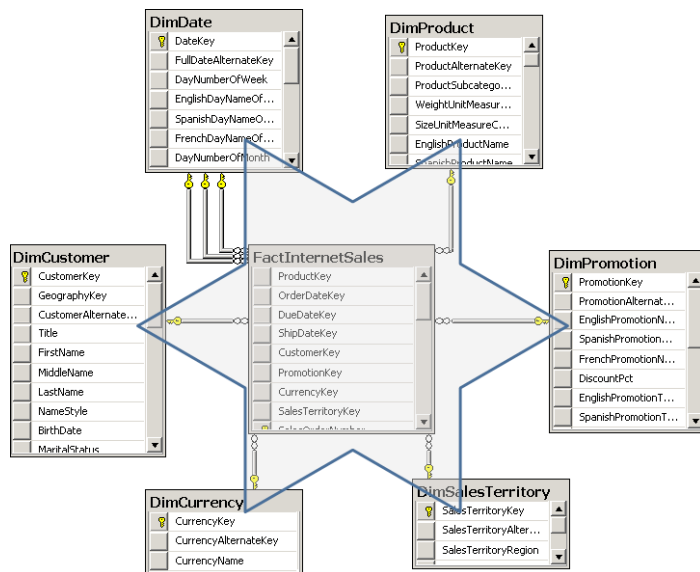


**FIGURE 1** Un diagramme des tables dont vous auriez besoin pour un simple rapport des ventes.

Trouver les tables appropriées et les colonnes dont vous avez besoin pour un rapport peut être une tâche très difficile dans une base de données normalisée, simplement en raison du nombre de tables impliquées. En outre, une requête qui joint les 10 tables, comme il faudrait pour faire des rapports de ventes par pays pendant des années, ne serait pas très rapide. La requête aurait aussi d'énormes quantités de données — ventes sur plusieurs années — ce qui interférerait avec le travail transactionnel régulier d'insertion et de mise à jour des données. Un autre problème dans cet exemple est le fait qu'il n'y a aucune table explicite pour les dates. Vous devez extraire les colonnes date/time d'une sales.tables, comme la colonne OrderDate de la table SalesOrderHeader.

## Explorer le schéma de la base AdventureWorksDW2012

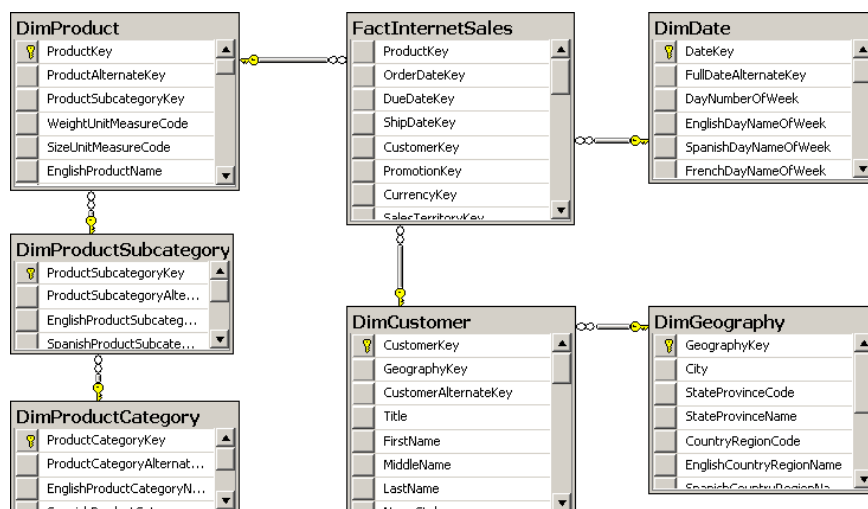
La base de données AdventureWorksDW2012 est un entrepôt de données fédérant plusieurs schémas en étoile/flocon. La figure 2 en est un exemple. Une étoile couvre un secteur d'activité. Dans ce cas, le schéma couvre les ventes sur Internet. Un entrepôt de données couvre plusieurs domaines d'activité et se compose de plusieurs schémas en étoiles (et / ou en flocon).



