Installation SQL server

Nous allons procéder à l'installation de SQL Express 2017 sur chacune de nos machines. L'installation de SQL Express est intéressante car elle présente le processus classique d'installation en se concentrant sur le moteur de base de données. Une installation d'un SQL Server complet sera semblable, proposera plus d'options mais celles spécifiques au moteur de BDD resteront identiques.

L'installation se décomposera en trois étapes :

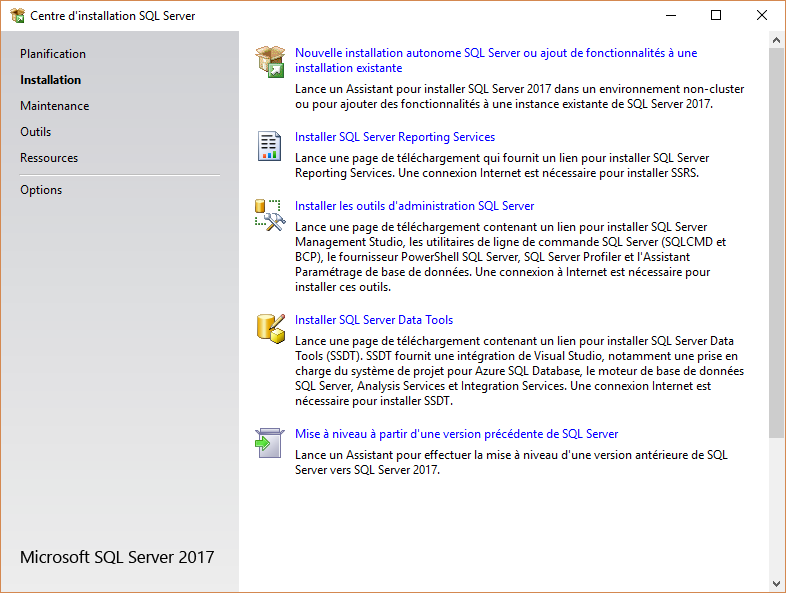
* L'installation de SQL server à proprement parler.
* La configuration réseau permettant l'accès à SQL Server depuis un autre PC que le vôtre.
* Puis celle des outils de gestion de SQL server (SQL Server Management Studio).

# Installation de SQL Server Express

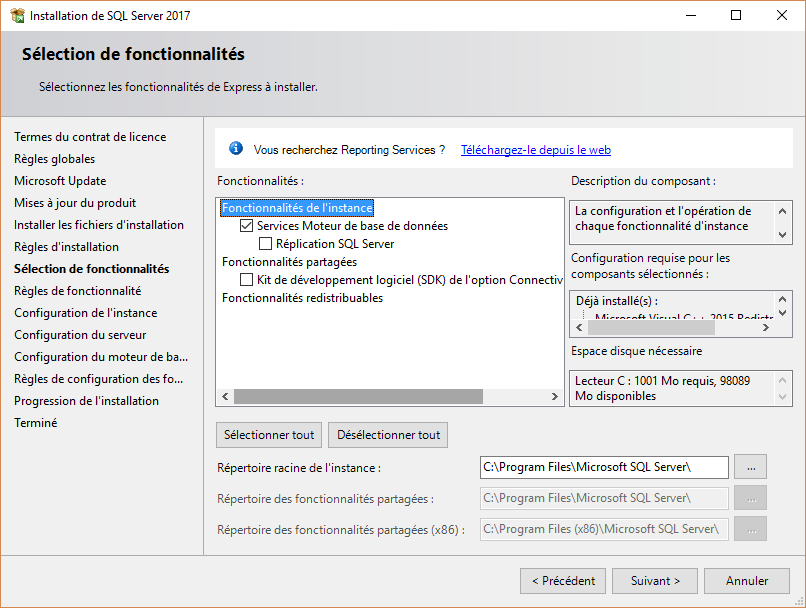
Depuis le répertoire commun, copiez SQLEXPR\_x64\_FRA.exe en local, lancez-le, spécifiez un répertoire qui contiendra les fichiers d'installation de SQL server, puis validez l'extraction

