

# Architecture serveur

Le choix de l'architecture serveur est un préalable nécessaire au projet pour plusieurs raisons :

- Estimer le coût des machines et du matériel nécessaire.
- Estimer le coût des licences.
- Laisser le temps à l'équipe exploitation de s'organiser, de se former et d'installer les environnements.

La mise en place d'un projet décisionnel nécessite en fait plusieurs environnements de travail. On dénombre généralement :

- Un environnement de développement : c'est sur cet environnement que seront réalisés notamment les développements ETL. Ce type de développement est assez gourmand en ressources machine, il s'agit alors de le désolidariser de l'environnement de travail des utilisateurs.
- Un environnement de recette : l'environnement de recette ou de pré-production est l'environnement de travail sur lequel on évalue la qualité du travail réalisé en développement. La recette est très importante et peut parfois prendre du temps, il est donc important que cet environnement soit distinct du développement et de la production afin de ne perturber ni les développeurs ni les utilisateurs.
- Un environnement d'intégration : l'environnement d'intégration est quant à lui un laboratoire sur lequel l'équipe exploitation peut tester, s'entraîner et préparer les travaux de maintenance ou de mises à jour applicatives.

*Par exemple, si vous partez tout d'abord sur SharePoint Foundation 2013, puis que vous souhaitez passer sous SharePoint Server 2013, l'environnement d'intégration permettra à l'équipe exploitation de réaliser une étude d'impact et d'écrire les procédures permettant d'éviter les écueils et de réduire les perturbations sur l'environnement de production.*

- Un environnement de production : c'est l'environnement de travail des utilisateurs. De fait, cet environnement doit avoir le moins possible d'interruptions de services tout en offrant avec constance les meilleures performances.

En début de projet décisionnel, deux environnements peuvent suffire : celui de développement et celui de production, l'environnement de production jouant dans un premier temps aussi le rôle d'environnement de recette et d'intégration.

Toutefois, une fois que le projet est lancé et que le premier lot projet va être en production et bientôt mis à disposition des utilisateurs, il devient indispensable de disposer a minima d'un environnement de recette, pouvant jouer aussi pendant un temps le rôle d'intégration.

Ensuite, quand le projet arrive à une certaine maturité et que le système décisionnel devient incontournable et donc critique, il convient alors de distinguer l'environnement de recette de l'environnement d'intégration.

L'environnement de recette doit répondre aux exigences des équipes de développement sans les contraintes de haute disponibilités et de performance, alors que l'environnement d'intégration doit répondre aux exigences des équipes d'exploitation sans toutefois obligatoirement disposer des dernières données ou des dernières versions des flux ou du métamodèle. Il n'y a pas de règles en la matière, ce qui compte surtout est de bien clarifier les rôles des équipes de développement et d'exploitation, et surtout leur donner les moyens de mener correctement leur travail en toute sérénité.

## 1. L'environnement de production

La performance d'un environnement de production est évaluée suivant deux axes :

- Sa capacité à monter en charge pour répondre efficacement à toutes les demandes utilisateurs. Rappelons que le système décisionnel doit présenter des chiffres dans des délais de l'ordre de la seconde. Un décideur n'attendra certainement pas plusieurs minutes que son tableau veuille bien s'afficher.
- Sa haute disponibilité. Par haute disponibilité, on entend sa capacité à offrir un service chaque fois qu'un utilisateur en a besoin. Un service qui est régulièrement indisponible fera fuir les utilisateurs. De plus, si le décisionnel devient l'outil de pilotage de l'entreprise, cela peut devenir catastrophique si celui-ci ne fonctionne pas lorsqu'on a besoin de lui pour prendre une décision importante.

