

TUTORIEL MS-QUERY V1.0

Auteur : Mickaël ROMAIN (mromain)

L'objectif de ce tutoriel est de présenter l'outil **Microsoft Query** qui permet d'extraire des données – en lecture donc – depuis diverses sources.

Dans le cadre de ce tutoriel, nous allons donc utiliser MS-Query pour effectuer des requêtes depuis Excel sur un autre fichier Excel. Le tutoriel se concentre sur ce type de connexion, bien que MS-Query puisse se connecter à un grand nombre de source de données, telles que des fichiers texte, bien sûr des bases de données, mais également des systèmes particuliers si un driver ODBC est fourni. Il est simplement plus axé sur l'utilisation de cet outil de requêtes.

La première partie est sous forme de didacticiel et présente un petit exercice permettant de prendre en main MS-Query. La seconde partie est plus simple et présente quelques limitations, bonnes pratiques et compléments à creuser.

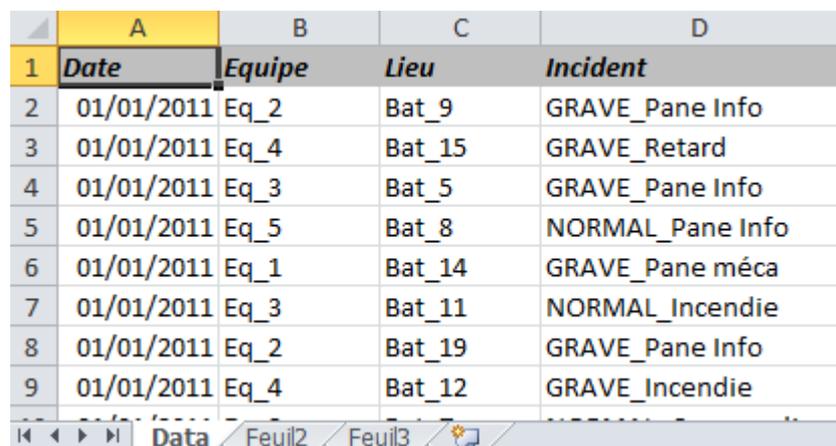
SOMMAIRE

1.	PRESENTATION DU CONTEXTE	2
1.1.	Fichier Excel source de données	2
1.2.	Objectif	2
1.3.	Base du fichier de consultation	2
2.	EXTRAIRE LA LISTE DES EQUIPES (EXTRACTION SIMPLE)	3
3.	EXTRAIRE LA LISTE DES LIEUX (EXTRACTION SIMPLE)	6
4.	EXTRAIRE LA LISTE DES EVENEMENTS (EXTRACTION + TRI + FILTRE)	8
5.	CONCLUSION / OUVERTURE	11
5.1.	Pour creuser plus loin.....	11
5.2.	Limitations	12
5.3.	Utilisation en réseau / bonne pratique	13

1. PRESENTATION DU CONTEXTE

1.1. Fichier Excel source de données

Chemin : « C:\ExempleBDD.xls »



	A	B	C	D
1	Date	Equipe	Lieu	Incident
2	01/01/2011	Eq_2	Bat_9	GRAVE_Pane Info
3	01/01/2011	Eq_4	Bat_15	GRAVE_Retard
4	01/01/2011	Eq_3	Bat_5	GRAVE_Pane Info
5	01/01/2011	Eq_5	Bat_8	NORMAL_Pane Info
6	01/01/2011	Eq_1	Bat_14	GRAVE_Pane méca
7	01/01/2011	Eq_3	Bat_11	NORMAL_Incendie
8	01/01/2011	Eq_2	Bat_19	GRAVE_Pane Info
9	01/01/2011	Eq_4	Bat_12	GRAVE_Incendie

Le fichier source est composé de trois feuilles. Seule la feuille *Data* contient des données, sur quatre colonnes.

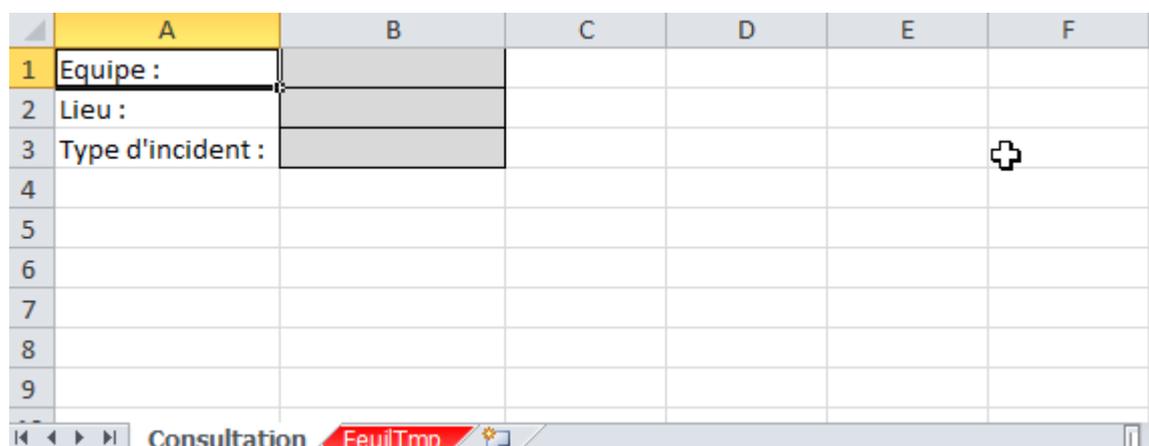
1.2. Objectif

L'objectif de cet exercice est de créer un fichier Excel de consultation qui permettra d'extraire les données de la source (ici du fichier Exemple BDD.xls) en fonction de l'équipe, du lieu, et du type d'incident (GRAVE, NORMAL, ou Tous).

1.3. Base du fichier de consultation

Il sera lui composé de deux feuilles :

- Une feuille *FeuilTmp* qui sera cachée à l'utilisateur.
- Une feuille *Consultation* qui permettra à l'utilisateur d'afficher les incidents en fonction de l'équipe et du lieu.