Dans le répertoire créé, lancez Setup.exe

Vous devriez arriver à la page suivante :

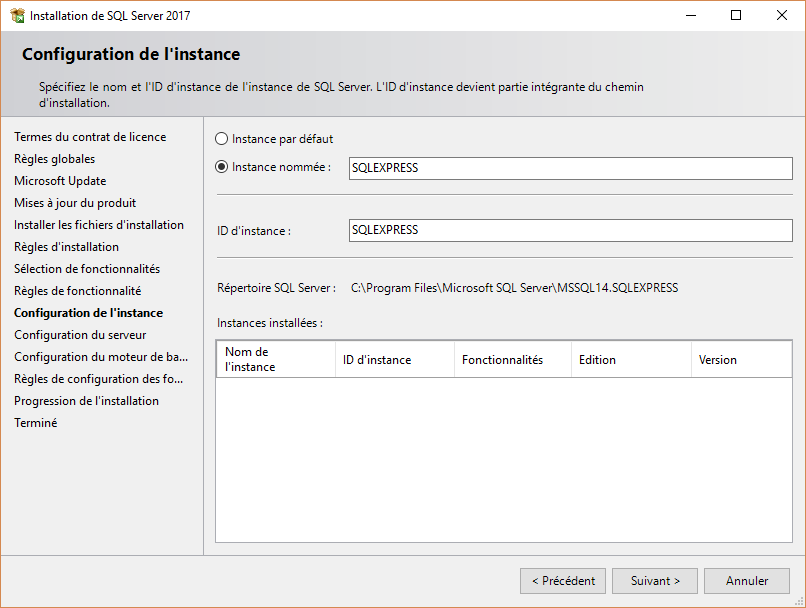


Dans celle-ci, sélectionnez la première option : "nouvelle installation autonome SQL Server"

Dans la fenêtre qui s'ouvrent, accepter les différentes options par défaut jusqu'à arriver à l'écran de sélection de fonctionnalités :

C'est dans cet écran que l'on peut choisir quelles briques logicielles on souhaite installer. SQL Server Express expose peu de possibilités, mais c'est à cet endroit sur une installation complète de SQL Server que l'on retrouverait SSRS, SSIS, SSAS etc. Dans notre cas, seul l'option "**Services Moteur de base de données**" nous intéresse.

Une fois les options voulues sélectionnées, validez les prochains écrans jusqu'à arriver sur l'écran de "configuration de l'instance"



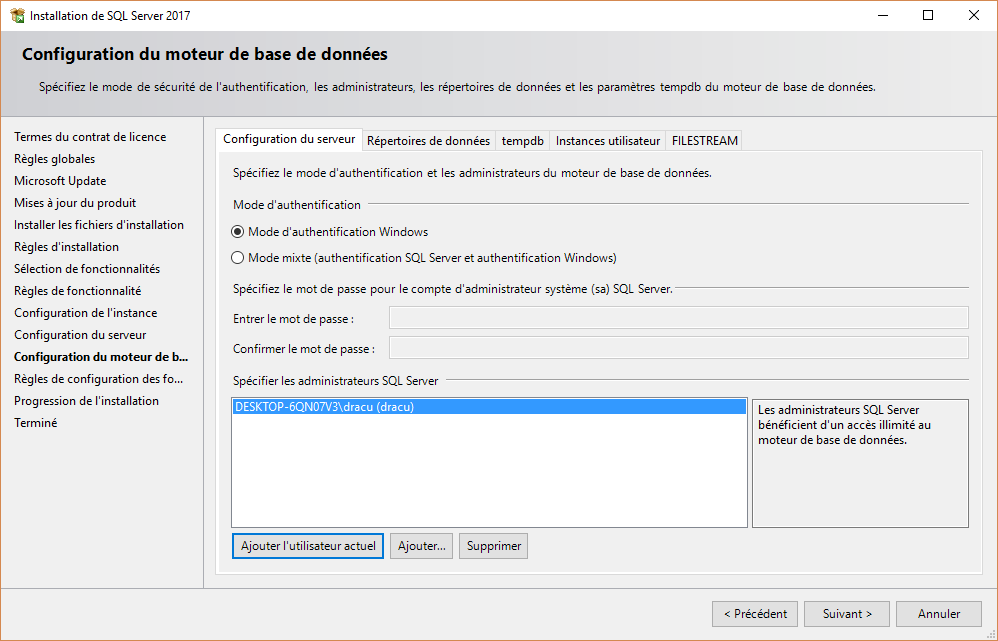
C'est sur cet écran que l'on doit sélectionner si l'on souhaite créer :