## Exercice : Exploration du DW AdventureWorksDW2012.

1. Lancer SSMS et se connecter à votre instance de SQL Server. Développez le dossier Bases de données, puis la base de données **AdventureWorksDW2012**.
2. Cliquer-droit sur le dossier diagrammes de base de données et sélectionnez l'option **Nouveau schéma de base de données**.
3. Dans la liste **Ajouter une table**, sélectionner les tables suivantes (cliquer sur chaque table et puis cliquer sur le bouton Ajouter) :
  - DimCustomer ; DimDate ; DimGeography
  - DimProduct ; DimProductCategory ; DimProductSubcategory
  - FactInternetSales

Le diagramme doit ressembler à la figure suivante



4. Analyser soigneusement les tables, les colonnes et les relations.

5. Enregistrer le diagramme avec le nom **DW\_02\_01\_InternetSales**.

## Analyser le diagramme

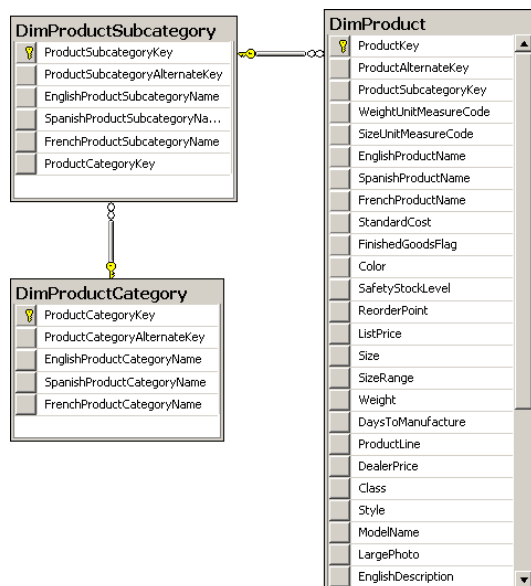
Examiner le schéma AdventureWorksDW2012 pour constater les faits suivants :

- La table de dimension **DimDate** n'a pas de tables **Lookup** supplémentaires qui lui sont associés et utilise donc un **schéma en étoile**.
- La table de dimension de **DimProduct** est snowflaked ; elle utilise la table lookup **DimProductSubcategory**, qui utilise encore la table lookup **DimProductCategory**.
- La table de dimension **DimCustomer** utilise un schéma hybride. le premier niveau du schéma est en snowflake seulement à travers la table lookup **DimGeography**. La table **DimGeography** est **dénormalisée**.

## Exploration des dimensions d'AdventureWorksDW2012

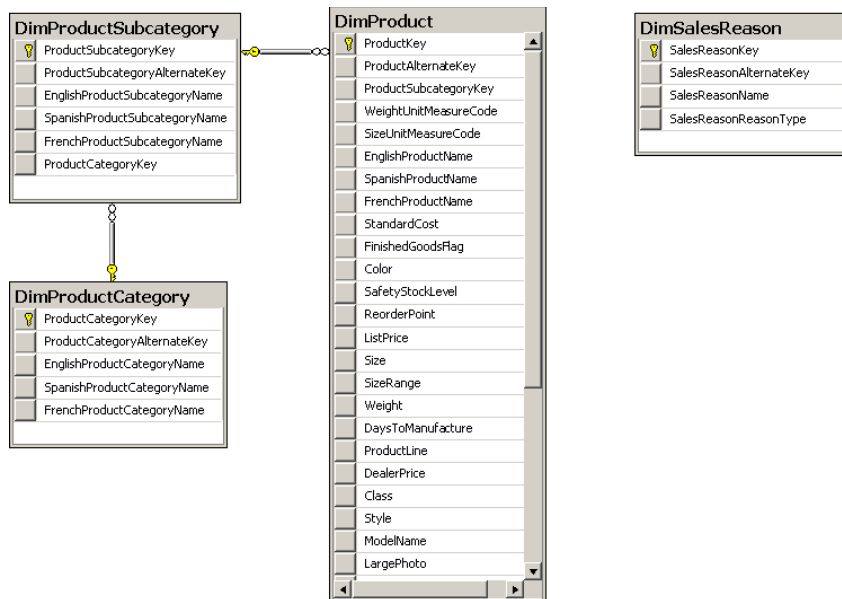
La base de données exemple **AdventureWorksDW2012** possède plusieurs dimensions. Dans ce TP, vous explorerez certaines d'entre elles.