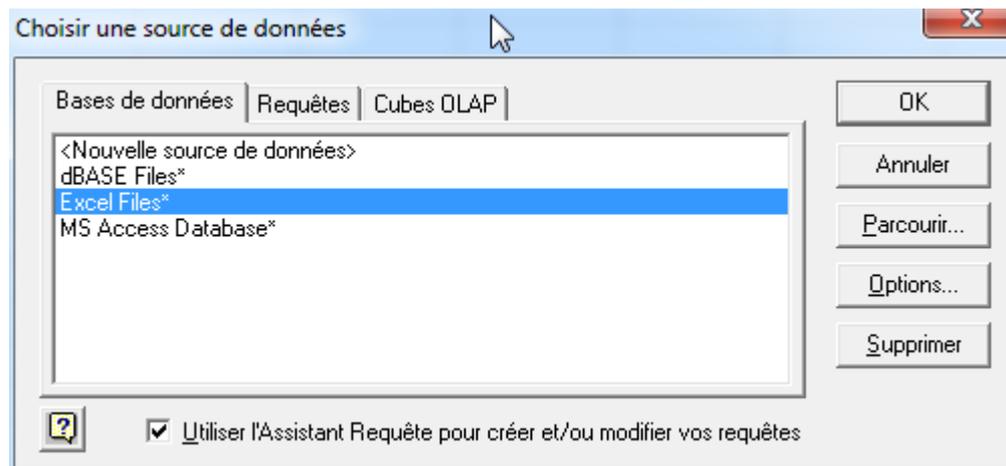


	A	B	C	D	E	F
1	Equipe :					
2	Lieu :					
3	Type d'incident :					
4						
5						
6						
7						
8						
9						

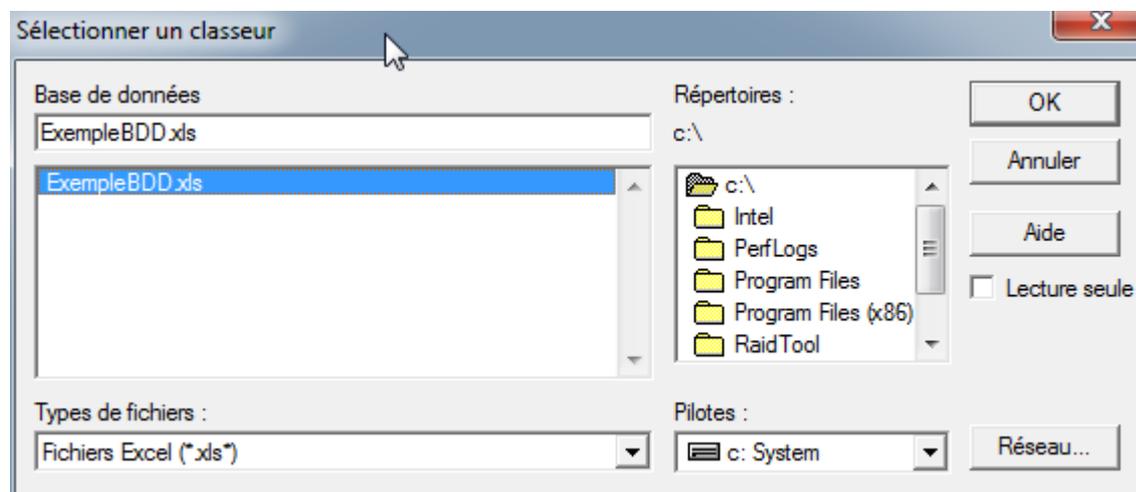
2. EXTRAIRE LA LISTE DES EQUIPES (EXTRACTION SIMPLE)

On va dans une première partie extraire la liste des équipes triée et sans doublons dans la feuille *FeuilTmp*.

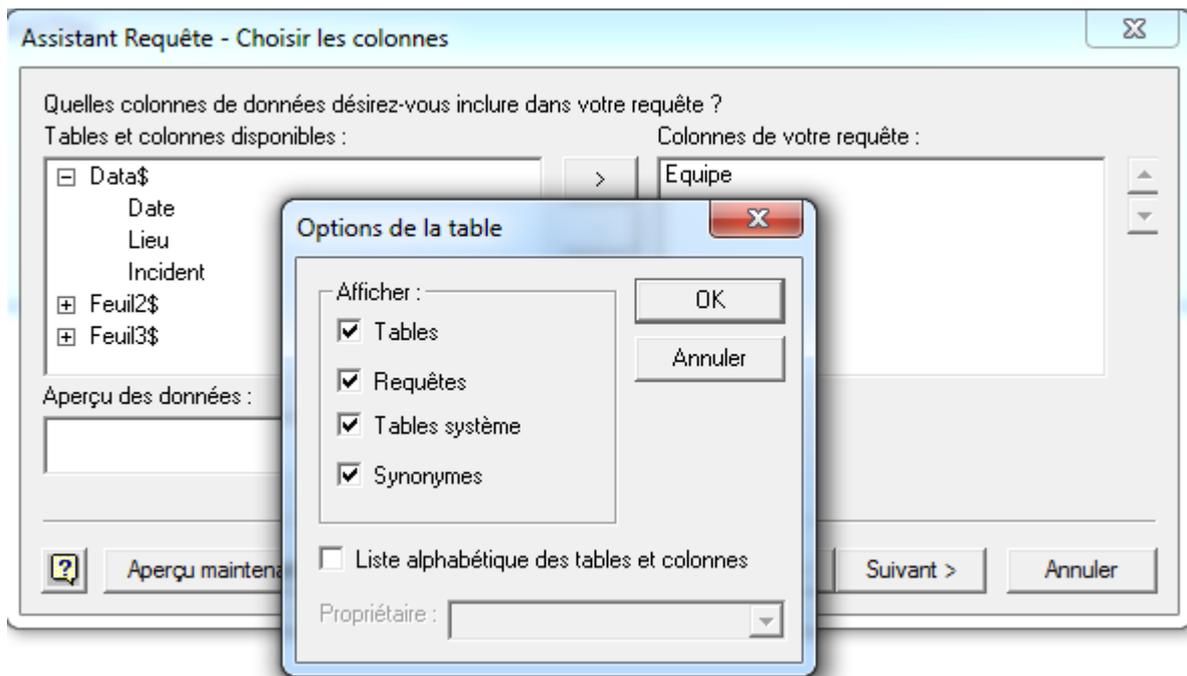
- Ouvrir MS-Query et choisir l'option *Excel File*
 - Interface Ruban : Données > Autres sources > Microsoft Query
 - Interface classique : Data > Import External Data > New Database Query



- Rechercher le fichier « C:\ExempleBDD.xls »