* Une instance par défaut
* Une instance nommée, auquel cas il est obligatoire de lui donner un nom
  + Par défaut, SQL Server Express s'installe en temps qu'instance nommée appelée "SQLExpress" et sera donc joignable à l'adresse suivante : "NomDeVotrePC\SQLExpress"

Pour plus de simplicité dans le groupe, gardez ce nommage par défaut. Puis validez l'écran ainsi que les suivants jusqu'à arriver sur le choix "**Configuration du moteur de base de données**"

En fonction des choix de fonctionnalités cochées plus haut, un certain nombre d'écrans seront alors proposés pour configurer chacune de ces dernières. Comme nous n'avons coché qu'une seule option, il n'y aura qu'un seul écran.

L'écran "Configuration du moteur de la base de données" est le plus important. Les options par défaut sont bonnes mais il vaut mieux les comprendre.



Il est constitué de 5 onglets :

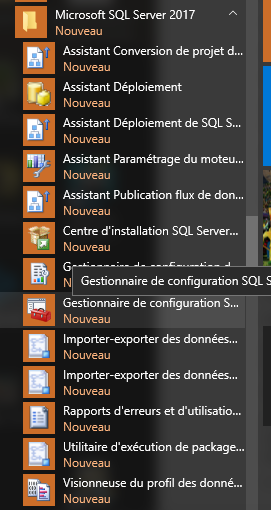
* Le premier, "Configuration du serveur" permet de gérer les options de connexion par défaut à la base de données, à savoir :
  + Activer ou non l'authentification SQL Server (l'authentification Windows est de toute façon activée et obligatoire)
  + Spécifier le mot de passe du compte "sa" (Superadministrateur) si l'option "Authentification SQL Server" est activée
  + Ajouter les comptes utilisateurs Windows qui seront administrateurs de l'instance SQL (le compte courant est ajouté par défaut sur SQL Express)
* Le second onglet, "Répertoire de données" permet de configurer où seront stockées les données liées à l'instance SQL Server, ainsi que là où seront proposées d'être stockées les bases de données que l'on créera plus tard (il ne s'agit que d'une configuration par défaut, on pourra la modifier au moment de la création)
* Les trois derniers onglets permettent de paramétrer des comportements spécifiques hors contexte dans ce cours.

Une fois bien compris, validez les écrans restants, l'installation va alors s'effectuer.

# Configuration réseau

## Côté SQL Server

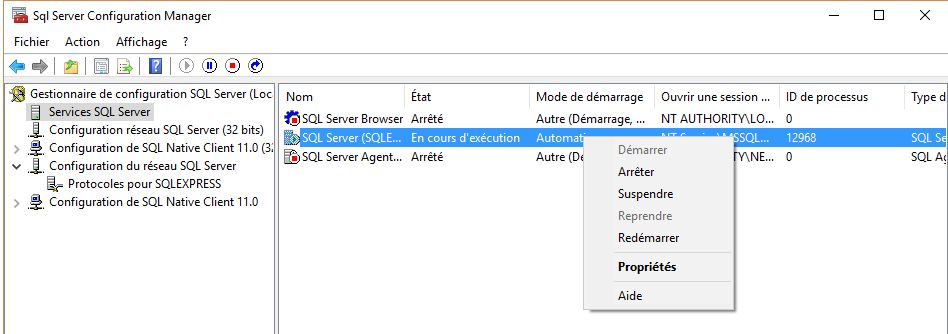
Maintenant que SQL Server est installé, nous allons le configurer pour accepter les requêtes provenant d'autres ordinateurs (ce n'est pas le cas par défaut). Dans le menu "Windows", sélectionnez "Microsoft SQL Server 2017", puis "Gestionnaire de configuration SQL Server"



