



M2 BI/Data Analyse

Datavisualisation dans Power BI

## M2 : BI/DATA ANALYSE – ETUDE DE CAS



### TABLE DES MATIERES

OBJECTIFS .....	2
demarche de conception d'une appication bi .....	3
Connexion au modèle HYPERMARCHE.....	4
Préparation des données .....	6
Création de la table Calendrier dans Power BI .....	6
Modèle final HYPERMARCHE .....	15
<b>Analyse et Création des mesures</b> .....	16
<b>Datavisualisation</b> .....	16

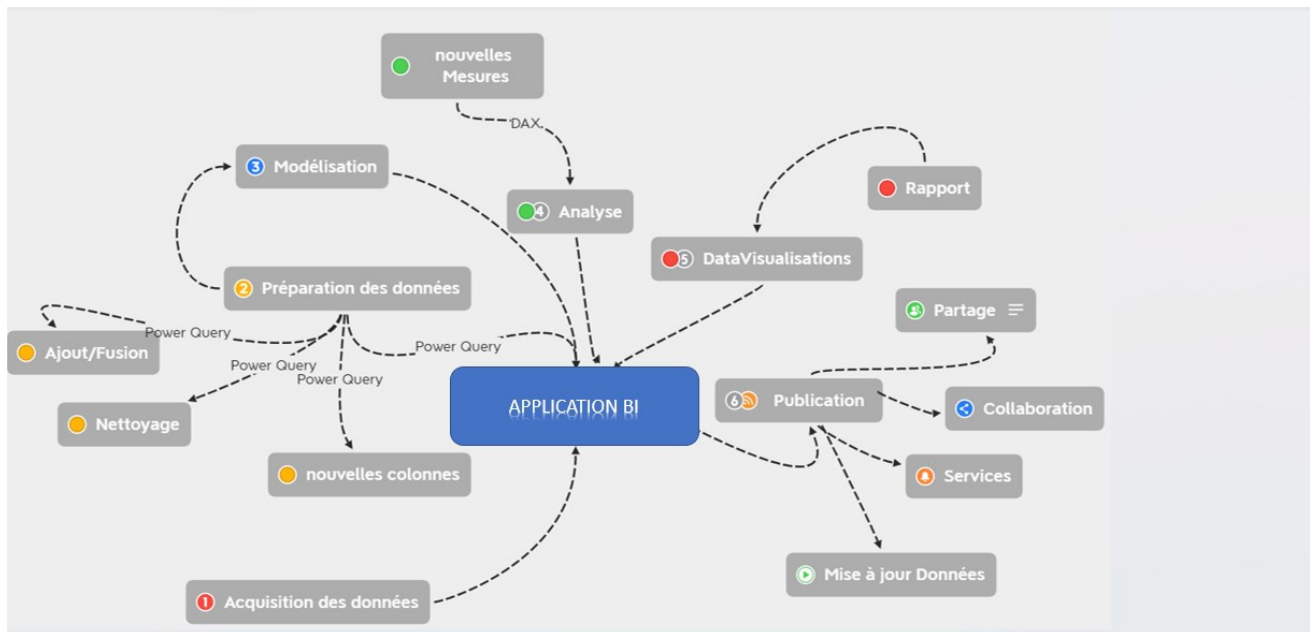
### OBJECTIFS

A partir du modèle HYPERMARCHE vous allez construire un rapport d'analyse permettant d'obtenir des indicateurs stratégiques sur l'activité des ventes .

**Ce rapport va s'appuyer sur le modèle HYPERMARCHE que vous alimentez depuis l'ETL PDI.**

## M2 : BI/DATA ANALYSE – ETUDE DE CAS

### DEMARCHE DE CONCEPTION D'UNE APPLICATION BI



L'ETL PDI nous a permis de réaliser les premières étapes :

- **ACQUISITION** des données : transformations correspondant à l'extraction depuis les fichiers xls et csv
- **PREPARATION** : Transformations PDI avec composants de transformation et d'agrégation
- **MODELISATION** : Alimentation des tables de dimension et de la table de FAIT pour optimiser la restitution

*\*Remarque : En général les règles de nettoyage, de préparation et de transformation de données sont réalisées dans l'ETL même si celles-ci peuvent être potentiellement effectuées dans Power Query. Le risque est de perdre ces règles si l'organisation est amenée à changer d'outil de Datavisualisation ce qui n'arrive pas en cas de changement d'ETL (beaucoup moins courant).*

POWER BI va nous permettre de réaliser les étapes suivantes :

- **ANALYSE** : Création de nouveaux KPI
- **DATAVISUALISATION** : Création de visuels pertinents pour comprendre le business

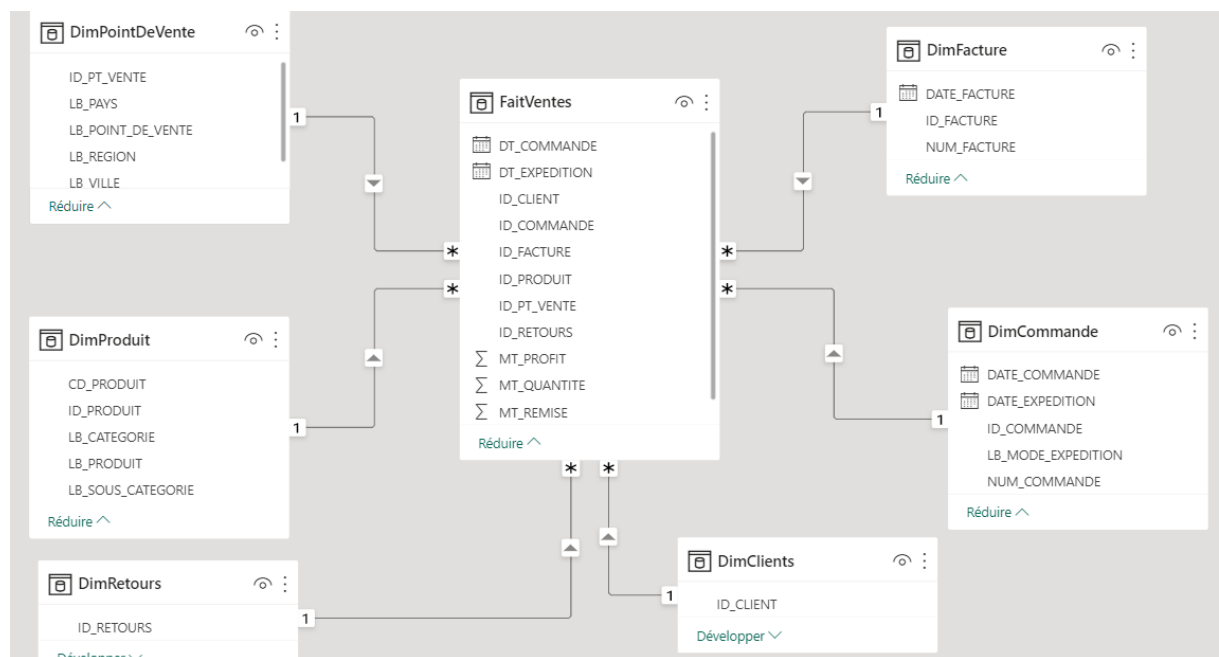
## M2 : BI/DATA ANALYSE – ETUDE DE CAS

### Connexion au modèle HYPERMARCHE

1 – Créer un nouveau rapport Power BI et depuis le bouton obtenir des données créer une connexion sur la base qui héberge le modèle Hypermarché.