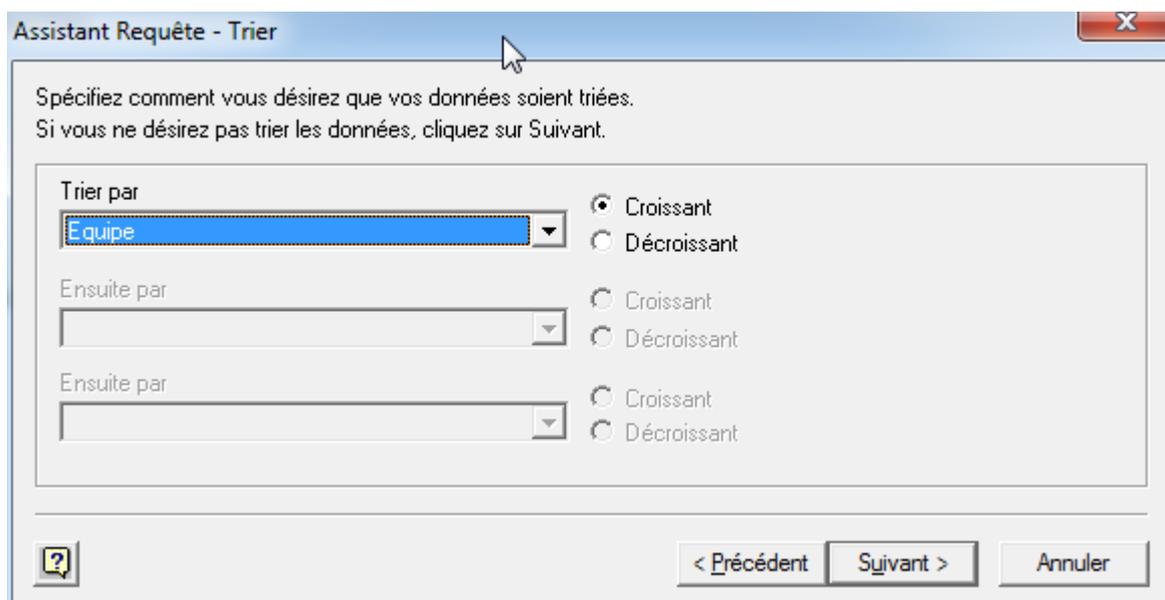


- La première fenêtre demande de sélectionner les colonnes à afficher
PS : Dans les options, cocher *Tables systèmes* ; cela permet de voir les feuilles dans la liste des tables.



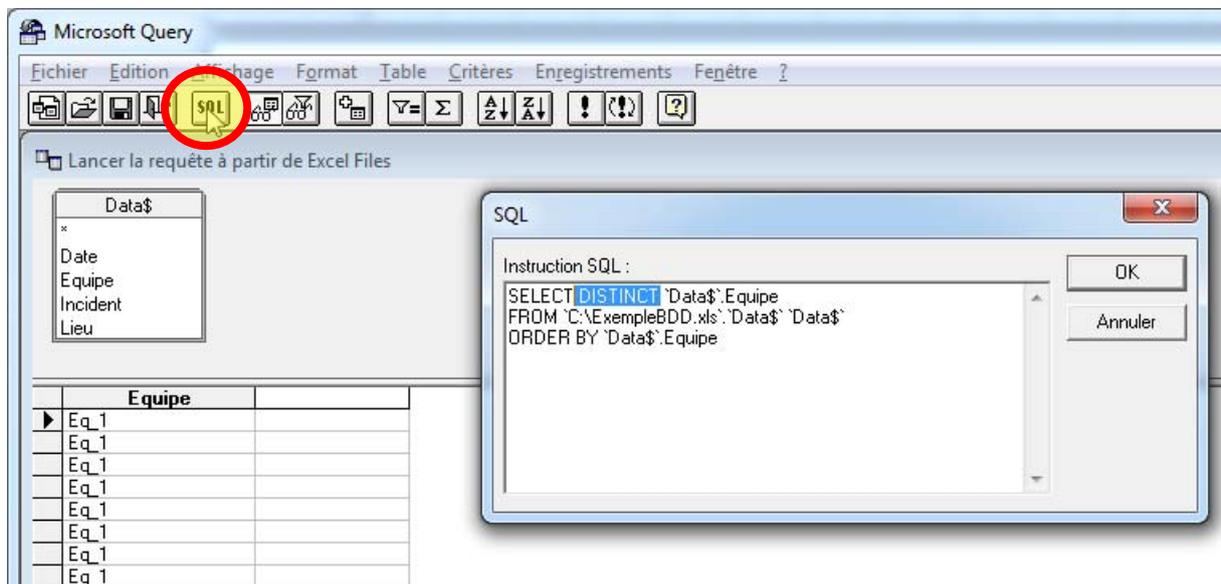
Notre objectif étant d'extraire la liste des équipes, on ne va sélectionner que la colonne *Equipe*.

- La fenêtre suivante sert à appliquer les filtres. On s'en servira plus tard dans le tutoriel.
- La fenêtre suivante sert à trier les données. Sélectionner *Equipe* et *Croissant*.



- Enfin, sur la dernière fenêtre, sélectionner l'option *Afficher les données ou modifier la requête dans Microsoft Query*, puis *Terminer*.

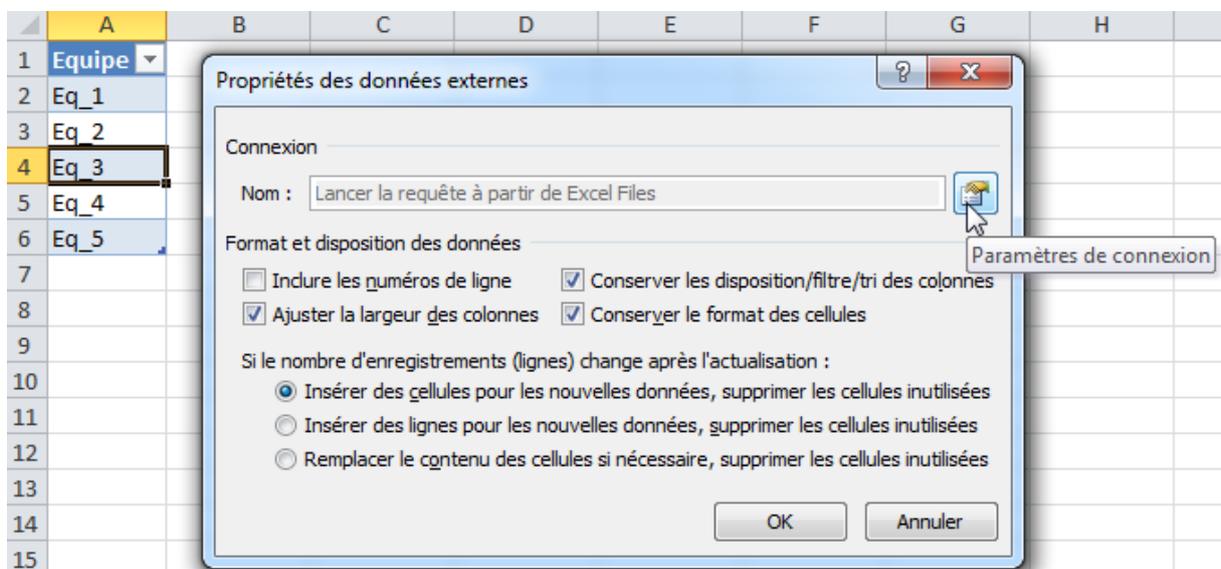
On arrive ici dans l'interface de Microsoft Query. On voit entre autre le résultat de la requête. Le problème actuellement est que la requête ne traite pas les doublons. On va donc la modifier légèrement en cliquant sur l'icône *Afficher l'interrogation SQL*.



