

**PRODUIT OFFICIEL DE FORMATION MICROSOFT**

**22462A**

**Administration de bases de données  
Microsoft® SQL Server® 2012**

**Volume 1**

Les informations contenues dans ce document, notamment les URL et les autres références aux sites Web, pourront faire l'objet de modifications sans préavis. Sauf mention contraire, les sociétés, produits, noms de domaines, adresses de messagerie, logos, personnes, lieux et événements utilisés dans les exemples sont fictifs et toute ressemblance avec des sociétés, produits, noms de domaines, adresses de messagerie, logos, personnes, lieux et événements réels est purement fortuite et involontaire. L'utilisateur est tenu d'observer la réglementation relative aux droits d'auteur applicable dans son pays. Aucune partie de ce document ne peut être reproduite, stockée ou introduite dans un système de restitution, ou transmise à quelque fin ou par quelque moyen que ce soit (électronique, mécanique, photocopie, enregistrement ou autre) sans la permission expresse et écrite de Microsoft Corporation.

Microsoft peut détenir des brevets, avoir déposé des demandes d'enregistrement de brevets ou être titulaire de marques, droits d'auteur ou autres droits de propriété intellectuelle portant sur tout ou partie des éléments qui font l'objet du présent document. Sauf stipulation expresse contraire d'un contrat de licence écrit de Microsoft, la fourniture de ce document n'a pas pour effet de vous concéder une licence sur ces brevets, marques, droits d'auteur ou autres droits de propriété intellectuelle.

Les noms de fabricants, de produits ou les URL sont fournis uniquement à titre indicatif et Microsoft ne fait aucune déclaration et exclut toute garantie légale, expresse ou implicite, concernant ces fabricants ou l'utilisation des produits avec toutes les technologies Microsoft. L'inclusion d'un fabricant ou produit n'implique pas l'approbation par Microsoft du fabricant ou du produit. Des liens vers des sites Web tiers peuvent être fournis. Ces sites ne sont pas sous le contrôle de Microsoft et Microsoft n'est pas responsable de leur contenu ni des liens qu'ils sont susceptibles de contenir, ni des modifications ou mises à jour de ces sites. Microsoft n'est pas responsable de la diffusion Web ou de toute autre forme de transmission reçue d'un site connexe. Microsoft fournit ces liens pour votre commodité, et l'insertion de n'importe quel lien n'implique pas l'approbation du site en question ou des produits qu'il contient par Microsoft.

© 2013 Microsoft Corporation. Tous droits réservés.

Microsoft et les marques commerciales figurant sur la page  
<http://www.microsoft.com/about/legal/en/us/IntellectualProperty/Trademarks/EN-US.aspx>  
sont des marques commerciales du groupe de sociétés Microsoft. Toutes les autres marques sont la propriété de leurs propriétaires respectifs.

Numéro de produit : 22462A

Numéro de référence : X18-84729

Date de publication : 02/2013

## TERMES DU CONTRAT DE LICENCE MICROSOFT

### COURS OFFICIELS DE FORMATION MICROSOFT – ÉDITION STAGIAIRE – Versions précommerciales et finales

Les présents termes ont valeur de contrat entre Microsoft Corporation et vous. Lisez-les attentivement. Ils portent sur le contenu sous licence visé ci-dessus, y compris le support sur lequel vous l'avez reçu, le cas échéant. Ce contrat porte également sur les produits Microsoft suivants :

- les mises à jour,
- les suppléments,
- les services Internet et
- les services d'assistance

de ce contenu sous licence à moins que d'autres termes n'accompagnent ces produits, auquel cas ces derniers prévalent.

**En utilisant le contenu sous licence, vous acceptez ces termes. Si vous êtes en désaccord avec ces termes, n'utilisez pas le contenu sous licence.**

**Si vous vous conformez aux présentes conditions de licence, vous disposez des droits stipulés ci-dessous.**

#### 1. PRÉSENTATION.

**Contenu sous licence.** Le contenu sous licence inclut le logiciel, la documentation imprimée, la documentation de formation (en ligne et sous forme électronique) et les supports associés.

**Modèle de licence.** Le contenu sous licence est concédé sous licence en vertu d'une licence par copie et par dispositif.

#### 2. INSTALLATION ET DROITS D'UTILISATION.

- a. **Dispositif concédé sous licence.** Le dispositif concédé sous licence est celui sur lequel vous utilisez le contenu sous licence. Vous êtes autorisé à installer et à utiliser une copie du contenu sous licence sur le dispositif sous licence.
- b. **Dispositif portable.** Vous êtes autorisé à installer une autre copie sur un dispositif portable à des fins d'utilisation par le seul utilisateur principal du dispositif concédé sous licence.
- c. **Dissociation de composants.** Les composants du contenu sous licence sont concédés sous licence en tant qu'unité unique. Vous n'êtes pas autorisé à dissocier les composants et à les installer sur différents dispositifs.
- d. **Programmes de tiers.** Le contenu sous licence peut également contenir des programmes tiers. L'utilisation de ces programmes tiers sera régie par les présents termes du contrat de licence, à moins que d'autres termes n'accompagnent ces programmes.

#### 3. VERSIONS PRÉCOMMERCIALES.

Si le contenu sous licence est une version précommerciale (« version bêta »), les présents termes s'appliquent en plus des termes de ce contrat :

- a. **Contenu sous licence en version précommerciale.** Ce contenu sous licence est une version précommerciale. Il peut ne pas contenir les mêmes informations et/ou ne pas fonctionner comme une version finale du contenu sous licence. Nous sommes autorisés à le changer pour la version commerciale finale. Nous sommes également autorisés à ne pas éditer de version commerciale. Vous devez informer clairement et de façon visible les Stagiaires qui participent à une Session de formation agréée et l'Instructeur qui dispense une Session de formation agréée de ce qui précède ; et vous ou Microsoft n'avez aucune obligation de leur fournir un contenu supplémentaire, notamment, de manière non limitative, la version finale du Contenu sous licence du Cours.
- b. **Commentaires.** Si vous acceptez de faire part à Microsoft de vos commentaires concernant le contenu sous licence, vous concédez à Microsoft, gratuitement, le droit d'utiliser, de partager et de commercialiser vos commentaires de quelque manière et à quelque fin que ce soit. Vous concédez également à des tiers, gratuitement, les droits de brevet nécessaires pour que leurs produits, technologies et services puissent être utilisés ou servir d'interface avec toute partie spécifique d'un logiciel, contenu sous licence ou service Microsoft qui inclut ces commentaires. Vous ne fournirez pas de commentaires faisant l'objet d'une licence qui impose à Microsoft de concéder sous licence son logiciel ou sa documentation à des tiers parce que nous y incluons vos commentaires. Ces droits survivent au présent contrat.

- c. **Informations confidentielles.** Le contenu sous licence, y compris la visionneuse, l'interface utilisateur, les fonctionnalités et la documentation pouvant être incluses dans le contenu sous licence, est confidentiel et la propriété de Microsoft et de ses fournisseurs.
- i. **Utilisation.** Pendant cinq ans après l'installation du contenu sous licence ou de sa commercialisation, selon la date la plus proche, vous n'êtes pas autorisé à divulguer des informations confidentielles à des tiers. Vous êtes autorisé à divulguer des informations confidentielles uniquement à vos employés et consultants qui en ont besoin. Vous devez avoir conclu avec eux des accords écrits qui protègent les informations confidentielles au moins autant que le présent contrat.
- ii. **Maintien en vigueur de certaines clauses.** Votre obligation de protection des informations confidentielles survit au présent contrat.
- iii. **Exclusions.** Vous êtes autorisé à divulguer des informations confidentielles conformément à une ordonnance judiciaire ou gouvernementale. Vous devez en informer par avance Microsoft afin de lui permettre de demander une ordonnance protectrice ou de trouver un autre moyen de protéger ces informations. Les informations confidentielles n'incluent pas les informations
- qui ont été portées à la connaissance du public sans qu'il y ait eu violation de l'obligation de confidentialité ;
  - que vous avez reçues d'un tiers qui n'a pas violé ses obligations de confidentialité à l'égard de Microsoft ou de ses fournisseurs ; ou
  - que vous avez développées en toute indépendance.
- d. **Durée.** Le présent contrat pour les versions précommerciales est applicable jusqu'à (i) la date d'expiration qui vous est communiquée par Microsoft pour l'utilisation de la version bêta, ou (ii) la commercialisation de la version finale du contenu sous licence, selon la date la plus proche (la « durée de la bêta »).
- e. **Utilisation.** Dès l'expiration ou la résiliation de la durée de la bêta, vous devrez cesser d'utiliser les copies de la version bêta et détruire toutes les copies en votre possession ou sous votre contrôle.
- f. **Copies.** Microsoft informera les Centres de formation agréés s'ils sont autorisés à effectuer des copies de la version bêta (version imprimée et/ou sur CD) et à les distribuer aux Stagiaires et/ou Instructeurs. Si Microsoft autorise cette distribution, vous devrez vous conformer aux termes supplémentaires fournis par Microsoft concernant lesdites copies et distribution.

4. **CONDITIONS DE LICENCE ET/OU DROITS D'UTILISATION SUPPLEMENTAIRES.**

- a. **Éléments multimédias et modèles.** Vous pouvez utiliser des photographies, images clip art, animations, sons, musiques, formes, clips vidéo et modèles inclus dans le contenu sous licence uniquement pour les besoins de votre formation personnelle. Si vous souhaitez utiliser les Éléments multimédias ou modèles à toute autre fin, accédez au site [www.microsoft.com/permission](http://www.microsoft.com/permission) pour savoir si leur utilisation est autorisée.
- b. **Documentation de formation.** Si le contenu sous licence inclut une documentation de formation (comme les livres blancs, les ateliers, les tests, les feuilles de données et les FAQ), vous êtes autorisé à copier et à utiliser cette documentation. Vous n'êtes pas autorisé à modifier la documentation de formation ni à imprimer un ouvrage (version électronique ou imprimée) dans son intégralité. Si vous reproduisez une documentation de formation, vous acceptez ces termes :
- Vous pouvez utiliser la documentation de formation aux seules fins de votre utilisation ou formation personnelle.
  - Vous ne pouvez pas rééditer ni publier la documentation de formation sur un ordinateur du réseau, ni la diffuser sur un support.
  - Vous devez ajouter la mention de droits d'auteur d'origine de la documentation de formation ou celle de Microsoft sous la forme ci-dessous :

**Forme de mention :**

© 2011 Réimprimé avec l'autorisation de Microsoft Corporation pour usage personnel uniquement. Tous droits réservés.

Microsoft et Windows sont soit des marques de Microsoft Corporation, soit des marques déposées de Microsoft Corporation aux États-Unis d'Amérique et/ou dans d'autres pays. Les autres noms de produits et de sociétés mentionnés dans les présentes sont des marques de leurs propriétaires respectifs.

c. **Code distribuable.** Le contenu sous licence peut contenir du code que vous êtes autorisé à distribuer dans des programmes que vous développez, sous réserve des conditions ci-après.

i. **Droit d'utilisation et de distribution.** Le code et les fichiers texte répertoriés ci-après constituent le « Code distribuable ».

- Fichiers REDIST.TXT. Vous êtes autorisé à copier et distribuer la version en code objet du code répertorié dans les fichiers REDIST.TXT.
- Code échantillon. Vous êtes autorisé à modifier, copier et distribuer les versions en code source et objet du « code échantillon ».
- Distribution par des tiers. Vous pouvez autoriser les distributeurs de vos programmes à copier et à distribuer le Code distribuable en tant que partie intégrante de ces programmes.

ii. **Conditions de distribution.** Pour tout Code distribuable que vous distribuez, vous devez :

- y ajouter des fonctionnalités importantes et principales au sein de vos programmes,
- exiger que les distributeurs et les utilisateurs finaux externes acceptent les conditions qui protègent le Code distribuable au moins autant que le présent contrat,
- afficher votre propre mention de droits d'auteur valable sur vos programmes et
- garantir et défendre Microsoft contre toute réclamation, y compris pour les honoraires d'avocats, qui résulterait de la distribution ou l'utilisation de vos programmes.

iii. **Restrictions de distribution.** Vous n'êtes pas autorisé à :

- modifier toute mention de droits d'auteur, de marques ou de droits de propriété industrielle pouvant figurer dans le Code distribuable,
- utiliser les marques de Microsoft dans les noms de vos programmes ou d'une façon qui suggère que vos programmes sont fournis ou recommandés par Microsoft,
- distribuer le Code distribuable en vue de son exécution sur une plateforme autre que la plateforme Windows,
- inclure le Code distribuable dans des programmes malveillants, trompeurs ou interdits par la loi, ou
- modifier ou distribuer le code source de tout Code distribuable de manière à ce qu'il fasse l'objet, en tout ou partie, d'une Licence Exclue. Une Licence Exclue implique comme condition d'utilisation, de modification ou de distribution, que :
  - le code soit divulgué ou distribué sous forme de code source, ou
  - d'autres aient le droit de le modifier.

5. **SERVICES INTERNET.** Microsoft peut fournir des services Internet avec le contenu sous licence. Ils peuvent être modifiés ou interrompus à tout moment. Vous n'êtes pas autorisé à utiliser ces services de quelque manière que ce soit qui pourrait leur porter atteinte ou perturber leur utilisation par un autre utilisateur. Vous n'êtes pas autorisé à tenter d'accéder de façon non autorisée aux services, données, comptes ou réseaux de toute autre manière.

6. **PORTEE DE LA LICENCE.** Le contenu sous licence n'est pas vendu, mais concédé sous licence. Le présent contrat vous confère certains droits d'utilisation du contenu sous licence. Microsoft se réserve tous les autres droits. Sauf si la loi en vigueur vous confère d'autres droits, nonobstant la présente limitation, vous n'êtes autorisé à utiliser le contenu sous licence qu'en conformité avec les termes du présent contrat. À cette fin, vous devez vous conformer aux restrictions techniques contenues dans le contenu sous licence qui vous permettent de l'utiliser d'une certaine façon. Vous n'êtes pas autorisé à :

- révéler à des tiers les résultats des tests d'évaluation du contenu sous licence sans l'accord écrit préalable de Microsoft ;
- contourner les restrictions techniques figurant dans le contenu sous licence ;
- reconstituer la logique du contenu sous licence, le décompiler ou le désassembler, sauf dans la mesure où ces opérations seraient expressément permises par la réglementation applicable nonobstant la présente limitation ;
- effectuer plus de copies du contenu sous licence que ce qui n'est autorisé dans le présent contrat ou par la réglementation applicable, nonobstant la présente limitation ;
- publier le contenu sous licence en vue d'une reproduction par autrui ;

- transférer le contenu sous licence portant la mention de « version bêta » ou « version précommerciale » à un tiers ;
  - autoriser d'autres utilisateurs à accéder au contenu sous licence ou à l'utiliser ;
  - louer ou prêter le contenu sous licence ; ou
  - utiliser le contenu sous licence pour des services d'hébergement commercial ;
  - Les droits d'accès au logiciel serveur pouvant être inclus avec le contenu sous licence, y compris les Disques durs virtuels, ne vous autorisent pas à exploiter des brevets appartenant à Microsoft ou tous autres droits de propriété intellectuelle de Microsoft sur le logiciel ou tous dispositifs qui peuvent accéder au serveur.
- 7. COPIE DE SAUVEGARDE.** Vous êtes autorisé à effectuer une copie de sauvegarde du contenu sous licence. Vous ne pouvez l'utiliser que dans le but de réinstaller le contenu sous licence.
- 8. TRANSFERT À UN AUTRE DISPOSITIF.** Vous êtes autorisé à désinstaller le contenu sous licence et à l'installer sur un autre dispositif pour les besoins de votre formation personnelle. Vous n'êtes pas autorisé à procéder de la sorte afin de partager cette licence entre plusieurs dispositifs.
- 9. TRANSFERT À UN TIERS.** Vous n'êtes pas autorisé à transférer des versions portant la mention « version bêta » ou « version précommerciale » à un tiers. Pour les versions finales, les termes suivants s'appliquent : Le premier utilisateur du contenu sous licence doit le transférer, ainsi que le présent contrat, directement à un tiers. Avant le transfert, le tiers doit reconnaître que cet accord s'applique au transfert et à l'utilisation du contenu sous licence. Le premier utilisateur doit désinstaller le contenu sous licence avant de le transférer séparément du dispositif. Le premier utilisateur n'est pas autorisé à en conserver une copie.
- 10. RESTRICTIONS À L'EXPORTATION.** Le contenu sous licence est soumis à la réglementation américaine en matière d'exportation. Vous devez vous conformer à toutes les réglementations nationales et internationales en matière d'exportation concernant le contenu sous licence. Ces réglementations comprennent des restrictions sur les pays destinataires, les utilisateurs finaux et les utilisations finales. Des informations supplémentaires sont disponibles sur le site [www.microsoft.com/exporting](http://www.microsoft.com/exporting).
- 11. LOGICIEL/CONTENU SOUS LICENCE EN REVENTE INTERDITE (« NOT FOR RESALE » OU « NFR »).** Vous n'êtes pas autorisé à vendre un logiciel ou un contenu sous licence portant la mention de revente interdite (« Not for Resale » ou « NFR »).
- 12. VERSION ÉDUCATION.** Pour utiliser un contenu sous licence portant la mention de Version Éducation (« Academic Edition » ou « AE »), vous devez avoir la qualité d'« Utilisateur Éducation Autorisé » (Qualified Educational User). Pour savoir si vous avez cette qualité, rendez-vous sur le site <http://www.microsoft.com/france/licences/education> ou contactez l'affilié Microsoft qui dessert votre pays.
- 13. INTÉGRALITÉ DES ACCORDS.** Le présent contrat ainsi que les termes concernant les suppléments, les mises à jour, les services Internet et d'assistance technique que vous utilisez constituent l'intégralité des accords en ce qui concerne le contenu sous licence et les services d'assistance technique.
- 14. DROIT APPLICABLE.**
- a. **États-Unis.** Si vous avez acquis le contenu sous licence aux États-Unis, les lois de l'État de Washington, États-Unis d'Amérique, régissent l'interprétation de ce contrat et s'appliquent en cas de réclamation pour rupture dudit contrat, sans donner d'effet aux dispositions régissant les conflits de lois. Les lois du pays dans lequel vous vivez régissent toutes les autres réclamations, notamment les réclamations fondées sur les lois fédérales en matière de protection des consommateurs, de concurrence déloyale et de délits.
  - b. **En dehors des États-Unis.** Si vous avez acquis le contenu sous licence dans un autre pays, les lois de ce pays s'appliquent.
- 15. EFFET JURIDIQUE.** Le présent contrat décrit certains droits légaux. Vous pouvez bénéficier d'autres droits prévus par les lois de votre État ou pays. Vous pouvez également bénéficier de certains droits à l'égard de la partie auprès de laquelle vous avez acquis le contrat sous licence. Le présent contrat ne modifie pas les droits que vous confèrent les lois de votre État ou pays si celles-ci ne le permettent pas.
- 16. EXCLUSIONS DE GARANTIE. LE CONTENU SOUS LICENCE EST CONCÉDÉ SOUS LICENCE « EN L'ÉTAT ». VOUS ASSUMEZ TOUS LES RISQUES LIÉS À SON UTILISATION. MICROSOFT N'ACCORDE AUCUNE GARANTIE OU CONDITION EXPRESSE. VOUS POUVEZ BENEFICIER DE DROITS DES CONSOMMATEURS SUPPLEMENTAIRES DANS LE CADRE DU DROIT LOCAL, QUE CE CONTRAT NE PEUT MODIFIER. LORSQUE CELA EST AUTORISE PAR LE DROIT LOCAL, MICROSOFT EXCLUT LES GARANTIES IMPLICITES DE QUALITE, D'ADEQUATION A UN USAGE PARTICULIER ET D'ABSENCE DE CONTREFAÇON.**

**17. LIMITATION ET EXCLUSION DE RECOURS ET DE DOMMAGES. VOUS POUVEZ OBTENIR DE MICROSOFT ET DE SES FOURNISSEURS UNE INDEMNISATION EN CAS DE DOMMAGES DIRECTS LIMITÉE À 5,00 \$ U.S. VOUS NE POUVEZ PRÉTENDRE À AUCUNE INDEMNISATION POUR LES AUTRES DOMMAGES, Y COMPRIS LES DOMMAGES INDIRECTS, DE PERTES DE BÉNÉFICES, SPÉCIAUX OU ACCESSOIRES.**

Cette limitation concerne :

- toute affaire liée au contenu sous licence, au logiciel, aux services ou au contenu (y compris le code) figurant sur des sites Internet tiers ou dans des programmes tiers ; et
- les réclamations pour rupture de contrat ou violation de garantie, les réclamations en cas de responsabilité sans faute, de négligence ou autre délit dans la limite autorisée par la loi en vigueur.

Elle s'applique également même si Microsoft connaissait l'éventualité d'un tel dommage. La limitation ou exclusion ci-dessus peut également ne pas vous être applicable, car votre pays n'autorise pas l'exclusion ou la limitation de responsabilité pour les dommages indirects, accessoires ou de quelque nature que ce soit.

# Bienvenue !

Merci de suivre notre formation. En collaboration avec nos sites Microsoft Certified Partners for Learning Solutions et nos centres Microsoft IT Academy, nous avons élaboré des formations de premier plan aussi bien destinées aux informaticiens souhaitant approfondir leurs connaissances qu'aux étudiants se destinant à une carrière informatique.

- **Formateurs et instructeurs Microsoft Certified**—Votre instructeur possède des compétences techniques et pédagogiques. Il répond aux exigences actuelles en matière de certification. En outre, si les instructeurs dispensent des formations sur l'un de nos sites Certified Partners for Learning Solutions, ils sont également évalués tout au long de l'année par les stagiaires et par Microsoft.
- **Avantages des examens de certification**—À l'issue d'une formation, pensez aux examens de certification Microsoft. Les certifications Microsoft valident vos compétences en matière de technologies Microsoft et peuvent faire la différence lors d'une recherche d'emploi ou pour faire progresser votre carrière. Une étude IDC indépendante a conclu que pour 75 % des responsables, les certifications sont importantes pour les performances des équipes<sup>1</sup>. Renseignez-vous auprès de votre instructeur pour connaître les promotions et remises auxquels vous pourriez avoir droit sur les examens de certification Microsoft.
- **Garantie de satisfaction du client**—Nos Certified Partners for Learning Solutions offrent une garantie de satisfaction et engagent leur responsabilité à ce sujet. À la fin du cours, nous vous demandons de bien vouloir remplir un formulaire d'évaluation sur votre expérience du jour. Vos commentaires sont les bienvenus !

Nous vous souhaitons une agréable formation et une carrière couronnée de succès.

Cordialement,

Microsoft Learning  
[www.microsoft.com/france/formation](http://www.microsoft.com/france/formation)



<sup>1</sup> IDC, Value of Certification: Team Certification and Organizational Performance, novembre 2006

# Remerciements

Formation Microsoft souhaite reconnaître la contribution apportée par les personnes citées ci-dessous à l'élaboration de ce titre et les en remercier. Elles ont en effet déployé des efforts aux différents stades de ce processus pour vous proposer une expérience de qualité en classe.

