Atelier 3:
Loading Data and Using Data Compression and Indexes

Objectif :

Dans cet atelier, vous allez alimenter le datawarehouse LightAdventureWorksDW depuis AdventureWorksDW2012 pour des raisons de simplicité. Vous allez utiliser SSMS et créer un package ETL qui extrait des données à partir de la base de données relationnelle AdventureWorksDW2012, et charge les données dans le datawarehouse LightAdventureWorksDW.

After this lesson, you will be able to:

- Load data using SSMS ;
- Determine the control flow of an SSIS package ;
- Plan the configuration of connection managers ;
- Understand package-scoped and project-scoped connection managers ;
- Determine the containers and tasks needed for an operation ;
- Implement the appropriate control flow task to solve a problem ;
- Use sequence containers and loop containers ;
- Use clustered and nonclustered indexes on a dimension and on a fact table ;
- Use data compression.

Exercice 1 : Loading data using SSMS

Dans cet exercice, vous allez charger les données de votre DW avec SSMS

1. Démarrez SSMS et connectez-vous à votre instance de SQL Server. Ouvrez une nouvelle fenêtre de requête en cliquant sur le bouton New Query button.
2. Connectez vous à la base de données LightAdventureWorksDW. Ecrire un script pour charger les données de la dimension Customers en utilisant les informations de u tableau ci-dessous.

Nom de la colonne	Type de donnée	Nullability	Remarques
CustomerDwKey	INT	NOT NULL	Clé de substitution ; attribuer des valeurs avec une séquence
CustomerKey	INT	NOT NULL	
FullName	NVARCHAR(150)	NULL	Concaténer *FirstName* et *LastName* de *DimCustomer*
EmailAddress	NVARCHAR(50)	NULL	
BirthDate	DATE	NULL	
MaritalStatus	NCHAR(1)	NULL	
Gender	NCHAR(1)	NULL	

Education	NVARCHAR(40)	NULL	*EnglishEducation* de *DimCustomer*
Occupation	NVARCHAR(100)	NULL	*EnglishOccupation* de *DimCustomer*
City	NVARCHAR(30)	NULL	*City* de *DimGeography*
StateProvince	NVARCHAR(50)	NULL	*StateProvinceName* de *DimGeography*
CountryRegion	NVARCHAR(50)	NULL	*EnglishCountryRegionName* de *DimGeography*
Age	Inherited	Inherited	Colonne calculée. Calculer la différence entre *BirthDate* et la date du jour, et la classifier en trois groupes : • Lorsque la différence <= 40, étiqueter "Younger" • Lorsque la différence> 50, étiqueter "Older" • Sinon étiqueter "Middle Age"
CurrentFlag	BIT	NOT NULL	1 par défaut

Le script utilisé est :

```
INSERT INTO LIGHTADVENTUREWORKSDW.dbo.Customers

(CustomerDwKey, CustomerKey, FullName, EmailAddress, Birthdate, MaritalStatus, Gender, Education, Occupation, City, StateProvince, CountryRegion)

SELECT

NEXT VALUE FOR LIGHTADVENTUREWORKSDW.dbo.SeqCustomerDwKey AS CustomerDwKey,

C.CustomerKey, C.FirstName + ' ' + C.LastName AS FullName, C.EmailAddress, C.BirthDate, C.MaritalStatus, C.Gender, C.EnglishEducation, C.EnglishOccupation, G.City, G.StateProvinceName, G.EnglishCountryRegionName

FROM AdventureWorksDW2012.dbo.DimCustomer AS C

INNER JOIN AdventureWorksDW2012.dbo.DimGeography AS G

ON C.GeographyKey = G.GeographyKey;

GO
```

Exercice2 : Loading data using Integration Services

Etape1 : Création du package

La première étape dans la création d'un package dans **Integration Services** est de créer un projet **Integration Services**.

3. Ouvrir **SQL Server Data Tools**
4. Dans le menu **Fichier**, pointer sur **Nouveau** et cliquer sur **Projet** pour créer un nouveau projet **Integration Services**.
5. Dans la boîte de dialogue Nouveau projet, sélectionner **Projet Integration Services** dans le volet Modèles.
6. Dans la zone Nom, remplacer le nom par défaut pour **LoadLightDWBySSIS**.
7. Cliquer sur **OK**.
8. Par défaut, un package vide, intitulé **Package.dtsx**, est créé et ajouté à votre projet.
9. Dans la barre d'outils Explorateur de solutions, cliquer droit sur Package.dtsx, cliquer sur **Renommer** et renommer le package par défaut en **LoadLightDWBySSIS.dtsx**.