Une fenêtre contenant la requête SQL générée en fonction des options choisies dans les étapes précédentes. Cette requête pouvant être modifiée, rajouter le mot **DISTINCT** après **SELECT**. Cela permet de ne pas renvoyer de doublons dans les résultats.

Pour finir, renvoyez le résultat de la requête à Excel en cliquant sur l'icône *Renvoyer les données* (📄↔️). Sélectionnez la cellule A1 de la feuille FeuilTmp, puis OK.

Le résultat est renvoyé sur la feuille. Faire clic droit sur une des cellules > Table > Propriété des données externes.



Cliquez ensuite sur l'icône Paramètre de connexion (en haut à droite). Une nouvelle fenêtre s'ouvre. Elle permet de choisir entre autre la fréquence d'actualisation des données. On va choisir de l'actualiser à l'ouverture du classeur.

On a donc en colonne A de la feuille *FeuilTmp* une liste triée sans doublons de toutes les équipes présentes dans le classeur « C:\ExempleBDD.xls ». Cette liste va servir de source de validation pour la cellule B1 de la feuille *Consultation* (Où l'utilisateur choisit l'équipe dont il veut consulter les incidents).

Définir le nom *ListeEquipes* : `=DECALER(FeuilTmp!A1;1;;NBVAL(FeuilTmp!$A:$A)-1)`.

Ajouter la validation en cellule B1 de la feuille *Consultation* (**Liste** > Source : `=ListeEquipes`)

A ce stade, on a créé en B1 une validation contenant la liste de toutes les équipes :

	A	B
1	Equipe :	Eq_3
2	Lieu :	Eq_1
3	Type d'incident :	Eq_2
4		Eq_3
		Eq_4
		Eq_5

3. EXTRAIRE LA LISTE DES LIEUX (EXTRACTION SIMPLE)

Refaire la même procédure pour extraire une liste sans doublons de tous les lieux dans la colonne B de la feuille *FeuilTmp*, créer un autre nom, et une validation en cellule B2 de la feuille *Consultation*.

FeuilTmp :

	A	B
1	Equipe	Lieu
2	Eq_1	Bat_1
3	Eq_2	Bat_10
4	Eq_3	Bat_11
5	Eq_4	Bat_12
6	Eq_5	Bat_13
7		Bat_14

Consultation :

	A	B
1	Equipe :	Eq_3
2	Lieu :	Bat_13
3	Type d'incident :	Bat_13
4		Bat_14
5		Bat_15
6		Bat_16
7		Bat_17
		Bat_18
		Bat_19
		Bat_2

Dans la cellule B3 de la feuille *Consultation*, Rajouter une validation de type liste avec comme source : **Grave;Normal;TOUS**.

	A	B
1	Equipe :	Eq_3
2	Lieu :	Bat_13
3	Type d'incident :	Normal
4		Grave
5		Normal
		TOUS

Dans la feuille *FeuilTmp*, saisir les formules suivantes dans les cellules :

- C1 : **=Consultation!B1 ;**
- C2 : **=Consultation!B2 ;**
- C3 : **=SI(Consultation!B3="TOUS";"";MAJUSCULE(Consultation!B3))&"%"**.

A ce stade, Le classeur de consultation possède trois cellules de choix pour l'utilisateur. Ces trois choix sont reportés en feuille *FeuilTmp*. Le type d'incident est « traduit » (*Normal* > *NORMAL%*; *Grave* > *GRAVE%* et *TOUS* > %).

PS : le % représente en SQL « n'importe quelle chaîne de caractère » (un peu comme le * en VBA)

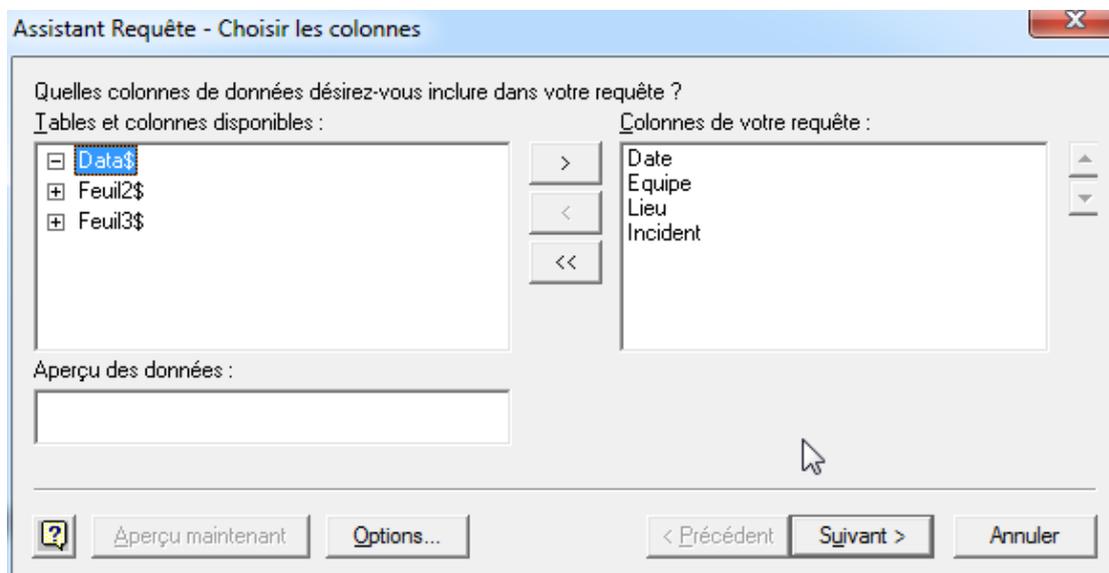
	A	B	C
1	Equipe ▾	Lieu ▾	Eq_3
2	Eq_1	Bat_1	Bat_13
3	Eq_2	Bat_10	NORMAL%
4	Eq_3	Bat_11	
5	Eq_4	Bat_12	
6	Eq_5	Bat_13	
7		Bat_14	