## Conception et développement

Ce cours a été conçu et développé par SolidQ. SolidQ est un prestataire de services global en expertise-conseil, tutorat et formation pour les plateformes de gestion des données, de business intelligence et de collaboration de Microsoft.

### Greg Low – Développeur senior

Le Dr Greg Low est certifié MVP pour SQL Server et certifié MCT, et il est directeur régional de Microsoft pour l'Australie. Il a travaillé avec SQL Server depuis la version 4.2 en tant que mentor, consultant et instructeur actifs. Il se décrit lui-même comme un « accro » de SQL Server et comme quelqu'un qui travaille dans le domaine du développement depuis que les dinosaures vivaient sur la Terre. Il a été instructeur dans le programme de certification Microsoft SQL Server Masters pendant plusieurs années et a été l'une des deux premières personnes à obtenir la certification SQL Server 2008 Master. Il est l'auteur de plusieurs livres blancs sur les sites Web Microsoft MSDN et TechNet et l'auteur de plusieurs ouvrages sur SQL Server. Il est basé à Melbourne, en Australie.

### Herbert Albert – Développeur de cours

Herbert Albert a débuté sa carrière en 1994. Il travaille comme instructeur, consultant et auteur dans le domaine des technologies SQL Server. Il exerce le rôle de mentor et est le PDG de SolidQ pour l'Europe centrale. Il est basé à Vienne, en Autriche. Il a plusieurs certifications Microsoft, notamment la certification MCT qu'il a obtenue depuis 1997. Il intervient régulièrement dans des conférences et est co-auteur des documents SQL Server 2012 Upgrade Technical Reference Guide et SQL Server 2005 Step-by-Step Applied Techniques. Avec Gianluca Hotz, il écrit une chronique régulière dans le journal de SolidQ.

### Marque Hions – Réviseur technique

La passion de Mark pour l'informatique et ses compétences de communicateur étaient parfaitement adaptées à son poste d'instructeur à Honeywell, au Canada, où il a commencé à travailler avec des mini-ordinateurs, des mainframes et des stagiaires matures en 1984. Il a rencontré Microsoft SQL Server pour la première fois quand il s'exécutait sur le système d'exploitation OS/2, et a dispensé depuis des formations sur chaque version. Certifié MCT et consultant indépendant depuis de nombreuses années, il est un présentateur très apprécié à TechEd, a conçu les examens SQL Server pour Microsoft et a donné des cours approfondis via le Microsoft Partner Channel. Il exerce maintenant le rôle d'instructeur et de consultant SQL Server à DesTech, qui est le plus grand fournisseur de formations SQL Server dans la région de Toronto.

### Chris Barker – Réviseur technique

Chris Barker est un formateur MCT qui intervient sur le marché de la Nouvelle-Zélande et actuellement employé comme formateur du personnel chez Auldhouse, l'un des centres de formation les plus importants du programme CPLS (Certified Partner for Learning Solutions) de Nouvelle-Zélande, basé à Wellington. Chris possède une expérience de programmation qui remonte au début des années 70 ; il a écrit son premier programme en langage assembleur et a effectué du débogage en binaire (au sens propre du terme). S'il se concentre plus particulièrement sur les formations de programmation (.NET pour la plupart) et de bases de données (Microsoft SQL Server en particulier), Chris est également formateur en infrastructures et détient les certifications réseau Novell et Microsoft.



# Sommaire

## Module 1 : Présentation de SQL Server 2012 et de son ensemble d'outils

Leçon 1 : Présentation de la plateforme SQL Server	1-3
Leçon 2 : Utilisation des outils SQL Server	1-15
Leçon 3 : Configuration de Services SQL Server	1-30
Atelier pratique 1 : Présentation de SQL Server et de son ensemble d'outils	1-38

## Module 2 : Préparation des systèmes pour SQL Server 2012

Leçon 1 : Vue d'ensemble de l'architecture SQL Server	2-3
Leçon 2 : Planification des besoins en ressources serveur	2-17
Leçon 3 : Tests de pré-installation pour SQL Server	2-29
Atelier pratique 2 : Préparation des systèmes pour SQL Server	2-35

## Module 3 : Installation et configuration de SQL Server 2012

Leçon 1 : Préparation pour l'installation de SQL Server	3-3
Leçon 2 : Installation de SQL Server	3-16
Leçon 3 : Mise à niveau et automatisation de l'installation	3-25
Atelier pratique 3 : Installation et configuration de SQL Server	3-33

## Module 4 : Utilisation des bases de données

Leçon 1 : Vue d'ensemble des bases de données SQL Server	4-3
Leçon 2 : Travailler avec des fichiers et des groupes de fichiers	4-15
Leçon 3 : Déplacer les fichiers de base de données	4-30
Atelier pratique 4 : Utilisation des bases de données	4-40

## Module 5 : Description des modes de récupération SQL Server 2012

Leçon 1 : Stratégies de sauvegarde	5-3
Leçon 2 : Description de la journalisation des transactions SQL Server	5-12
Leçon 3 : Planification d'une stratégie de sauvegarde SQL Server	5-24
Atelier pratique 5 : Description des modes de récupération SQL Server	5-34

## Module 6 : Sauvegarde des bases de données SQL Server 2012

Leçon 1 : Sauvegarde des bases de données et des journaux de transactions	6-3
Leçon 2 : Gestion des sauvegardes de base de données	6-15
Leçon 3 : Utilisation des options de sauvegarde	6-22
Atelier pratique 6 : Sauvegarde des bases de données SQL Server	6-28

**Module 7 : Restauration des bases de données SQL Server 2012**

Leçon 1 : Présentation du processus de restauration	7-4
Leçon 2 : Restauration des bases de données	7-10
Leçon 3 : Utilisation de la récupération limitée dans le temps	7-21
Leçon 4 : Restauration des bases de données système et des fichiers individuels	7-29
Atelier pratique 7 : Restauration des bases de données SQL Server 2012	7-37

**Module 8 : Importation et exportation de données**

Leçon 1 : Transfert des données vers/de SQL Server	8-3
Leçon 2 : Importation et exportation des données de table	8-15
Leçon 3 : Insertion de données en bloc	8-20
Atelier pratique 8 : Importation et exportation de données	8-29

**Module 9 : Authentification et autorisation des utilisateurs**

Leçon 1 : Authentification des connexions à SQL Server	9-3
Leçon 2 : Autorisation des connexions pour accéder aux bases de données	9-13
Leçon 3 : Autorisation sur plusieurs serveurs	9-23
Atelier pratique 9 : Authentification et autorisation des utilisateurs	9-31

**Module 10 : Attribution de rôles de serveur et de base de données**

Leçon 1 : Utilisation de rôles serveur	10-3
Leçon 2 : Utilisation de rôles de base de données fixes	10-12
Leçon 3 : Création de rôles de base de données définis par l'utilisateur	10-18
Atelier pratique 10 : Attribution de rôles de serveur et de base de données	10-26

**Annexe : Corrigés des ateliers pratiques**

Atelier pratique du module 1 : Présentation de SQL Server 2012 et de son ensemble d'outils	L1-1
Atelier pratique du module 2 : Préparation des systèmes pour SQL Server	L2-7
Atelier pratique du module 3 : Installation et configuration de SQL Server	L3-13
Atelier pratique du module 4 : Utilisation des bases de données	L4-23
Atelier pratique du module 5 : Description des modes de récupération SQL Server	L5-31
Atelier pratique du module 6 : Sauvegarde des bases de données SQL Server	L6-35
Atelier pratique du module 7 : Restauration des bases de données SQL Server 2012	L7-43
Atelier pratique du module 8 : Importation et exportation de données	L8-47
Atelier pratique du module 9 : Authentification et autorisation des utilisateurs	L9-53
Atelier pratique du module 10 : Attribution de rôles de serveur et de base de données	L10-59

# À propos de ce cours

Cette section décrit brièvement le cours et ses objectifs, le public visé, ainsi que les connaissances préalables requises.

## Description du cours

Ce cours dirigé par un instructeur et réparti sur cinq journées permet aux stagiaires d'acquérir les connaissances et les compétences nécessaires à l'administration de bases de données Microsoft® SQL Server® 2012. L'objectif de ce cours consiste à apprendre aux stagiaires à utiliser les fonctionnalités et outils SQL Server 2012 permettant d'assurer la maintenance d'une base de données.

## Public visé

Ce cours est destiné en priorité aux personnes chargées de l'administration et de la maintenance des bases de données SQL Server. Il peut également être utile aux personnes qui développent des applications qui fournissent du contenu provenant de bases de données SQL Server.

## Connaissances préalables

Pour suivre ce cours, vous devez disposer de connaissances pratiques dans les domaines suivants :

- avoir des connaissances de base du système d'exploitation Microsoft Windows et de ses fonctionnalités principales ;
- avoir une expérience professionnelle de Transact-SQL ;
- avoir une expérience pratique des bases de données relationnelles ;
- avoir de l'expérience dans la conception de bases de données ;
- avoir suivi le cours 22461 : Interrogation de Microsoft SQL Server 2012.

## Objectifs du cours

À la fin de ce cours, les stagiaires seront à même d'effectuer les tâches suivantes :

- expliquer l'architecture de SQL Server 2012 et les ressources requises et vérifier les sous-systèmes d'E/S avant l'installation ;
- planifier, installer et configurer SQL Server 2012 ;
- sauvegarder et restaurer des bases de données ;
- importer et exporter des assistants et expliquer leur relation avec SSIS ;
- utiliser BCP et BULK INSERT pour importer des données ;
- gérer la sécurité ;
- attribuer et configurer des rôles de base de données fixes et créer et attribuer des rôles de base de données définis par l'utilisateur ;
- configurer et attribuer des autorisations ;
- implémenter des audits SQL Server 2012 ;

- gérer SQL Server 2012 Agent et les travaux ;
- configurer la messagerie de base de données, les alertes et les notifications ;
- assurer la maintenance des bases de données ;
- configurer le Générateur de profils SQL et Trace SQL et utiliser l'Assistant Paramétrage du moteur de base de données ;
- surveiller les données à l'aide des vues de gestion dynamique ;
- exécuter des requêtes multiserveurs et configurer un serveur de gestion centralisée ;
- déployer une application de la couche Données ;
- résoudre des problèmes courants.

## Plan du cours

Cette section présente le plan du cours :

**Module 1** : Présentation de SQL Server 2012 et de son ensemble d'outils

**Module 2** : Préparation des systèmes pour SQL Server 2012

**Module 3** : Installation et configuration de SQL Server 2012

**Module 4** : Utilisation des bases de données

**Module 5** : Description des modes de récupération SQL Server 2012

**Module 6** : Sauvegarde des bases de données SQL Server 2012

**Module 7** : Restauration des bases de données SQL Server 2012

**Module 8** : Importation et exportation de données

**Module 9** : Authentification et autorisation des utilisateurs

**Module 10** : Attribution de rôles de serveur et de base de données

**Module 11** : Autorisation de l'accès des utilisateurs aux ressources

**Module 12** : Audit des environnements SQL Server

**Module 13** : Automatisation de la gestion de SQL Server 2012

**Module 14** : Configuration de la sécurité pour SQL Server Agent

**Module 15** : Surveillance de SQL Server 2012 avec alertes et notifications

**Module 16** : Exécution de la maintenance régulière des bases de données

**Module 17** : Trace de l'accès à SQL Server 2012

**Module 18** : Surveillance de SQL Server 2012

**Module 19** : Gestion de plusieurs serveurs

**Module 20** : Résolution des problèmes courants liés à l'administration de SQL Server 2012

## Documents de cours

Votre kit de cours contient les documents suivants :

- **Manuel du cours.** Guide de formation succinct qui fournit toutes les informations techniques importantes dans un format concis et très ciblé, parfaitement adapté à l'apprentissage en classe.
- **Leçons** : vous guident dans les objectifs de formation et fournissent les points clés essentiels pour un apprentissage en classe réussi.
- **Ateliers pratiques** : fournissent une plateforme qui vous permettra de mettre en application les connaissances et compétences acquises dans le module.
- **Contrôles des acquis et éléments à retenir** : fournissent une documentation de référence pratique qui favorise la mémorisation des connaissances et compétences.
- **Corrigés des ateliers pratiques** : fournissent des instructions pas à pas que vous pourrez consulter à tout moment au cours d'un atelier pratique.
- **Leçons** : incluent des informations détaillées pour chaque rubrique en complément du contenu du Manuel du cours.
- **Ateliers pratiques** : incluent des informations exhaustives sur les exercices et les corrigés des ateliers pratiques sous forme numérique, à utiliser pendant les ateliers pratiques.
- **Ressources** : incluent des ressources supplémentaires présentées par catégories qui vous donnent un accès immédiat à du contenu utile et à jour disponible sur TechNet, MSDN, Microsoft Press
- **Fichiers de cours destinés aux stagiaires** : incluent le fichier exécutable à extraction automatique Allfiles.exe, lequel contient les fichiers requis pour les ateliers pratiques et les démonstrations.
- **Évaluation du cours.** À la fin du cours, vous aurez l'occasion de remplir une fiche d'évaluation en ligne pour faire part de vos commentaires sur le cours, le centre de formation et l'instructeur.

Pour adresser d'autres commentaires ou remarques sur le cours, envoyez un message électronique à l'adresse support@mscourseware.com. Pour obtenir des informations sur le programme MCP (Microsoft Certification Program), envoyez un message électronique à l'adresse mchelp@microsoft.com.

## Environnement d'ordinateurs virtuels

Cette section fournit les informations nécessaires pour configurer l'environnement de la classe afin de prendre en charge le scénario d'entreprise du cours.

### Configuration des ordinateurs virtuels

Dans ce cours, vous allez utiliser Microsoft Hyper-V pour effectuer les ateliers pratiques.

Le tableau suivant montre le rôle de chaque ordinateur virtuel utilisé dans ce cours :

Ordinateur virtuel	Rôle
1077XA-MIA-DC	Contrôleur de domaine
1077XA-MIA-SQL	Ordinateur virtuel SQL Server pour les modules 1 à 20

### Configuration logicielle

Le logiciel suivant est installé sur chaque ordinateur virtuel :

- SQL Server 2012 (sur les ordinateurs virtuels SQL Server)

### Fichiers du cours

Des fichiers sont associés aux ateliers pratiques de ce cours. Ces fichiers des ateliers se trouvent dans le dossier D:\22462A\_Labs de l'ordinateur de chaque stagiaire.

### Configuration de la classe

L'ordinateur virtuel sera configuré de la même façon sur tous les ordinateurs de la classe.

### Niveau des éléments matériels du cours

Pour garantir une expérience satisfaisante, les formations Microsoft requièrent une configuration matérielle minimale pour les ordinateurs de l'instructeur et des stagiaires dans toutes les classes Microsoft CPLS (Certified Partner for Learning Solutions) dans lesquelles les cours officiels de formation Microsoft sont utilisés. Ce cours requiert le niveau matériel 6.

# Module 1

## Présentation de SQL Server 2012 et de son ensemble d'outils

### Sommaire :

Leçon 1 : Présentation de la plateforme SQL Server	1-3
Leçon 2 : Utilisation des outils SQL Server	1-15
Leçon 3 : Configuration de Services SQL Server	1-30
Atelier pratique 1 : Présentation de SQL Server et de son ensemble d'outils	1-38

## Vue d'ensemble du module

- Présentation de la plateforme SQL Server
- Utilisation des outils SQL Server
- Configuration des services SQL Server

Avant de commencer à utiliser Microsoft® SQL Server® dans un rôle de développement ou d'administration, il est important de comprendre la plateforme SQL Server dans sa globalité. En particulier, il est utile de comprendre que SQL Server n'est pas simplement un moteur de base de données, mais qu'il s'agit d'une plateforme complète pour gérer les données de l'entreprise.

Grâce à sa plateforme performante, SQL Server offre une série d'outils qui simplifient la gestion du produit et fournissent une excellente cible pour le développement d'applications.

Les différents composants de SQL Server peuvent s'exécuter dans des contextes de sécurité distincts. La configuration correcte des services SQL Server est importante pour les entreprises qui ont recours à une stratégie d'autorisations limitées.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- Décrire la plateforme SQL Server.
- Utiliser les outils SQL Server.
- Configurer les services SQL Server.

## Leçon 1

# Présentation de la plateforme SQL Server

- Architecture SQL Server
- Composants SQL Server
- Instances de SQL Server
- Éditions SQL Server
- Versions de SQL Server

SQL Server est une plateforme de développement d'applications de gestion axée sur les données. Au lieu d'être une application monolithique, SQL Server est structuré comme une série de composants. Il est important de comprendre l'utilisation de chaque composant.

Plusieurs copies de SQL Server peuvent être installées sur un serveur. Chacune de ces copies est appelée « instance », elle peut être configurée et gérée séparément.

SQL Server est fourni dans diverses éditions, chacune avec un ensemble varié de fonctions. Il est important de comprendre les analyses de cas d'activité ciblées pour chacune des éditions de SQL Server et comment SQL Server a évolué au fil des séries d'améliorations apportées aux versions depuis de nombreuses années. C'est une plateforme stable et fiable.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- Décrire la plateforme SQL Server dans sa globalité.
- Expliquer le rôle de chacun des composants qui constituent la plateforme SQL Server.
- Décrire la fonctionnalité fournie par les instances de SQL Server.
- Présenter les éditions de SQL Server disponibles.
- Expliquer comment SQL Server a évolué au fil d'une série de versions.

## Architecture SQL Server

- Ensemble complet de technologies et d'outils pour l'entreprise
  - Plus qu'un simple système de gestion de base de données relationnelle
  - Gère les données d'entreprise
  - Valorise les informations contenues dans les données
  - Performances élevées
  - Haute disponibilité
  - Sécurisé
  - Évolutif
- Coût total de possession faible
- Plateforme hautement intégrée
  - Outilage productif de gestion et de développement

### Points clés

SQL Server est une plateforme intégrée prête pour l'entreprise destinée à la gestion des données, dont le coût total d'acquisition est faible.

**Question :** avec quelle autre plateforme de base de données avez-vous travaillé ?

### Prête à l'emploi dans le milieu de l'entreprise

SQL Server est beaucoup plus qu'un système de gestion de base de données relationnelle, car il fournit un système très sécurisé, fiable et stable. SQL Server permet de gérer les données organisationnelles et de fournir une analyse et des ajouts à ces données.

Le moteur de base de données est l'un des moteurs les plus performants de sa génération et ses fonctionnalités figurent régulièrement en tête des tests d'évaluation de performance de l'industrie. Vous pouvez consulter les tests d'évaluation et les notes attribuées par l'industrie à cette adresse <http://www.tpc.org>.

### Haute disponibilité

Des performances impressionnantes sont nécessaires mais pas au détriment de la disponibilité. Les entreprises ont un besoin constant d'accéder à leurs données. De nombreuses entreprises recherchent à présent un accès disponible 24 heures sur 24, 7 jours sur 7. La plateforme SQL Server a été conçue avec le souci de fournir les niveaux de disponibilité les plus élevés qui soient. À chaque version de produit commercialisée sont ajoutées de plus en plus de fonctions en vue de réduire les temps d'arrêt potentiels.

### Sécurité

Autre exigence cruciale pour les responsables d'entreprise est la nécessité de sécuriser les données de l'organisation. La sécurité ne peut pas être sacrifiée lors de la conception d'une application ou d'un produit. SQL Server a été généré dès le départ avec pour objectif des niveaux de sécurité les plus élevés qui soient.

## Évolutivité

Les organisations ont un réel besoin de fonctions de gestion des données pour les systèmes de toutes tailles. SQL Server répond aux besoins les plus petits jusqu'aux plus grands grâce à une série d'éditions offrant des fonctions croissantes.

## Coût d'acquisition

De nombreux systèmes de gestion de base de données d'accès concurrentiel sont coûteux à l'achat et la maintenance. SQL Server offre un coût total d'acquisition très faible. L'outillage SQL Server (en termes de gestion et de développement) s'appuie sur les connaissances Windows existantes. La plupart des utilisateurs sont susceptibles de se familiariser avec les outils assez rapidement. Le niveau de productivité obtenu en utilisant les outils est amélioré par le degré élevé d'intégration entre les outils. Par exemple, de nombreux outils SQL Server ont des liens pour ouvrir et préconfigurer d'autres outils SQL Server.

## Composants SQL Server

- Plus qu'un moteur de base de données
- Composants relationnels et de Business Intelligence

Composants SQL Server	
Moteur de base de données	Analysis Services
Integration Services	Reporting Services
Master Data Services	StreamInsight
Exploration de données	Recherche en texte intégral
PowerPivot	RéPLICATION
Data Quality Services	Power View

### Points clés

SQL Server est un très bon moteur de base de données relationnelle, mais en tant que plateforme de données, il offre beaucoup plus qu'un simple moteur de base de données relationnelle. SQL Server est une plateforme constituée de plusieurs composants.

Composant	Fonction
Moteur de base de données	Moteur de base de données relationnelle sur le langage SQL.
Analysis Services	Moteur de traitement analytique en ligne (OLAP) qui fonctionne avec les cubes analytiques.
Integration Services	Outil permettant d'orchestrer le déplacement des données entre les composants SQL Server et les systèmes externes (dans les deux sens).
Reporting Services	Moteur de création de rapports basé sur des services Web et fournit un portail Web et les outils de création de rapports de l'utilisateur.
Master Data Services	Fournit l'outillage et un concentrateur de gestion des données master ou de référence.
StreamInsight	Plateforme qui permet de générer des applications de traitement des événements à haut débit.
Exploration de données	Fournit l'outillage et un moteur d'inférence pour dériver les connaissances et les analyses des données OLAP ou des données relationnelles existantes.
Recherche en texte intégral	Permet de générer des options de recherche sophistiquées dans des applications. SQL Server 2012 ajoute la recherche sémantique sophistiquée à la recherche en texte intégral existante.

(suite)

Composant	Fonction
PowerPivot	Permet aux utilisateurs, aux utilisateurs avec pouvoir et les analystes d'entreprise d'analyser rapidement d'importants volumes de données à différents emplacements.
RéPLICATION	Permet le déplacement de données entre serveurs pour répondre aux besoins de distribution des données.
Data Quality Services	Permet de générer ou de se connecter à une base de connaissances pour le nettoyage des données.
Power View	Permet la visualisation rapide des données d'utilisateurs.

**Question :** avec quels composants de SQL Server avez-vous travaillé ?

## Instances de SQL Server

- De nombreux composants de SQL Server sont dépendants de l'instance
  - Moteur de base de données/Data Quality Services
  - Analysis Services/Exploration de données
  - Reporting Services/Integration Services
- Les instances permettent d'isoler
  - Configuration de la sécurité et administration
  - Performance, contrats de niveau de service
  - Versions, classements
- Assistance dans les scénarios de mise à niveau
- Deux types d'instance
  - Instances par défaut
  - Instances nommées

### Points clés

Il est parfois utile d'installer plusieurs copies d'un composant SQL Server sur le même serveur. De nombreux composants SQL Server peuvent être installés à plusieurs reprises comme instances distinctes.