Dans l'application qui s'ouvre, dans la colonne de gauche, plusieurs choix vous sont proposés, notamment :

* Services SQL Server

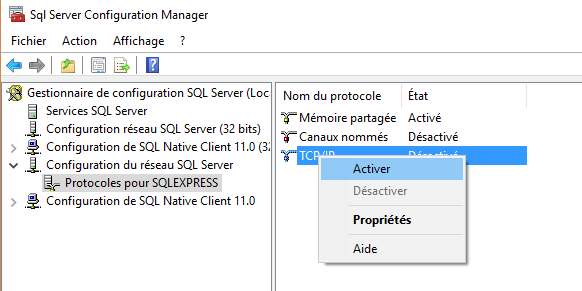
Cet écran vous permettra de gérer les services Windows qui hébergent SQL Server, notamment pour les arrêter, les redémarrer etc. Les services "SQL Server Browser" et "SQL Server Agent" ne nous intéressent pas dans ce cours.



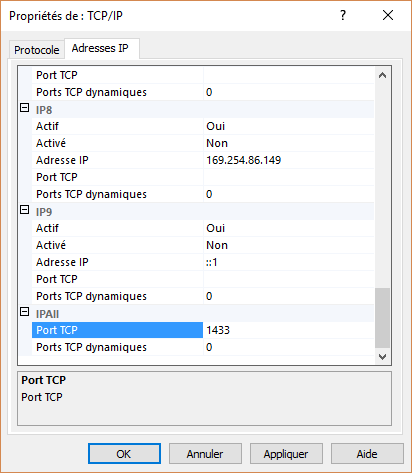
* Configuration du réseau SQL Server

Ce choix permet de configurer les différents protocoles exposés par SQL Server pour pouvoir être interrogés. Par défaut, seul "Mémoire partagée" est activé et ce protocole ne permet que la connexion depuis un client local, mais pas depuis un client distant.

Le protocole qui nous intéresse est "TCP/IP". Il est désactivé par défaut, mais on peut l'activer en faisant un clic-droit, puis "activer".



Puis, il faut ouvrir les propriétés du protocole (double-clic sur TCP/IP ou clic-droit puis "Propriétés"), puis dans l'onglet "Adresses IP", tout à la fin, saisir le port 1433 dans "Port TCP" sous le choix "IPAll"



Cela rend accessible l'instance SQL Server sur le port 1433 (port utilisé pour SQL Server par convention) sur toutes les connexions réseau du PC. Il faut alors en suite redémarrer le service (vous devriez savoir le faire maintenant 😊)

## Côté Firewall

Néanmoins, ce port est bloqué par défaut par le Firewall de Windows, il faut donc l'ouvrir. Pour se faire :

* Pressez la touche "Windows ", puis tapez "wf.msc", puis pressez la touche "Entrée".
* Puis dans la fenêtre qui est apparue, cliquez-droit (c'est un verbe ça ?) sur "Règles de trafic entrant", puis sélectionnez "Nouvelle règle"
* Dans la fenêtre qui s'ouvre, sélectionnez l'option "Port"
* Puis dans l'écran suivant "TCP" et saisissez "1433" dans le port à ouvrir"
* Puis dans l'écran suivant, sélectionnez "Autoriser la connexion"
* Laissez les options par défaut dans l'onglet "Profil"
* Puis donnez un nom à la règle (par exemple "SQL Server - Port 1433") dans l'écran "Nom"

# Installation des outils de gestion de SQL Server Express

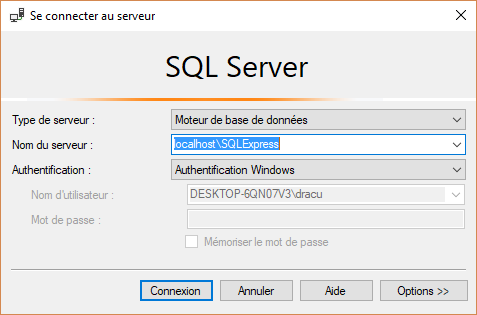
L'installation de SQL Server est maintenant fonctionnelle, néanmoins aucun outil n'est installé par défaut. Visual Studio permet des manipulations de base sur SQL Server, mais dans l'industrie, on préfèrera installer un outil bien plus puissant appelé "SQL Server Management Studio", abrégé en "SSMS".