4. EXTRAIRE LA LISTE DES EVENEMENTS (EXTRACTION + TRI + FILTRE)

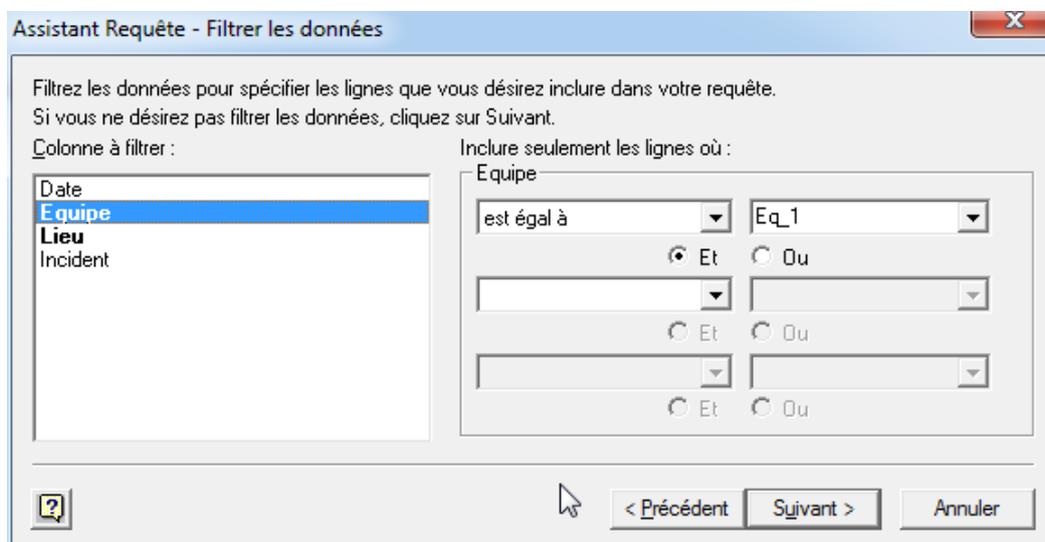
Il nous reste la dernière requête à créer – la plus intéressante.

Elle doit renvoyer sur la feuille consultation les évènements présents dans le classeur « C:\ExempleBDD.xls » en fonction des choix de l'utilisateur (Equipe, Lieux et Type).

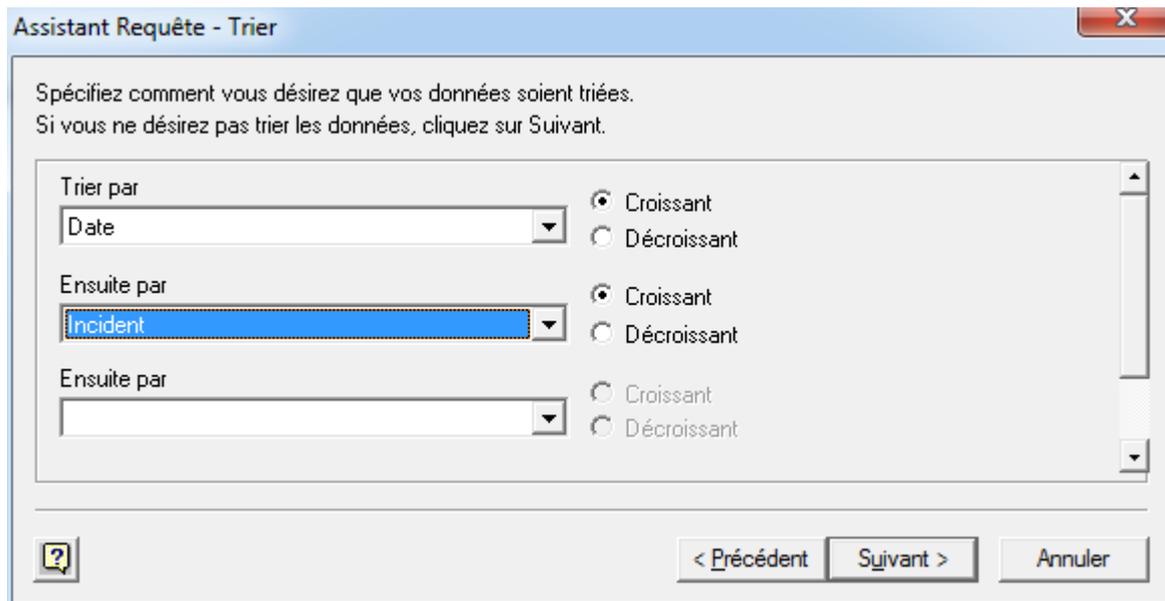
- Créer une nouvelle connexion au fichier « C:\ExempleBDD.xls »
- Sélectionner les colonnes à utiliser dans la requête : ici, les quatre colonnes de la feuille *Data*.