- Importez toutes les tables (6 tables de Dimension 1 Table de Fait)
- Redéfinissez les jointures entre les différentes tables : Celles-ci se font directement si les clés étrangères portent les mêmes noms que les clés primaires des tables de dimension
- Vérifiez les directions des filtres et les cardinalités
- Au besoin vérifiez la cohérence du modèle en croisant quelques dimensions avec des Faits

Vue du modèle sous power BI :



Application des formats de présentation :

Appliquer un format de présentation Short Date pour les dates de commande et d'expédition de la table de FAIT :

DT_COMMANDE	DT_EXPEDITION
15/04/2011	17/04/2011
16/08/2011	21/08/2011
16/08/2011	21/08/2011
14/05/2013	18/05/2013
27/02/2013	04/03/2013

## M2 : BI/DATA ANALYSE – ETUDE DE CAS

Appliquer des formats de Devis € pour le montant des ventes et le profit :

MT_VENTE	MT_QUANTITE	MT_PROFIT
35,76 €	2	12,12 €
12,06 €	2	3,72 €
48,00 €	2	5,76 €
37,86 €	2	13,62 €
127,72 €	2	19,00 €

Appliquer un format « % » pour le montant Remise :

MT_REMISE
10,00 %
40,00 %
40,00 %

Modifier la fonction d'agrégation de la remise en appliquant par défaut une moyenne plutôt qu'une somme :

Moyenne	
Ne pas résumer	
Somme	
Moyenne	
Minimum	
Maximum	
Nombre	
Nombre (éléments distincts)	

Appliquer les rôles géographiques adéquats aux colonnes : Pays, Région, Ville

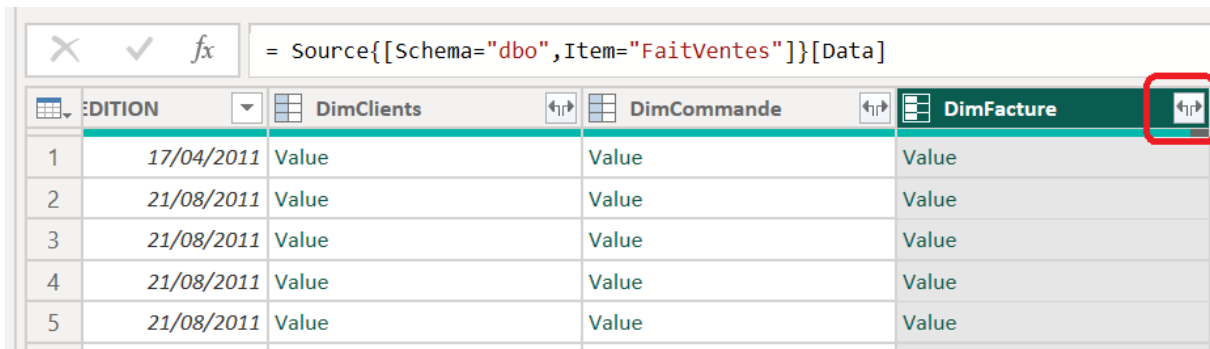
DimPointDeVente
ID_PT_VENTE
LB_PAYS
LB_POINT_DE_VENTE
LB_REGION
LB_VILLE
LB_ZONE_GEOGRAP

## M2 : BI/DATA ANALYSE – ETUDE DE CAS

### Préparation des données

Nous allons créer dans la table de FAIT une nouvelle colonne permettant d'identifier les commandes qui n'ont pas été facturées : 1 si la commande est facturée, 0 si ce n'est pas le cas.

Dans Power Query au niveau de la requête FAIT\_VENTE, rajouter la colonne NUM\_FACTURE depuis DIM\_FACTURE en cliquant sur les 2 flèches qui divergent :



EDITION	DimClients	DimCommande	DimFacture
1	17/04/2011	Value	Value
2	21/08/2011	Value	Value
3	21/08/2011	Value	Value
4	21/08/2011	Value	Value
5	21/08/2011	Value	Value

ajouter une nouvelle colonne personnalisée basée sur DIM\_FACTURE.NUM\_FACTURE avec la formule suivante :

*if [DimFacture.NUM\_FACTURE] is null then 1 else 0*

Renommer cette colonne « *identification commande non facturée* »

### CREATION DE LA TABLE CALENDRIER DANS POWER BI

#### Création du calendrier en langage « M »

La dimension Temps/Calendrier est une dimension quasiment obligatoire dans tous les modèles BI.

Celle-ci va permettre aux utilisateurs de construire tous les tableaux et graphiques permettant de suivre l'évolution de son activité

De façon générale, cette dimension est déjà présente dans les systèmes BI mais lorsque les utilisateurs sont amenés à créer leurs propres modèles, celle-ci doit être créée. La qualité du Calendrier est importante car pour utiliser les fonctions de Time Intelligence de Power BI, le calendrier doit être complet entre sa date de début et sa date de fin. La table calendrier se conçoit depuis un script « M » fourni par le professeur : « Calendar\_M.txt ». Elle est à créer dans Power Query à partir d'une requête vide et **dans l'éditeur avancé** :

## M2 : BI/DATA ANALYSE – ETUDE DE CAS

requête

Options d'affichage ▾

```
let
CreateDateTable = (StartDate as date, EndDate as date, optional Culture as nullable text) as table =>
let
DayCount = Duration.Days(Duration.From(EndDate - StartDate)),
Source = List.Dates(StartDate, DayCount, #duration(1,0,0,0)),
TableFromList = Table.FromList(Source, Splitter.SplitByNothing()),
ChangedType = Table.TransformColumnTypes(TableFromList,{{"Column1", type date}}),
RenamedColumns = Table.RenameColumns(ChangedType,{{"Column1", "Date"}}),
InsertYear = Table.AddColumn(RenamedColumns, "Année", each Date.Year([Date]), Int64.Type),
InsertQuarter = Table.AddColumn(InsertYear, "Trimestre", each Date.QuarterOfYear([Date]), Int64.Type),
InsertMonth = Table.AddColumn(InsertQuarter, "MoisNombre", each Date.Month([Date]), Int64.Type),
InsertDay = Table.AddColumn(InsertMonth, "Jour", each Date.Day([Date]), Int64.Type),
InsertDayInt = Table.AddColumn(InsertDay, "DateNombre", each [Année] * 10000 + [MoisNombre] * 100 + [Jour]),
InsertMonthName = Table.AddColumn(InsertDayInt, "MoisNom", each Date.ToText([Date], "MMMM", Culture), type text),
InsertCalendarMonth = Table.AddColumn(InsertMonthName, "MoisCalendrier", each (try(Text.Range([MoisNom],0,3)) otherwise [MoisNom]) & " " & Number.ToText([Trimestre]) & " " & Number.ToText([Année])),
InsertCalendarQtr = Table.AddColumn(InsertCalendarMonth, "TrimestreCalendrier", each "T" & Number.ToText([Trimestre]) & " " & Number.ToText([Année])),
InsertDayWeek = Table.AddColumn(InsertCalendarQtr, "JourSemaine", each Date.DayOfWeek([Date])),
InsertDayName = Table.AddColumn(InsertDayWeek, "NomJourSemaine", each Date.ToText([Date], "dddd", Culture), type text),
InsertWeekEnding = Table.AddColumn(InsertDayName, "FinSemaine", each Date.EndOfWeek([Date]), type date)
in
CreateDateTable
```

```
let
CreateDateTable = (StartDate as date, EndDate as date, optional Culture as nullable text) as table =>
let
DayCount = Duration.Days(Duration.From(EndDate - StartDate)),
Source = List.Dates(StartDate, DayCount, #duration(1,0,0,0)),
TableFromList = Table.FromList(Source, Splitter.SplitByNothing()),
ChangedType = Table.TransformColumnTypes(TableFromList,{{"Column1", type date}}),
RenamedColumns = Table.RenameColumns(ChangedType,{{"Column1", "Date"}}),
InsertYear = Table.AddColumn(RenamedColumns, "Année", each Date.Year([Date]), Int64.Type),
InsertQuarter = Table.AddColumn(InsertYear, "Trimestre", each Date.QuarterOfYear([Date]), Int64.Type),
InsertMonth = Table.AddColumn(InsertQuarter, "MoisNombre", each Date.Month([Date]), Int64.Type),
InsertDay = Table.AddColumn(InsertMonth, "Jour", each Date.Day([Date]), Int64.Type),
InsertDayInt = Table.AddColumn(InsertDay, "DateNombre", each [Année] * 10000 + [MoisNombre] * 100 + [Jour]),
InsertMonthName = Table.AddColumn(InsertDayInt, "MoisNom", each Date.ToText([Date], "MMMM", Culture), type text),
InsertCalendarMonth = Table.AddColumn(InsertMonthName, "MoisCalendrier", each (try(Text.Range([MoisNom],0,3)) otherwise [MoisNom]) & " " & Number.ToText([Trimestre]) & " " & Number.ToText([Année])),
InsertCalendarQtr = Table.AddColumn(InsertCalendarMonth, "TrimestreCalendrier", each "T" & Number.ToText([Trimestre]) & " " & Number.ToText([Année])),
InsertDayWeek = Table.AddColumn(InsertCalendarQtr, "JourSemaine", each Date.DayOfWeek([Date])),
InsertDayName = Table.AddColumn(InsertDayWeek, "NomJourSemaine", each Date.ToText([Date], "dddd", Culture), type text),
InsertWeekEnding = Table.AddColumn(InsertDayName, "FinSemaine", each Date.EndOfWeek([Date]), type date)
in
InsertWeekEnding
in
CreateDateTable
```