Les critères entrant en jeu lors de la définition de l'environnement de production sont :

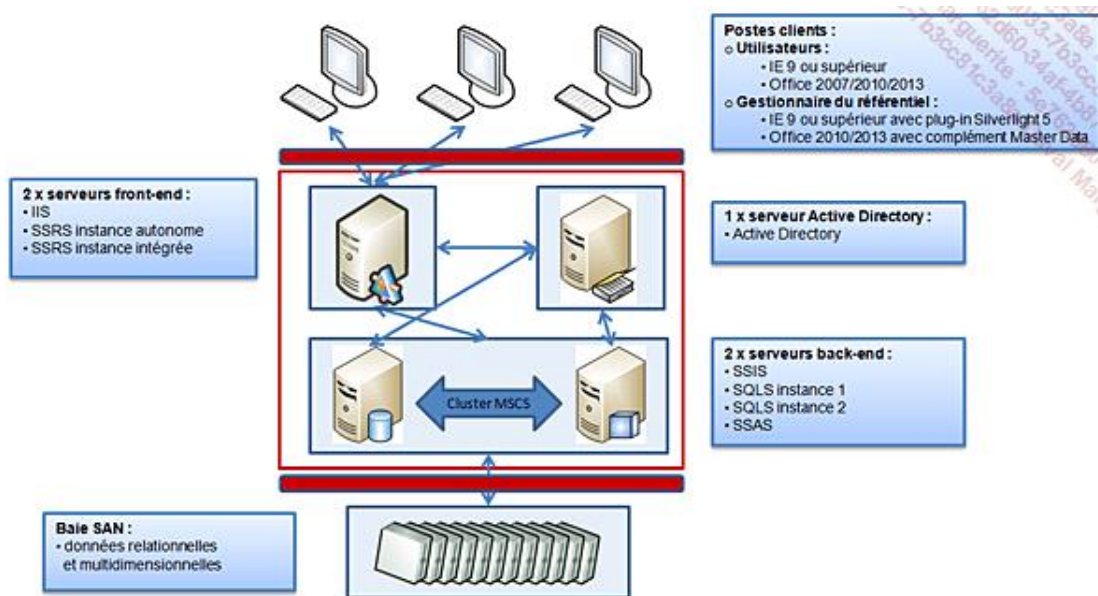
- La volumétrie : le volume de données influe bien évidemment sur la taille des machines et des disques.

➤ Lorsque l'on parle d'estimation du volume de données, il faut sous-entendre l'ordre de grandeur de l'entrepôt de données : parle-t-on de Mo, de Go, de centaines de Go ou de To ? Il ne faut pas chercher à faire une estimation exacte. Généralement, pour estimer l'ordre de grandeur d'un entrepôt de données, on se base sur l'estimation de la volumétrie des tables de faits qui seront susceptibles d'être les plus volumineuses. Cette estimation devra être réalisée à partir d'hypothèses précises, chiffrées et validées.

➤ Par volumétrie, il est aussi vraiment très important de prendre en considération le volume des données de référence. Si vous disposez d'une donnée de référence de plusieurs millions de membres (les abonnés d'une société de télécom ou les produits d'un grand supermarché par exemple), ce volume de données aura forcément un impact un jour ou l'autre sur le cache nécessaire au niveau de l'ETL (mémoire RAM) ou sur la probable émergence d'une table de type photo gigantesque.

- Le nombre d'utilisateurs au total et le nombre d'analystes : vous comprendrez aisément l'importance de ce critère si vous réalisez un suivi commercial de votre réseau de plusieurs milliers d'agents... Le nombre d'analystes est quant à lui important, car ces utilisateurs ont un mode de consommation du système décisionnel qui pèsera beaucoup plus sur ses performances qu'un utilisateur lambda qui va recevoir une fois par mois un rapport dans sa boîte mail.
- Les prévisions d'évolution et de montée en charge du système : l'architecture devra bien évidemment prendre en compte la croissance prévisionnelle du périmètre fonctionnel de l'entrepôt de données, une augmentation du périmètre fonctionnel ayant forcément un impact sur le nombre d'utilisateurs et sur la volumétrie du système.
- La haute disponibilité : parmi tous les critères, l'exigence en haute disponibilité est sûrement le critère le plus contraignant et celui qui a le plus d'impact dans l'élaboration de l'environnement de production. On mesure la disponibilité avec des taux de disponibilité composés de 9 :
  - Un taux de disponibilité de 99% correspondant à un arrêt de service de 3,65 jours dans l'année.
  - Un taux de 99,9% à un arrêt de 8,75 heures par an.
  - Un taux de 99,99% à un arrêt de 52 minutes par an... et ainsi de suite.