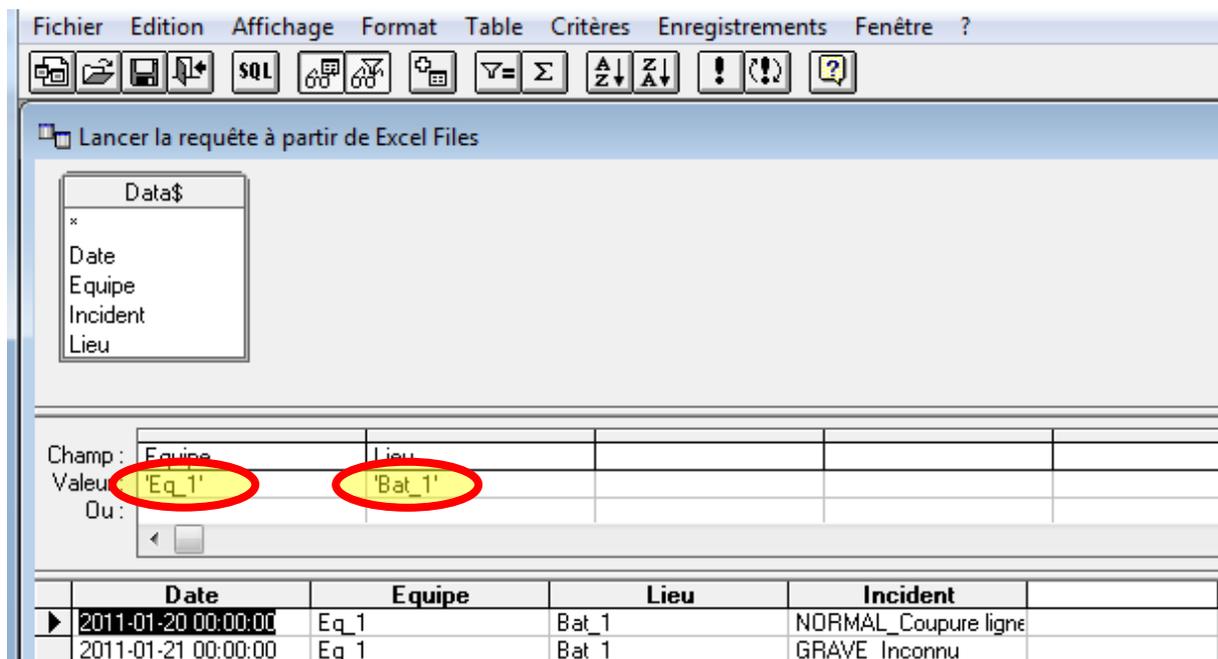
- Rajouter un filtre sur Equipe (**Est égal à Eq_1**) et sur Lieu (**Est égal à Bat_1**).



- Trier par **Date**, puis par **Incident**.



- Enfin, sur la dernière fenêtre, sélectionner l'option *Afficher les données ou modifier la requête dans Microsoft Query*, puis *Terminer*.



On peut voir que les critères de filtre ('Eq_1' et 'Bat_1') sont « en dur ». Pour associer plus tard des valeurs de cellules à ces critères (sélection de l'utilisateur), on va créer des variables CrEquipe et CrLieu qu'on va insérer dans la requête à la place des valeurs « en dur ».

La syntaxe est simple : il faut remplacer les valeurs par le nom des variables entre crochet [].

Champ :	Equipe	Lieu
Valeur :	[CrEquipe]	[CrLieu]

Si vous re-exécutez la requête (), une fenêtre vous demandera les valeurs pour ces critères.

On va rajouter un troisième critère de filtre sur la colonne *Incident* afin de ne récupérer que ceux commençant par GRAVE, ou NORMAL, ou tous.

Rajouter un troisième Champ (à droite de Lieu) sur *Incident* ayant pour valeur **LIKE [CrType]** (Normalement traduit automatiquement par **Comme [CrType]** si la version d'Excel est en français).

Champ :	Equipe	Lieu	Incident	
Valeur :	[CrEquipe]	[CrLieu]	Comme [CrType]	
Ou :				

Date	Equipe	Lieu	Incident
2011-01-20 00:00:00	Eq_1	Bat_1	NORMAL_Coupure ligne
2011-01-21 00:00:00	Eq_1	Bat_1	GRAVE_Inconnu

PS : Le tableau suivant liste les différents opérateurs utilisables dans les requêtes de MS-Query

Opérateur	Signification	Exemple	
=	Est égal à	Trouver tous les enregistrements dont le champ est égal à <i>toto</i>	= 'toto'
<>	Est différent de	Trouver tous les enregistrements dont le champ est différent de <i>titi</i>	<> 'titi'
>	Est supérieur à	Trouver tous les enregistrements dont le champ est supérieur à 20	>20
<	Est inférieur à	Trouver tous les enregistrements dont le champ est inférieur à 30	<30
>=	Est supérieur ou égal à	Trouver tous les enregistrements dont le champ est supérieur ou égal à 40	>=40
<=	Est inférieur ou égal à	Trouver tous les enregistrements dont le champ est inférieur ou égal à 40	<=50
And	Permet d'ajouter un critère (l'enregistrement doit remplir les deux critères pour être retourné)	Trouver tous les enregistrements dont le champ est supérieur à 60 et inférieur à 80	>60 And <80
Or	Permet d'ajouter un critère (l'enregistrement doit remplir un des deux critères pour être retourné)	Trouver tous les enregistrements dont le champ est égal à <i>toto</i> ou <i>titi</i>	toto' Or 'titi'
Not	Négation du critère (l'enregistrement ne doit pas remplir le critère pour être retourné)	Trouver tous les enregistrements dont le champ est différent de <i>titi</i>	Not 'titi'
Between	Permet de donner une plage de valeur pour le critère	Trouver tous les enregistrements dont le champ est compris entre 60 et 80	Between 60 And 80
In	Permet de donner un tableau de valeur pour le critère	Trouver tous les enregistrements dont le champ est égal à <i>tata</i> ou <i>toto</i> ou <i>titi</i>	In('tata', 'toto', 'titi')
Is Null	Est Null (retourne l'enregistrement si le champ ne contient pas d'info)	Trouver tous les enregistrements dont le champ est vide	Is Null
Is Not Null	N'est pas Null (retourne l'enregistrement si le champ contient une info)	Trouver tous les enregistrements dont le champ n'est pas vide	Is Not Null
Like	Like est utilisé avec le caractère % pour comparer les valeurs	Trouver tous les enregistrements dont le champ commence par <i>Fr</i>	Like 'Fr%'

Ensuite, on va supprimer dans MS-Query les colonnes *Equipe* et *Lieu* afin qu'elles ne soient pas renvoyées par la requête.

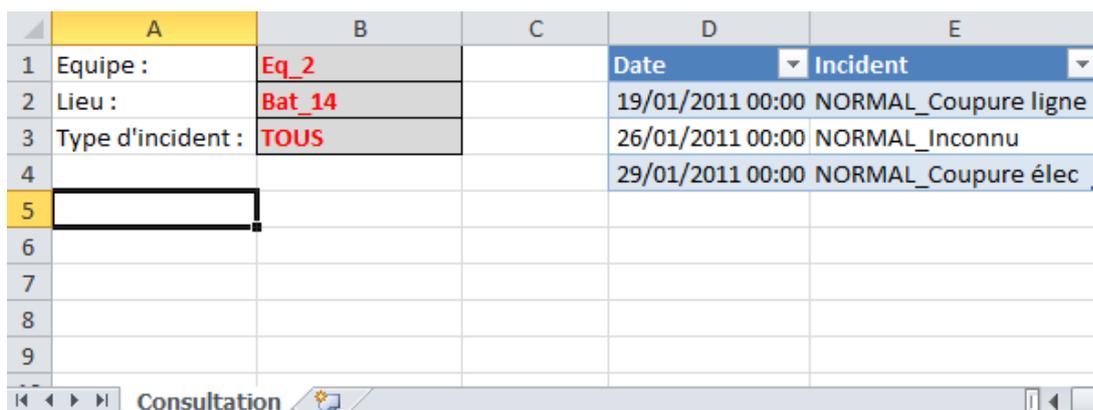
Pour finir avec MS-Query, renvoyer les données () en cellule *D1* de la feuille *Consultation*. Une fenêtre s'ouvrira pour saisir chaque paramètre :

- *CrEquipe* : **Eq_1** ;
- *CrLieu* : **Bat_1** ;
- *CrType* : **NORMAL%**.