Étape 2 : Ajout et configuration du gestionnaire de connexions OLE DB source

Dans cette étape, vous ajoutez un **gestionnaire de connexions de fichiers** au package que vous venez de créer. Un gestionnaire de connexions OLE DB permet à un package d'extraire ou de charger des données à partir d'une source de données OLE DB conforme. En utilisant le gestionnaire de connexions OLE DB, vous pouvez spécifier le serveur, la méthode d'authentification, et la base de données par défaut de la connexion.

Dans cette étape, vous allez créer un gestionnaire de connexions OLE DB qui utilise l'authentification Windows pour se connecter à l'instance locale d'**AdventureWorksDW2012**. Le gestionnaire de connexions OLE DB que vous créez sera également référencé par d'autres composants que vous allez créer plus tard dans ce TP.

Pour ajouter et configurer un gestionnaire de connexions OLE DB

1. Cliquer-droit n'importe où dans la zone **Gestionnaires de connexion**, puis cliquer sur **Nouvelle connexion OLE DB**.
2. Dans la boîte de dialogue **Configurer le gestionnaire de connexion OLE DB**, cliquer sur **Nouveau**.
3. Pour le nom du serveur, entrer **localhost**.
4. Vérifier que l'option **Utiliser l'authentification Windows** est sélectionnée.
5. Dans le connecter à un groupe de base de données, dans la zone Sélectionner ou entrer un nom de base de données, taper ou sélectionner **AdventureWorksDW2012**.
6. Cliquer sur **Tester la connexion** pour vérifier que les paramètres de connexion que vous avez spécifiées sont valides.
7. Cliquez sur **OK**.
8. Cliquez sur **OK**.
9. Dans le volet Connexions de données de la boîte de dialogue **Configurer le gestionnaire de connexion OLE DB**, vérifier que **localhost.AdventureWorks2012** est sélectionné.

10. Cliquez sur **OK**

Étape 3 : Ajout et configuration du gestionnaire de connexions OLE DB destination

Après avoir ajouté un gestionnaire de connexions pour se connecter à la source de données, la tâche suivante consiste à ajouter un **gestionnaire de connexions OLE DB** pour se connecter à la destination. Un gestionnaire de connexions OLE DB permet à un package d'extraire ou de charger des données à partir d'une source de données OLE DB conforme. En utilisant le gestionnaire de connexions OLE DB, vous pouvez spécifier le serveur, la méthode d'authentification, et la base de données par défaut de la connexion.

Dans cette étape, vous allez créer un gestionnaire de connexions OLE DB qui utilise l'authentification Windows pour se connecter à l'instance locale de **LightAdventureWorksDW**. Le gestionnaire de connexions OLE DB que vous créez sera également référencé par d'autres composants que vous allez créer plus tard dans ce TP.

Pour ajouter et configurer un gestionnaire de connexions OLE DB

11. Cliquer-droit n'importe où dans la zone **Gestionnaires de connexion**, puis cliquer sur **Nouvelle connexion OLE DB**.
12. Dans la boîte de dialogue **Configurer le gestionnaire de connexion OLE DB**, cliquer sur **Nouveau**.
13. Pour le nom du serveur, entrer **localhost**.
14. Vérifier que l'option **Utiliser l'authentification Windows** est sélectionnée.
15. Dans le connecter à un groupe de base de données, dans la zone Sélectionner ou entrer un nom de base de données, taper ou sélectionner **LightAdventureWorksDW**.
16. Cliquer sur **Tester la connexion** pour vérifier que les paramètres de connexion que vous avez spécifiées sont valides.
17. Cliquez sur **OK**.
18. Cliquez sur **OK**.
19. Dans le volet Connexions de données de la boîte de dialogue **Configurer le gestionnaire de connexion OLE DB**, vérifier que **localhost. LightAdventureWorksDW** est sélectionné.
20. Cliquez sur **OK**

Ajout d'une tâche SQLTASK loadCustomers

1. Commencez par supprimer les données que vous avez chargées à l'aide de SSMS dans la table customers
2. Cliquer sur l'onglet **Flux de contrôle**.
3. Dans la boîte à outils, glisser une **exécuter une tâche SQL** sur la surface de dessin de l'onglet Flux de contrôle.
4. Cliquer-droit sur la tâche de flux de données nouvellement ajouté, cliquez sur **Renommer** et modifier le nom à **LoadCustomersScript**.
5. Spécifier la connexion et dans SQL Statement mettre le script décrit en haut.