### Instances de SQL Server

La possibilité d'installer plusieurs instances des composants SQL Server sur un serveur unique est utile dans un certain nombre de cas :

- Il peut être nécessaire d'avoir différents administrateurs ou environnements de sécurité pour des ensembles de bases de données. Chaque instance de SQL Server peut être gérée et sécurisée séparément.
- Les applications qui doivent être prises en charge par une organisation peuvent exiger des configurations de serveur qui sont incohérentes ou incompatibles avec la configuration requise pour le serveur d'applications. Chaque instance de SQL Server est configurable séparément.
- Les bases de données d'application doivent être prises en charge avec des niveaux de service en particulier, par rapport à la disponibilité. Les instances de SQL Server peuvent être utilisées pour séparer les charges de travail des contrats de niveau de service différents (SLAs) devant être exécutés.

- Il se pourrait que différentes versions et éditions de SQL Server doivent être prises en charge.
- Les applications peuvent nécessiter plusieurs classements au niveau du serveur. Alors que chaque base de données peut avoir des classements différents, une application peut dépendre du classement de la base de données tempdb si l'application utilise des objets temporaires.

**Question :** pourquoi auriez-vous intérêt à séparer les bases de données en fonction du contrat de niveau de service ?

Plusieurs versions de SQL Server peuvent généralement être installées côté à côté avec plusieurs instances. Cela peut permettre de tester des scénarios de mise à niveau ou d'exécuter des mises à niveau.

### Instances par défaut et instances nommées

Avant SQL Server 2000, une seule copie de SQL Server pouvait être installée sur un système hôte. SQL Server était traité par le nom du serveur. Pour assurer la compatibilité descendante, ce mode de connexion est toujours pris en charge et est appelé instance « par défaut ».

Les instances supplémentaires de SQL Server requièrent un nom d'instance en plus du nom du serveur et sont appelées instances « nommées ». Aucune instance par défaut ne doit être installée avant d'installer les instances nommées. Tous les composants SQL Server peuvent être installés dans plusieurs instances. Une différence majeure dans SQL Server 2012 permet de prendre en charge plusieurs instances de SQL Server Integration Services.

Il n'est pas nécessaire d'installer les outils et les utilitaires SQL Server plusieurs fois. Une seule installation des outils peut gérer et configurer toutes les instances.

## Éditions SQL Server

Éditions Premium	
Parallel Data Warehouse	Enterprise
Éditions Core	
Business Intelligence	Standard
Autres éditions	
Express	Compact
Developer	Base de données SQL Windows Azure
Web	

### Points clés

SQL Server est disponible dans une grande variété d'éditions, avec des prix et des niveaux de fonctionnalités différents.

## Éditions SQL Server

Toutes les éditions de SQL Server sont destinées à un cas d'utilisation professionnelle spécifique comme indiqué dans le tableau suivant :

Édition	Cas d'utilisation professionnelle
Parallel Data Warehouse	Utilisation du traitement parallèle massif (MPP, Massively Parallel Processing) pour exécuter des requêtes sur une quantité importante de données rapidement. Les systèmes d'entrepôt de données en parallèle sont vendus comme « appareil » complet plutôt que des licences logicielles standard.
Enterprise	Fournit le plus haut niveau de fiabilité en termes de charges de travail exigeantes.
Business Intelligence	Ajoute Business Intelligence aux offres de Standard Edition.
Standard	Fournit une plateforme de gestion des données complète et fiable.
Express	Édition gratuite simplifiée pour les petites applications serveur et Web.
Compact	Édition gratuite pour les applications mobiles autonomes et occasionnellement connectées, optimisée pour un très faible encombrement mémoire.

(suite)

Édition	Cas d'utilisation professionnelle
Developer	Permet la génération, le test et la démonstration de toutes les fonctionnalités de SQL Server.
Web	Fournit une plateforme sécurisée, rentable et évolutive pour les sites et les applications Web publics.
Base de données SQL Windows Azure	Permet la génération et l'extension des applications SQL server à une plateforme en nuage.

**Question :** quel serait un bon exemple d'entreprise qui aurait l'usage d'un service en nuage ?

## Versions de SQL Server

Version	Année de la version
1.0	1989
1.1	1991
4.2	1992
4.21	1994
6.0	1995
6.5	1996
7.0	1998
2000	2000
2005	2005
2008	2008
2008 R2	2010
2012	2012

### Points clés

Soulignez le fait que SQL Server est une plateforme à l'histoire riche en innovation tout en conservant une très grande stabilité. SQL Server est disponible depuis de nombreuses années, mais il évolue rapidement avec de nouvelles capacités et fonctionnalités.

### Versions antérieures

Les versions antérieures (1.0 et 1.1) étaient basées sur le système d'exploitation OS/2.

Les versions 4.2 et les versions ultérieures ont fait l'objet d'une migration vers le système d'exploitation Windows, initialement sur le système d'exploitation Windows NT.

### Versions ultérieures

La version 7.0 a vu une réécriture significative du produit. Des avancées significatives ont été apportées en réduisant la charge de travail d'administration du produit. Les services OLAP (qui sont devenus par la suite Analysis Services) ont été introduits.

SQL Server 2000 offrait une prise en charge de plusieurs instances et classements. Il a également introduit la prise en charge de l'exploration de données. SQL Server Reporting Services a été introduit après la commercialisation du produit sous la forme d'une amélioration additionnelle du produit, avec la prise en charge des processeurs 64 bits.

SQL Server 2005 a fourni une réécriture significative de nombreux aspects du produit. Il a introduit les prises en charge suivantes :

- Données non relationnelles stockées et interrogées au format XML.
- SQL Server Management Studio) a été débloqué pour remplacer plusieurs outils d'administration précédents.

- SQL Server Integration Services a remplacé un outil précédent appelé Data Transformation Services (DTS).
- Un autre ajout majeur au produit a été l'introduction de la prise en charge des objets créés par le Common Language Runtime (CLR).
- L'instruction T-SQL a été considérablement améliorée, notamment la gestion des exceptions structurées.
- Les vues et les fonctions de gestion dynamique ont été introduites pour activer le contrôle d'intégrité détaillé, le réglage des performances et le dépannage.
- Des améliorations de haute disponibilité significatives ont été incluses au produit. La mise en miroir des bases de données a été introduite.
- La prise en charge du chiffrement des colonnes a été introduite.

SQL Server 2008 a également fourni de nombreuses améliorations :

- Les technologies « SQL Server Always On » ont été introduites pour réduire les temps d'arrêt potentiels.
- La prise en charge FILESTREAM a amélioré la gestion des données structurées et semi-structurées.
- Les types de données spatiales ont été introduites.
- Les technologies de compression et de chiffrement de base de données ont été ajoutées.
- Les types de données liées à la date et l'heure ont été introduites, y compris la prise en charge des fuseaux horaires à l'aide des données Datetime.
- L'indexation de texte intégral a été intégrée directement dans le moteur de base de données. (Précédemment l'indexation de texte intégral était basée sur les interfaces des services de niveau système d'exploitation).
- Une infrastructure de gestion basée sur les stratégies a été introduite pour améliorer les méthodes de gestion déclarative plutôt que les méthodes réactives.
- Un fournisseur PowerShell pour SQL Server a été introduit.

Les améliorations et ajouts apportés au produit dans SQL Server 2008 R2 comprenaient :

- Des améliorations significatives ont été effectuées dans SQL Server Reporting Services.
- L'introduction des fonctions analytiques avancées avec PowerPivot.
- Des fonctions de gestion multi-serveur améliorées ont été ajoutées.
- La prise en charge de la gestion des données de référence a été fournie avec l'introduction de Master Data Services.
- StreamInsight permet d'interroger des données traitées avec une grande rapidité, avant d'enregistrer les données dans une base de données.
- Les applications de la couche Données assistent des applications de base de données d'empaquetage dans les projets de développement d'applications.

## SQL Server 2012

Les améliorations et ajouts apportés au produit dans SQL Server 2012 comprenaient :

- D'autres améliorations significatives ont été effectuées dans SQL Server Reporting Services.
- Améliorations significatives dans SQL Server Integration Services.
- Introduction des modèles de données tabulaires dans SQL Server Analysis Services.
- Migration des projets Business Intelligence dans Visual Studio 2010.
- Introduction des améliorations « Always on » à la haute disponibilité SQL Server.
- Introduction de Data Quality Services.
- Améliorations importantes apportées au langage T-SQL, comme l'ajout de séquences, de capacités de gestion des erreurs, ainsi que l'ajout de nouvelles fonctions de fenêtre.
- Introduction de FileTable.
- Introduction de la recherche sémantique statistique.
- De nombreuses améliorations d'outillage générales.

**Question** : avec quelle(s) version(s) de SQL Server avez-vous travaillé ?

## Leçon 2

# Utilisation des outils SQL Server

- Connexion à partir des clients et des applications
- Couches logicielles pour les connexions
- SQL Server Management Studio
- Démonstration 2A : Utilisation de SQL Server Management Studio
- SQL Server Data Tools
- Démonstration 2B : Utilisation de SQL Server Data Tools
- Documentation en ligne
- Démonstration 2C : Utilisation de la documentation en ligne

Travailler efficacement avec SQL Server requiert une connaissance des outils utilisés conjointement avec SQL Server. Avant qu'un outil puisse être connecté à SQL Server, une connexion réseau au serveur doit être établie. Dans cette leçon, vous allez voir comment ces connexions sont générées, puis rechercher les outils qui sont les plus couramment utilisés lors de l'utilisation de SQL Server.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- Se connecter à partir des clients et des applications.
- Décrire les rôles des Couches logicielles pour les connexions.
- Utiliser SQL Server Management Studio.
- Utiliser les outils de données SQL Server.
- Utiliser la documentation en ligne.

## Connexion à partir des clients et des applications

- Les applications clientes et les outils se connectent aux points de terminaison via des protocoles
  - Protocole TDS (Tabular Data Stream)
  - SQL Native Access Client (SNAC)
  - Transport par protocole TCP/IP, canaux nommés ou mémoire partagée
- Les utilisateurs (ou les applications) doivent être authentifiés par SQL Server
  - Connexions Windows
  - Groupes Windows
  - Connexions SQL Server
- La connexion est établie avec un serveur et une base de données
  - Peut être le nom du serveur ou une instance nommée

### Points clés

Les applications clientes se connectent aux points de terminaison. Plusieurs protocoles de communication sont disponibles pour l'établissement de connexions. En outre, les utilisateurs doivent être identifiés avant qu'ils ne soient autorisés à utiliser le serveur.

### Connectivité

Le protocole que les applications clientes utilisent pour se connecter au moteur de base de données relationnelle SQL Server s'appelle le protocole TDS (Tabular Data Stream). Il définit la manière dont les requêtes sont exécutées et la façon dont les résultats sont retournés.

D'autres composants de l'utilisation de SQL Server alternent des protocoles. Par exemple, les clients à SQL Server Analysis Services communiquent via le protocole XML/A (XML for Analysis). Toutefois, dans ce mode, vous êtes principalement concerné par le moteur de base de données relationnelle.

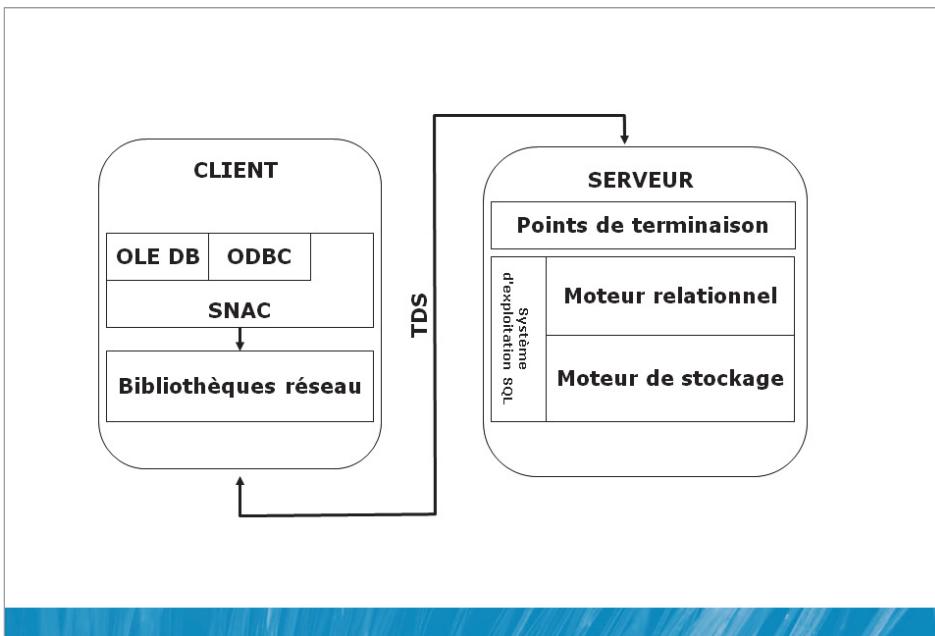
TDS est un protocole de haut niveau qui est transporté par les protocoles de niveau inférieur. Il est plus souvent transporté par le protocole TCP/IP, le protocole des canaux nommés ou implémenté sur une connexion de mémoire partagée.

## Authentification

Pour la majorité des applications et des organisations, les données doivent être sécurisées et l'accès aux données est basé sur l'identité de l'utilisateur qui tente d'accéder aux données. Le processus de vérification de l'identité d'un utilisateur (ou plus formellement, de toute individu), est appelé l'authentification. SQL Server 2005 prend en charge deux formes d'authentification. Il peut stocker les informations de connexion pour les utilisateurs directement dans ses propres bases de données système. Ces connexions sont des connexions SQL. Sinon, il peut être configuré de manière à approuver l'authentificateur Windows (par exemple Active Directory). Dans ce cas, un utilisateur Windows peut se voir accorder l'accès au serveur, soit directement, soit via les apparténances de son groupe Windows.

Lorsqu'une connexion est établie, l'utilisateur est connecté à une base de données, appelée base de données « par défaut ».

## Couches logicielles pour les connexions



### Points clés

Les connexions à SQL Server sont générées grâce à une série de couches logicielles. Il est important de comprendre la manière dont chacune de ces couches interagit. Ces informations vous seront utiles lorsque vous devrez effectuer une configuration ou un dépannage.

#### Bibliothèques clientes

Les applications clientes utilisent les bibliothèques de programmes pour simplifier leur accès aux bases de données telles que SQL Server.

Open Database Connectivity (ODBC) est une bibliothèque couramment utilisée. Elle s'exécute comme une couche de conversion qui protège l'application des détails du moteur de la base de données sous-jacente. En changeant la configuration ODBC, une application peut être modifiée pour utiliser un moteur de base de données différent, sans avoir besoin de modifier l'application. La connectivité JDBC (Java database connectivity) est la bibliothèque Java équivalente à ODBC.

OLEDB a initialement représenté la liaison et incorporation d'objets pour les bases de données, toutefois, son intérêt n'est plus très pertinent. OLE DB est une bibliothèque qui ne convertit pas les commandes. Lorsqu'une application envoie une commande SQL, OLEDB la transmet au serveur de base de données sans modification.

SQL Server Native Access Component (SNAC) est une couche logicielle qui encapsule les commandes émises par des bibliothèques telles que OLEDB, ODBC et JDBC dans les commandes qui peuvent être incluses par SQL Server, et inclut les résultats retournés par SQL Server prêts à être utilisés par ces bibliothèques. Cela implique essentiellement le renvoi à la ligne des commandes et des résultats dans le protocole TDS.

## Bibliothèques réseau

SQL Server présente les points de terminaison auxquels les applications clientes peuvent se connecter. Le point de terminaison est utilisé pour transmettre les commandes et les données à partir et vers le moteur de base de données.

La couche logicielle SNAC se connecte à ces points de terminaison via les bibliothèques réseau, comme TCP/IP ou les canaux nommés. Pour les applications clientes qui s'exécutent sur le même ordinateur que le service SQL Server, la connexion réseau de mémoire « partagée » spéciale est également disponible.

## Couches de logiciel SQL Server

SQL Server reçoit les commandes via des points de terminaison et envoie des résultats aux clients via des points de terminaison. Les clients interagissent avec le moteur relationnel qui utilise ensuite le moteur de stockage pour gérer le stockage des bases de données. Le système d'exploitation SQL Server (système d'exploitation SQL) est une couche logicielle qui fournit une couche d'abstraction entre le moteur relationnel et les ressources serveur disponibles.

## SQL Server Management Studio

- Environnement intégré pour l'utilisation de SQL Server
  - Interrogation de données
  - Configuration des serveurs
  - Administration des serveurs
  - Administration des bases de données
- Outils graphiques pour l'utilisation de nombreux composants SQL Server
  - Connexion au moteur de base de données, à Analysis Services, à Reporting Services, à Integration Services, à SQL Server Compact
- Système de projet avec des éditeurs de script performants
  - Scripts pour automatiser les tâches courantes et le déploiement

### Points clés

SQL Server Management Studio (SSMS) est le principal outil fourni par Microsoft pour interagir avec les services SQL Server.

### SQL Server Management Studio

SSMS est un environnement intégré qui a été créé dans le shell de la plateforme Microsoft Visual Studio. SSMS partage de nombreuses fonctionnalités communes avec Visual Studio.

L'environnement SSMS est utilisé pour exécuter des requêtes et des résultats, mais il est également en mesure d'aider les utilisateurs à analyser des requêtes. Il offre des éditeurs de texte enrichi pour différents types de documents (fichiers .sql, les fichiers .xml, etc.). Lors de l'utilisation de fichiers .sql, SSMS fournit Intellisense pour aider à l'écriture des requêtes.

Alors que toutes les tâches de gestion de base de données relationnelle SQL Server peuvent être effectuées à l'aide de l'instruction T-SQL, de nombreux utilisateurs préfèrent les outils d'administration graphique car ils sont généralement plus faciles à utiliser que les commandes T-SQL. SSMS fournit des interfaces graphiques pour configurer les bases de données et les serveurs.

SSMS est en mesure de se connecter à divers services SQL Server, notamment le moteur de base de données, Analysis Services, Integration Services, Reporting Services et SQL Server Compact Edition.

## Démonstration 2A : Utilisation de SQL Server Management Studio

Dans cette démonstration, vous allez apprendre à utiliser SQL Server Management Studio



### Procédure de démonstration

1. Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
3. Dans la fenêtre Se conn. au serveur, vérifiez que **Type de serveur** a la valeur **Moteur de base de données**.
4. Dans la zone de texte **Nom du serveur**, tapez **(local)**.
5. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn.**.
6. Si nécessaire, dans le menu **Affichage**, cliquez sur l'**Explorateur d'objets**.
7. Dans l'Explorateur d'objets, développez **Bases de données**, développez **AdventureWorks** et **Tables**. Vérifiez les objets de la base de données.
8. Cliquez avec le bouton droit sur la base de données **AdventureWorks** et sélectionnez **Nouvelle requête**.
9. Tapez la requête illustrée dans cet extrait de code ci-dessous :

```
SELECT * FROM Production.Product ORDER BY ProductID;
```
10. Remarquez la présence de Intellisense lors de la saisie, puis cliquez sur **Exécuter** dans la barre d'outils. Notez comment les résultats sont retournés.
11. Dans le menu **Fichier**, cliquez sur **Enregistrer SQLQuery1.sql**. Notez qu'il enregistre la requête dans un fichier. Dans la fenêtre Enregistrer sous, cliquez sur **Annuler**.

12. Dans l'onglet **Résultats**, cliquez avec le bouton droit sur la cellule pour **ProductID 1** (première ligne et première cellule) et cliquez sur **Enregistrer les résultats sous...**. Dans la zone de texte **Nom de fichier**, tapez **Demonstration2AResults**, puis cliquez sur **Enregistrer**. Notez que cela a pour effet d'enregistrer les résultats de la requête dans un fichier.
13. Dans le menu **Requête**, cliquez sur **Afficher le plan d'exécution estimé**. Notez que SSMS peut faire davantage que la simple exécution de requêtes.
14. Dans le menu **Outils**, cliquez sur **Options**.
15. Dans le volet **Options**, développez **Résultats de la requête**, développez **SQL Server**, puis cliquez sur **Général**. Examinez les options de configuration disponibles puis cliquez sur **Annuler**.
16. Dans le menu **Fichier**, cliquez sur **Fermer**. Dans la fenêtre Microsoft SQL Server Management Studio, cliquez sur **Non**.
17. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**.
18. Dans la fenêtre Ouvrir un projet, sélectionnez le projet **D:\22462A\_Labs\22462A\_02\_PRJ\22462A\_02\_PRJ.ssmssln**.
19. Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Notez le contenu de l'Explorateur de solutions. Des projets SQL Server sont fournis pour chaque module du cours ; ils contiennent les étapes de démonstration et des solutions suggérées pour l'atelier pratique, ainsi que tout code nécessaire pour l'installation ou l'arrêt du module.
20. Dans l'Explorateur de solutions, cliquez sur **X** pour le fermer.
21. Dans Explorateur d'objets, notez grâce à l'icône **Conneter** de la barre d'outils les autres composants auxquels SQL Server peut se connecter :
  - Moteur de base de données, Analysis Services, Integration Services, Reporting Services
22. Dans le menu **Fichier**, cliquez sur **Nouveau**, puis sur **Requête de moteur de base de données** pour ouvrir une nouvelle connexion.
23. Dans la fenêtre Se connecter au moteur de base de données, tapez **(local)** dans la zone de texte **Nom du serveur**.
24. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn.**
25. Dans la liste déroulante **Bases de données disponibles**, cliquez sur la base de données **tempdb**. Notez que cela a pour effet de changer la base de données concernée par la requête.
26. Cliquez avec le bouton droit dans la fenêtre de requête et cliquez sur **Connexion**, puis cliquez sur **Modifier la connexion...** Remarque : cette opération permet de reconnecter la requête à une autre instance de SQL Server. Dans la fenêtre Se connecter au moteur de base de données, cliquez sur **Annuler**.
27. Dans le menu **Afficher**, cliquez sur **Serveurs inscrits**.
28. Dans la fenêtre Serveurs inscrits, développez **Moteur de base de données**, cliquez avec le bouton droit sur **Groupes de serveurs locaux**, puis cliquez sur **Nouveau groupe de serveurs...**
29. Dans la fenêtre Propriétés du nouveau groupe de serveurs, tapez **Dev Servers** dans la zone de texte **Nom du groupe**, puis cliquez sur **OK**.