La contrainte de haute disponibilité étant généralement présente, de ce fait, la typologie de l'environnement de production est bien souvent celle présentée ci-après.



*Typologie de l'architecture de production en haute disponibilité*

Cette architecture nécessite :

- 2 serveurs back-end en cluster, qui devront être deux grosses machines physiques.
- 2 serveurs front-end en load balancing (ou équilibrage de charge), qui peuvent être deux petites machines virtuelles.
- 1 serveur Active Directory : il s'agira vraisemblablement d'un serveur existant. Sachez cependant que le service Active Directory sera nécessaire pour installer et configurer correctement la solution BI Microsoft.
- Une baie SAN : l'optimisation d'une architecture serveur passe nécessairement par un plan de découpage et d'isolement des bases et des partitions sur des disques physiques distincts :
  - Il s'agit tout d'abord de séparer les fichiers Data, Logs et Temp du service SQL Server Database Services sur des disques distincts.
  - Il s'agit ensuite qu'au sein d'un même flux le disque contenant la base ou la table source soit distinct de celui contenant la base ou la table de destination. Et ce, afin d'éviter qu'un disque physique travaille à la fois en lecture et en écriture au cours de l'exécution d'un flux.

Si vous ne disposez pas de SAN et que vous ne souhaitez pas investir, prévoyez un maximum de disques durs.

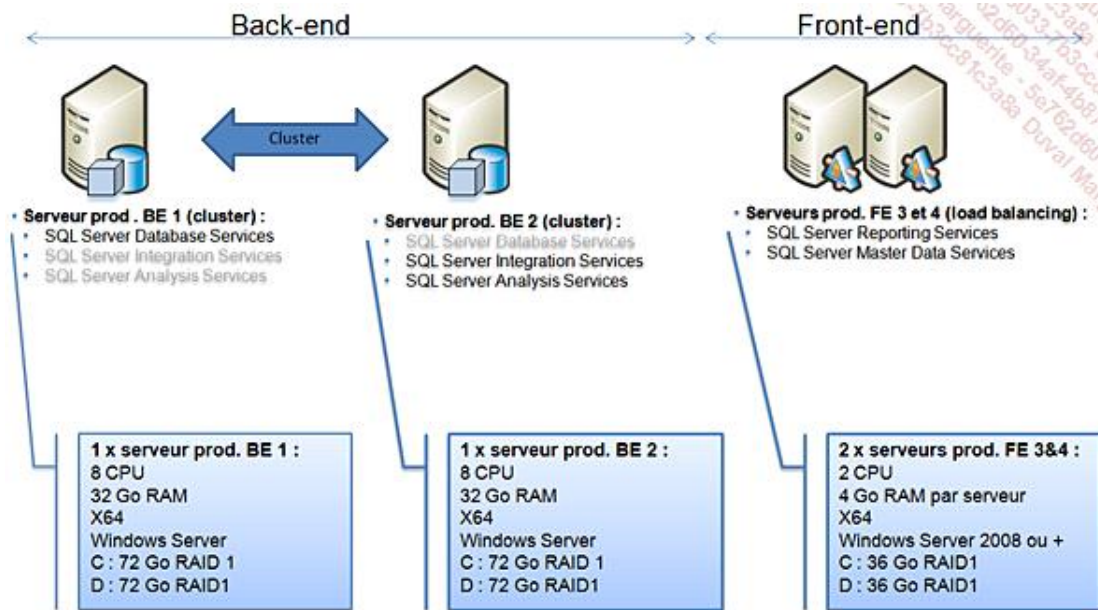
Cet ouvrage n'a pas vocation à être un livre blanc sur l'installation de la solution BI. Toutefois, voilà quelques remarques ou points d'attention :

- Tous les services applicatifs seront installés sur les serveurs front-end mais auront leur base de données de référence installée sur le cluster des serveurs de back-end.
- Installez au moins deux instances SQL Server Database Services : l'une pour accueillir les bases décisionnelles (DW et SA - cf. chapitre Réaliser son premier système décisionnel) et l'autre pour accueillir les bases de données de référence des applicatifs, telles que Reporting Services ou SharePoint.
- Tous les services seront installés sans exception avec des comptes de service qui seront exclusivement des comptes de domaines. Avoir deux comptes est vivement conseillé : un pour l'installation de tous les services de back-end, un autre pour tous les services de front-end. D'autres comptes pourront être indispensables notamment en ce qui concerne la configuration de SharePoint.
- Enfin, cette architecture étant distribuée, elle requerra forcément la configuration de la délégation de sécurité Kerberos. Il s'agit en fait de la difficulté de l'installation. Nous vous engageons alors à contacter d'une part votre équipe exploitation et d'autre part l'éditeur ou un prestataire spécialisé pour vous faire accompagner dans la mise en

place de Kerberos.