Étape 4 : Ajout d'une tâche de flux de données LoadProduct au package

Après avoir créé les gestionnaires de connexions pour les données source et destination, la tâche suivante consiste à ajouter une tâche de flux de données à votre package. La tâche de flux de données encapsule le moteur de flux de données qui se déplace entre les données sources et destinations, et fournit la fonctionnalité pour la **transformation**, le **nettoyage** et la **modification** des données pendant leur déplacement. La tâche de flux de données est où la plupart des tâches d'extraction, de transformation et de chargement (**ETL**) se produisent.

Pour ajouter une tâche de flux de données

6. Cliquer sur l'onglet **Flux de contrôle**.
7. Dans la boîte à outils, glisser une **Tâche de flux de données** sur la surface de dessin de l'onglet Flux de contrôle.
8. Cliquer-droit sur la tâche de flux de données nouvellement ajouté, cliquez sur **Renommer** et modifier le nom à **Product Data Flow**.

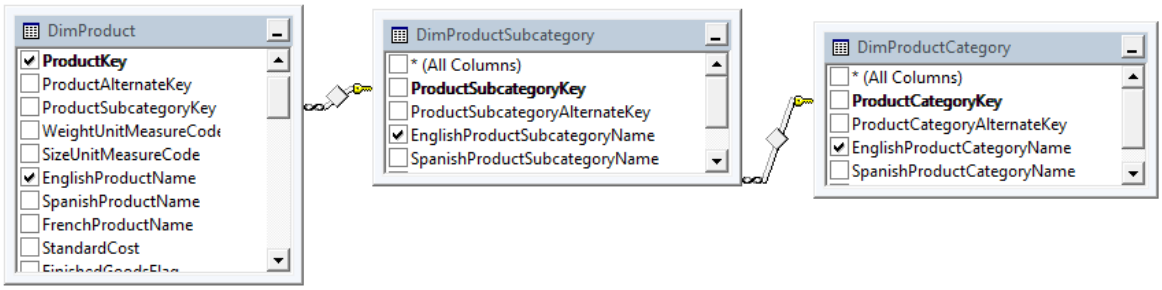
Étape 5 : Ajout et configuration de la source OLE DB

Dans cette étape, vous allez ajouter et configurer le fichier source de votre package. Un fichier source est un composant de flux de données qui utilise les métadonnées définies par un gestionnaire de connexions de fichiers pour spécifier le format et la structure des données à extraire du fichier par un processus de transformation.

Pour ce TP, vous allez configurer le fichier source pour utiliser le gestionnaire de connexions de fichiers que vous avez créé précédemment.

Pour ajouter et configurer la source OLE DB

1. Ouvrir le **concepteur de flux de données**, soit en double-cliquant sur la tâche de flux de données **Product Data Flow** ou en cliquant sur l'onglet **Flux de données**.
2. Dans la boîte à outils, glisser une **source OLE DB** sur la surface de dessin de l'onglet Flux de données.
3. Cliquer-droit sur la source de données nouvellement ajoutée, cliquer sur **Renommer** et modifier le nom en **Extract Product Data**.
4. Double-cliquer sur **Extract Product Data**.
5. Dans la boîte de dialogue **Éditeur de source OLE DB**, veiller à ce que **localhost.AdventureWorksDW2012** soit sélectionné dans la boîte de gestionnaire de connexions OLE DB.
6. Dans le **Mode d'accès aux données**, sélectionner **Commande SQL**
7. Avec Build query construire la requête comme suit



sinon copiez le script suivant:

```
SELECT P.ProductKey, P.EnglishProductName, P.Color,
      P.Size, S.EnglishProductSubcategoryName,
      C.EnglishProductCategoryName
FROM        AdventureWorksDW2012.dbo.DimProduct AS P
INNER JOIN
            AdventureWorksDW2012.dbo.DimProductSubcategory AS S
ON
            P.ProductSubcategoryKey = S.ProductSubcategoryKey
INNER JOIN
            AdventureWorksDW2012.dbo.DimProductCategory AS C
ON
            S.ProductCategoryKey = C.ProductCategoryKey;
```

8. Cliquer sur **Colonnes** et vérifier que les noms des colonnes sont corrects.
9. Cliquer sur **OK**.

Étape 7 : Ajout et configuration la destination OLE DB

Votre package peut maintenant extraire des données à partir de la base de données source et transformer ces données dans un format qui est compatible avec la destination. L'étape suivante consiste à charger effectivement les données transformées dans la destination. Pour charger les données, vous devez ajouter une destination OLE DB pour le flux de données. La destination OLE DB peut utiliser une table de base de données, une vue ou une commande SQL pour charger des données les bases de données.