30. Cliquez avec le bouton droit sur **Dev Servers** et cliquez sur **Nouvelle inscription de serveur...**
31. Dans la fenêtre Nouvelle inscription de serveur, cliquez sur la liste déroulante **Nom du serveur**, tapez **(local)** et cliquez sur **Enregistrer**.
32. Cliquez avec le bouton droit sur **Dev Servers** et cliquez sur **Nouvelle inscription de serveur...**
33. Dans la fenêtre Nouvelle inscription de serveur, cliquez sur la liste déroulante **Nom du serveur**, tapez **.MKTG** et cliquez sur **Enregistrer**.
34. Dans la fenêtre Serveurs inscrits, cliquez avec le bouton droit sur le groupe **Dev Servers** et sélectionnez **Nouvelle requête**.
35. Tapez la requête comme indiqué dans l'extrait de code ci-dessous et cliquez sur l'icône **Exécuter** de la barre d'outils :

```
SELECT @@version;
```

36. Fermez SQL Server Management Studio.
37. Cliquez sur **Non** dans la fenêtre SQL Server Management Studio.

**Question** : à quel moment l'affichage d'un plan d'exécution estimé pourrait être utile ?

## SQL Server Data Tools

- Environnement intégré pour le développement d'objets BI
- Basé sur Visual Studio 2010 avec des modèles de projet
  - Projet Analysis Services
  - Projet Integration Services
  - Projet Reporting Services

### Points clés

La plateforme SQL Server comprend plusieurs composants. Les projets pour plusieurs composants Business Intelligence associés sont créés et modifiés à l'aide de SQL Server Data Tools (SSDT).

### SQL Server Data Tools

SSDT est une série de modèles de projets qui ont été ajoutés à l'environnement Microsoft Visual Studio® 2010. Les modèles permettent la création et la modification des projets pour Analysis Services, Integration Services et Reporting Services.

Visual Studio n'a pas besoin d'être installé avant SQL Server. Si une installation existante de Visual Studio 2010 est présente, le programme d'installation de SQL Server ajoute des modèles de projet à l'environnement. Si aucune installation existante de Visual Studio 2010 n'est spécifiée, le programme d'installation de SQL Server installera d'abord l'édition « partenaire » de Visual Studio 2010 et ajoutera ensuite les modèles de projet requis. L'édition partenaire de Visual Studio 2010 est essentiellement un environnement Visual Studio vide avec un modèle pour « une solution vierge ».

## Démonstration 2B : Utilisation de SQL Server Data Tools

Dans cette démonstration, vous allez apprendre à utiliser SQL Server Data Tools



### Procédure de démonstration

1. Si la démonstration 2A n'a pas été exécutée, rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, cliquez sur **Microsoft SQL Server 2012**, puis sur **SQL Server Data Tools**. Dans le menu **Fichier**, cliquez sur **Nouveau projet**. Notez les modèles de projet disponibles (également si d'autres langues sont installées).
3. Dans le volet **Modèles installés**, cliquez sur **Projet Report Server**, puis cliquez sur **OK**.
4. Dans l'Explorateur de solutions, cliquez avec le bouton droit sur **Rapports** et cliquez sur **Ajouter un nouveau rapport**.
5. Dans la fenêtre Assistant Rapport, cliquez sur **Suivant**.
6. Dans la fenêtre Sélectionner la source de données, cliquez sur **Modifier**.
7. Dans la fenêtre Paramètres de connexion, tapez **(local)** pour le **Nom du serveur** et dans la liste déroulante **Se connecter à une base de données**, sélectionnez **AdventureWorks**, puis cliquez sur **OK**.
8. Dans la fenêtre Sélectionner la source de données, cliquez sur **Suivant** dans la fenêtre Concevoir la requête, de la zone de texte **Chaîne de requête**, tapez la requête suivante comme indiqué dans l'extrait de code ci-dessous, puis cliquez sur **Suivant**.

```
SELECT ProductID, Name, Color, Size FROM Production.Product ORDER BY ProductID;
```

9. Dans la fenêtre Sélectionner le type du rapport, cliquez sur **Suivant**.
10. Dans la fenêtre Générer le tableau, cliquez sur **Détails** quatre fois, puis cliquez sur **Terminer>>|**.

11. Dans la fenêtre Fin de l'Assistant, cliquez sur **Terminer**.
12. Dans l'onglet **Report1.rdl [Design]\***, cliquez sur **Aperçu** et notez le rapport généré.
13. Cliquez sur l'onglet **Concevoir**, dans le menu **Fichier**, cliquez sur **Quitter**. Notez de ne pas enregistrer les modifications si vous y êtes invité.

**Question** : pouvez-vous suggérer une situation où la possibilité de programmer l'exécution d'un rapport serait utile ?

## Documentation en ligne

- Référence principale de SQL Server
- Peut être installée en mode hors connexion et en ligne
- Mise à jour continuellement
- La plupart des commandes sont illustrées d'exemples
- Pages disponibles pour les versions antérieures

### Points clés

La Documentation en ligne (Books Online, BOL) est une référence principale de SQL Server. Elle peut être installée hors connexion (pour une utilisation lors de la déconnexion à Internet) et peut également être utilisée en ligne directement à partir du site Web Microsoft MSDN (via une connexion Internet).

La version la plus récente est toujours la version en ligne. Il s'agit également de la version la plus souple car elle permet la comparaison simple entre les versions SQL Server de la documentation. SQL Server 2012 comporte un gestionnaire de bibliothèque d'aide qui permet de basculer entre les modes en ligne et hors connexion de la documentation en ligne et permet également d'installer des mises à jour de la version hors connexion.

### Documentation en ligne

La Documentation en ligne doit être considérée comme la référence technique principale de SQL Server.

Une erreur courante en installant la Documentation en ligne localement sur une installation de SQL Server consiste à négliger sa mise à jour régulière. Pour éviter que la taille des fichiers téléchargés soit excessive, la Documentation en ligne n'est pas incluse dans le Service Pack SQL Server ni les mises à jour cumulatives. La Documentation en ligne est régulièrement mise à jour et de ce fait, une vérification permanente doit être effectuée pour les mises à jour.

Pour la plupart des commandes T-SQL, de nombreux utilisateurs rechercheront les exemples fournis plus simples à suivre que la définition formelle de la syntaxe. Notez qu'en consultant la page de référence pour une instruction, la syntaxe formelle est affichée en haut de la page et les exemples sont généralement en bas de la page.

La Documentation en ligne est disponible pour toutes les versions de SQL Server prises en charge. Il est important de s'assurer que vous utilisez les pages conçues pour la version de SQL Server avec laquelle vous travaillez. De nombreuses pages de la documentation en ligne fournissent des liens vers des pages concernant les autres versions du produit.

## Démonstration 2C : Utilisation de la documentation en ligne

Dans cette démonstration vous allez découvrir l'utilisation de la Documentation en ligne SQL Server Business

### Procédure de démonstration

1. Si la démonstration 2A n'a pas été exécutée, rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, cliquez sur **Tous les programmes**, cliquez sur **Microsoft SQL Server 2012**, cliquez sur **Documentation et communauté**, puis cliquez sur **Documentation SQL Server**.
3. Agrandissez la fenêtre Visionneuse d'aide Microsoft et notez les options de navigation basiques disponibles.
4. Sur **l'ordinateur virtuel**, cliquez sur **Démarrer**, sur **Tous les programmes**, cliquez sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
5. Dans la fenêtre **Se conn au serveur**, vérifiez que **Type de serveur** a la valeur **Moteur de base de données**.
6. Dans la zone de texte **Nom du serveur**, tapez **(local)**.
7. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn**.
8. Dans le menu **Fichier**, cliquez sur **Nouveau**, puis cliquez sur **Requête avec la connexion actuelle**.
9. Dans l'onglet **SQLQuery1.sql**, tapez la requête comme indiqué dans l'extrait de code ci-dessous et cliquez sur l'icône **Exécuter** de la barre d'outils :

```
SELECT SUBSTRING('test string',2,7);
```
10. Cliquez sur le nom de la fonction **SUBSTRING**, puis frappez la touche F1 pour ouvrir la rubrique de la documentation en ligne de SUBSTRING.

11. Notez le contenu de la page et défilez vers le bas de la page pour consulter les exemples, puis fermez la fenêtre Visionneuse d'aide Microsoft.
12. Fermez SQL Server Management Studio sans enregistrer les modifications.
13. Si votre système hôte dispose d'un accès Internet, ouvrez **Internet Explorer®** et naviguez jusqu'à la page de Documentation en ligne SQL Server : <http://go.microsoft.com/fwlink/?LinkId=233780> (*Certains de ces sites adressées dans ce cours sont en anglais*) et remarquez les options accessibles en ligne.

## Leçon 3

# Configuration de Services SQL Server

- Gestionnaire de configuration SQL Server
- Services SQL Server
- Ports réseau et écouteurs
- Création d'alias de serveur
- Autres outils SQL Server
- Démonstration 3A : Utilisation du Générateur de profils SQL Server

Chaque service SQL Server peut être configuré individuellement. En fournissant l'option de configuration individuelle des services, les organisations ont la possibilité de réduire les autorisations attribuées aux comptes de service, dans le cadre d'une stratégie d'exécution de privilèges minimaux. Le gestionnaire de configuration SQL Server permet de configurer des services, notamment les comptes sous lesquels les services fonctionnent, puis les bibliothèques réseau utilisées par les services SQL Server.

SQL Server fournit également divers outils et utilitaires. Il est important de connaître la fonction de ces outils et utilitaires.

## Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- Utiliser le Gestionnaire de configuration SQL Server.
- Utiliser les Services SQL Server.
- Utiliser les ports réseau et les écouteurs.
- Créer des alias de serveur.
- Utiliser d'autres outils SQL Server.

## Gestionnaire de configuration SQL Server

- Gère les services associés à SQL Server
- Gère les protocoles réseau et les ports exposés par SQL Server
- Gère les protocoles réseau et les ports utilisés par les applications clientes sur le serveur lors de la connexion à ce serveur ou à d'autres



### Points clés

Le gestionnaire de configuration (SSCM) SQL Server permet de configurer des services SQL Server, de configurer les bibliothèques réseau exposées par les services SQL Server et de configurer la façon dont les connexions clientes sont établies avec SQL Server.

### Gestionnaire de configuration SQL Server

SSCM peut être utilisé pour trois objectifs différents :

- Gestion des services – chaque service peut être contrôlé (démarré ou arrêté) et configuré.
- Gestion des protocoles serveur – il est possible de configurer les points de terminaison exposés par les services SQL Server. Cela inclut les protocoles et les ports utilisés.
- Gestion des protocoles clients – lorsque des applications clientes telles que (SSMS) sont installées sur un serveur, il est nécessaire de configurer la façon dont les rapports de ces outils sont établis avec SQL Server. SSCM peut être utilisé pour configurer les protocoles requis et peut être utilisé pour créer des alias pour les serveurs afin de simplifier la connectivité.

**Question :** pourquoi un système serveur a-t-il besoin d'un node de configuration client ?

## Services SQL Server

- De nombreux composants SQL Server fournissent des services du système d'exploitation
- Les composants dépendant d'une instance assurent un service par instance
- Les services ont une identité dans Windows
  - Éviter les priviléges excessifs
- Les services peuvent être configurés pour démarrer automatiquement

Nom	État	Mode de démarrage
SQL Server Integration Services 11.0	En cours d'exécution	Automatique
SQL Server (MKTG)	En cours d'exécution	Automatique
SQL Full-text Filter Daemon Launcher (MSSQLSE...)	En cours d'exécution	Manuel
SQL Full-text Filter Daemon Launcher (MKTG)	En cours d'exécution	Manuel
SQL Server (MSSQLSERVER)	En cours d'exécution	Automatique
SQL Server Analysis Services (MSSQLSERVER)	En cours d'exécution	Automatique
SQL Server Reporting Services (MKTG)	En cours d'exécution	Automatique
Agent SQL Server (MKTG)	Arrêté	Manuel
SQL Server Browser	Arrêté	Autre (Démarrage, S...
Agent SQL Server (MSSQLSERVER)	En cours d'exécution	Automatique

### Points clés

Le gestionnaire de configuration SQL Server peut être utilisé pour configurer les services fournis par SQL Server. De nombreux composants fournis par SQL Server sont implémentés en tant que services du système d'exploitation. Les composants SQL Server que vous sélectionnez lors de l'installation déterminent les services SQL Server qui sont installés.

## Services SQL Server

Ces services s'exécutent au sein d'une identité Windows spécifique. Si l'existe un besoin de modifier l'identité affectée pour un service, SSCM doit être utilisé pour apporter cette modification. Une erreur courante consiste à utiliser l'applet des services dans les outils d'administration de serveur pour modifier l'identité du service. Bien que cette applet permettra de modifier l'identité du service, elle ne mettra pas à jour les autres autorisations et listes de contrôle d'accès nécessaires pour que le service fonctionne correctement. Lorsque des identités de service sont modifiées à partir de SSCM, les autorisations et les listes de contrôle d'accès requises sont également modifiées.

Chaque service dispose d'un mode de démarrage. Ce mode peut être défini comme automatique, manuel ou désactivé. Les services définis en mode de démarrage automatique sont lancés automatiquement lorsque le système d'exploitation démarre. Les services qui sont définis en mode de démarrage manuel peuvent être démarrés manuellement. Les services définis sur le mode de démarrage désactivé, ne peuvent pas être démarrés.

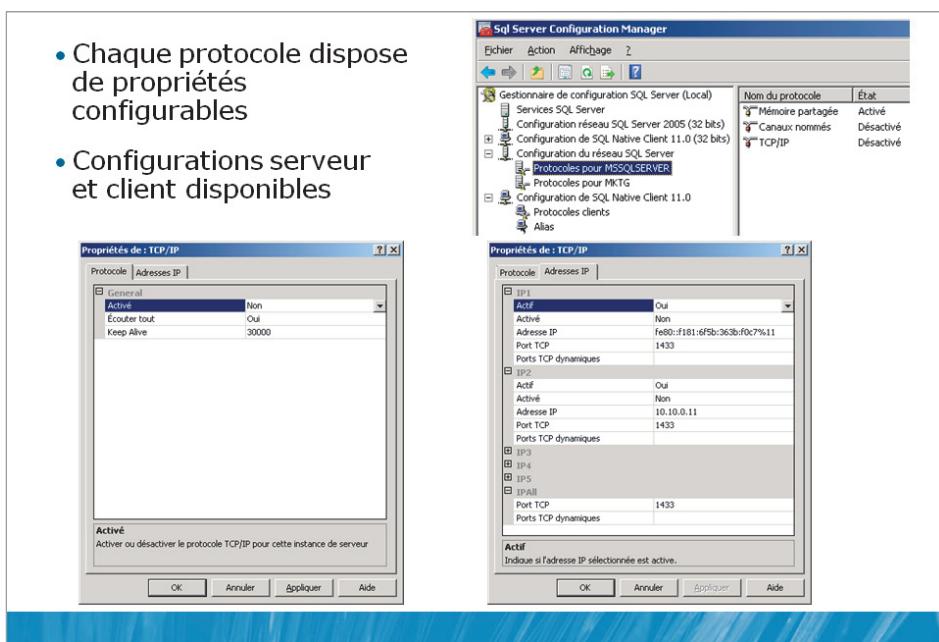
## Instances

De nombreux composants SQL Server sont des instances et peuvent être installés plusieurs fois sur un seul serveur. Lorsque SSCM répertorie chaque service, il pointe vers l'instance associée de SQL Server entre parenthèses après le nom du service.

Dans l'exemple de la diapositive, il existe une seule instance du moteur de base de données installé. MSSQLSERVER est le nom par défaut affecté à l'instance du moteur de base de données SQL Server par défaut.

## Ports réseau et écouteurs

- Chaque protocole dispose de propriétés configurables
- Configurations serveur et client disponibles



### Points clés

Le gestionnaire de configuration SQL Server peut être utilisé pour configurer les protocoles et les ports clients.

### Ports réseau et écouteurs

SSCM fournit deux ensembles de configurations réseau. Chaque point de terminaison réseau qui est exposé par une instance de SQL Server peut être configuré. Cela inclut la détermination des bibliothèques réseau qui sont activées et, pour chaque bibliothèque, la configuration de la bibliothèque réseau. En général, il impliquera des paramètres tels que des numéros de port de protocole. Vous devez discuter de la configuration du protocole réseau requis de SQL Server avec votre administrateur réseau.

De nombreux protocoles fournissent plusieurs niveaux de configuration. Par exemple, la configuration du protocole TCP/IP autorise différents paramètres sur chaque adresse IP configurées si nécessaire ou un ensemble général de configurations qui s'appliquent à toutes les adresses IP.

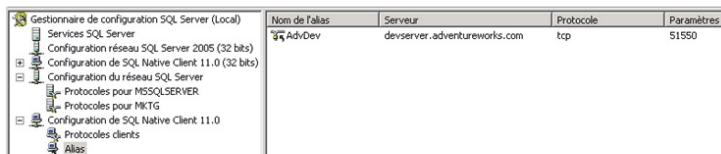
### Configurations clientes

Chaque ordinateur qui possède un client SNAC (SQL Native Access Client) doit avoir la possibilité de configurer la façon dont cette bibliothèque accède aux services SQL Server.

Un nom SNAC est installé sur le serveur ainsi que sur les systèmes client. Lorsque SSMS est installé sur le serveur, il utilise la bibliothèque de clients SNAC pour établir des connexions aux services SQL Server qui se trouvent sur le même système. Les nœuds de configuration cliente dans SSCM peuvent être utilisés pour configurer la façon dont ces rapports sont générés. Notez que deux ensembles de configuration cliente sont fournies et ils ne s'appliquent qu'à l'ordinateur sur lequel ils sont configurés. Un ensemble est utilisé pour les applications 32 bits ; l'autre ensemble est utilisé pour les applications 64 bits. SSMS est une application 32 bits, même si SQL Server est installé en tant qu'application 64 bits.

## Création d'alias de serveur

- Les alias peuvent servir d'abstraction pour un client
  - Nom du serveur
  - Protocole
  - Configuration réseau ou de port
- Plusieurs alias peuvent pointer vers le même serveur



Nom de l'alias	Serveur	Protocole	Paramètres
AdvDev	devserver.adventureworks.com	tcp	51550

### Points clés

La connexion à un service SQL Server peut impliquer plusieurs paramètres tels que l'adresse, le protocole et le port du serveur. Pour faciliter les applications clientes et fournir un niveau de redirection disponible, des alias peuvent être créés pour les serveurs.

### Alias

Les informations de connexion de codage pour un serveur, protocole et port spécifiques dans une application ne sont pas souhaitables, puisqu'elles vont évoluer au fil du temps.

Un alias de serveur peut être créé et associé à un serveur, un protocole et un port (si nécessaire). Les applications clientes peuvent se connecter ensuite à l'alias sans être concernées par la façon dont ces rapports sont générés.

Chaque système client qui utilise un client SNAC (notamment le serveur lui-même) peut comporter un ou plusieurs alias configurés. Les alias pour les applications 32 bits sont configurés indépendamment des alias pour les applications 64 bits.

Dans l'exemple ci-dessous de la diapositive, l'alias « AdvDev » a été créé pour le serveur « devserver.adventureworks.com » et le protocole TCP/IP avec un port TCP personnalisé de 51550 ont été utilisés. Le client doit ensuite se connecter uniquement au nom « AdvDev ».

## Autres outils SQL Server

- SQL Server Profiler
- Assistant Paramétrage du moteur de base de données
- Gestionnaire de configuration Master Data Services
- Gestionnaire de configuration de Reporting Services
- Client Data Quality Services
- Crédit de rapports d'erreur et d'utilisation SQL Server
- Fournisseur PowerShell
- SMO (SQL Server Management Objects)

### Points clés

SQL Server fournit un ensemble complet d'outils et d'utilitaires pour faciliter l'utilisation du produit.

Le tableau suivant répertorie les outils les plus couramment utilisés :

Outil	Rôle
SQL Server Profiler	Trace l'activité des applications clientes vers SQL Server. Prend en charge le moteur de base de données et Analysis Services.
Assistant Paramétrage du moteur de base de données	Génère des index et des statistiques pour améliorer les performances de la base de données, en fonction de l'analyse des charges de travail de trace.
Gestionnaire de configuration Master Data Services	Configure et gère le Master Data Services SQL Server.
Gestionnaire de configuration de Reporting Services	Configure et gère SQL Server Reporting Services.
Client Data Quality Services	Configure et gère les projets et bases de connaissances de Data Quality Services.
Création de rapports d'erreur et d'utilisation SQL Server	Configure le niveau d'information automatisée (rapports) fournies à l'équipe de produit SQL Server sur les erreurs qui se produisent lors de l'utilisation de différents aspects du produit.
Fournisseur PowerShell	Autorise la configuration et l'interrogation de SQL Server à l'aide de PowerShell.
SMO (SQL Server Management Objects)	Fournit une bibliothèque basée sur le .NET détaillé pour utiliser les aspects de la gestion de SQL Server directement à partir du code d'application.

## Démonstration 3A : Utilisation du Générateur de profils SQL Server

Dans cette démonstration, vous verrez comment SQL Server Profiler peut capturer les traces d'instructions exécutées

### Procédure de démonstration

1. Si la démonstration 2A n'a pas été exécutée, rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
3. Dans la fenêtre Se conn au serveur, vérifiez que **Type de serveur** a la valeur **Moteur de base de données**.
4. Dans la zone **Nom du serveur**, tapez **local**, puis dans la liste déroulante **Authentification**, sélectionnez **Authentification Windows** et cliquez sur **Se conn**.
5. Dans le menu **Outils**, cliquez sur **SQL Server Profiler**.
6. Dans la fenêtre Se conn au serveur, vérifiez que **Type de serveur** a la valeur **Moteur de base de données**.
7. Dans la zone **Nom du serveur**, tapez **local**, puis dans la liste déroulante **Authentification**, sélectionnez **Authentification Windows** et cliquez sur **Se conn**.
8. Dans la fenêtre Propriétés de la trace, cliquez sur **Exécuter**. Remarquez que cela initie une nouvelle trace avec les options par défaut.
9. Basculez vers SQL Server Management Studio, cliquez sur l'icône **Nouvelle requête** de la barre d'outils.

10. Dans la fenêtre Requête, tapez la requête comme indiqué dans l'extrait de code ci-dessous et cliquez sur l'icône **Exécuter** de la barre d'outils :

```
USE AdventureWorks;
GO
SELECT * FROM Person.Contact ORDER BY FirstName;
GO
```

11. Basculez vers SQL Server Profiler. Notez la trace d'instruction qui se produit dans SQL Server Profiler, puis dans le menu **Fichier** et cliquez sur **Arrêter la trace**.
12. Dans la grille de **résultats**, cliquez sur les instructions individuelles pour en voir les détails s'afficher dans le volet inférieur.
13. Fermez SQL Server Management Studio et SQL Server Profiler sans enregistrer les modifications.

**Question** : dans quel cas utiliseriez-vous des fichiers de trace capturés ?

## Atelier pratique 1 : Présentation de SQL Server 2012 et de son ensemble d'outils

- Exercice 1 : Vérifier l'installation des composants de SQL Server
- Exercice 2 : Modifier les comptes de service de la nouvelle instance
- Exercice 3 : Activer le protocole des canaux nommés pour les deux instances
- Exercice 4 : Créer un alias pour AdvDev
- Exercice difficile 5 : Vérifier que le navigateur SQL est désactivé et configurer un port TCP/IP fixe (si le temps le permet)

Informations d'ouverture de session

Ordinateur virtuel	22462A-MIA-SQL1
Nom d'utilisateur	AdventureWorks\Administrateur
Mot de passe	Pa\$\$w0rd

Durée approximative : 45 minutes



### Configuration de l'atelier pratique

Pour cet atelier pratique, vous utiliserez l'environnement d'ordinateur virtuel disponible. Avant de débuter cet atelier pratique, vous devez procéder aux étapes suivantes :

- Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.