1. Cliquer-droit sur le dossier diagrammes de base de données et sélectionnez l'option **Nouveau schéma de base de données**.
2. Dans la liste **Ajouter une table**, sélectionner les tables suivantes (cliquer sur chaque table et puis cliquer sur le bouton Ajouter) :
  - DimProduct
  - DimProductCategory
  - DimProductSubcategory
3. Le diagramme doit ressembler à la figure suivante



4. Essayer de trouver les colonnes utilisées dans les buts suivants
  - clés
  - noms
  - traductions
  - attributs
5. Ajouter la table **DimSalesReason** au schéma

- Le diagramme doit ressembler à la figure suivante
- Analyser soigneusement les tables, les colonnes et les relations.
- Enregistrer le diagramme avec le nom **DW\_02\_02\_InternetSales**.



## Exploration des tables de faits d'AdventureWorksDW2012

La base de données exemple **AdventureWorksDW2012** possède de nombreuses tables de faits ainsi, afin de montrer toutes les mesures possibles. Dans ce TP, vous allez examiner l'une d'elle.

- Cliquer-droit sur le dossier diagrammes de base de données et sélectionnez l'option **Nouveau schéma de base de données**.
- Dans la liste **Ajouter une table**, sélectionner les tables suivantes (cliquer sur chaque table et puis cliquer sur le bouton Ajouter) :
  - DimProduct
  - DimDate
  - FactProductInventory
- Le diagramme doit ressembler à la figure suivante
- Analyser soigneusement les tables, les colonnes et les relations.
- Enregistrer le diagramme avec le nom **DW\_02\_03\_InternetSales**.



## Slowly Changing Dimensions

Il y a un problème courant avec les dimensions dans un entrepôt de données : les données des dimensions changent au fil du temps. Ce n'est généralement pas un problème dans une application OLTP ; quand une donnée change il suffit de la mettre à jour dans l'OLTP. Toutefois, dans un DW, vous devez conserver l'historique. La question qui se pose est de savoir *comment* entretenir cet historique. Voulez-vous mettre à jour uniquement les données non modifiées, comme dans une application OLTP et prétendre que la valeur était toujours ça? Ce problème est connu dans le jargon de la DW par le problème du *Slowly Changing Dimension (SCD)*.

Le problème s'explique dans un exemple. Tableau 1 montre les données originales de OLTP source pour un client.

TABLE 1 Original OLTP Data for a Customer

CustomerId	FullName	City	Occupation
17	Bostjan Strazar	Vienna	Professional

Le client vit à Vienne, en Autriche et est un professionnel. Maintenant, imaginez que le client se déplace à Ljubljana, en Slovénie. Dans une base de données OLTP, vous aurez simplement à mettre à jour la colonne City, ayant pour résultat les valeurs indiquées dans le tableau 2.

TABLE 2 OLTP Data for a Customer After the City Change

CustomerId	FullName	City	Occupation
17	Bostjan Strazar	Ljubljana	Professional

Si vous créez un rapport, toutes les ventes historiques pour ce client sont maintenant attribués à la ville de Ljubljana et (à un niveau supérieur) à la Slovénie. Le fait que ce client a contribué au chiffre d'affaires à Vienne et en Autriche, dans le passé aurait disparu. Pour capturer le fait que le client a contribué au chiffre d'affaires dans une autre ville, pays ou région, vous ne pouvez pas simplement écraser les données; vous devez insérer une nouvelle ligne contenant les nouvelles données. Cependant, cela crée un problème concernant la valeur de la clé, la clé ne serait plus unique. Par conséquent, il vaut mieux ne pas utiliser la clé de l'entreprise comme clés primaire. La solution est d'introduire une nouvelle clé, une clé de l'entrepôt de données. Dans la terminologie DW, ce genre de clé est appelée une clé de substitution.