Dans ce TP, vous ajoutez et configurez une destination OLE DB à utiliser le gestionnaire de connexions OLE DB que vous avez créé précédemment.

Pour ajouter et configurer la destination OLE DB

1. Dans **la boîte à outils SSIS**, glisser **destination OLE DB** sur la surface de dessin de l'onglet **Flux de données**. Placer la destination OLE DB directement en dessous de la transformation **Sort Data**.
2. Renommer le composant de destination en **Load Product Data**.
3. Double-cliquer sur **Load Product Data**.
4. Dans la boîte de dialogue **Éditeur de destination OLE DB**, veiller à ce que **localhost.LightAdventureWorksDW** soit sélectionnée dans la boîte de gestionnaire de connexions OLE DB.

5. Dans le nom de la table ou de la vue, taper ou sélectionner **[dbo].[Products]**.
6. Cliquez sur **Mappages**.
7. Vérifier que les différentes colonnes d'entrée sont mappées correctement aux colonnes de destination.
8. Cliquer sur **OK**.

Flux de contrôle

Ajout d'une tâche de flux de données LoadDate au package

Selon la même procédure que la transformation précédente, réaliser l'extraction et le chargement des données vers la dimension **Date** du datawarehouse **LightAdventureWorksDW en utilisant le script :**

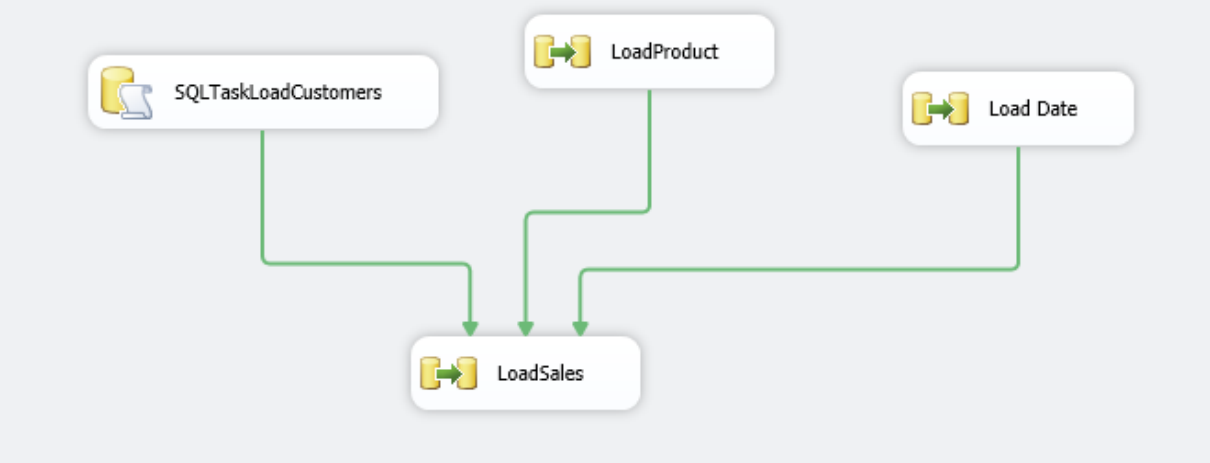
```
SELECT DateKey, FullDateAlternateKey as FullDate,
SUBSTRING(CONVERT(CHAR(8), FullDateAlternateKey, 112), 5, 2)
+ ' ' + EnglishMonthName as MonthNumberName,
CalendarQuarter, CalendarYear
FROM AdventureWorksDW2012.dbo.DimDate;
```

Ajout d'une tâche de flux de données LoadInternetSale au package

1. Selon la même procédure que la transformation précédente, réaliser l'extraction et le chargement des données vers la table de faits **InternetSales** en utilisant le script :

```
SELECT C.CustomerDwKey,
FIS.ProductKey, FIS.OrderDateKey,
FIS.OrderQuantity, FIS.SalesAmount,
FIS.UnitPrice, FIS.DiscountAmount
FROM AdventureWorksDW2012.dbo.FactInternetSales AS FIS
INNER JOIN LightAdventureWorksDW.dbo.Customers AS C
ON FIS.CustomerKey = C.CustomerKey;
```

2. Utiliser un conteneur de séquences pour ordonnancer l'exécution des flux de données comme dans le schéma suivant (les dimensions doivent être alimentées avant la table de faits) :



Exécuter les transformations

Dans le menu Déboguer, cliquez sur Démarrer le débogage ou sur exécuter.