### Scénario de l'atelier pratique

AdventureWorks est un fournisseur mondial, vendeur en gros et au détail de produits de cycle. Les propriétaires de l'entreprise ont décidé d'ouvrir une nouvelle filiale spécialisée dans le marketing direct. Elle a donné naissance à une nouvelle société nommée Proseware Inc. Bien qu'elle ait été conçue comme une entreprise indépendante; elle bénéficie de certains services informatiques fournis par la société AdventureWorks et reçoit un sous-ensemble des données d'entreprise de AdventureWorks. La plateforme SQL Server existante de la société AdventureWorks a été déplacée vers un nouveau serveur qui peut prendre en charge à la fois la charge de travail existante et la charge de travail de la nouvelle société. Dans cet atelier pratique, vous allez vérifier que l'instance supplémentaire de SQL Server a été configurée correctement et vous allez apporter à sa configuration un certain nombre de modifications requises.

## Exercice 1 : Vérifier l'installation des composants de SQL Server

Une nouvelle instance de SQL Server a été installée par le service informatique chez AdventureWorks. Elle est utilisée par la nouvelle entreprise de marketing direct. L'instance nommée de SQL Server s'appelle MKTG. Dans le premier exercice, vous devrez vérifier que les composants SQL Server requis ont bien été installés.

Les tâches principales de cet exercice sont les suivantes :

1. Vérifier que le Moteur de base de données et Reporting Services ont été installés pour l'instance MKTG.
2. Notez les services installés pour l'instance par défaut.
3. Vérifier que tous les services nécessaires comprenant l'Agent SQL Server sont lancés et configurés pour démarrer automatiquement dans les deux instances.

► **Tâche 1 : Vérifier que le Moteur de base de données et Reporting Services ont été installés pour l'instance MKTG**

- Ouvrez SQL Server Configuration Manager.
- Vérifiez la liste des services installés pour l'instance de MKTG et vérifiez que le moteur de base de données et Reporting Services ont été installés pour l'instance de MKTG.

► **Tâche 2 : Notez les services installés pour l'instance par défaut**

- Notez la liste des services installés pour l'instance par défaut.

► **Tâche 3 : Vérifier que tous les services nécessaires comprenant l'Agent SQL Server sont lancés et configurés pour démarrer automatiquement dans les deux instances**

- Vérifiez que tous les services de MKTG sont lancés et configurés pour démarrer automatiquement. (Ignorez le démon de filtre de texte intégral pour le moment).
- Vérifiez que tous les services de l'instance par défaut sont configurés pour démarrer automatiquement. (Ignorez le démon de filtre de texte intégral pour le moment).

**Résultats :** À la fin de cet exercice, vous aurez vérifié que les services SQL Server nécessaires sont installés, lancés et configurés pour démarrer automatiquement.

## Exercice 2 : Modifier les comptes de service de la nouvelle instance

### Scénario

Les services SQL Server de l'instance de MKTG ont été configurés pour s'exécuter sous le compte de service AdventureWorks\SQLService. Dans cet exercice, vous allez configurer les services à exécuter sous le compte de service AdventureWorks\PWService.

Les tâches principales de cet exercice sont les suivantes :

1. Modifier le compte de service du moteur de base de données MKTG.
2. Modifier le compte de service pour l'Agent SQL Server MKTG.

► **Tâche 1 : Modifier le compte de service du moteur de base de données MKTG**

- Remplacez le compte de service du moteur de base de données MKTG par AdventureWorks\PWService à l'aide de la page des **propriétés** du service.

► **Tâche 2 : Modifier le compte de service pour l'Agent SQL Server MKTG**

- Remplacez le compte du service de l'Agent SQL Server MKTG par AdventureWorks\PWService à l'aide de la page des **propriétés** du service, puis redémarrez le service.

**Résultats** : À la fin de cet exercice, vous aurez configuré les comptes de service de l'instance de MKTG.

## Exercice 3 : Activer le protocole des canaux nommés pour les deux instances

### Scénario

Les applications clientes installées sur le serveur se connectent au moteur de base de données à l'aide du protocole de canaux nommés. Dans cet exercice, vous allez vérifier que le protocole des canaux nommés est activé pour les deux instances du moteur de base de données.

Les tâches principales de cet exercice sont les suivantes :

1. Activer le protocole des canaux nommés pour l'instance par défaut.
2. Activer le protocole des canaux nommés pour l'instance de MKTG.
3. Redémarrer les services de moteur de base de données des deux instances.

#### ► Tâche 1 : Activer le protocole des canaux nommés pour l'instance par défaut

- Si nécessaire, activez le protocole des canaux nommés pour l'instance du moteur de base de données par défaut à l'aide de la fenêtre Protocoles.

#### ► Tâche 2 : Activer le protocole des canaux nommés pour l'instance de MKTG

- Si nécessaire, activez le protocole des canaux nommés pour l'instance du moteur de base de données MKTG à l'aide de la fenêtre Protocoles.

#### ► Tâche 3 : Redémarrer les deux services Moteur de base de données

- Si nécessaire, redémarrez l'instance du moteur de base de données par défaut.
- Si nécessaire, redémarrez l'instance de moteur de base de données MKTG.
- Activez l'option pour garantir que les deux instances ont été redémarrées correctement.

**Résultats :** À la fin de cet exercice, vous devez vous être assuré que le protocole des canaux nommés est activé pour les deux instances du moteur de base de données.

## Exercice 4 : Créer un alias pour AdvDev

### Scénario

Une application de surveillance mal écrite a été installée sur le serveur. Malheureusement, elle requiert un nom de serveur codé de manière irréversible d'AdvDev. Dans cet exercice, vous allez tester et configurer un alias pour AdvDev qui indique le serveur MKTG. Utilisez le protocole des canaux nommés. Les alias de 32 et 64 bits doivent être configurés. Vous utiliserez SQL Server Management Studio pour tester l'alias après sa configuration.

Les tâches principales de cet exercice sont les suivantes :

1. Créer un alias 32 bits (AdvDev) pour l'instance de MKTG.
2. Créer un alias 64 bits (AdvDev) pour l'instance de MKTG.
3. Utiliser SQL Server Management Studio pour se connecter à l'alias et s'assurer qu'il fonctionne correctement.

#### ► Tâche 1 : Créer un alias 32 bits (AdvDev) pour l'instance de MKTG

- Créez un alias 32 bits pour l'instance de MKTG. Appelez l'alias AdvDev et connectez-vous via des canaux nommés. Utilisez le nom serveur « .\MKTG ».

#### ► Tâche 2 : Créer un alias 64 bits (AdvDev) pour l'instance de MKTG

- Créez un alias 64 bits pour l'instance de MKTG. Appelez l'alias AdvDev et connectez-vous via des canaux nommés. Utilisez le nom serveur « .\MKTG ».

#### ► Tâche 3 : Utiliser SQL Server Management Studio pour se connecter à l'alias et s'assurer qu'il fonctionne correctement

- Ouvrez SQL Server Management Studio.
- Connectez-vous à l'alias AdvDev.

**Résultats** : À la fin de cet exercice, vous devez avoir créé et testé un alias pour l'instance nommée.

## Exercice difficile 5 : Vérifier que le navigateur SQL est désactivé et configurer un port TCP/IP fixe (si le temps le permet)

### Scénario

Les applications clientes doivent se connecter à l'instance du moteur de base de données de MKTG via le protocole TCP/IP. Lorsque les connexions doivent parcourir un pare-feu, le port utilisé pour les connexions ne peut pas être configuré en tant que port dynamique. Le numéro de port ne doit pas changer. La stratégie d'entreprise à AdventureWorks est que les instances nommées doivent être accessibles via les ports IP fixes et que le service de SQLBrowser doit être désactivé. Dans cet exercice, vous apporterez des modifications de configuration pour vous conformer aux spécifications. Une exception de pare-feu a déjà été créée pour le port 51550, pour être utilisée avec l'instance du moteur de base de données de MKTG.

Les tâches principales de cet exercice sont les suivantes :

1. Configurer le port TCP pour l'instance du moteur de base de données MKTG sur 51550.
2. Désactiver le service SQLBrowser.

#### ► Tâche 1 : Configurer le port TCP pour l'instance du moteur de base de données MKTG sur 51550

- Dans la page de propriétés du protocole de serveur TCP/IP, configurez l'utilisation du port fixe 51550. (Vérifiez que vous désactivez le port dynamique).
- Redémarrez l'instance de moteur de base de données MKTG.
- Vérifiez que l'instance du moteur de base de données de MKTG a été redémarré correctement.

#### ► Tâche 2 : Désactiver le service SQLBrowser

- Arrêtez le service SQLBrowser.
- Définissez le mode de démarrage pour le service SQL Browser sur Désactivé.

**Résultats** : À la fin de cet exercice, vous aurez configuré un port TCP fixe pour l'instance du moteur de base de données de MKTG et désactivé le service de SQLBrowser.

## Contrôle des acquis du module et éléments à retenir

- Questions de contrôle des acquis
- Meilleures pratiques

### Questions de contrôle des acquis

1. Quelle est la différence entre une version et une édition SQL Server ?
2. Quelle est la fonction de SQL Server Data Tools ?
3. Visual Studio doit-il être installé préalablement à SSDT ?

### Meilleures pratiques

1. Vérifiez que personne n'utilise de licences d'édition développeur dans les environnements de production.
2. Effectuez le développement avec le moins possible de privilèges afin d'éviter de développer des applications qui ne fonctionneraient pas pour des utilisateurs ordinaires.
3. Si vous utilisez une version hors-connexion de la documentation en ligne, veillez à ce qu'elle soit maintenue à jour.
4. Veillez à ce que les comptes de service bénéficient du moins d'autorisations possible pour fonctionner.

# Module 2

## Préparation des systèmes pour SQL Server 2012

### Sommaire :

Leçon 1 : Vue d'ensemble de l'architecture SQL Server	2-3
Leçon 2 : Planification des besoins en ressources serveur	2-17
Leçon 3 : Tests de pré-installation pour SQL Server	2-29
Atelier pratique 2 : Préparation des systèmes pour SQL Server	2-35

## Vue d'ensemble du module

- Vue d'ensemble de l'architecture SQL Server
- Planification des besoins en ressources serveur
- Tests de pré-installation pour SQL Server

Avant de déployer Microsoft® SQL Server®, il est important d'organiser le déploiement de manière appropriée. Comme pour presque tous les types de projets, le travail d'installation est toujours plus facile s'il est accompagné d'une bonne planification.

Pour comprendre les exigences de SQL Server, vous devez avoir une certaine connaissance de son architecture. Dans ce module, vous pouvez voir comment SQL Server est structuré et les spécifications qu'il place sur la plateforme serveur sous-jacente.

Au lieu d'implémenter le nouveau matériel et les logiciels sur un serveur, d'effectuer l'installation de SQL Server, puis d'espérer que tout fonctionne comme prévu, une meilleure approche consiste à tester la plateforme serveur avant de déployer SQL Server. Dans ce module, vous verrez également comment utiliser les outils qui vous permettent de précontraindre la plateforme serveur pour déterminer si elle est réellement en mesure de prendre en charge les applications SQL server. Une approche qui inclut le pré-test pour éviter de résoudre les problèmes de plateforme une fois que SQL Server est déployé en production.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- Décrire l'architecture SQL Server.
- Planifier les besoins en ressources serveur.
- Effectuer le test de contrainte de pré-installation de SQL Server.

## Leçon 1

# Vue d'ensemble de l'architecture SQL Server

- Architecture SQL Server
- Utilisation de l'unité centrale (UC) par SQL Server
- Parallélisme
- Serveurs 32 bits et serveurs 64 bits
- Vue d'ensemble de la mémoire SQL Server
- E/S physiques et logiques
- Démonstration 1A : Configurations de l'UC et de la mémoire dans SSMS

Avant d'aborder les ressources que SQL Server tire de la plateforme serveur sous-jacente, vous devez comprendre comment SQL Server s'exécute, afin de pouvoir comprendre la fonction de chaque ressource requise. Pour maîtriser la Documentation SQL Server, vous devez également vous familiariser avec une partie de la terminologie utilisée pour décrire le fonctionnement du produit.

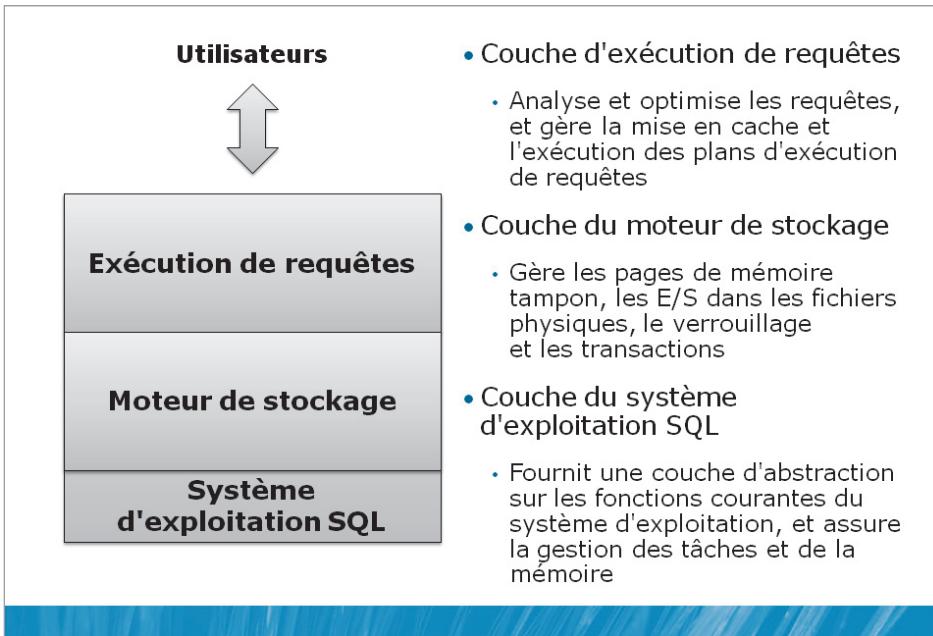
Les ressources les plus importantes que SQL Server utilise sur la plateforme serveur sont l'unité centrale (UC), la mémoire et les fonctionnalités d'entrée/sortie (E/S). Dans cette leçon, vous verrez comment SQL Server est structuré et comment il utilise chacune de ces ressources.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- Décrire l'architecture SQL Server.
- Expliquer l'utilisation de l'unité centrale (UC) par SQL Server.
- Décrire le rôle du parallélisme.
- Expliquer comment les serveurs 32 et 64 bits diffèrent par rapport à SQL Server.
- Décrire comment SQL Server utilise la mémoire.
- Décrire la différence entre les E/S physique et logique.

## Architecture SQL Server



### Points clés

SQL Server est construit à partir d'une série de nombreux petits composants qui fonctionnent ensemble. Trois catégories générales de composants existent dans le moteur de base de données et sont structurées sous forme de couches : l'Exécution de la requête (ou également appelée Moteur relationnel), le Moteur de stockage et le système d'exploitation SQL.

#### Couche Exécution de la requête

En plus de la gestion du processus d'optimisation de requête, la couche Exécution de la requête gère les connexions et la sécurité. Une série de sous-composants l'aide à déterminer le mode d'exécution des requêtes :

- L'analyseur vérifie que vous avez suivi les règles du langage T-SQL et génère une arborescence de syntaxe, qui est une représentation simplifiée des requêtes à exécuter. L'analyseur génère ce que vous souhaitez obtenir dans vos requêtes.
- L'algébriseur convertit l'arborescence de syntaxe en une arborescence algébrique relationnelle, où les opérations sont représentées par des objets logiques plutôt que des mots. L'objectif de cette phase consiste à répertorier les éléments que vous souhaitez obtenir et de les convertir en une série d'opérations logiques constituant le travail qui sera effectué.

- L'optimiseur de requête considère les différentes façons dont les requêtes peuvent être exécutées et recherche un plan acceptable basé sur les coûts de chaque opération. Les coûts sont basés sur l'opération requise et le volume de données à traiter qui est calculée en fonction des statistiques de distribution. Par exemple, l'optimiseur de requête considère les données devant être extraites d'une table et les index qui sont disponibles sur la table, pour déterminer comment accéder aux données dans la table. Il est important de savoir que l'optimiseur de requête ne trouve pas toujours le plan le moins coûteux. Dans certains cas, la recherche du plan le moins coûteux peut être très longue. A la place, l'optimiseur de requête trouve un plan qu'il considère satisfaisant. L'optimiseur de requête traite également un cache du plan de requête afin qu'il puisse éviter toute surcharge lors de l'exécution du travail, lorsqu'une autre requête similaire est reçue pour l'exécution.

### Couche Moteur de stockage

La couche Moteur de stockage gère les données conservées dans les bases de données. Les principales responsabilités de la couche du moteur de stockage consistent à gérer la façon dont les données sont stockées sur le disque et en mémoire, gérer la façon dont les données sont mises en cache pour une réutilisation et gérer la cohérence des données via des verrous et des transactions. Les sous-composants principaux de la couche du moteur de stockage sont les suivants :

- Le composant Méthodes d'accès permet de gérer le mode d'accès aux données. Par exemple, le composant Méthodes d'accès utilise les analyses, les accès et les recherches.
- Le Cache de page gère le stockage des copies mises en cache relatives aux pages de données. La mise en cache des pages de données permet de réduire le temps nécessaire pour accéder aux pages de données. Le Cache de la page place des pages de données en mémoire, pour qu'elles soient accessibles en cas de besoin lors de l'exécution de la requête.
- Les composants de verrouillage et de gestion des transactions fonctionnent ensemble pour assurer la cohérence de vos données, y compris la maintenance de l'intégrité transactionnelle à l'aide du fichier de base de données.

### Couche Système d'exploitation SQL

Le système d'exploitation SQL est la couche de SQL Server qui fournit la fonctionnalité du système d'exploitation aux composants SQL Server. Tous les composants SQL Server utilisent des interfaces de programmation fournies par le système d'exploitation SQL pour accéder à la mémoire, planifier des tâches ou exécuter les E/S.

La couche d'abstraction fournie par le système d'exploitation SQL évite la présence de code ressources dans le code du moteur de base de données SQL Server. Les fonctions les plus importantes fournies par cette couche sont la planification et la gestion de la mémoire. Ces deux aspects sont présentés plus en détails ultérieurement dans cette leçon.

**Question** : pourquoi SQL Server doit-il optimiser les requêtes ?

**Question** : à votre avis, que signifie l'optimisation basée sur les coûts ? Pouvez-vous indiquer d'autres méthodes d'optimisation des requêtes (hormis l'optimisation basée sur les coûts) ?

## Utilisation de l'unité centrale (UC) par SQL Server

- Windows utilise la planification préemptive de threads
- Un planificateur pour chaque UC logique créée dans le système d'exploitation SQL
  - Gère les threads récupérés de Windows et affecte des tâches aux threads
  - Réduit les changements de contexte via la planification coopérative
- La disponibilité de l'UC peut être configurée sans redémarrage
  - Les planificateurs peuvent être activés ou désactivés
  - Le masque d'affinité de l'UC peut être défini
- Les tâches en attente sur une ressource sont déplacées vers une liste d'attente
  - Le type et le temps d'attente sont enregistrés
  - Les détails sont utiles pour la surveillance et la résolution des problèmes

### Points clés

Tout le travail effectué au sein du système d'exploitation Windows® est basé sur l'exécution des threads. Windows utilise la planification des threads de manière anticipée pour conserver le contrôle. Au lieu d'exiger des threads qu'ils abandonnent volontairement le contrôle de l'UC, les systèmes préemptifs ont une horloge qui permet d'interrompre les threads lorsque leur partage alloué du temps processeur est terminé. Les threads subissent un changement de contexte du fait de l'anticipation.

### Threads SQL Server

SQL Server extrait des threads de Windows. Un paramètre de configuration SQL Server (Nombre maximum de threads de travail) détermine le nombre de threads à récupérer. SQL Server a son propre système de planification interne, distinct de la planification qui est effectuée par le système d'exploitation. Au lieu d'utiliser les threads Windows directement, SQL Server crée un pool de threads de travail qui sont mappés à des threads Windows chaque fois que le travail doit être effectué.

Chaque fois qu'un composant SQL Server doit exécuter du code, le composant crée une tâche qui représente l'unité de travail à effectuer. Par exemple, si vous envoyez une série de commandes T-SQL sur le serveur, il est probable que votre série de commandes soit exécutée dans une tâche.

Lorsqu'une tâche est créée, elle reçoit le thread de travail disponible suivant qui n'est pas utilisé. Si aucun thread de travail n'est disponible, SQL Server essaie de récupérer un autre thread Windows, jusqu'au seuil défini par la limite de configuration du nombre maximum de threads de travail. À ce stade, la nouvelle tâche doit attendre de récupérer un thread de travail.

Toutes les tâches sont planifiées par le planificateur SQL Server jusqu'à ce qu'elles soient effectuées.

## Masque d'affinité

Les planificateurs peuvent être activés ou désactivés en définissant le masque d'affinité de l'UC sur l'instance. Le masque d'affinité est un bitmap configurable qui détermine les processeurs du système hôte devant être utilisés pour SQL Server et pouvant être modifiés sans redémarrage. Par défaut, SQL Server suppose qu'il peut utiliser tous les processeurs du système hôte. Bien qu'il est possible de configurer le bitmap de masque d'affinité directement à l'aide de sp\_configure, utilisez la fenêtre des propriétés de l'instance de serveur dans SSMS pour modifier l'affinité du processeur.

 **Remarque** Chaque fois que le terme UC (unité centrale) est utilisé ici par rapport à l'architecture interne SQL Server, il fait référence à tout processeur logique, que le noyau ou les unités centrales d'hyperthreading soient utilisés ou non. Notez également que le programme de licence SQL Server est actuellement basé sur les sockets UC physiques plutôt que sur les UC logiques. Si les paramètres d'affinité sont utilisés pour violer les limites du programme de licence SQL Server, le système le détecte au démarrage et stocke les détails du journal de violation dans le journal d'événements.

## Attente de ressources

Un concept qu'il peut être difficile à saisir au début, est que la plupart des tâches SQL Server passent la plus grande partie de leur temps à attendre qu'un événement externe se produise. La plupart du temps, elles peuvent attendre des E/S ou la version des verrous, mais il peut s'agir d'autres ressources système.

Lorsqu'une tâche doit attendre une ressource, elle est placée dans une liste d'attente jusqu'à ce que la ressource soit disponible. Lorsque la ressource est disponible, la tâche est signalée pour la poursuite du traitement, mais elle doit toujours attendre un autre partage de temps processeur. Cette allocation des processeurs à des ressources est une fonction du système d'exploitation SQL.