Pour ce faire, installez SSMS via SSMS-Setup-FRA.exe présent dans le répertoire "commun" (Copiez le en local d'abord, ne le lancez pas depuis le répertoire commun). Les options par défaut de l'installateur sont bonnes.

*Pour information, SQL Server Management Studio se base sur une petite partie de Visual Studio (appelée "Visual Studio Shell") qui sera installée automatiquement. Une fois SSMS installé, il y aura donc potentiellement deux Visual Studio sur la machine, dans des versions différentes, ce qui n'est pas problématique en soi, mais si vous vous demandez d'où vient cette autre version de Visual Studio qui pourrait apparaitre plus tard, l'explication vient de là.*

Puis lancez SQL Server management studio en tapant la touche "Windows", puis "SSMS" puis "Entrée"

Le premier écran vous demander de saisir les informations permettant de se connecter à l'instance SQL, à savoir :



* Le nom du serveur
  + Dans notre cas, comme il s'agit d'une instance nommée, elle est de la forme :

NomDeVotrePC\NomDeLinstanceNommee

Donc par exemple

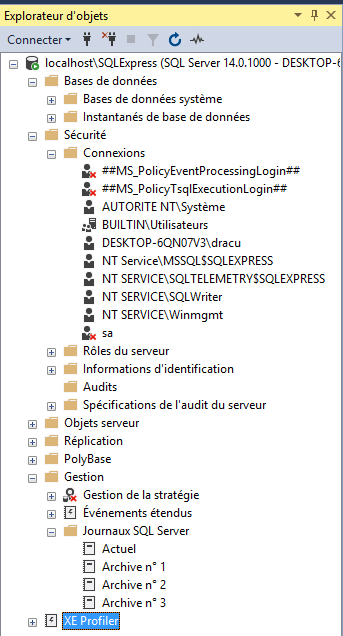
localhost\SQLExpress

* + Si l'instance était une instance par défaut, il suffirait ici de saisir "NomDeVotrePC" ou même plus simplement "."
* Les informations de connexion
  + Par défaut, SSMS tente de se connecter avec l'utilisateur Windows courant
  + Il est possible de se connecter avec un autre utilisateur Windows
  + Ou de se connecter avec un compte de connexion SQL.
  + Dans notre cas, on se connectera toujours avec l'utilisateur Windows courant.

SSMS se connecte alors au serveur, une nouvelle entrée apparait dans le panneau "Explorateur d'objet".

# Découverte de SSMS

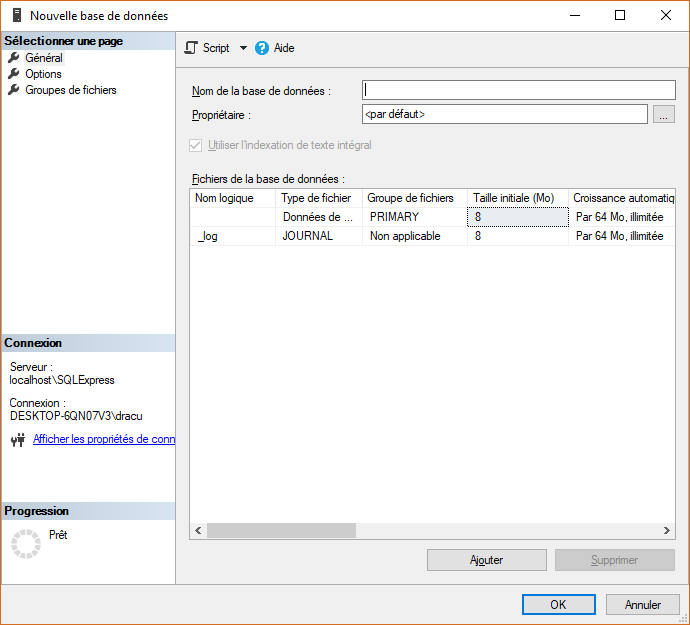
Une fois connecté à une instance SQL Serveur, on peut remarquer un certain nombre de choses (j'ai déplié certains nœuds pour plus d'explication)