Appeler la fonction avec les paramètres suivants :

Entrer des paramètres

StartDate  
01/01/2011

EndDate  
31/12/2020

Culture (facultatif)  
Exemple : abc

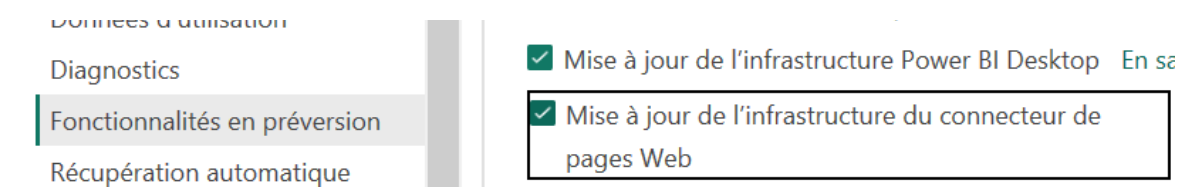
Appeler Effacer

La table calendrier est générée, renommez la requête en calendrier

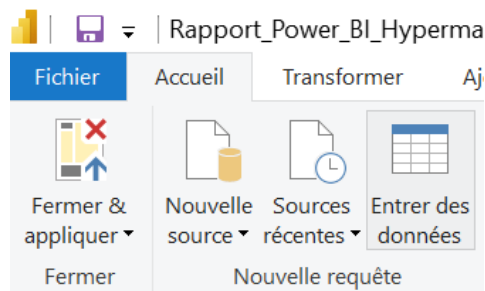
### Récupération des jours fériés depuis une page Web

Nous allons les télécharger depuis un site Web, pour cela vous devez aller dans les options pour configurer l'option suivante :

## M2 : BI/DATA ANALYSE – ETUDE DE CAS



Créer une table avec les années présentes dans le calendrier (de 2010 à 2020) via l'option « entrer des données » (dans Power Query) :



Renseignez les années de 2010 à 2020 :

A <sup>B</sup> C Année
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020

Modifier le type de la colonne « Année » en texte.

Sélectionner l'option « Obtenir des données à partir d'une source Web »

Rentrer l'URL suivant :

<https://kalendrier.ouest-france.fr/jours-feries/2013.html>

Choisissez la table suggérée :



## M2 : BI/DATA ANALYSE – ETUDE DE CAS

Options d'affichage ▾

- Tables HTML [1]
  - ☐ Voici la liste chronologique de tous les jours fériés 2010...
- Tables suggérées [1]
  - ☒ Table 1
- Texte [2]
  - Table 1
  - Code HTML
  - ☐ Texte affiché

Table 1

Column1	Column2	Column3
Jour de l'an en France	Jour de l'an	Vendredi 1
Dimanche de Pâques en France	Dimanche de Pâques	Dimanche
Lundi de Pâques en France	Lundi de Pâques	Lundi 5 avr
Fête du travail en France	Fête du travail	Samedi 1e
Victoire 1945 en France	Victoire 1945	Samedi 8 m
Jeudi de l'Ascension en France	Jeudi de l'Ascension	Jeudi 13 m
Dimanche de Pentecôte en France	Dimanche de Pentecôte	Dimanche

Chargez les données puis renommez les colonnes et supprimez la dernière colonne (France) :

AB C LIBELLE JOUR	AB C LIBELLE 2 JOUR	AB C DATE TEXTE
Jour de l'an en France	Jour de l'an	Mardi 1er janvier 2013
Dimanche de Pâques en France	Dimanche de Pâques	Dimanche 31 mars 2013
Lundi de Pâques en France	Lundi de Pâques	Lundi 1er avril 2013
Fête du travail en France	Fête du travail	Mercredi 1er mai 2013
Victoire 1945 en France	Victoire 1945	Mercredi 8 mai 2013
Jeudi de l'Ascension en France	Jeudi de l'Ascension	Jeudi 9 mai 2013
Dimanche de Pentecôte en France	Dimanche de Pentecôte	Dimanche 19 mai 2013
Lundi de Pentecôte en France	Lundi de Pentecôte	Lundi 20 mai 2013
Fête nationale en France	Fête nationale	Dimanche 14 juillet 2013
Assomption en France	Assomption	Jeudi 15 août 2013
Toussaint en France	Toussaint	Vendredi 1er novembre 2013
Armistice 1918 en France	Armistice 1918	Lundi 11 novembre 2013
Noël en France	Noël	Mercredi 25 décembre 2013

Toujours dans Power Query Créer un nouveau paramètre « paramètre année » :

### Gérer les paramètres

Nouveau

AB C Année

Nom

Année

Description

☒ Obligatoire

Type

Texte

Valeurs suggérées

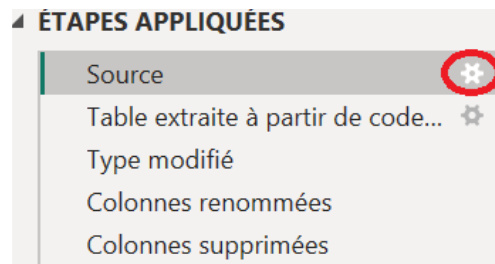
Tout

Valeur actuelle

2013

- Allez au niveau de la requête Table1 qui liste les jours fériés pour l'année 2013.
- Accédez à la première étape source et cliquez sur l'engrenage :

## M2 : BI/DATA ANALYSE – ETUDE DE CAS



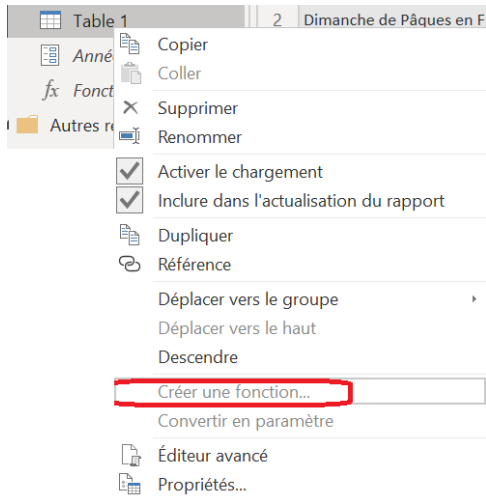
- Nous allons décomposer l'adresse html avec une partie fixe et une partie variable (le paramètre année) :

Vérifier que la requête est toujours bien alimentée avec la décomposition de l'adresse web :

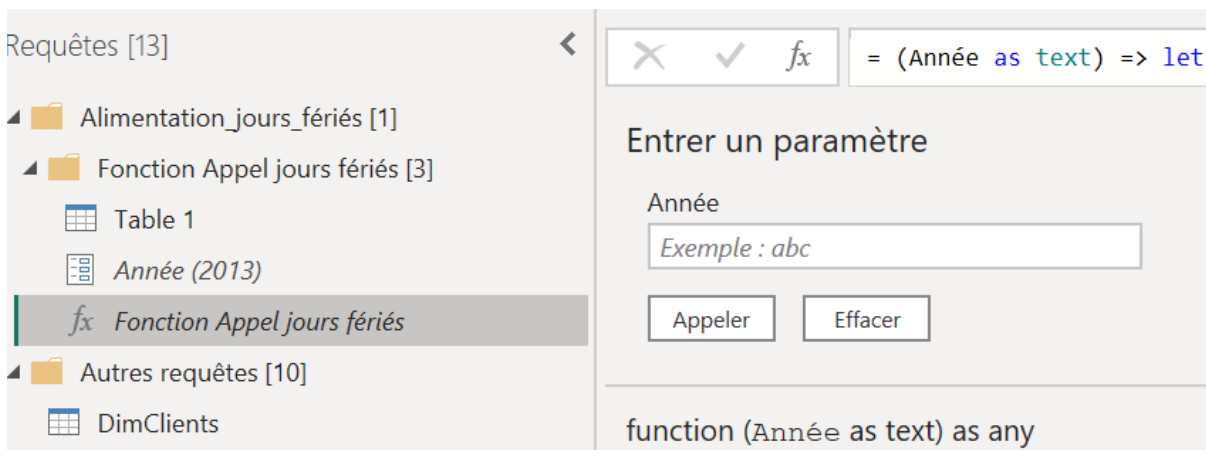
AB_C LIBELLE JOUR	AB_C LIBELLE 2 JOUR	AB_C DATE TEXTE
Jour de l'an en France	Jour de l'an	Mardi 1er janvier 2013
Dimanche de Pâques en France	Dimanche de Pâques	Dimanche 31 mars 2013
Lundi de Pâques en France	Lundi de Pâques	Lundi 1er avril 2013
Fête du travail en France	Fête du travail	Mercredi 1er mai 2013
Victoire 1945 en France	Victoire 1945	Mercredi 8 mai 2013
Jeudi de l'Ascension en France	Jeudi de l'Ascension	Jeudi 9 mai 2013
Dimanche de Pentecôte en France	Dimanche de Pentecôte	Dimanche 19 mai 2013
Lundi de Pentecôte en France	Lundi de Pentecôte	Lundi 20 mai 2013
Fête nationale en France	Fête nationale	Dimanche 14 juillet 2013
Assomption en France	Assomption	Jeudi 15 août 2013
Toussaint en France	Toussaint	Vendredi 1er novembre 2013
Armistice 1918 en France	Armistice 1918	Lundi 11 novembre 2013
Noël en France	Noël	Mercredi 25 décembre 2013