SQL Server conserve les enregistrements internes détaillés de la durée d'attente des tâches et des types de ressources qu'elles attendent. Vous pouvez afficher ces détails en interrogeant les vues système suivantes :

```
sys.dm_os_waiting_tasks;  
sys.dm_os_wait_stats;
```

L'utilisation de ces vues sera décrite dans le module 18.

**Question** : pourquoi l'analyse des attentes est-elle un important instrument de surveillance des performances SQL Server ?

## Parallélisme

**Le parallélisme est l'utilisation de plusieurs processeurs pour exécuter une requête unique en même temps**

- SQL Server peut choisir de distribuer des requêtes à plusieurs tâches
  - Les tâches peuvent s'exécuter en parallèle
  - L'exécution globale est plus rapide
  - La surcharge de synchronisation est réduite
  - Le parallélisme est uniquement utilisé pour les plans coûteux
- Le **degré maximum de parallélisme** définit comment plusieurs UC peuvent être utilisées pour exécuter une requête parallèle
  - Peut être remplacé en utilisant l'indicateur de requête MAXDOP
- Le **seuil de coût pour le parallélisme** définit le coût minimal des plans parallèles

### Points clés

SQL Server effectue normalement des requêtes avec des plans de séquence dans une seule tâche. Pour réduire le temps d'exécution globale, SQL Server peut décider de distribuer des requêtes sur plusieurs tâches afin qu'il puisse exécuter des tâches en parallèle.

### Plans d'exécution parallèles

L'exécution parallèle implique la surcharge de synchronisation des tâches et leur surveillance. En raison de cette surcharge, SQL Server considère uniquement des plans parallèles pour les requêtes les plus coûteuses, où les avantages sont supérieurs à la charge supplémentaire.

L'optimiseur de requête choisit si un plan parallèle doit être utilisé, selon deux aspects de la requête et de la configuration du système :

- Une valeur nommée Degré maximal de parallélisme (MAXDOP, Maximum Degree of Parallelism) détermine la limite du nombre d'UC pouvant être utilisés pour exécuter une requête.
- Une autre valeur appelée Seuil de coût pour le parallélisme détermine le coût qu'une requête doit couvrir avant qu'un plan de requête parallèle ne soit considéré.

Si une requête est relativement coûteuse pour prendre en compte un plan parallèle, SQL Server peut toujours décider d'utiliser un plan séquentiel qui est inférieur au coût global.

## Contrôle du parallélisme

L'optimiseur de requête crée uniquement un plan parallèle mais n'est pas impliqué dans le choix de la valeur MAXDOP. La valeur MAXDOP peut être configurée au niveau serveur et peut être remplacée au niveau de la requête via un indicateur de requête. Même si l'optimiseur de requête crée un plan de requête parallèle, le moteur d'exécution peut décider d'utiliser une seule UC, en fonction des ressources disponibles lors de l'exécution de la requête.

 **Remarque** Dans les versions antérieures de SQL Server, il était commun de désactiver les requêtes parallèles sur les systèmes qui étaient principalement utilisés pour le traitement des transactions. Cette restriction a été implémentée en définissant le paramètre serveur pour MAXDOP à la valeur 1. Dans la version actuelle de SQL Server, elle n'est généralement plus considérée comme une bonne pratique.

**Question** : pourquoi les plans parallèles impliquent-ils une surcharge ?

## Serveurs 32 bits et serveurs 64 bits

- L'espace d'adressage virtuel est la mémoire qui peut être allouée aux applications telles que SQL Server
  - 4 Go sur les systèmes 32 bits (2 à 3 Go disponibles pour l'application)
  - 4 Go pour les applications 32 bits s'exécutant sur WOW sur les systèmes d'exploitation 64 bits
  - 8 To pour les systèmes 64 bits
- L'extension AWE ne peut plus être utilisée pour accéder à la mémoire supplémentaire sur les systèmes 32 bits
  - Peut être un problème important pour les mises à niveau 32 bits vers SQL Server 2012
- Les processeurs Itanium ne sont plus pris en charge
- Les performances SQL Server dépendent fortement de la mémoire
  - L'installation des versions 64 bits est préférable
  - Options 64 bits disponibles pour toutes les versions de SQL Server

### Points clés

L'espace d'adressage virtuel (VAS, Virtual Address Space) est la quantité totale de mémoire à laquelle une application comme SQL Server peut faire référence dans Windows. Le VAS dépend de la configuration du serveur.

### Systèmes 32 bits

Les systèmes 32bits ont un VAS de 4 Go. Par défaut, la moitié de l'espace d'adressage (2 Go) est réservée au système et est appelée espace d'adressage du mode noyau. L'autre moitié du VAS (2 Go) est disponible pour l'utilisation de l'application et est appelée espace d'adressage du mode utilisateur. Il est possible de modifier cette proportion à l'aide d'un commutateur /3GB dans le fichier boot.ini du système d'exploitation (sur les systèmes d'exploitation antérieurs) ou à l'aide du programme bcdedit pour modifier la banque de données de configuration de démarrage (sur les systèmes d'exploitation plus récents). Une fois que le commutateur /3GB a été configuré, 1 Go du VAS est alloué au noyau et 3 Go du VAS est alloué aux applications.



**Remarque** Un contrôle plus affiné de l'espace alloué peut être effectué à l'aide du commutateur /USERVA à la place du commutateur /3GB. Le commutateur /USERVA autorise la configuration de n'importe quelle valeur entre 2 Go et 3 Go pour les applications des utilisateurs.

## Systèmes 64 bits

Les systèmes de base de données ont tendance à utiliser une grande quantité de données et une grande quantité de mémoire pour que les systèmes puissent servir un grand nombre d'utilisateurs simultanés. SQL Server a besoin d'une grande quantité de mémoire pour mettre en cache les requêtes et les pages de données. Une limite de 2 Go à 3 Go pour le VAS est une contrainte importante sur la plupart des systèmes actuels. Comme c'est le cas avec la plupart des moteurs de base de données, l'installation de SQL Server est plus performante sur une version Windows de 64 bits plutôt qu'une version de 32 bits.

Il convient d'installer une version 64 bits de SQL Server sur une version de Windows 64 bits. Les systèmes 64 bits complets offrent un VAS de 8 To.

Il est possible (mais moins souhaitable) d'installer une version 32 bits de SQL Server sur une version de Windows 64 bits. Cette configuration fournit un VAS de 4 Go complet pour les applications 32 bits et est basée sur la technologie d'émulation Windows on Windows (WOW).

## Limites du système 64 bits

Tous les systèmes actuels ne peuvent être implémentés comme des systèmes 64 bits. La limitation la plus courante concerne la disponibilité des fournisseurs de données. Si des fournisseurs de données spécifiques sont nécessaires pour connecter SQL Server à d'autres systèmes (notamment via la fonctionnalité de serveurs liés), il est important de vérifier que les fournisseurs de données requis sont disponibles dans les versions 64 bits. Par exemple, le fournisseur du moteur JET n'est actuellement disponible qu'en tant que fournisseur 32 bits.

**Question :** pourquoi la mise en cache des données est-elle si importante pour les moteurs de base de données tels que SQL Server ?

## Vue d'ensemble de la mémoire SQL Server

- Le pool de mémoires tampons est le principal objet de mémoire de SQL Server
  - Contient le cache de données
  - Fournit de la mémoire pour les autres composants SQL Server
  - Est divisé en pages de 8 Ko
- Le système d'exploitation SQL alloue automatiquement autant de mémoire que nécessaire
  - Utilise un mécanisme pour éviter une insuffisance de mémoire sur le système
  - Peut être configuré à l'aide des options de mémoire minimum et maximum du serveur

### Points clés

L'objet de mémoire centrale de SQL Server s'appelle le pool de mémoires tampons. Le pool de mémoires tampons est divisé en pages de 8 Ko ; la même taille que les pages de données dans une base de données. Le pool de mémoires tampons se compose de trois sections :

- Pages libres : pages qui ne sont pas encore utilisées et qui sont conservées en vue de répondre aux nouvelles demandes de mémoire.
- Pages occultées : pages qui sont utilisées (anciennement connues sous le nom de « pages volées ») par d'autres composants SQL Server, tels que l'optimiseur de requête et le moteur de stockage.
- Cache de données : pages utilisées pour mettre en cache les données des pages de base de données. Toutes les opérations de données sont effectuées dans le cache de données. Si une requête souhaite sélectionner les données d'une page de données spécifique, la page de données doit d'abord être déplacée dans le cache de données. En outre, la modification de données n'est jamais exécutée en mémoire. Les modifications ne sont jamais exécutées directement sur les fichiers de données. Les modifications appliquées aux pages génèrent des pages considérées comme incorrectes. Ces pages incorrectes sont écrites dans la base de données par un processus en arrière-plan appelé point de contrôle.

Le cache de données implémente un algorithme le moins récemment utilisé (LRU, Least Recently Used) pour déterminer les pages candidates à supprimer du cache en cas d'espace insuffisant, après qu'elles ont été vidées du disque (si nécessaire) par le processus de point de contrôle. Le processus qui effectue la tâche de suppression des pages est nommé outil d'écriture différée. L'outil d'écriture différée remplit deux principales fonctions. En supprimant les pages du cache tampon, l'outil d'écriture différée tente de conserver l'espace libre suffisant dans le cache de la mémoire tampon pour que SQL Server puisse fonctionner. L'outil d'écriture différée contrôle également la taille générale de la mémoire cache tampon pour éviter de trop consommer la mémoire du système d'exploitation Windows.

## Mémoire SQL Server et mémoire Windows

Le gestionnaire de mémoire du système d'exploitation SQL alloue et contrôle la mémoire utilisée par SQL Server. Il y parvient en vérifiant la mémoire disponible sur le système Windows, en calculant une valeur de mémoire cible, qui est la mémoire maximale que SQL Server peut utiliser à un instant donné, pour éviter une insuffisance de mémoire au niveau du système d'exploitation Windows. SQL Server est conçu pour répondre aux signaux du système d'exploitation Windows qui indiquent une modification des besoins en mémoire.

Tant que SQL Server reste dans la limite de la mémoire cible définie, il peut demander de la mémoire supplémentaire à Windows en cas de besoin. Si la valeur de la mémoire cible est atteinte, le gestionnaire de mémoire répond aux demandes de mémoire provenant des composants en libérant de la mémoire des autres composants. Cela peut impliquer la suppression de pages des caches. La valeur de mémoire cible peut être contrôlée via des options de mémoire maximum et minimum du serveur.

**Question :** quelle est l'utilité d'un algorithme le moins récemment utilisé ?

## E/S physiques et logiques

Type d'E/S	Description
E/S physique	Une E/S physique se produit lorsque la page demandée n'est pas disponible dans le cache de tampons et doit être lue à partir du fichier de données dans le cache de tampons pour qu'elle puisse être fournie, ou lorsqu'une page modifiée est écrite dans le fichier de données
E/S logique	Une E/S logique se produit lorsque la page demandée est disponible dans le cache de tampons

### Points clés

Une E/S logique se produit lorsqu'une tâche peut récupérer une page de données dans le cache de la mémoire tampon. Lorsque la page demandée n'est pas présente dans le cache de la mémoire tampon, elle doit d'abord être lue dans le cache de la mémoire tampon de la base de données. Cette opération de lecture dans la base de données est appelée E/S physique.

### Minimisation d'E/S physique

Du point de vue des performances globales du système, deux aspects d'E/S doivent être réduits :

- Vous devez réduire le nombre d'opérations d'E/S physiques.
- Vous devez réduire le temps requis par chaque opération d'E/S qui est encore nécessaire.

La réduction du nombre d'opérations d'E/S physiques s'effectue en :

- fournissant suffisamment de mémoire pour le cache de données ;
- optimisant la disposition physique et logique des bases de données, y compris les index ;
- optimisant des requêtes pour demander le moins d'opérations d'E/S possible.

Un des objectifs principaux de l'optimisation de la requête est de réduire le nombre d'opérations d'E/S logiques. L'effet secondaire est la réduction du nombre d'opérations d'E/S physiques.

 **Remarque** Il peut être difficile d'interpréter le nombre d'E/S logiques car certaines opérations peuvent provoquer le gonflage artificiel du nombre, en raison de plusieurs accès à la même page. Cependant, les petites quantités sont préférées aux plus grandes quantités.

## Gestion des opérations d'E/S de requête

Le nombre d'opérations d'E/S logiques et physiques sont visibles pour chaque requête en définissant l'option de connexion suivante :

```
SET STATISTICS IO ON;
```

Les opérations d'E/S physiques globales qui se produisent sur le système peuvent être affichées en interrogeant la fonction système sys.dm\_io\_virtual\_file\_stats. Les valeurs rentrées par cette fonction sont cumulatives à partir du point auquel le système a effectué le dernier redémarrage.

**Question** : pourquoi les E/S logiques et pas les E/S physiques doivent être optimisées lors de l'optimisation des requêtes ?

## Démonstration 1A : Configurations de l'UC et de la mémoire dans SSMS

Dans cette démonstration, vous allez apprendre à

- Configurer l'UC et la mémoire SQL Server à l'aide de SSMS
- Examiner les options de configuration à l'aide de T-SQL

### Procédure de démonstration

1. Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se conn au serveur, tapez **Proseware**, puis cliquez sur **Se conn**. Dans le menu **Fichier** cliquez sur **Ouvrir**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_02\_PRJ\22462A\_02\_PRJ.ssmssln** puis cliquez sur **Ouvrir**.
3. Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** à partir de l'**Explorateur de solutions**.
4. Ouvrez le fichier de script **11 – Demonstration 1A.sql**.
5. Suivez les instructions contenues dans les commentaires du fichier de script.

## Leçon 2

# Planification des besoins en ressources serveur

- Présentation de la planification des besoins en ressources de serveur
- Discussion : Expérience passée dans la planification de ressources
  - Planification des besoins en UC
  - Planification des besoins en mémoire
  - Planification des besoins en réseau
  - Planification des besoins en stockage et E/S

À présent que vous comprenez mieux comment SQL Server exécute des requêtes, il est temps d'organiser les ressources pertinentes pour prendre en charge l'installation de SQL Server sur une plateforme serveur.

Installer SQL Server est beaucoup plus facile si vous avez déjà planifié l'installation de manière appropriée.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- Décrire le processus de planification.
- Planifier les besoins en UC.
- Planifier les besoins en mémoire.
- Planifier les besoins liés au réseau.
- Planifier les besoins en stockage et E/S.

## Présentation de la planification des besoins en ressources de serveur

- Il n'existe pas de formule simple pour calculer les besoins en ressources
- La planification implique
  - Consultation des éditeurs de logiciels indépendants ou des développeurs
    - Demandez des études de cas et des installations de référence
    - Demandez des détails sur les expériences passées avec l'application
  - Exécution de tests avec des charges de travail réelles
  - Définition des objectifs et évaluation des résultats par rapport aux objectifs
  - Planification et prévision d'une nouvelle augmentation de la charge de travail et de la base de données
- Contrôlez les résultats après la mise en production

### Points clés

La planification des spécifications de ressources n'est pas une tâche simple. Il n'existe aucune formule simple qui permette de calculer les ressources requises basées sur les mesures, telles que la taille de la base de données et le nombre de connexions, même si vous pouvez obtenir des références quelquefois. Certains éditeurs d'applications fournissent des outils de dimensionnement qui peuvent donner un point de référence, mais même ces outils doivent être utilisés prudemment.

### Fournisseur d'application

Dans un premier temps, il est toujours correct de communiquer avec le fournisseur de l'application qui doit être installée sur le serveur. Si le fournisseur est un éditeur de logiciels indépendant (ISV), le fournisseur a souvent les installations de référence disponibles qui peuvent être utilisées comme point de départ pour dimensionner les ressources requises. En particulier, les études de cas sont souvent aussi utiles que la documentation des plateformes serveur sur lesquelles elles ont été effectuées.

### Test de production simulé

La deuxième étape de la planification de ressources doit inclure le test des différentes configurations. Pour y parvenir, il est important de définir les objectifs à atteindre. Ces objectifs doivent être définis au niveau de l'entreprise comme par exemple, « Combien de temps faut-il pour enregistrer une nouvelle commande ? » plutôt que de ramener la question au niveau système avec une question du genre « Quelle est la durée d'une opération d'E/S ? ».

Une partie de cette phase de test doit également inclure le test de l'évolution espérée. Ne testez pas seulement les systèmes dans la configuration initiale proposée. Testez comment les systèmes se comportent avec une base de données remplie à la taille maximale souhaitée, pendant la durée de vie de la plateforme serveur.

### Contrôle de la production

Le processus de planification ne doit pas se terminer une fois le système installé. Il est important de générer un processus de surveillance constant et des actions correctives, pour s'assurer que les objectifs souhaités sont toujours respectés.

**Question :** pourquoi est-il important d'exécuter des tests de planification des capacités ?

## Discussion : Expérience passée dans la planification de ressources

- La planification des ressources est une partie importante des nouvelles installations
  - Quelle est votre expérience passée en matière de planification de nouveaux systèmes ?
  - Quelle a été l'efficacité de la planification ?



### Points clés

La planification des ressources est une partie importante des nouvelles installations.

**Question :** quelle est votre expérience en matière de planification de nouveaux systèmes ?

**Question :** quelle a été l'efficacité de la planification ?

## Planification des besoins en UC

- La planification de processeur est relativement simple
- Effectuez des tests sur une charge de travail classique et contrôlez l'utilisation de l'UC
  - L'UC doit être inférieure à 30 pour cent en moyenne
  - Des pics peuvent se produire mais ne doivent pas durer trop longtemps
- Un plus grand nombre de processeurs fournissent de meilleures options de parallélisme
  - Prennent en charge un grand nombre de connexions simultanées
  - Prennent en charge des charges d'entrepôt de données élevées
- Un serveur de base de données dédié doit utiliser toutes les UC disponibles
- De nombreux nouveaux systèmes utilisent l'architecture NUMA
  - Consultez le fournisseur de matériel pour une configuration SQL Server optimale

### Points clés

L'utilisation du processeur dépend principalement des types de requêtes qui s'exécutent sur le système. La planification du processeur est généralement considérée comme relativement simple car très peu d'architectures système fournissent un contrôle précis des ressources processeur disponibles. Le test avec des charges de travail réalistes est la meilleure solution.



**Remarque** Un grand nombre de livres blancs techniques suggèrent qu'une charge de processeur soutenue de 50 % est acceptable. L'expérience des auteurs du cours est que les utilisations moyennes supérieures à 30 % aboutissent à des systèmes très longs, une fois combinées avec des charges de travail courantes. Les pics qui ne sont pas trop longs sont acceptables, quel que soit le pourcentage cible.

L'augmentation du nombre d'UC disponibles fournit à SQL Server de meilleures options pour créer des plans de requêtes parallèles. Même sans plan de requête parallèle, les charges de travail de SQL Server permettent optimiser l'utilisation de plusieurs processeurs lorsque vous utilisez des charges de travail de requêtes simples provenant d'un grand nombre d'utilisateurs simultanés. Les plans de requêtes parallèles sont particulièrement utiles lorsqu'un grand nombre de données sont analysées dans des entrepôts de données.

Essayez autant que possible que votre serveur soit dédié à SQL Server. La plupart des serveurs qui exécutent des charges de production sur SQL Server ne doivent pas avoir d'autres services significatifs qui s'exécutent sur le même système. Cela s'applique en particulier à d'autres applications serveur telles que Microsoft Exchange Server.

## NUMA

De nombreux nouveaux systèmes sont basés sur les architectures NUMA (Non-Uniform Memory Access). Dans un système traditionnel de multitraitements symétrique (SMP, Symmetric Multiprocessing), tous les processeurs et la mémoire sont liés à un même bus système. Le bus peut devenir un goulot d'étranglement lorsque des processeurs supplémentaires sont ajoutés. Sur un système NUMA, chaque ensemble d'unités centrales a son propre bus associé à la mémoire locale. Sur certains systèmes, le bus local peut également inclure des canaux d'E/S distincts. Ces ensembles d'UC sont appelés nœuds NUMA. Chaque nœud NUMA peut accéder à la mémoire des autres nœuds, mais l'accès à la mémoire locale est beaucoup plus rapide. De meilleures performances sont obtenues si les unités centrales accèdent essentiellement à leur propre mémoire locale. Windows et SQL Server ont adopté la technologie NUMA et tentent d'utiliser ses atouts.

La configuration NUMA optimale dépend fortement du matériel. Des fonctionnalités spéciales dans le système BIOS peuvent être nécessaires pour optimiser les performances. Il est essentiel de vérifier auprès du fabricant du matériel quelle est la configuration optimale pour installer SQL Server sur du matériel doté de la technologie NUMA.

## Planification des besoins en mémoire

- SQL Server dépend fortement de la mémoire
  - Les caches permettent de réduire les E/S et l'UC
    - Le cache de données réduit les E/S
    - Le cache du plan réduit la nécessité de recompiler les requêtes
  - Une mémoire de stockage et de traitement des données est requise pendant l'exécution des requêtes
  - Une insuffisance de mémoire peut entraîner une pression sur l'UC et les E/S
- Prévoyez suffisamment de mémoire
  - Possibilité de mettre en cache les données souvent consultées
  - Tenez compte de l'utilisation de mémoire des autres services qui s'exécutent sur le système

### Points clés

Vous venez de découvrir précédemment dans le module comment SQL Server est dépendant de la mémoire disponible pour optimiser ses performances. Les caches sont utilisés pour éviter les E/S et l'utilisation de l'UC, ainsi que pour stocker les plans de requête afin que le coût de compilation des requêtes pour produire des plans de requête soit évité, en réutilisant les requêtes déjà générées.

SQL Server utilise également la mémoire pendant l'exécution des requêtes. Chaque utilisateur qui exécute une requête requiert un espace mémoire séparé appelé contexte exécution, qui contient les données implantées en mémoire locale pour cet utilisateur.

Un manque de mémoire peut être perçu comme manque d'autres ressources. Par exemple, un manque de mémoire peut entraîner l'exécution des mécanismes de mise en cache qui se traduira par une utilisation inefficace. Cela peut entraîner des niveaux élevés de la charge du processeur et d'E/S, puisque le système recompile constamment des requêtes qui ont été expulsées du cache du plan ou lorsque le système relit en permanence les données de la base de données, après la suppression des pages de données du cache de données. Cela signifie qu'un manque de mémoire disponible, peut apparaître à la surface, comme un manque de spécifications d'UC ou d'E/S.

## Planification des besoins en mémoire

La planification de la mémoire requise sur un système implique la planification non seulement pour l'instance SQL Server, mais aussi pour les autres services qui s'exécutent sur le système. Les éditeurs de logiciels peuvent souvent fournir des recommandations sur les besoins en mémoire de leurs produits. Chaque système se comporte différemment, même lors de l'exécution de la même application, donc les expériences d'installations similaires peuvent être utilisées en tant que directives.

Comme indiqué précédemment dans cette leçon, le système doit être testé dans un scénario de production réel avec d'autres services installés et exécutés sur le système de production.