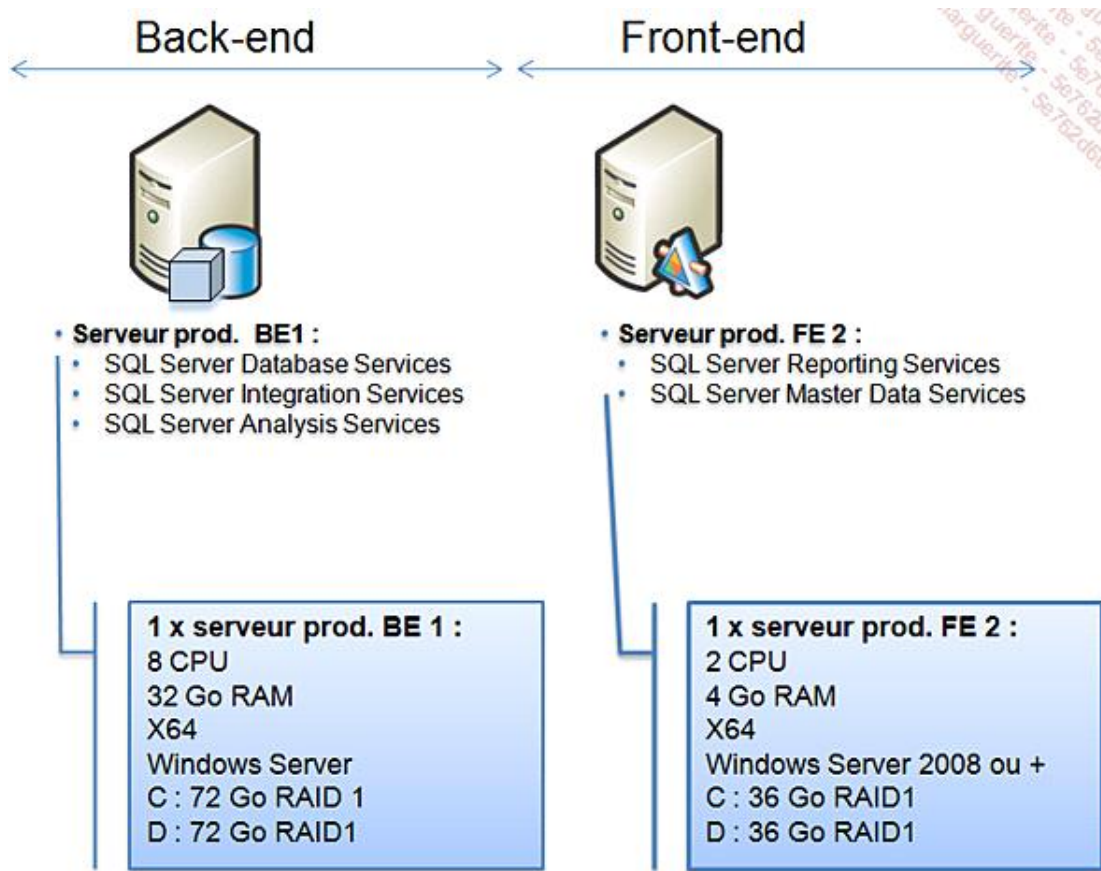
Enfin, pour être concret, nous vous proposons ci-dessous deux configurations types :

- La configuration de production intégrant la haute disponibilité :



*Exemple de configuration pour un environnement de production en haute disponibilité*

- La configuration de production sans haute disponibilité :



*Exemple de configuration de production sans haute disponibilité*

➤ Attention toutefois de bien comprendre qu'une interruption de service de plus de 24 heures est envisageable avec une configuration n'intégrant pas la haute disponibilité.