Une fois revenu sur Excel, en faisant clic droit sur la zone de données renvoyé par la requête, puis *Table*, puis *Paramètres...*, **modifier les options de chaque paramètre** :

- *Obtenir la valeur de la cellule suivante* : sélectionner la cellule appropriée sur la feuille *FeuilTmp* ;
- Cocher l'option *Actualiser automatiquement lorsque la valeur de la cellule est modifiée*.

La feuille *FeuilTmp* peut ensuite être cachée à l'utilisateur final.



	A	B	C	D	E
1	Equipe :	Eq_2		Date	Incident
2	Lieu :	Bat_14		19/01/2011 00:00	NORMAL_Coupure ligne
3	Type d'incident :	TOUS		26/01/2011 00:00	NORMAL_Inconnu
4				29/01/2011 00:00	NORMAL_Coupure élec
5					
6					
7					
8					
9					

Voilà, vous avez créé un outil de consultation vers une source de données sans une ligne de code. N'est-ce pas merveilleux ?

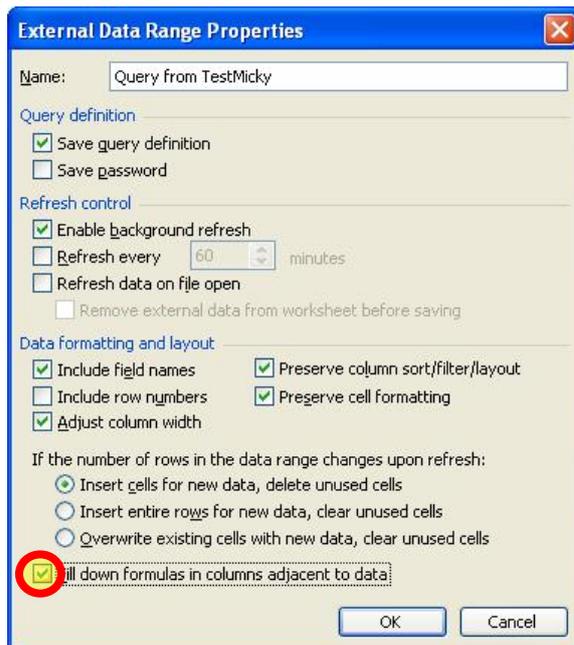
5. CONCLUSION / OUVERTURE

5.1. Pour creuser plus loin

Au terme de ce tutoriel, on s'aperçoit qu'on peut très facilement filtrer et trier des données depuis des sources de données. Les données renvoyées par MS-Query ont un nombre fixe de colonnes, mais un nombre variables de lignes. Il faut savoir qu'Excel permet de rajouter des sortes de « colonnes calculées » supplémentaires (colonnes contenant des formules effectuant des calculs ou du traitement avec les données des colonnes renvoyées par la requête).

F2 fx ="Le "&TEXTE(D2;"jjjj jj mmmm aaaa")& ", l'équipe "&B\$1&" a repéré l'incident ""&E2&"" dans "&B\$2						
	A	B	C	D	E	F
1	Equipe :	Eq_1		Date	Incident	Phrase Générée
2	Lieu :	Bat_13		09/01/2011 00:00	NORMAL_Inconnu	Le dimanche 09 janvier 2011, l'équipe Eq_1 a repéré l'incident "NORMAL_Inconnu" dans Bat_13
3	Type d'incident :	Normal		30/01/2011 00:00	NORMAL_Pane Info	Le dimanche 30 janvier 2011, l'équipe Eq_1 a repéré l'incident "NORMAL_Pane Info" dans Bat_13

Il suffit de rajouter une troisième colonne avec une formule pour le premier élément. Excel 2007 et 2010 auront automatiquement rajouté une colonne au Tableau (pour Excel 2003, il faut cocher l'option *Compléter les formules dans les colonnes adjacentes aux données* dans les propriétés de la Table). Ça y est vous avez rajouté une « colonne calculée ». Au prochain rafraichissement de la requête, tous les éléments auront la formule saisie dans cette colonne



A travers ce tutoriel, on a vu également que MS-Query permet d'effectuer des requêtes SQL sur des sources de données. Les requêtes SQL n'ont pas été très poussées dans ce document, mais sachez qu'il est possible de traiter plusieurs tables et d'effectuer des jointures entre elles grâce à MS-Query.

5.2. Limitations

Cet outil est donc parfaitement adapté à la lecture de données. Cependant, ne nous emballons pas trop vite. Cette méthode, bien que très pratique, performante et sans-code (souvent source d'erreurs) possède une forte limitation (lorsque la source de données est un fichier Excel – comme dans ce tutoriel) dont il faut avoir conscience.

Les connexions créées pointent vers le classeur « C:\ExempleBDD.xls ». Si ce classeur est déplacé, les connexions ne se feront plus.

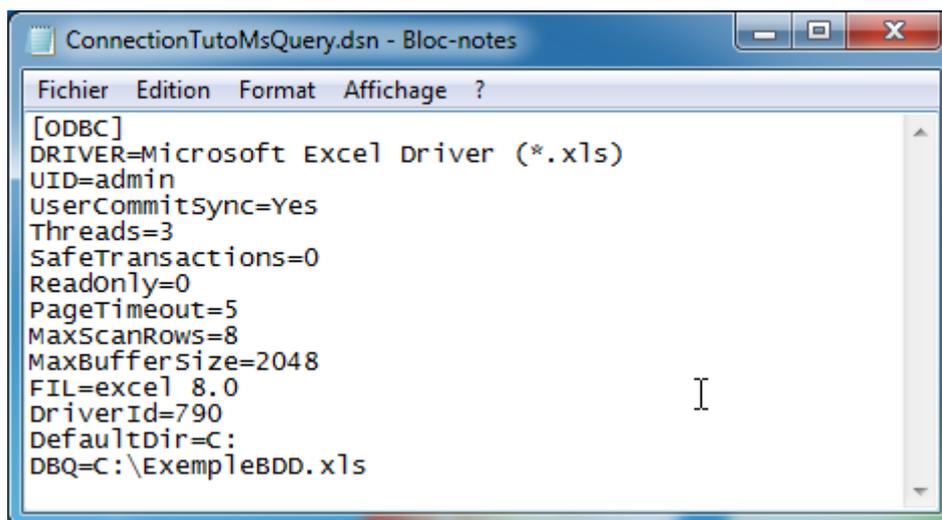
5.3. Utilisation en réseau / bonne pratique