**Remarque** Certaines applications externes peuvent s'exécuter occasionnellement mais demandent beaucoup de mémoire lorsqu'elles sont lancées en premier. SQL Server Integration Services et les outils d'analyse des antivirus en sont des exemples classiques.

**Question** : quels sont les symptômes potentiels d'une insuffisance de mémoire ?

## Planification des besoins en réseau

- Planifiez et testez le débit requis pour l'application
  - Prévoyez d'utiliser plusieurs cartes réseau
- Prévoyez de dédier une connexion réseau à des fins d'administration
  - Écriture des sauvegardes sur les unités réseau
  - Transfert des données vers d'autres instances à l'aide de SSIS
- Utilisez une connexion réseau dédiée lors de l'utilisation de la mise en miroir de bases de données
- Choisissez et configurez les protocoles réseau utilisés
  - Vérifiez les configurations du pare-feu
  - Effectuez des tests dans un environnement réseau semblable au système de production cible

### Points clés

Le débit de réseau nécessaire doit être planifié et testé. Comme indiqué pour d'autres ressources, tester les besoins dans des environnements de production simulés est important. Pour de nombreux serveurs, une carte réseau unique ne permet pas de fournir un débit suffisant, vous devez alors considérer l'utilisation de plusieurs cartes réseau.

### Débit de la carte réseau

En examinant les caractéristiques des cartes réseau, gardez à l'esprit que la plupart des protocoles réseau de bas niveau (comme le protocole Ethernet) fonctionnent sur une base où les collisions peuvent se produire, mais sont détectées et récupérées. Cela signifie que les réseaux fonctionnent correctement lorsqu'ils sont légèrement chargés, mais sont rapidement inutilisables à mesure que des charges augmentent car les collisions sont fréquentes. Résultat : davantage de temps de récupération est consacré aux collisions plutôt qu'à l'exécution des E/S réseau. Par exemple, la plupart des réseaux Ethernet commencent à saturer à environ 10 % de leur vitesse indiquée.

### Autres services et applications

Outre la planification et le test du débit de réseau nécessaires aux clients et serveurs d'applications, il est également important de planifier pour d'autres opérations qui nécessitent le réseau sur le système. Souvent ces opérations sont oubliées lors de la planification et l'évaluation de nouveaux systèmes. Cependant, des opérations telles que l'exécution de sauvegardes sur les partages réseau ou le déplacement des unités de sauvegarde sur les stockages de sauvegarde sont des opérations qui génèrent des volumes d'E/S de réseau élevés. Cela peut ralentir les connexions réseau sur les clients et serveurs d'applications. Si le ralentissement est grave, les applications clientes peuvent commencer à souffrir des délais d'attente et des erreurs.

Par conséquent, ces opérations doivent également être planifiées et testées. Allouer une connexion réseau dédiée à ces opérations d'administration est une option intéressante à prendre en compte.

D'autres fonctionnalités SQL Server telles que la mise en miroir de bases de données et la réplication peuvent également avoir besoin de connexions dédiées à effectuer efficacement.



**Remarque** La mise en miroir et la réplication des bases de données sont des fonctionnalités avancées traitées dans ce cours.

**Question** : pourquoi les sauvegardes peuvent-elles interrompre ou ralentir les charges de travail de l'utilisateur ?

## Planification des besoins en stockage et E/S

- Les besoins en E/S doivent être déterminées et testées
- Éléments à prendre en compte pour le stockage
  - Stockage dédié et SAN
  - Systèmes RAID utilisés
  - Nombre de piles impliquées (souvent plus important que la taille de disque)
  - Configuration de la mise en cache d'E/S (cache de disque et d'écriture)

### Points clés

La planification et le test du stockage figurent parmi les tâches les plus importantes lors de la phase de planification. Les systèmes d'E/S actuels sont très complexes par nature. La planification et le test impliqueront souvent une équipe de personnes, chacune avec un ensemble de compétences spécifiques. Les performances de SQL Server sont étroitement liées aux performances du sous-système d'E/S qu'elles utilisent.

### Détermination des besoins des utilisateurs

Pendant la première phase de planification, les spécifications de l'application doivent être déterminées, y compris les modèles d'E/S qui doivent être respectés. Ces modèles d'E/S incluent la fréquence et la taille des lectures et d'écritures transmises par l'application. En règle générale, les systèmes OLTP produisent un grand nombre d'opérations d'E/S aléatoires dans les fichiers de données et les opérations d'écriture séquentielles sur les fichiers journaux de la base de données. En comparaison, les applications basées sur l'entrepôt de données sont susceptibles de générer des analyses sur les fichiers de données.

### Styles de stockage

La seconde phase de planification consiste à déterminer le style de stockage à utiliser. Avec le stockage en attachement direct (DAS, Direct Attached Storage), il est plus facile d'obtenir de bonnes performances prévisibles. Sur les systèmes SAN (Storage Area Network), plus de travail est souvent requis pour obtenir de bonnes performances, mais le stockage SAN fournit généralement une grande variété de fonctions de gestion.

L'une des préoccupations particulières des administrateurs SQL Server est que les administrateurs SAN sont généralement plus concernés par l'espace disque alloué aux applications plutôt que par les exigences en matière de performances des fichiers individuels. Au lieu de débattre des structures de fichiers avec un administrateur SAN, essayez de vous concentrer sur vos besoins en matière de performances pour des fichiers spécifiques. Laissez à l'administrateur SAN le soin de décider comment atteindre ces objectifs. Autrement dit, concentrez-vous sur ce qui est nécessaire au cours des discussions plutôt que sur la façon d'atteindre le résultat.

### Systèmes RAID

Sur les systèmes SAN, vous ne serez pas souvent concerné par les niveaux RAID utilisés. Si vous avez spécifié les performances requises sur une base de fichier, l'administrateur SAN doit sélectionner les niveaux RAID appropriés et les dispositions de disque physique pour le faire.

Pour le stockage DAS, tenez compte des niveaux RAID. Lorsque d'autres niveaux RAID existent, les niveaux RAID 1, 5 et 10 sont les niveaux RAID courants utilisés sur les systèmes SQL Server.

### Nombre de broches

Pour la plupart des systèmes actuels, le nombre de lecteurs (ou broches, bien que le terme est maintenant obsolète), importera plus que la taille du disque. Il est facile de rechercher les grands disques qui contiendront les bases de données significatives, mais souvent un seul grand disque ne peut pas fournir les opérations d'E/S suffisantes par seconde ou un débit de données (mo/s) suffisant à exploiter. Les systèmes SSD (Solid State Drive) changent rapidement les options disponibles dans cette zone.

### Mise en cache des lecteurs

La mise en cache de lecture dans des lecteurs de disque n'est pas particulièrement utile si SQL Server gère déjà son propre système de mise en cache et il est peu probable que SQL Server relira une page qu'il a récemment écrite, à moins que le système soit à court de mémoire. Les caches en écriture peuvent considérablement améliorer les performances d'E/S de SQL Server, mais assurez-vous que les caches de matériel permettent une lecture, même après une défaillance du système. Plusieurs caches en écriture du lecteur ne peuvent pas survivre à des échecs et peuvent produire des corruptions dans la base de données.



**Remarque** Le placement des fichiers sur des systèmes de disque sera présenté dans un module ultérieurement.

**Question** : pourquoi est-il préférable de répartir une base de données sur plusieurs disques au lieu d'un disque volumineux ?

## Leçon 3

# Tests de pré-installation pour SQL Server

- Vue d'ensemble des tests de pré-installation
- Effectuer des pré-vérifications des sous-système d'E/S
- Présentation de SQLIOSIM
- Présentation de SQLIO
- Démonstration 3A : Utilisation de SQLIOSIM et SQLIO

L'un des problèmes courants dans les déploiements de serveurs de base de données se produit lorsque la plateforme serveur est livrée, installée et mandatée dès que SQL Server y est installé.

Il existe de nombreux problèmes subtils qui peuvent provoquer des erreurs sur les systèmes serveur, notamment les problèmes liés aux sous-systèmes d'E/S. Ces sous-systèmes d'E/S sont souvent très complexes et il peut être difficile d'évaluer l'interaction entre les composants du système.

L'option par défaut consiste à réaliser un test de contrainte de pré-installation des systèmes, sur la base de l'activité similaire à l'activité générée par SQL Server. Des outils tels que SQLIOSIM permettent de créer des charges d'E/S représentatives sans avoir à exécuter des applications SQL Server. Dans cette leçon, vous verrez comment utiliser SQLIOSIM pour le test général et utiliser un autre utilitaire appelé SQLIO, pour les types de test d'E/S.

## Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- Effectuer le test de pré-installation.
- Effectuer des pré-vérifications des sous-système d'E/S.
- Utiliser SQLIOSIM.
- Utiliser SQLIO.

## Vue d'ensemble des tests de pré-installation

- La planification et les tests sont fortement couplés
- Après la première phase de planification, des tests doivent être effectués
  - Vérifiez si les prévisions sont atteintes
  - Modifiez les prévisions en fonction des résultats
  - Vérifiez si les objectifs fixés dans la phase de planification sont atteints
- Testez différentes configurations impliquant
  - Disposition du stockage et des E/S
  - Configurations de la mémoire, de l'UC et du réseau
- Documentez chaque série de tests pour une utilisation ultérieure

### Points clés

Tout test que vous réalisez avant de mandater un nouveau système doit ressembler autant que possible à l'utilisation de la sortie finale. En particulier, vérifiez que la taille de la base de données et les charges de travail sont comparables aux objectifs attendus au cours de la durée de vie du système.

La planification n'est pas une activité unique. Une fois le premier test terminé, les résultats doivent être exploités dans un autre tentative de processus de planification.

### Documentation

Un aspect important qui est souvent absent dans de nombreuses organisations est la documentation des procédures de planification et de test. Si par la suite, vous avez besoin de procéder à des modifications parce que le système a changé, vous serez heureux d'avoir documenté les détails du processus de chaque scénario de test que vous avez effectué.

**Question :** pourquoi est-il important de documenter et d'archiver les tests exécutés sur le système ?

## Effectuer des pré-vérifications des sous-système d'E/S

- Testez les différents types et tailles d'E/S
  - Petites lectures et écritures aléatoires dans les fichiers de données pour les systèmes OLTP
  - Grandes lectures et écritures séquentielles pour les systèmes OLAP
  - Écritures séquentielles pour le stockage de journal
- Effectuez des tests avec différents nombres de fichiers impliqués
  - Utilisez des fichiers de test similaires à la configuration cible
- Déterminez le point de saturation d'un sous-système d'E/S en augmentant progressivement la charge
- Validez les résultats par rapport aux résultats attendus
  - En fonction des objectifs définis
  - Comparaison avec les fonctions d'E/S potentielles du système

### Points clés

Comme pour d'autres aspects du test de pré-installation, il est important que le test du sous-système d'E/S soit représentatif de la charge prévue. En particulier, il est important de s'assurer que les types d'opérations d'E/S attendues font partie de vos tests.

Les systèmes de traitement transactionnel en ligne (OLTP, Online Transaction Processing) sont susceptibles d'avoir un grand nombre de plus petits accès en lecture et écriture aléatoires. Les systèmes de traitement analytique en ligne (OLAP) sont susceptibles d'avoir des accès en lecture et écriture plus volumineux et plus séquentiels que les systèmes OLTP.

N'effectuez pas le test sur des fichiers uniques. Vérifiez que le nombre de fichiers que vous utilisez pour le test est également identique à celui de la configuration de production.

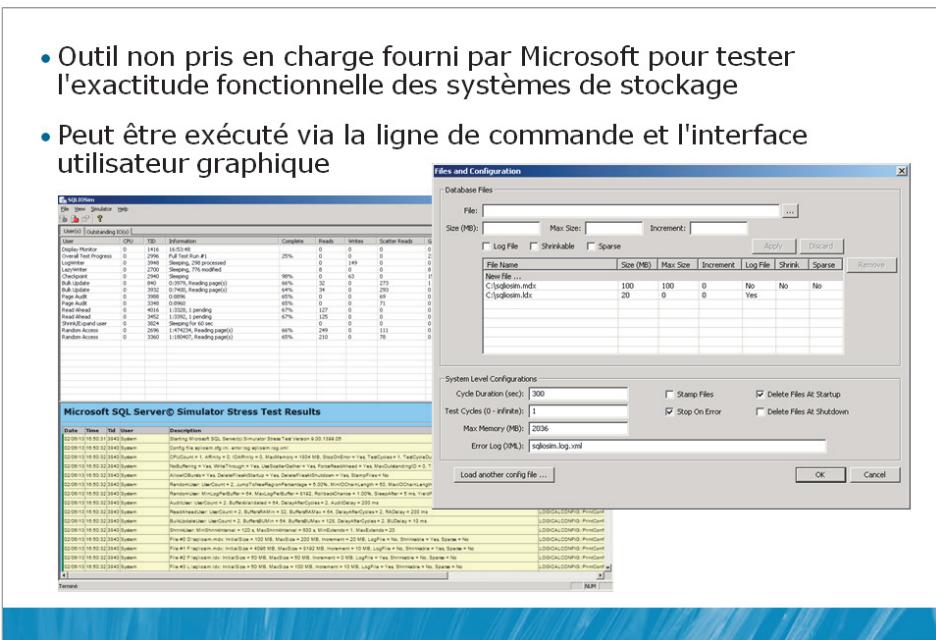
### Détermination des points de saturation

Pour la plupart des ressources, le point de saturation de la ressource apparaît lorsque le temps de récupération des résultats commence à augmenter, mais que la quantité du travail en cours d'exécution reste stationnaire.

**Question :** pourquoi est-il préférable de répartir une base de données sur plusieurs disques plutôt que sur un seul grand disque ?

## Présentation de SQLIOSIM

- Outil non pris en charge fourni par Microsoft pour tester l'exactitude fonctionnelle des systèmes de stockage
- Peut être exécuté via la ligne de commande et l'interface utilisateur graphique



### Points clés

SQLIOSIM est un utilitaire non pris en charge qui peut être téléchargé à partir de <http://download.microsoft.com>. SQLIOSIM est conçu pour simuler l'activité générée par SQL Server sans recourir à l'installation de SQL Server. Cette fonction fait de SQLIOSIM un bon outil de pré-test des systèmes serveur qui sont des cibles d'exécution SQL Server.

SQLIOSIM est un outil autonome qui peut être copié sur le système et exécuté. Il n'a pas besoin d'être installé sur le système cible via un programme d'installation. SQLIOSIM possède une interface utilisateur graphique et des options d'exécution de ligne de commande.



**Remarque** SQLIOSIM remplace plusieurs outils précédents par des noms tels que SQLStress et SQLIOStress.

Alors que SQLIOSIM est utile pour les systèmes soumis au test de contrainte, il n'est pas utile pour le test des performances générales. Les tâches qu'il effectue varient entre chaque exécution de l'utilitaire, donc inutile de comparer les résultats de plusieurs exécutions directes, notamment en termes de temps.

SQLIO est conçu pour des tests de performances renouvelables et doit être utilisé à la place de SQLIOSIM à cet effet.

## Présentation de SQLIO

- Outil non pris en charge fourni par Microsoft pour tester les performances des systèmes de stockage
- Outil de ligne de commande qui peut être configuré sur un modèle d'E/S spécifique

```
C:\>Program Files <x86>\SQLIO>sqlio.exe -kW -s60 -frandom -o8 /b8 -LS -Fparam.txt
sqlio v1.5.SG
using system counter for latency timings, 10000000 counts per second
parameter file used: param.txt
    file d:\testfile.dat with 2 threads <0-1> using mask 0x0 <0>
2 threads writing for 60 secs to file d:\testfile.dat
    using 2KB random I/Os
    enabling multiple I/Os per thread with 8 outstanding
using specified size: 100 MB for file: d:\testfile.dat
initialization done
CUMULATIVE DATA:
throughput metrics:
I/Os/sec: 91.03
MBs/sec: 0.17
latency metrics:
Min_Latency<ns>: 0
Avg_Latency<ns>: 175
Max_Latency<ns>: 1371
histogram:
ms: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24+
z: 0 1 2 2 2 2 2 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 83
C:\>Program Files <x86>\SQLIO>timeout /T 60
```

### Points clés

SQLIO est autre utilitaire non pris en charge qui peut être téléchargé à partir de <http://download.microsoft.com>. Contrairement à SQLIOSIM qui est utilisé pour le test de contrainte non renouvelable, SQLIO est un outil qui est conçu pour créer des modèles d'E/S entièrement renouvelables. Un fichier de configuration permet de déterminer les types d'opérations d'E/S qui doivent être testées, ce qui permet ensuite à SQLIO de tester les types d'opérations spécifiques.

Lors de la tentative de résolution des problèmes, une méthode courante d'utilisation conjointe de SQLIOSIM et SQLIO consiste à utiliser SQLIOSIM pour rechercher les problèmes dans certains types d'opérations d'E/S, puis d'utiliser SQLIO pour générer ces types de problèmes spécifiques d'opérations d'E/S. SQLIO est également un outil autonome qui ne requiert pas l'installation de SQL Server sur le système. En outre, il est conseillé de réaliser ces tests avant l'installation de SQL Server. SQLIO contrôle uniquement un seul type d'E/S à la fois. Ainsi, l'interprétation des résultats est la tâche la plus importante.

**Question :** quels types d'E/S SQL Server produit-il principalement pour les fichiers de données d'un système OLTP classique ?

## Démonstration 3A : Utilisation de SQLIOSIM et SQLIO

Dans cette démonstration, vous allez apprendre à

- Configurer SQLIOSim pour exécuter un test de contrainte sur un serveur
- Tester différents modèles d'E/S à l'aide de l'utilitaire SQLIO

### Procédure de démonstration

1. Si la démonstration 1A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se conn au serveur, tapez **Proseware**, puis cliquez sur **Se conn**. Dans le menu **Fichier** cliquez sur **Ouvrir, sur Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_02\_PRJ\22462A\_02\_PRJ.ssmssln** puis cliquez sur **Ouvrir**.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** à partir de l'**Explorateur de solutions**.
2. Ouvrez le fichier de script **31 – Démonstration 3A.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

## Atelier pratique 2 : Préparation des systèmes pour SQL Server

- Exercice 1 : Ajuster la configuration de la mémoire
- Exercice 2 : Tests de contrainte de pré-installation
- Exercice difficile 3 : Vérifier des opérations d'E/S spécifiques (si le temps le permet)

Informations d'ouverture de session

Ordinateur virtuel	22462A-MIA-SQL1
Nom d'utilisateur	AdventureWorks\Administrateur
Mot de passe	Pa\$\$w0rd

**Durée approximative : 45 minutes**

### Configuration de l'atelier pratique

Pour cet atelier pratique, vous utiliserez l'environnement d'ordinateur virtuel disponible. Avant de débuter cet atelier pratique, vous devez procéder aux étapes suivantes :

1. Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
3. Dans la fenêtre Se connecter au serveur, tapez **Proseware** dans la zone de texte **Nom du serveur**.
4. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn.**
5. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**.
6. Dans la fenêtre Ouvrir un projet, sélectionnez le projet **D:\22462A\_Labs\22462A\_02\_PRJ\22462A\_02\_PRJ.ssmssln**.
7. Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Dans l'Explorateur de solutions, double-cliquez sur le fichier de requête **00 - Setup.sql**. Lorsque la fenêtre de requête s'affiche, cliquez sur **Exécuter** dans la barre d'outils.

## Scénario de l'atelier pratique

Vous avez examiné l'instance supplémentaire de SQL Server. Un administrateur système chez AdventureWorks se demande si le serveur existant a suffisamment de mémoire ou de capacité d'E/S pour prendre en charge cette instance SQL Server ; il étudie la solution de mettre en place un nouveau sous-système d'E/S. En tant qu'administrateur de base de données, vous devez vérifier la mémoire disponible du serveur et la mémoire allouée à chacune des instances de SQL Server. Vous devez vérifier que le sous-système d'E/S du nouveau serveur peut exécuter correctement SQL Server et la charge de travail requise.

### Documentation

#### Configuration de la mémoire requise

- 1,5 Go réservé pour le système d'exploitation.
- 60 % de la mémoire restante comme valeur maximale pour l'instance de serveur AdventureWorks.
- 40 % de la mémoire restante comme valeur maximale pour l'instance de serveur Proseware.
- Configurez la mémoire minimale à zéro pour les deux instances.

#### Configuration SQLIOSIM requise

- Lecteur D doté d'un fichier de données de 100 Mo qui augmente avec des incrément de 20 Mo pour atteindre une taille maximale 200 Mo.
- Lecteur L doté d'un fichier journal de 50 Mo qui augmente avec des incrément de 10 Mo pour atteindre une taille maximale de 100 Mo.
- Durée des cycles (s) définie à 60 secondes.
- L'option Supprimer les fichiers lors de l'arrêt doit être sélectionnée.

#### Tests SQLIO nécessaires

- Le lecteur à tester est D.
- Test des lectures séquentielles 64 Ko pendant 60 secondes.
- Test des écritures aléatoires 8 Ko pendant 60 secondes.

## Exercice 1 : Ajuster la configuration de la mémoire

### Scénario

Le serveur Marketing d'Adventure Works dispose d'une instance Microsoft SQL Server 2012 par défaut installée et de la nouvelle instance MKTG. Vous devez activer la mémoire totale disponible sur le serveur et la quantité de mémoire a été allouée à chacune des deux instances de SQL Server. Vous devez ensuite déterminer si l'allocation de mémoire est appropriée. Si ce n'est pas le cas, apportez les modifications nécessaires à la configuration de la mémoire.

Les tâches principales de cet exercice sont les suivantes :

1. Vérifier la mémoire totale du serveur.
2. Vérifier la mémoire allouée à l'instance par défaut.
3. Vérifier la mémoire allouée à l'instance MKTG.
4. Déterminer si l'allocation de mémoire est appropriée. Si ce n'est pas le cas, apportez les modifications nécessaires à la configuration de la mémoire.

#### ► Tâche 1 : Vérifier la mémoire totale du serveur

- Récupérez la valeur de la mémoire (RAM) installée à partir des propriétés de l'ordinateur.

#### ► Tâche 2 : Vérifier la mémoire allouée à l'instance par défaut

- À l'aide des propriétés de l'instance de serveur AdventureWorks dans SSMS, récupérez les paramètres de mémoire serveur minimum et maximum.

#### ► Tâche 3 : Vérifier la mémoire allouée à l'instance MKTG

- À l'aide des propriétés de l'instance de serveur Proseware dans SSMS, récupérez les paramètres de mémoire serveur minimum et maximum.

#### ► Tâche 4 : Déterminer si l'allocation de mémoire est appropriée Si ce n'est pas le cas, apportez les modifications nécessaires à la configuration de la mémoire

- Examinez la configuration de la mémoire requise dans la documentation associée.
- Modifiez la configuration de la mémoire pour les deux instances de SQL Server conformément aux besoins. Vous devez déterminer la quantité de mémoire devant être utilisée pour toutes les instances de SQL Server et répartir la mémoire en fonction des spécifications de la documentation associée.