* Le nœud "Bases de données" ne contient aucun élément autre que "Bases de données système" et "Instantanées de base de données" car aucune base de données n'est créée par défaut. Lorsque l'on créera des bases de données dans notre instance, elles apparaitront ici.
* Dans le nœud "Sécurité" puis "Connexions" apparaissent un certain nombre de comptes déjà existants, notamment l'utilisateur courant, ainsi que "sa" qui est probablement désactivé. Si vous double-cliquez sur votre login, puis que vous allez dans "Rôles du serveur", vous pourrez voir les rôles de sécurité que vous avez, notamment "sysadmin" (le plus important, il vous donne tous les droits).
  + Profitez-en pour ajouter un nouveau compte de connexion pour votre voisin de table en faisant un clic-droit sur "Connexions" puis "Nouvelle connexion…"
  + Dans l'onglet "Général", renseignez son nom d'accès sous la forme "DomainActiveDirectory\LoginUtilisateur" (celui utilisé pour se connecter à sa machine), puis validez.
  + Demandez à votre collègue de se connecter à votre instance SQL Server. Normalement, il devrait pouvoir s'y connecter, mais il n'y verra rien car son rôle par défaut ne lui permet rien (c'est "voulu").
* Le nœud "Gestion" puis "Journaux SQL Server" contient les journaux ("logs") de SQL Server et peuvent être très pratiques pour rechercher la cause de certains soucis, par exemple ceux de connexion.

# Création d'une première base de données

Nous allons créer notre première base de données. Pour ce faire, effectuez un clic-droit sur le nœud "Base de données", puis cliquez sur "Nouvelle base de données". L'écran suivant vous permettra de configurer la nouvelle base de données à créer.



Renseignez donc le nom puis notez que cet écran vous permet aussi de spécifier où seront disposés les fichiers qui contiendront plus tard :

* Les données à proprement parler (un fichier avec une extension ".mdf")
* Le journal de la base de données : concrètement la liste de toutes les opérations effectuées depuis la création de la base de données, permettant de revenir dans l'état d'une base de données dans le passé, de faire des sauvegardes incrémentales etc. (un fichier avec l'extension ".ldf")

Avant de cliquer sur OK, cliquez plutôt sur le bouton "Script" en haut de la fenêtre. Cela vous ouvrira un fichier texte qui comprendra une requête SQL vous permettant de créer la base de données telle que vous l'avez configuré dans la fenêtre. En fait, valider la fenêtre avec le bouton "OK" ne ferait qu'exécuter l'équivalent de la requête que vous avez généré. L'intérêt d'avoir ce script plutôt que de juste cliquer sur "OK" est que vous pouvez conserver ce script quelque part pour le relancer à l'envie plus tard, sans avoir à réeffectuer toutes les opérations de configuration manuellement.

*Il en sera de même plus tard, les différentes manipulations que SSMS vous proposera d'effectuer (création de table, de lignes, de vues, manipulation de droits, suppressions, sauvegarde et restauration etc.) ne sont en général que des interfaces graphiques qui vous permettent de générer du SQL en "sous-main". Il est toujours possible de récupérer ce code SQL, tout comme il est possible d'effectuer toutes ces opérations directement en SQL.*

Maintenant que vous avez à votre disposition le script SQL à exécuter pour générer votre première base de données, cliquez sur le bouton "Exécuter" dans la barre d'outils. L'onglet "Résultats" devrait alors afficher "Commandes réussies.". La nouvelle base de données ne s'affichera pas immédiatement dans l'explorateur d'objets car il faut rafraichir la liste pour qu'elle apparaisse. Vous pouvez fermer puis rouvrir le nœud "Base de données" ou effectuer un clic droit sur ce nœud et cliquer sur "Rafraichir". Il en sera de même pour tous les objets que l'on va créer par la suite (tables, utilisateurs etc.)

# La Suite

La suite, on la verra tous ensemble 😊