Convertissez la requête en fonction :

## M2 : BI/DATA ANALYSE – ETUDE DE CAS

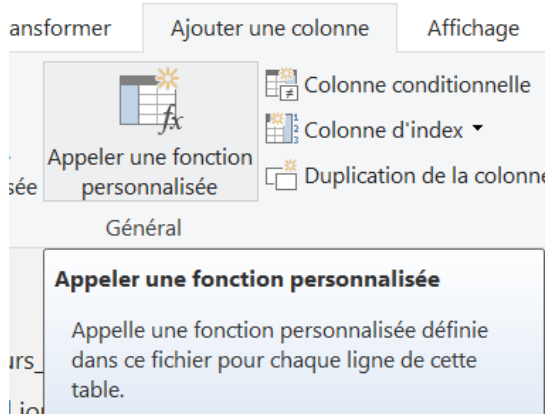


Que vous nommerez fonction appel jours fériés :



Depuis la colonne Année de la requête « année » Appelez cette fonction de la manière suivante :

wer\_BI\_Hypermarché - Editeur Power Query



Choisissez la fonction « appel jours fériés »

## M2 : BI/DATA ANALYSE – ETUDE DE CAS

Une nouvelle colonne apparaît avec la possibilité de ramener toutes les colonnes de la page web.  
Cliquez sur les 2 flèches divergentes et choisissez toutes les colonnes à récupérer

ABC 123	Année	ABC 123	Fonction Appel jours fériés	
	2010		Table	
	2011		Table	
	2012		Table	
	2013		Table	
	2014		Table	
	2015		Table	
	2016		Table	
	2017		Table	
	2018		Table	
	2019		Table	
	2020		Table	

☒ Développer ☐ Agréger

☒ (Sélectionner toutes les colonnes)  
☒ LIBELLE JOUR  
☒ LIBELLE 2 JOUR  
☒ DATE TEXTE

☒ Utiliser le nom de la colonne d'origine comme préfixe

OK Annuler

La requête affiche tous les jours fériés de toutes les années.

Créez ensuite une nouvelle colonne permettant d'obtenir une date à partir de la colonne

« Fonction Appel jours fériés. DATE TEXTE » :

**Attention** : il faut remplacer les « 1<sup>er</sup> » par « 1 » sinon la conversion est en erreur pour les dates au 1/MM/AAAA

ABC 123	Fonction Appel jours fériés.DATE TEXTE	Date jour férié
	Vendredi 1er janvier 2010	01/01/2010
	Dimanche 4 avril 2010	04/04/2010
	Lundi 5 avril 2010	05/04/2010
	Samedi 1er mai 2010	01/05/2010
	Samedi 8 mai 2010	08/05/2010
	Jeudi 13 mai 2010	13/05/2010
	Dimanche 23 mai 2010	23/05/2010
	Lundi 24 mai 2010	24/05/2010
	Mercredi 14 juillet 2010	14/07/2010
	Dimanche 15 août 2010	15/08/2010
	Lundi 1er novembre 2010	01/11/2010

## M2 : BI/DATA ANALYSE – ETUDE DE CAS

Renommez la requête « Année » en « Table jours fériés »

Nous allons maintenant fusionner cette requête avec la table calendrier de manière à rapatrier la colonne jour férié. Dans Power Query, sélectionnez le calendrier puis allez sur Combiner/Fusionner des requêtes.

Choisissez « Table des jours fériés »

**Fusionner**

Sélectionnez une table et les colonnes correspondantes pour créer une table fusionnée.

Calendrier

Date	Année	Trimestre	MoisNombre	Jour	DateNombre	MoisNom	MoisCalendrier	Actualiser
01/01/2011	2011	1	1	1	20110101	janvier	jan 2011	T1 2011
02/01/2011	2011	1	1	2	20110102	janvier	jan 2011	T1 2011
03/01/2011	2011	1	1	3	20110103	janvier	jan 2011	T1 2011
04/01/2011	2011	1	1	4	20110104	janvier	jan 2011	T1 2011

Table jours fériés

Année	Fonction Appel jours fériés.LIBELLE JOUR	Fonction Appel jours fériés.LIBELLE 2 JOUR	Fonction Appel jours f
2010	Jour de l'an en France	Jour de l'an	Vendredi 1er janvier 2010
2010	Dimanche de Pâques en France	Dimanche de Pâques	Dimanche 4 avril 2010
2010	Lundi de Pâques en France	Lundi de Pâques	Lundi 5 avril 2010
2010	Fête du travail en France	Fête du travail	Samedi 1er mai 2010

Type de jointure

Externe gauche (toutes à partir de la première, corres...) ▼

☐ Utiliser la correspondance approximative pour effectuer la fusion

Options de correspondance approximative

OK Annuler

Récupérez la colonne « Libellé jour » et renommez la « Nom jour férié » :

**Développer Table jours fériés**

Sélectionnez les colonnes à développer.

(Sélectionner toutes les colonnes)

- ☒ Année
- ☒ Fonction Appel jours fériés.LIBELLE JOUR
- ☐ Fonction Appel jours fériés.LIBELLE 2 JOUR
- ☐ Fonction Appel jours fériés.DATE TEXTE
- ☐ Date jour férié

Préfixe de nom de colonne par défaut (facultatif)

Table jours fériés

OK Annuler

Vous obtenez ainsi une nouvelle colonne permettant d'identifier si la date correspond à un jour férié :

## M2 : BI/DATA ANALYSE – ETUDE DE CAS

3	jeudi	16/01/2011	null
4	vendredi	16/01/2011	null
5	samedi	16/01/2011	null
6	dimanche	24/04/2011	Dimanche de Pâques en France
6	dimanche	16/01/2011	null
0	lundi	01/05/2011	Lundi de Pâques en France
0	lundi	23/01/2011	null

Toujours au niveau de la requête Calendrier ajouter une nouvelle colonne permettant d'identifier les jours ouvrés :

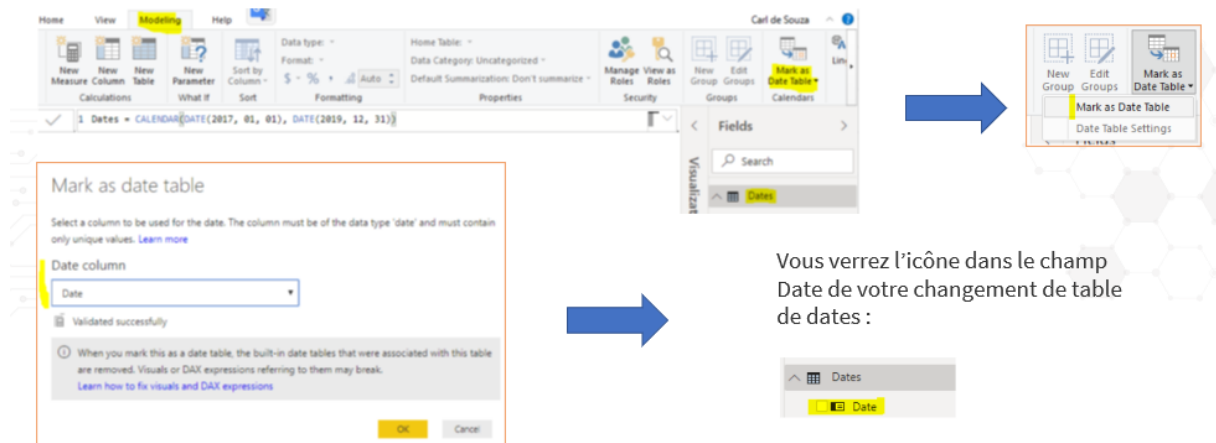
*If [JourSemaine] < 5 AND [Nom jour férié] is null then 1 else 0*

Appliquez-lui un type Nombre entier

Désactiver le chargement des requêtes Table 1 et Table jour fériés dont les colonnes qui nous intéressent ont été rapatriées dans la table calendrier .