On a vu que la limitation vient du fait que la connexion à la source de données est écrite « en dur » dans le fichier Excel (si on déplace le fichier source, les requêtes ne fonctionnent plus). Une solution afin de prévenir ce problème est de créer un driver ODBC vers la source de données et d'utiliser ce driver dans le fichier Excel pour se connecter à la source de données.

Vous trouverez [ici](#) un tutoriel pour créer le driver ODBC (vers le fichier Excel ou vers une autre base de données).

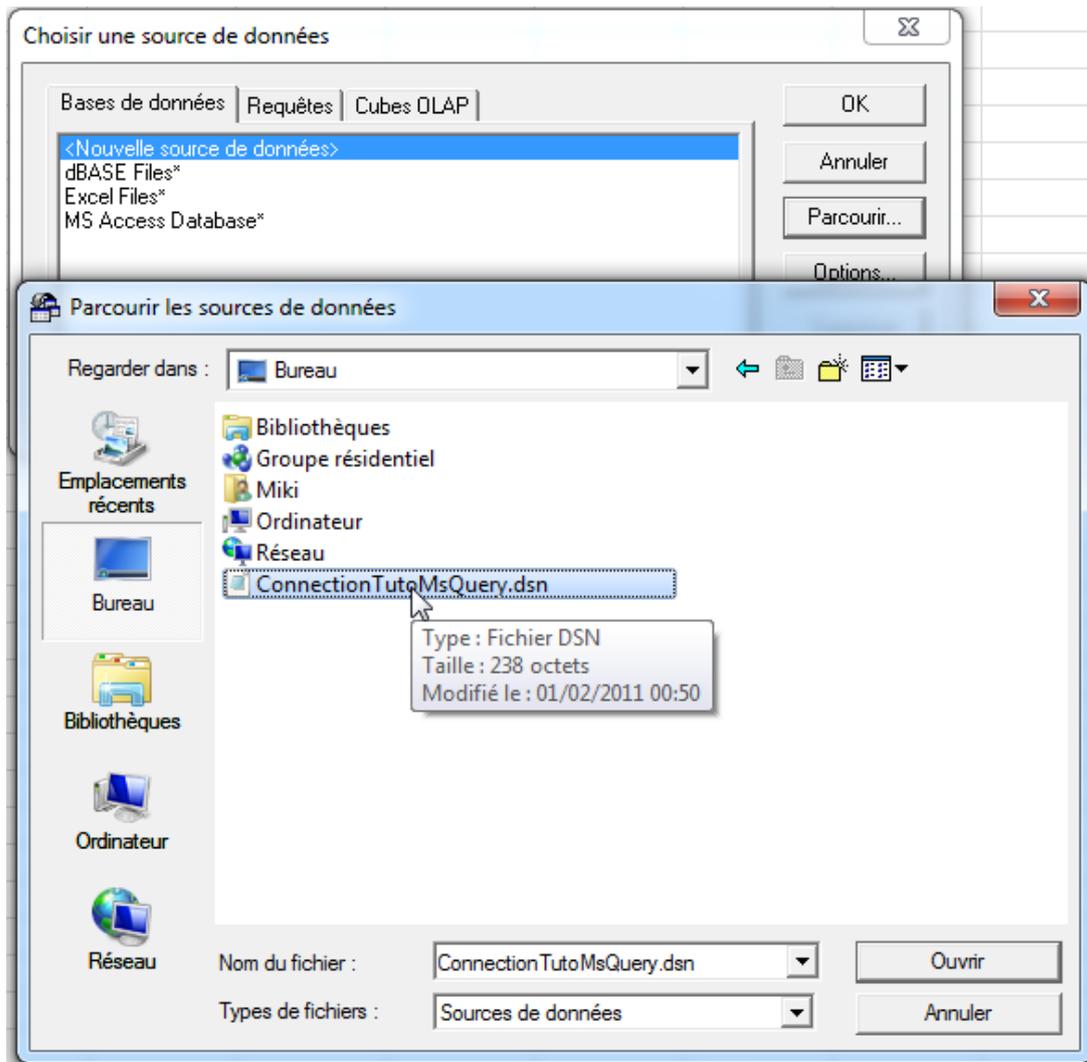
Une fois le driver créé, vous obtenez un fichier *.dsn* contenant les paramètres de connexions à la source de données. Mettez ce fichier dans un « répertoire sûr », afin d'être sûr qu'il ne sera pas déplacé ou effacé.

Si on ouvre le fichier *.dsn* avec l'éditeur de texte, on s'aperçoit que la source de données est de type *Excel*, située dans le répertoire *C :* et que le path du fichier Excel est *C:\ExempleBDD.xls*.



```
ConnectionTutoMsQuery.dsn - Bloc-notes
Fichier Edition Format Affichage ?
[ODBC]
DRIVER=Microsoft Excel Driver (*.xls)
UID=admin
UserCommitsync=Yes
Threads=3
SafeTransactions=0
ReadOnly=0
PageTimeout=5
MaxScanRows=8
MaxBufferSize=2048
FIL=excel 8.0
DriverId=790
DefaultDir=C:
DBQ=C:\ExempleBDD.xls
```

Cela permet dans l'assistant requête de MS-Query de récupérer ce fichier comme source de données :



On pourrait envisager avoir sur le réseau un répertoire contenant toutes les connexions vers les sources de données de l'entreprise (qui peuvent être de plusieurs types) sous forme de drivers ODBC.

Ce répertoire servirait de référence vers toutes les sources de données. Lorsqu'on utiliserait MS-Query dans des fichiers Excel, on pointerait vers ces fichiers de connexion (sur le réseau).

Ainsi, si une source de données est déplacée, il suffit de modifier le fichier *.dsn* – driver ODBC – correspondant (avec le bloc note). L'ensemble des applications utilisant cette source de données fonctionnera alors normalement.