Exercice 3 : Applying Data Compression

Dans cet exercice, vous appliquez la compression de données sur la table de faits InternetSales

1. Utilisez la procédure stockée système sp_spaceused pour calculer l'espace utilisé par la table InternetSales. Utilisez le code suivant.

```
EXEC sp_spaceused N'dbo.InternetSales', @updateusage = N'TRUE';
GO
```

2. La table doit utiliser environ 3, 080 KB pour l'espace réservé. Maintenant, utilisez l'instruction ALTER TABLE pour compresser la table. Utilisez la compression de page, comme illustré dans le code suivant.

```
ALTER TABLE dbo.InternetSales
REBUILD WITH (DATA_COMPRESSION = PAGE);
GO
```

3. Calculer l'espace réservé à nouveau.

```
EXEC sp_spaceused N'dbo.InternetSales', @updateusage = N'TRUE';
GO
```

4. La table devrait maintenant utiliser environ 1, 096 KB pour l'espace réservé. Vous pouvez voir que vous avez épargné près des deux tiers de l'espace à l'aide de la compression de page.

   Pour plus d'informations, veuillez consulter les white papers à l'adresse http://msdn.microsoft.com/en-us/library/dd894051(SQL.100).aspx

Exercice 4 : using index

5. La requête suivante agrège la colonne SalesAmount selon la colonne ProductKey de la table FactInternetSales dans la base de données AdventureWorksDW2012. Le code définit également STATISTICS IO our calculer les IO.

```
USE AdventureWorksDW2012;
GO
```

```
SET STATISTICS IO ON;
GO
SELECT ProductKey,
SUM(SalesAmount) AS Sales,
COUNT_BIG(*) AS NumberOfRows
FROM dbo.FactInternetSales
GROUP BY ProductKey;
GO
```

Cette commande devrait donner les résultats suivant dans l'onglet Message

```
(158 row(s) affected)
Table 'Worktable'. Scan count 0, logical reads 0, physical reads 0, read-ahead reads 0, lob
logical reads 0, lob physical reads 0, lob read-ahead reads 0.
Table 'FactInternetSales'. Scan count 1, logical reads 2062, physical reads 0, read-ahead
reads 0, lob logical reads 0, lob physical reads 0, lob read-ahead reads 0.
```

6. La requête fait 2062 lectures logiques dans la table FactInternetSales. Vous pouvez créer une vue de cette requête et l'indexer, comme illustré dans le code suivant.

```
CREATE VIEW dbo.SalesByProduct
WITH SCHEMABINDING AS
SELECT ProductKey,
SUM(SalesAmount) AS Sales,
COUNT_BIG(*) AS NumberOfRows
FROM dbo.FactInternetSales
GROUP BY ProductKey;
GO
CREATE UNIQUE CLUSTERED INDEX CLU_SalesByProduct
ON dbo.SalesByProduct (ProductKey);
GO
```

(COUNT_BIG(*) returns the number of items in a group. This includes NULL values and duplicates)

7. Notez que la vue doit être créée avec l'option SCHEMABINDING, si vous voulez l'indexer. En outre, vous devez utiliser la fonction d'agrégat COUNT_BIG. Après avoir créé la vue et l'index, exécutez à nouveau la requête.

```
USE AdventureWorksDW2012;
GO
SET STATISTICS IO ON;
GO
SELECT ProductKey,
SUM(SalesAmount) AS Sales,
COUNT_BIG(*) AS NumberOfRows
FROM dbo.FactInternetSales
GROUP BY ProductKey;
GO
```

8. Cette fois ci la même commande donne comme résultat

```
(158 row(s) affected)
Table 'SalesByProduct'. Scan count 1, logical reads 2, physical reads 0, read-ahead reads
0, lob logical reads 0, lob physical reads 0, lob read-ahead reads 0.
```

Le nombre de lecture logique est bien réduit

9. Supprimer la vue

```
Drop view dbo.SalesByProduct
```