Fermez en appliquant les modifications sur le modèle

Une dernière étape de ce processus consiste à marquer la table en tant que table de dates, afin que Power BI sache utiliser la table à cette fin. Sélectionnez votre table de dates dans Power BI Desktop et accédez à l'onglet Modélisation, sélectionnez « Marquer comme table de dates » :



Sélectionnez la colonne de date dans la table de dates et cliquez sur OK

Vous avez terminé les phases d'acquisition et de préparation des données.

### Modèle final HYPERMARCHÉ

La modélisation a été réalisée en amont, mais il nous reste à relier la table calendrier au modèle. La table de FAIT contient 2 dates :

- Date de commande
- Date d'expédition

Si on relie le calendrier à la date de commande, les indicateurs seront filtrés sur la date de commande. Si on relie le calendrier avec la date d'expédition, les indicateurs seront filtrés sur la date d'expédition. Il faut trouver le moyen d'interagir avec le modèle pour obtenir les 2 : Les indicateurs filtrés sur la date de commande ET sur la date d'expédition (le montant commandé n'est pas forcément égal au montant expédié si l'entreprise facture après l'expédition).

Le schéma ci-dessous permettra de mieux comprendre :

### Time intelligence

### Fonction USERELATIONSHIP

Une seule relation possible entre 2 tables .  
L'autre relation existante (illustrée par des pointillés) est inactive.

**Objectif :** Utiliser la table 'Calendar' afin de suivre aussi bien le CA généré (Order Date) vs le CA encaissé (Payment Date).

**Solution : UseRelationship.** cette fonction « utilise » une relation présente dans le modèle de données. Plus précisément, elle l'active localement (e.g au sein de la mesure). Voici sa Syntaxe:  
UseRelationship (<columnName1>;<columnName2>)

<columnName1> est la colonne du côté 'Many' de la relation,  
<columnName2> est la colonne du côté 'One' de la relation.

Total Sales = Sum(Sales[Amount])  
Total Sales Collected = Calculate ( [Total Sales] , Userelationship(Sales[Payment Date]; Calendar[Dates] ) )

C'est grâce à la fonction USERELATIONSHIP que l'on pourra sélectionner la bonne jointure à activer pour utiliser soit la date de commande soit la date d'expédition pour calculer les indicateurs sur l'une de ces dates .

### Analyse et Création des mesures

Liste des mesures à Créer en langage DAX

- **Total Vente** : Sum(montant des ventes)
- **Total Quantité** : sum (quantité)
- **Total Profit** : sum (profit)
- **Total Vente (A-1)** : Correspond au total vente pour la même période l'année dernière  
*CALCULATE (TOTAL\_VENTE, SAMEPERIODLASTYEAR (Calendrier.date))*
- **Total Vente Cumulée Annuel** : Correspond au cumul des Ventes sur une année (fonction DATESYTD)
- **Total Profit Cumulé Annuel** : Correspond au cumul des Profits sur une année
- **Total Profit Net Cumulé annuel** : Correspond au cumul des Profits sur une année moins le montant remisé
- **Variation Annuelle** : Correspond à la variation entre le total vente de l'année A et le total vente de l'année A-1. Utilisez la fonction DIVIDE (plus Safe) S'exprime en %
- **Nombre de commandes** : Correspond au nombre unique de ID\_COMMANDE de la table Achat (DISTINCTCOUNT)
- **Nombre de factures** : Correspond au nombre de ID\_COMMANDE de la table FACTURE
- **Nombre de factures non honorées** : correspond au Nombre distinct de commandes ou la colonne [Identification commande non facturée] =1
- **Total Expédié** par rapport à la date d'expédition
- **Délai jours ouvrés** : correspond à la somme du nombre de jours ouvrés du calendrier entre la première date de commande et la dernière date d'expédition
- **Classement produit (vente)** = N° rang correspondant au classement des produits par rapport au total vente (le classement devra se faire sur le nom du produit)  
*RANKX(ALL(DimProduit[LB\_PRODUIT]), [total vente], , DESC)*

Regrouper ces mesures dans une table unique appelée « Table Mesure »

Au besoin aidez vous du glossaire DAX disponible sur le site de MICROSOFT :

<https://learn.microsoft.com/fr-fr/dax/dax-glossary>

### Datavisualisation

#### Page Carte

Réaliser une première visualisation permettant d'identifier les villes sur un fond de carte .



## M2 : BI/DATA ANALYSE – ETUDE DE CAS

Vous constatez que certaines villes apparaissent sur le continent américain ce qui est une erreur car il s'agit d'homonymes (exemple Derby).

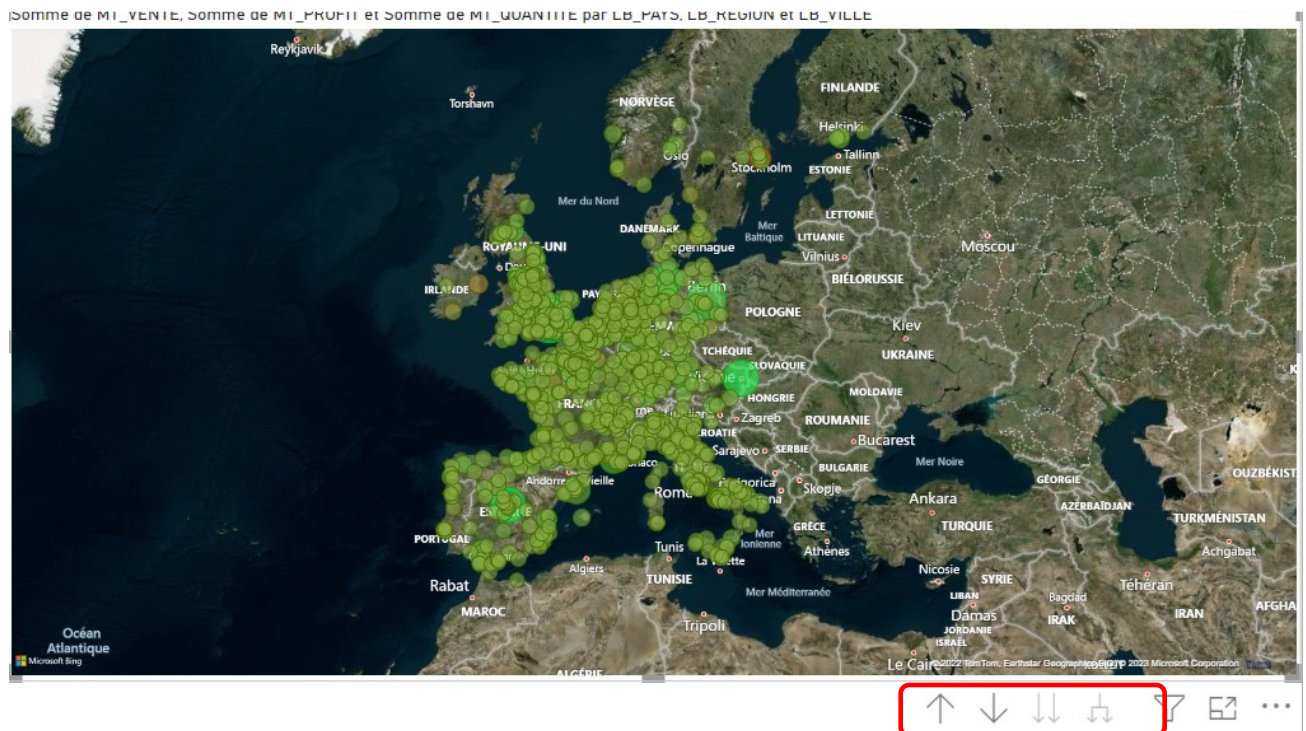
La solution pour corriger cela est de créer une hiérarchie Pays/Région/Ville ou l'emplacement de la ville sera canalisé par la hiérarchie.