La taille des machines devra bien évidemment être adaptée à votre contexte (volumétrie, utilisateurs, budget...) et aux prix pratiqués par le marché. Discutez et échangez autour de la configuration avec votre équipe informatique chargée de l'exploitation.

## 2. L'environnement de développement

L'environnement de développement n'est pas soumis aux contraintes de haute disponibilité. Il s'agit toutefois de réaliser un environnement robuste et performant afin que l'équipe de développement puisse travailler avec tout le confort nécessaire.

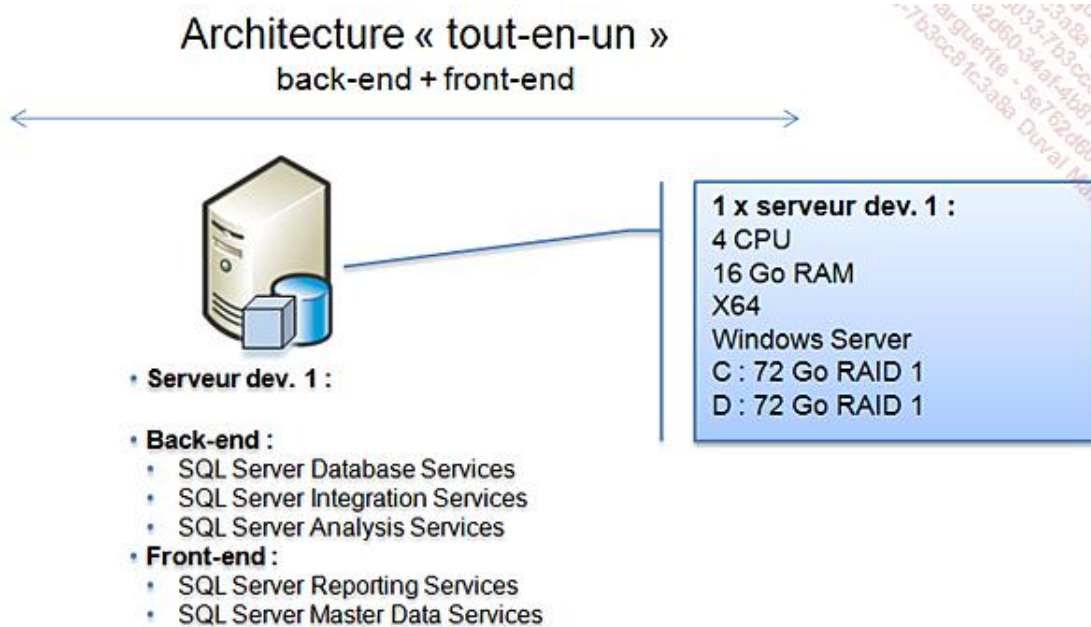
Afin d'améliorer la productivité de vos équipes, nous vous engageons à investir dans un outil de contrôle de code source. Cet outil vous permettra de partager vos projets, historiser les changements, tracer les modifications et permettre de réaliser une sauvegarde régulière et quotidienne de l'ensemble.

➤ Dans la solution Microsoft, le serveur et l'outil de contrôle de code source est Team Foundation Server et Visual Studio Team System.

En termes d'architecture serveur, une architecture tout-en-un comprenant l'ensemble des serveurs présents en production est suffisante. Prévoyez tout simplement une bonne machine.

À titre d'exemple, nous vous proposons cette configuration, qui devra elle aussi être adaptée à votre contexte (au

nombre de développeurs, au nombre de projets...) :



*Exemple de configuration pour un environnement de développement*

### 3. Les environnements de recette et d'intégration

En termes de configuration, l'environnement de recette n'a pas de contrainte particulière et peut ressembler par exemple à l'environnement de développement.

L'environnement d'intégration quant à lui devra si possible ressembler autant que possible à l'environnement de production pour y intégrer les mêmes contraintes.

En termes de budget, cela peut revenir bien évidemment très cher. Un compromis moyen peut être trouvé en considérant au moins un environnement de recette/intégration avec une architecture de production sans la haute disponibilité.

Tout est affaire de budget et de négociation avec l'équipe chargée de l'exploitation de vos machines.