Créer la hiérarchie géographique Pays/Région/Ville et remplacez dans emplacement la ville par la nouvelle hiérarchie .

➔ Il n'y a plus de villes ailleurs que sur le continent européen.

Passez en style aérien et reproduisez la carte suivante :

Notez tous les détails de la carte (La taille de la bulle varie en fonction du montant et la couleur de la bulle varie en fonction du profit) :



- Remarquez les flèches qui apparaissent grâce à la hiérarchie.
- Jouez avec les flèches et essayer de définir le rôle de chacune d'entre elles
- Ajouter dans l'infobulle le profit et la quantité

### Page Synthèse

Réaliser la page ci-dessous

## M2 : BI/DATA ANALYSE – ETUDE DE CAS



### Détails



Filtre Sur l'année du calendrier présentée au format Vignette



Jauge permettant de comparer le nombre de commandes VS le nombre de factures

**3** **Loupe** permet de se débrancher sur les commandes qui n'ont pas été honorées à l'aide d'un signet :

## M2 : BI/DATA ANALYSE – ETUDE DE CAS

Nb de commandes non facturées



57

Cliquer sur la loupe pour accéder  
aux commandes non facturées

### Liste des commandes non facturées



Nb de commandes non facturées	NUM_COMMANDE	LB_NOM_CLIENT	LB_NOM_SEGMENT	LB_PAYS	LB_REGION	LB_VILLE	total vente
1	ES-2012-1158066	Almy Garnier	Grand public	France	Provence-Alpes-Côte d'Azur	Marseille	138,75 €
1	ES-2012-1158066	Barthélémy Rochon	Entreprise	France	Alsace-Champagne-Ardenne-Lorraine	Mulhouse	211,09 €
1	ES-2012-1919871	Martine Plourde	Entreprise Individuelle	Espagne	Galice	Vigo	122,70 €
1	ES-2012-1929989	Beltane Lereau	Grand public	Royaume-Uni	Angleterre	Bristol	3 019,14 €
1	ES-2012-1929989	Xarles Devost	Grand public	Royaume-Uni	Angleterre	Londres	334,61 €
1	ES-2012-2224678	Camille Cloutier	Grand public	Italie	Vénétie	Rovigo	3 014,43 €
1	ES-2012-2874029	Émilie Chabot	Grand public	France	Île-de-France	Cholsy-le-Roi	155,18 €
1	ES-2012-4527901	Gérard Parrot	Grand public	Allemagne	Rhénanie-du-Nord-Westphalie	Bochum	573,19 €
1	ES-2012-4997791	Corinne Lépicier	Grand public	Royaume-Uni	Angleterre	Eastbourne	203,25 €
1	ES-2013-1013413	Félicienne Garreau	Grand public	Royaume-Uni	Angleterre	Londres	74,60 €

4

Loupe permettant d'accéder au délai moyen de livraison . La valeur affichée correspond au délai d'expédition moyen

Délai moyen de livraison (en  
jours ouvrés)



0,22



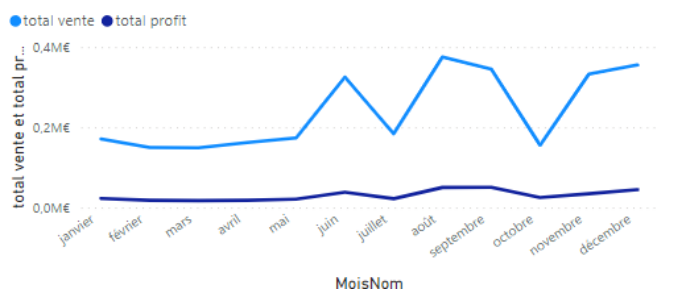
Cliquer sur la loupe pour accéder  
aux commandes dont le délai de  
livraison excède 5 jours

NUM_COMMANDE	DATE_COMMANDE	DATE_EXPEDITION	Délai jours ouvrés
IT-2014-5992832	jeudi 14 août 2014	dimanche 17 août 2014	1
IT-2014-5989338	samedi 12 avril 2014	vendredi 18 avril 2014	5
IT-2014-5984498	mercredi 1 octobre 2014	lundi 6 octobre 2014	4
IT-2014-5975833	samedi 19 juillet 2014	mercredi 23 juillet 2014	3
IT-2014-5966070	mardi 28 janvier 2014	samedi 1 février 2014	4
IT-2014-5965314	mardi 5 août 2014	dimanche 10 août 2014	4
IT-2014-5950843	vendredi 27 juin 2014	mercredi 2 juillet 2014	34
IT-2014-5936992	mardi 15 avril 2014	lundi 21 avril 2014	4
IT-2014-5935536	vendredi 7 février 2014	dimanche 9 février 2014	1
IT-2014-5928156	lundi 17 février 2014	mardi 18 février 2014	2
IT-2014-5919161	mercredi 8 janvier 2014	dimanche 12 janvier 2014	3

5

Graphique en courbes permettant de visualiser l'évolution des ventes VS l'évolution des profits sur l'exercice sélectionné

total vente et total profit par MoisNom



6

« Score Card » permettant d'afficher le total des ventes et le total des profits sous forme de KPI

## M2 : BI/DATA ANALYSE – ETUDE DE CAS

CA :

2,88M€

Profits :

364K €

7

Signets permettant d'accéder à différentes pages « détail »

Analyse indicateurs financiers



Analyse Points de vente



Analyse remise



Analyse produits



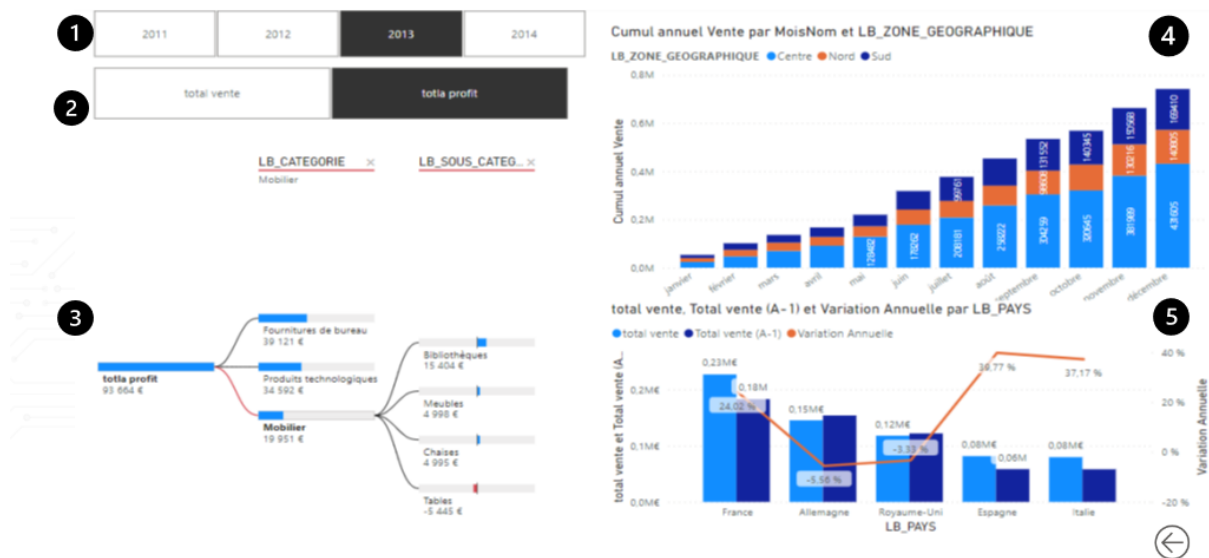
Analyse clients



- Page « Analyse indicateurs financier »
- Page « Analyse point de vente »
- Page « Analyse remise »
- Page « Analyse Produits »
- Page « Analyse Clients »

*Page Analyse indicateurs financiers*

## M2 : BI/DATA ANALYSE – ETUDE DE CAS



1

: Segment basé sur l'année présentée en mode « diapositive »

2

: Paramètre Champ permettant de switcher sur l'affichage des ventes ou des profits

3

: Arborescence de décomposition permettant de descendre dans le détail du Profit par axes d'analyse

4

: Histogramme cumulé permettant de suivre le total des ventes mensuel cumulé par Zones géographique

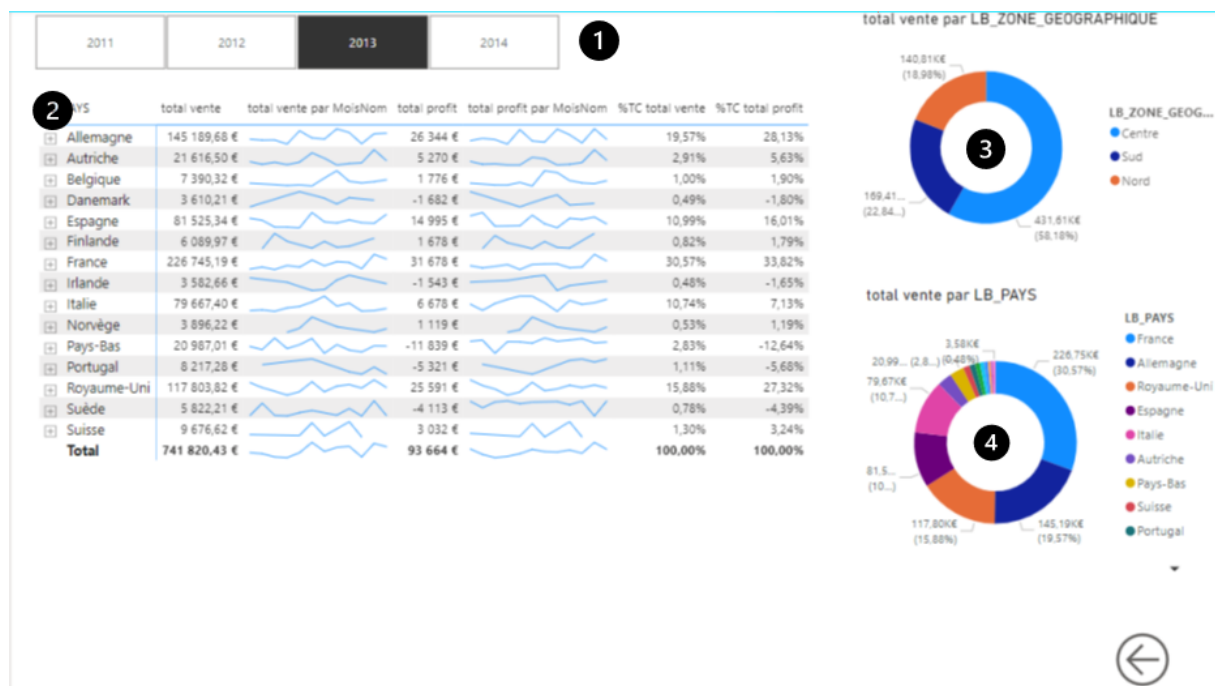
5

: Histogramme vertical représentant les 5 meilleurs pays en termes de Ventes avec :

- Le total vente de l'année sélectionnée
- Le total vente (A-1) par rapport à l'année sélectionnée
- La courbe d'évolution de la variation annuelle

*Page Analyse Points de vente*

## M2 : BI/DATA ANALYSE – ETUDE DE CAS



1

: Segment basé sur l'année présentée en mode « diapositive »

2

: Matrice contenant la hiérarchie géographique (Pays/Région/Ville), Le total des Ventes, un sparkline permettant de suivre l'évolution des ventes depuis le début de l'année sélectionnée, le total des profits, un sparkline permettant de suivre l'évolution des profits depuis le début de l'année sélectionnée

3

: Graphique en anneau permettant de voir la répartition des ventes par zones géographiques

4

: Graphique en anneau permettant d'analyser la répartition des ventes par Pays

## M2 : BI/DATA ANALYSE – ETUDE DE CAS

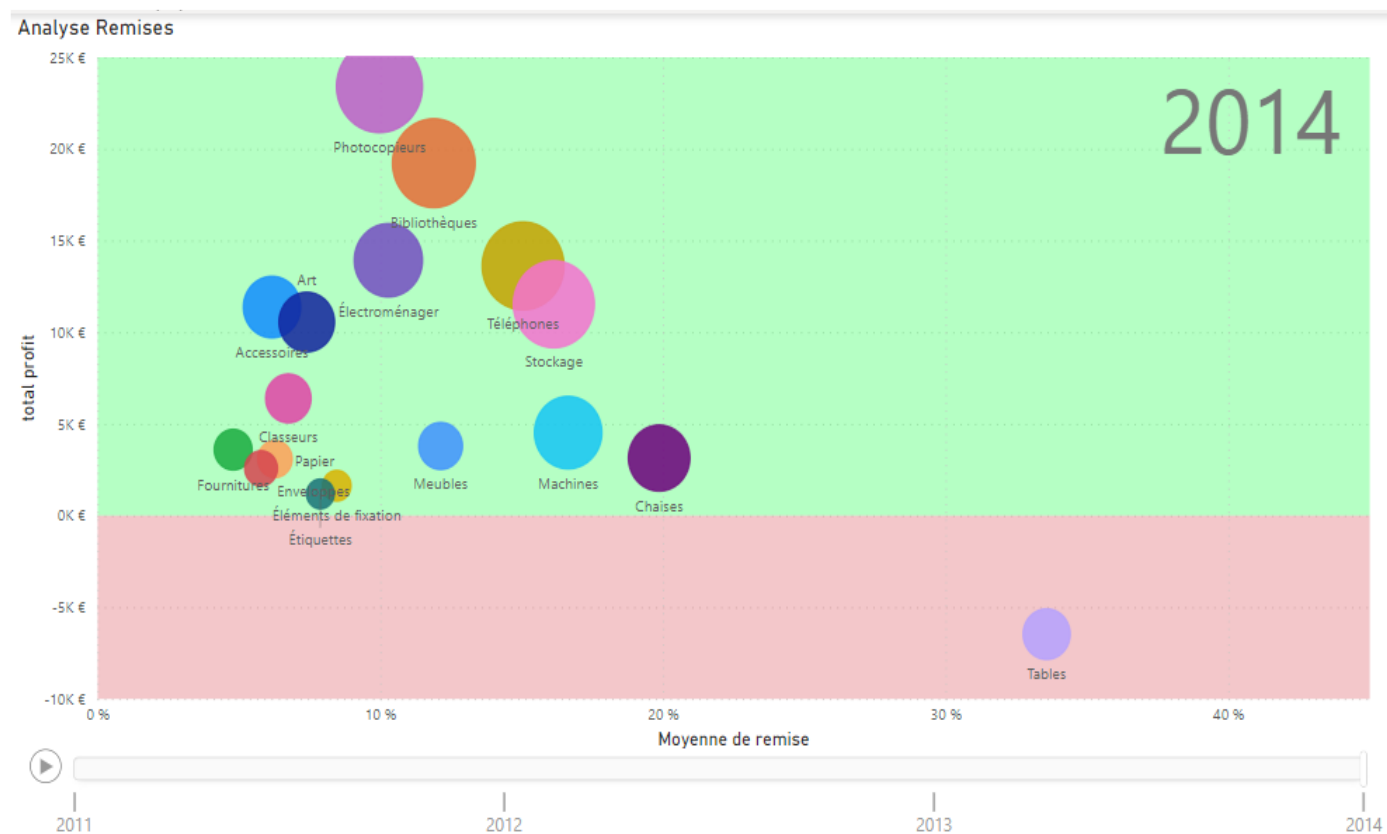
### Page Analyse Remise

Nuage de point permettant de faire la corrélation entre les Remises appliquées et les profits générés par sous-catégorie .

La taille de la bulle représentant la sous-catégorie varie en fonction des ventes.

La position dans la bulle dans le nuage de points nous donne une indication sur la relation existante entre le profit et la remise . Les sous catégories les plus intéressantes se situent dans la partie supérieure gauche du graphique (peu de remise et beaucoup de profit) alors que celles qui sont les moins intéressantes sont situées dans la partie inférieure droite du graphique

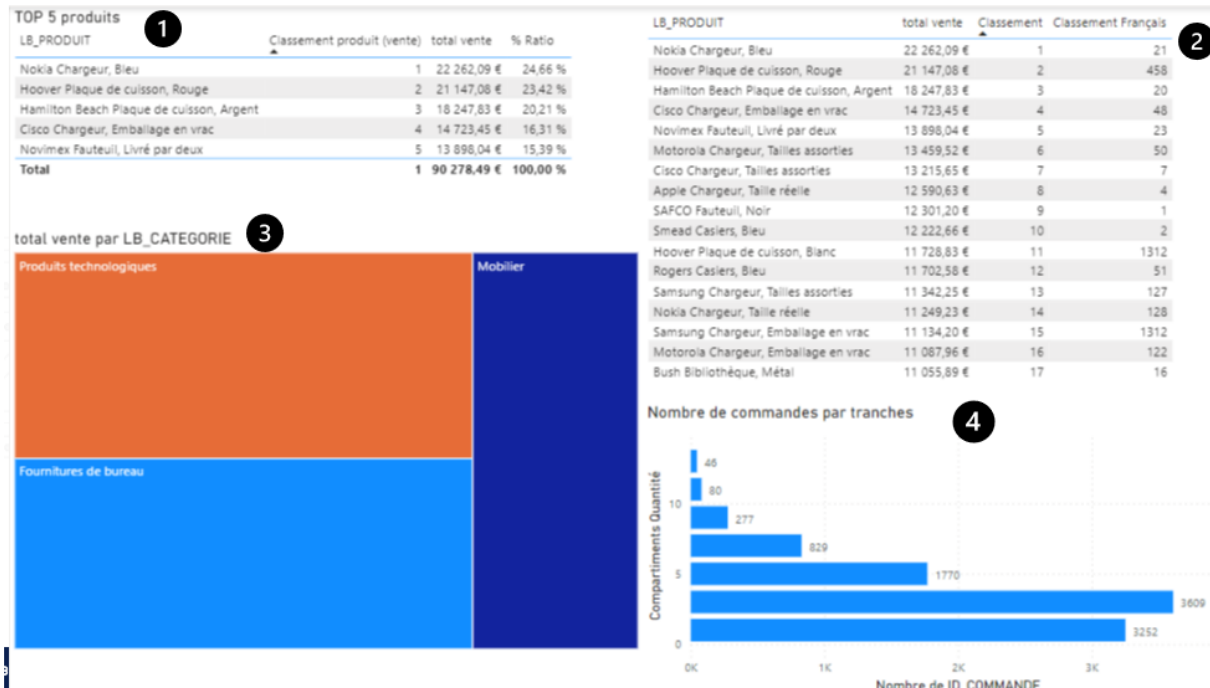
Notez le « player » situé dans la partie inférieure du graphique permettant de lancer une animation par années





# M2 : BI/DATA ANALYSE – ETUDE DE CAS

## Page Analyse Produit



1

TOP 5 Produit : Affichage des 5 meilleurs produits en termes de vente avec :

- Le libellé du produit
- Son classement toutes ventes confondues
- La contribution des ventes du produit par rapport au total « TOP 5 »

2

Tableau qui affiche l'ensemble des produits et leur classement toutes ventes confondues ainsi que le classement en **France**

3

: Tree map permettant de comparer les composantes de l'axe hiérarchique produit : Catégorie/Sous Catégorie/Produit . Ce graphique doit permettre de faire du « Drill down »

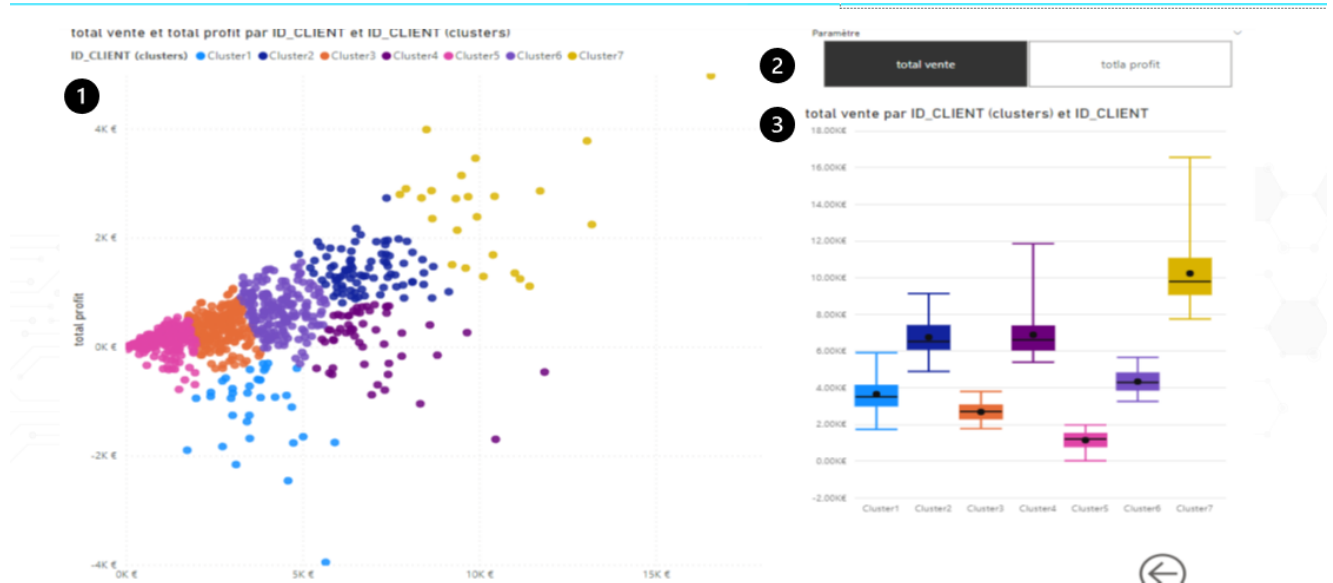
4

: Répartition du nombre de commandes par tranches quantité . Les quantités ont été réparti sur 7 compartiments différents avec un écart de 2



## M2 : BI/DATA ANALYSE – ETUDE DE CAS

### Page Analyse Clients



1

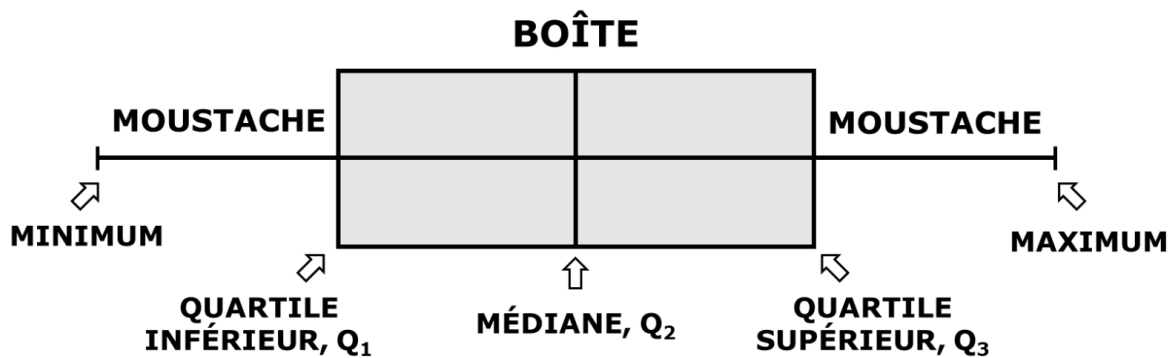
Nuage de points avec intégration de cluster . Chaque point correspond à un client (utiliser la clé métier) . En abscisse on retrouve le total des ventes et en ordonnée le total des profits . Les clients sont ensuite répartis dans 7 clusters différents en fonction de leur performance (on prendra une attention particulière pour les clients représentés par des points jaunes qui génèrent des profits élevés)

2

Paramètre champ permettant de « switcher » sur l’affichage des profits ou des ventes .

3

Graphique « boîte à moustache » appelé « Box plot » sur le market place Microsoft d'où vous pourriez le télécharger. Dans la zone « Category » on retrouve les 7 clusters générés par le nuage de points et dans la zone sampling la clé métier du client (CD\_CLIENT)



- Les côtés gauche et droit de la boîte sont les quartiles inférieur et supérieur. La boîte couvre donc l'intervalle interquartile, là où se situent 50 % des données.
- La ligne verticale qui sépare la boîte en deux représente la médiane. Parfois, la moyenne est également indiquée par un point ou une croix sur la boîte à moustaches.
- Les moustaches sont les deux lignes horizontales à l'extérieur de la boîte qui s'étendent du minimum jusqu'au quartile inférieur (le début de la boîte) et du quartile supérieur (la fin de la boîte) jusqu'au maximum.
- Le diagramme est habituellement accompagné d'un axe qui indique les valeurs (non montré à la figure 4.5.2.1).
- La boîte à moustache peut être présentée horizontalement, comme à la figure 4.5.2.1, ou verticalement.