 **Remarque** Alors que la réduction de la mémoire serveur maximale pourrait nécessiter le redémarrage de SQL Server, vous n'avez pas besoin de redémarrer le serveur à ce stade de l'exercice.

**Résultats** : À la fin de cet exercice, vous devez avoir configuré la mémoire pour les instances de SQL Server.

## Exercice 2 : Tests de contrainte de pré-installation

### Scénario

Après avoir examiné la mémoire allouée sur le serveur, vous devez tester si le nouveau sous-système d'E/S est capable d'exécuter SQL Server. Dans cet exercice, vous devez employer l'utilitaire SQLIOSIM pour le test de contrainte afin de vérifier la stabilité des performances de SQL Server.

Les tâches principales de cet exercice sont les suivantes :

1. Configurer SQLIOSIM.
2. Exécuter SQLIOSIM.
3. Examiner les résultats de l'exécution SQLIOSIM.

#### ► Tâche 1 : Configurer SQLIOSIM

- Installez SQLIOSIM à partir du fichier F:\22462A\_Labs\22462A\_02\_PRJ\sqliosimx64.exe (vérifiez que vous utilisez l'option Exécuter en tant qu'administrateur).
- Configurer SQLIOSIM conformément aux exigences de la documentation associée.

#### ► Tâche 2 : Exécuter SQLIOSIM

- Exéutez SQLIOSIM basé sur les paramètres configurés.

#### ► Tâche 3 : Examiner les résultats de l'exécution SQLIOSIM

- Si des erreurs sont retournées en rouge, consultez les erreurs.
- Localisez le résumé final de chaque lecteur et notez la durée d'E/S moyenne en millisecondes.

**Résultats :** À la fin de cet exercice, vous avez exécuté un test de contrainte à l'aide de SQLIOSIM.

## Exercice difficile 3 : Vérifier des opérations d'E/S spécifiques (si le temps le permet)

### Scénario

Vous avez identifié une liste de charges de travail d'E/S et planifié de tester les performances du nouveau sous-système d'E/S. Dans cet exercice, vous devez exécuter l'utilitaire SQLIO pour vérifier les charges de travail d'E/S.

Les tâches principales de cet exercice sont les suivantes :

1. Installer l'utilitaire SQLIO.
2. Configurer et exécuter l'utilitaire SQLIO.

#### ► Tâche 1 : Installer l'utilitaire SQLIO

- Configurez l'utilitaire SQLIO à partir du fichier D:\22462A\_Labs\22462A\_02\_PRJ\SQLIO.msi (choisissez l'installation pour tous les utilisateurs).

#### ► Tâche 2 : Configurer et exécuter l'utilitaire SQLIO

- Examinez la documentation associée pour plus d'informations sur les tests de SQLIO à effectuer.
- Configurez le fichier C:\Program Files (x86)\SQLIO\param.txt conformément aux exigences de la documentation associée.
- Exécutez le programme sqlio.exe à partir d'une fenêtre de commande, pour tester les types d'E/S comme demandé dans la documentation associée. Pour chaque test, stockez les IOP et le débit obtenus. Notez également l'écart minimal, maximal et la latence moyenne pour chaque test.

**Résultats :** À la fin de cet exercice, vous devez avoir exécuté l'utilitaire SQLIO pour tester des fonctionnalités d'E/S spécifiques.

## Contrôle des acquis du module et éléments à retenir

- Questions de contrôle des acquis
- Meilleures pratiques

### Questions de contrôle des acquis

1. Quelle est la principale raison qui justifie que SQL Server ait besoin d'une grande quantité de mémoire ?
2. Pourquoi la planification et les tests avant l'installation sont-ils importants ?

### Meilleures pratiques

1. Comprendre l'architecture SQL Server.
2. Planifier les besoins en processeur, réseau et E/S pour le système spécifique.
3. Tester ces configurations sur le matériel disponible.

# Module 3

## Installation et configuration de SQL Server 2012

### Sommaire :

Leçon 1 : Préparation pour l'installation de SQL Server	3-3
Leçon 2 : Installation de SQL Server	3-16
Leçon 3 : Mise à niveau et automatisation de l'installation	3-25
Atelier pratique 3 : Installation et configuration de SQL Server	3-33

## Vue d'ensemble du module

- Préparation pour l'installation de SQL Server
- Installation de SQL Server
- Mise à niveau et automatisation de l'installation

Dans le module précédent, vous venez de découvrir comment préparer vos systèmes matériels pour vérifier leur compatibilité avec Microsoft® SQL Server®. Dans ce module, vous verrez les étapes de l'installation de SQL Server. Comme avec presque tous les projets, une bonne organisation est essentielle. Vous découvrirez comment préparer une installation puis comment réaliser l'installation.

Les grandes entreprises ont souvent besoin d'installer de nombreuses instances de SQL Server tout en veillant à ce que les installations soient faites de manière cohérente. Vous découvrirez des options pour automatiser l'installation de SQL Server, qui peuvent également être utiles pour l'installation en mode silencieux du produit lorsqu'il est nécessaire d'installer SQL Server en arrière-plan pour prendre en charge une application.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- préparer l'installation de SQL Server ;
- installer SQL Server ;
- mettre à niveau et automatiser l'installation de SQL Server.

## Leçon 1

# Préparation pour l'installation de SQL Server

- Configuration matérielle - généralités
- Configuration matérielle - mémoire
- Configuration logicielle - systèmes d'exploitation
- Configuration logicielle - généralités
- Détermination de l'emplacement des fichiers
- Configuration requise du compte de service
- Utilisation des classements
- Démonstration 1A : Utilisation des classements

Avant de commencer la procédure d'installation de SQL Server, il est important de savoir comment se conformer aux prérequis d'une installation réussie. Dans cette leçon, vous découvrirez les exigences spécifiques à SQL Server relativement aux plateformes matérielles et logicielles sur lesquelles il s'exécute, vous découvrirez comment déterminer l'emplacement des bases de données et comment configurer les comptes de service. L'objectif des comptes de service consiste à s'assurer qu'ils ont suffisamment de privilèges de fonctionner tout en réduisant l'allocation de toutes les autres autorisations aux comptes de service. Les applications peuvent requérir une configuration spécifique des classements. Vous découvrirez le fonctionnement des classements et la façon dont vos choix de configuration d'un classement du serveur peuvent affecter votre utilisation ultérieure du système.

## Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- décrire la configuration matérielle générale requise pour SQL Server ;
- expliquer la mémoire requise pour SQL Server ;
- décrire le système d'exploitation requis pour SQL Server ;
- décrire les autres logiciels requis pour prendre en charge SQL Server ;
- déterminer où placer des bases de données ;
- déterminer les autorisations appropriées et les privilèges des comptes de service SQL Server ;
- configurer les classements.

## Configuration matérielle - Généralités

- Processeurs
  - Pratiquement tous les processeurs sont maintenant suffisamment rapides
  - Le nombre de processeurs doit être pris en compte
- Disque
  - L'espace disque requis pour SQL Server est minime par rapport aux tailles de lecteur actuelles
  - La taille de la base de données utilisateur est un facteur déterminant
  - Les performances d'E/S disque sont critiques pour les performances de SQL Server
- Virtualisation
  - La virtualisation de SQL Server est maintenant très courante

### Points clés

Dans les versions antérieures de SQL Server, il était nécessaire de prendre en compte la configuration requise minimale du processeur, de l'espace disque et de la mémoire. De nos jours, il est inutile de se préoccuper de la vitesse des processeurs ou des contraintes d'espace disque pour les composants SQL Server. Même le processeur le plus lent d'un portable récent est suffisamment rapide pour répondre aux exigences minimales de SQL Server.

### Processeurs

Dans les environnements d'entreprise, le nombre de processeurs est maintenant un problème beaucoup plus sérieux. Bien qu'il puisse sembler souhaitable d'ajouter autant d'UC que possible, il est important de considérer un compromis entre le nombre de processeurs et les coûts de licence. En outre, toutes les architectures ne prennent pas en charge l'ajout de processeurs. L'ajout de ressources de processeur peut impliquer par la suite des mises à niveau de l'architecture des systèmes informatiques, et pas simplement des processeurs ajoutés.

Alors que la prise en charge de l'utilisation des processeurs Itanium était possible pour certains composants des versions antérieures de SQL Server, SQL Server 2012 ne prend pas en charge l'utilisation de ces processeurs.

### Disque

La configuration matérielle requise pour SQL Server indique l'espace disque nécessaire pour installer le produit. Ces valeurs sont toutefois rarement pertinentes car la taille des bases de données utilisateur rend généralement l'espace occupé par le produit complètement négligeable.

Les performances du sous-système de disque sont cependant essentielles. Un système SQL Server « standard » est aujourd’hui axé sur les E/S, s'il est configuré correctement et fonctionne normalement. Notez qu'un goulot d'étranglement n'est pas en soi-même une mauvaise chose. Tout système informatique affecté à n'importe quelle tâche présente un goulot d'étranglement quelque part. Si un autre composant du serveur est le goulet d'étranglement (plutôt que le sous-système d'E/S), il y a généralement un autre problème sous-jacent à résoudre. Il peut s'agir d'un manque de mémoire ou d'une affaire plus subtile comme un problème de recompilation (autrement dit, une situation où SQL Server recompile constamment le code). La mémoire requise par SQL Server est présentée dans la rubrique suivante.

### **Virtualisation**

Il existe sur le marché une forte résistance à la virtualisation de SQL Server. Bien que cette résistance diminue rapidement, une grande partie en est périmée et mal orientée. Souvent, la résistance est basée sur des hypothèses de virtualisation du sous-système d'E/S entier. Alors que la virtualisation du serveur peut fournir de bons résultats, virtualiser le sous-système d'E/S entier est rarement une bonne solution lors de l'utilisation SQL Server. La virtualisation de SQL Server sera présentée dans un module ultérieur.

**Question :** si vous devez continuer de prendre en charge une version plus ancienne de SQL Server, quelle serait la bonne méthode à adopter ?

## Configuration matérielle - Mémoire

Édition SQL Server 2012	Mémoire maximale
Enterprise	Maximum du système d'exploitation
Business Intelligence	64 Go
Web	64 Go pour le moteur de base de données, 4 Go pour Reporting Services
Express	1 Go
Express with Advanced Services	1 Go pour le moteur de base de données, 4 Go pour Reporting Services

- Les serveurs 32 bits ont limité l'espace d'adresse « visible »

### Points clés

La disponibilité d'une grande quantité de mémoire utilisable par SQL Server est l'un des facteurs les plus importants du dimensionnement des systèmes.

Bien que SQL Server s'exécute avec des quantités de mémoire relativement petites, lorsque des problèmes de configuration de la mémoire surviennent, ces quantités ont tendance à prendre les valeurs maximales et non les valeurs minimales. Par exemple, SQL Server Express Edition n'utilisera pas plus de 1 Go de mémoire, indépendamment de la quantité de mémoire physique installée dans le système.

### Systèmes et mémoire 64 bits et 32 bits

La majorité des serveurs installés aujourd'hui sont des serveurs 64 bits. Les serveurs 64 bits ont un espace d'adressage unique qui peut accéder directement à de grandes quantités de mémoire.

La plus grande difficulté que l'on rencontre avec les serveurs 32 bits réside dans le fait que la mémoire externe à l'espace d'adressage « visible » de 4 Go (la mémoire accessible directement) est accessible via Address Windowing Extensions (AWE).

 **Remarque** Alors que les versions antérieures de SQL Server permettaient l'utilisation de la mémoire accessible par AWE pour la mise en cache des pages de données, SQL Server 2012 ne prend plus en charge l'utilisation de la mémoire AWE.

**Question** : pouvez-vous suggérer une raison pour laquelle il peut s'avérer nécessaire de mettre en œuvre des serveurs 32 bits ?

## Configuration logicielle - Systèmes d'exploitation

- Système d'exploitation (récapitulatif général)
  - Windows Server 2008 R2 SP1
  - Windows Server Core 2008 R2 SP1
  - Windows Server 2008 SP2
  - Windows 7 SP1
  - Windows Vista SP2
- Évitez l'installation sur les contrôleurs de domaine
- Utilisez plutôt SQL 64 bits sur les systèmes d'exploitation 64 bits mais WOW est pris en charge

### Points clés

Sur cette diapositive, la liste présente un récapitulatif général des systèmes d'exploitation pris en charge par SQL Server. Bien qu'il soit possible d'installer une version de SQL Server sur différents systèmes d'exploitation, toutes les éditions de SQL Server ne peuvent être installées sur n'importe quel système d'exploitation.

### Systèmes client

Bien qu'il soit possible d'installer des versions de SQL Server sur des systèmes d'exploitation clients, tels que Windows 7® (SP1) et Windows Vista® (SP2), le produit est réellement conçu pour une utilisation sur les systèmes d'exploitation serveurs tels que la série Windows Server®.

### Configuration requise de la version

La plupart des éditions haut de gamme de SQL Server nécessitent également des éditions plus haut de gamme de Windows. Consultez la documentation en ligne pour obtenir une liste exacte des versions et éditions prises en charge.

Il est possible d'installer des applications 32 bits sur des systèmes d'exploitation 64 bits via le système d'émulation Windows on Windows (WOW) qui est disponible sur les systèmes d'exploitation 64 bits. Il est possible d'installer des versions 32 bits de SQL Server sur des systèmes d'exploitation 64 bits, mais il est préférable d'installer des versions 64 bits de SQL Server sur les systèmes 64 bits.

Le programme d'installation de SQL Server sur Windows Server Core est maintenant pris en charge.

## Contrôleurs de domaine

Il est fortement recommandé d'éviter d'installer SQL Server sur un contrôleur de domaine. Si vous essayez d'installer SQL Server sur un contrôleur de domaine, le programme d'installation ne sera pas bloqué mais des restrictions s'y appliqueront.

- Il est impossible d'exécuter des services SQL Server sur un contrôleur de domaine sous un compte de service local ou un compte de service réseau.
- Une fois SQL Server installé sur un ordinateur, vous ne pourrez pas changer l'ordinateur de membre de domaine en contrôleur de domaine. Vous devrez désinstaller SQL Server avant de changer l'ordinateur - hôte en contrôleur de domaine.
- Une fois SQL Server installé sur un ordinateur, vous ne pourrez pas changer l'ordinateur de contrôleur de domaine en membre de domaine. Vous devrez désinstaller SQL Server avant de changer l'ordinateur - hôte en membre de domaine.
- Les instances de basculement de cluster SQL Server ne sont pas prises en charge lorsque les nœuds du cluster sont des contrôleurs de domaine.
- Le programme d'installation de SQL Server ne peut pas créer de groupes de sécurité ni configurer de comptes de service SQL Server sur un contrôleur de domaine en lecture seule. Dans ce scénario, le programme d'installation échoue.

## Configuration logicielle – Généralités

- .NET Framework
  - .NET Framework 4.0 est requis
  - Certains composants utilisent également .NET Framework 3.5 SP1
- PowerShell 2.0
- SQL Server Native Client
- Fichiers de prise en charge du programme d'installation de SQL Server
- Windows Installer 4.5 ou version ultérieure
- Internet Explorer 6 SP1 ou version ultérieure
- Logiciels réseau
  - Mémoire partagée
  - Canaux nommés
  - TCP/IP

### Points clés

Dans les versions antérieures, le programme d'installation de SQL Server préinstallait la plupart des outils nécessaires dans le cadre du processus d'installation. Ce n'est plus le cas aujourd'hui ; .NET Framework et Powershell doivent être préinstallés avant d'exécuter le programme d'installation. Le programme d'installation de SQL Server installe SQL Server Native Client (SNAC) et les fichiers de support du programme d'installation de SQL Server. Toutefois, pour réduire la durée d'installation de SQL Server, en particulier dans les environnements de production très actifs, il est utile de préinstaller ces composants pendant un temps d'arrêt planifié disponible. Les composants tels que .NET Framework requièrent souvent un redémarrage après l'installation. La pré-installation de ces composants peut donc réduire le temps d'inactivité causé par les installations ou aux mises à niveau.

### Configuration logicielle générale

Le programme d'installation de SQL Server est basé sur la technologie Windows Installer 4.5. Vous devez envisager d'installer Windows Installer 4.5 avant l'installation de SQL Server, afin de réduire le temps d'installation de ce dernier.

Plusieurs composants de SQL Server impliquent le navigateur Internet Explorer®. Ces composants sont le complément Microsoft Management Console (MMC), SQL Server Management Studio (SSMS), Business Intelligence Design Studio (BIDS), le concepteur de rapports de BIDS ainsi que toute forme d'aide HTML.

Les communications de SQL Server peuvent être basées sur la mémoire partagée, les canaux nommés ou les protocoles TCP/IP. Le protocole VIA n'est plus pris en charge.

## Détermination de l'emplacement des fichiers

Type de fichier	Considération
Fichiers de données primaires et secondaires	<ul style="list-style-type: none"><li>Si vous avez des tables qui sont fréquemment utilisées ensemble, vous devez les placer dans des groupes de fichiers sur des lecteurs physiques distincts</li></ul>
Journaux des transactions	<ul style="list-style-type: none"><li>Créez le journal des transactions sur un contrôleur RAID ou un disque physiquement séparé</li></ul>
tempdb	<ul style="list-style-type: none"><li>Placez la base de données tempdb sur un sous-système du lecteur de disque rapide pour garantir des performances optimales</li></ul>

**Planification des disques et emplacement des fichiers**

**Les éléments les plus importants à prendre en compte sont le nombre de disques ou piles accessibles par un volume particulier et la vitesse des lecteurs impliqués**

### Points clés

Le placement des bases de données est présenté plus loin dans le module 4, de même que les informations sur l'utilisation des groupes de fichiers ; mais avant d'installer SQL Server, il est utile de comprendre les principaux concepts liés à l'emplacement des fichiers. Les groupes de fichiers sont des noms donnés aux jeux de fichiers.

#### Fichiers de données primaires et secondaires

De manière générale, si vous avez des tables qui sont fréquemment utilisées ensemble, vous devez les placer dans des groupes de fichiers et des lecteurs physiques distincts. Vous devrez peut-être altérer cette recommandation si les fichiers ne sont pas similaires en taille.

Bien qu'un gros disque soit susceptible de gérer toutes vos données, il ne peut exécuter qu'une seule E/S à la fois. Par conséquent, vous devez considérer non seulement combien d'espace disque est disponible mais aussi quel est le nombre de lecteurs physiques (ou « piles ») nécessaires pour obtenir le nombre d'opérations d'E/S par seconde requis par vos applications.

#### tempdb

La base de données tempdb sert à gérer des objets temporaires dans un système SQL Server. Placez la base de données tempdb sur un sous-système d'E/S rapide pour garantir des performances optimales. Répartissez la base de données tempdb sur plusieurs disques pour obtenir de meilleures performances. Si possible, assurez-vous que la base de données tempdb se trouve sur des disques distincts des bases de données utilisateur. Bien que la base de données tempdb puisse se trouver située au même endroit que les données dans de nombreux cas, pour les plus grands systèmes qui font une utilisation intensive de tempdb il est préférable de regrouper cette base de données sur un ensemble de disques distinct afin d'optimiser les performances.

En général vous devez faire correspondre le nombre de fichiers tempdb au nombre de processeurs du système (jusqu'à une limite raisonnable de huit), car cela évite tout problème de contention avec les structures système dans les fichiers.

## Journaux des transactions

Créez le journal des transactions sur un contrôleur RAID ou un disque physiquement distinct. Les journaux des transactions sont écrits de manière séquentielle. Par conséquent, l'utilisation d'un disque dédié indépendant garantit que les têtes de disque restent en place pour l'opération d'écriture suivante. Pour cette raison, les plus petits systèmes pourront utiliser un disque en miroir unique pour le journal des transactions. Un disque physique en miroir unique doit pouvoir prendre en charge jusqu'à 1000 transactions par seconde environ, selon la vitesse du disque. Les systèmes nécessitant une plus grande prise en charge doivent répartir le journal des transactions sur un contrôleur RAID 10 pour obtenir des performances optimales.

**Question :** comment planifiez-vous les lecteurs et l'emplacement des fichiers pour votre organisation ?

## Configuration requise du compte de service

- Prenez en compte la configuration requise du compte de service
  - Créez des comptes de service qui ont le moins de priviléges
  - Choisissez généralement des comptes de domaine
  - Le compte Service local peut convenir dans certains environnements
- Si les comptes sont définis à l'aide du programme d'installation de SQL Server, les autorisations minimales sont configurées automatiquement
- Chaque compte de service a des besoins en autorisation différents
  - Consultez la documentation en ligne pour obtenir des détails sur les autorisations spécifiques pour chaque compte
  - Utilisez un compte différent pour chaque service

### Points clés

Il n'existe pas de règle pour déterminer la meilleure manière de configurer les comptes de service de SQL Server, mais il existe un ensemble général de principes à suivre.

### Principe des privilèges minimum

Le principe de base est le suivant : ne pas allouer aux comptes de service plus de possibilités qu'ils n'en ont besoin. Bien qu'il soit facile de faire pour chaque service un compte administrateur, cela n'est pas raisonnable du point de vue de la sécurité. Si vous créez un compte avec peu de priviléges, le programme d'installation de SQL Server le configurera automatiquement avec les autorisations et priviléges requis par le compte de service. SQL Server effectue cette opération en créant des groupes, en assignant les autorisations requises aux groupes, puis en ajoutant les comptes de service en tant que membres de ces groupes.

### Comptes de domaine contre comptes locaux

Dans la plupart des cas, vous devez utiliser un compte de domaine. Il existe toutefois des circonstances dans lesquelles il peut être plus approprié d'utiliser un compte local. Cela correspond à des situations où vous souhaitez spécifiquement limiter l'accès du compte à l'ordinateur local.

Si vous décidez d'utiliser un compte local au lieu d'un compte de domaine, il est important de savoir que le compte de service local n'est pas identique au compte système local. Le compte de service local est configuré avec les mêmes autorisations que les autres utilisateurs authentifiés, tandis que le compte système local dispose de priviléges administratifs et ne doit en principe pas être utilisé comme compte de service pour les services SQL Server.

## Utilisation des classements

**Un classement code les règles qui régissent l'utilisation appropriée de caractères pour une langue, telle que le grec, ou un alphabet, tel que le Latin1\_General**

- Classements Windows
  - Basés sur les règles relatives à des paramètres régionaux Windows associés
- Classements SQL Server
  - Mise en correspondance du numéro de page de code et de l'ordre de tri qui peuvent avoir été spécifiés dans les versions antérieures de SQL Server
  - Classement et règles de tri par défaut
    - Le classement par défaut s'applique si vous ne définissez pas de règles de classement ni de tri

### Points clés

Microsoft SQL Server 2012 prend en charge de nombreux classements. Un classement code les règles qui régissent l'utilisation appropriée de caractères pour une langue, telle que le grec ou le polonais, ou un alphabet, tel que le Latin1\_General (l'alphabet utilisé par les langues de l'Europe occidentale).

### Sélection d'un classement de serveur

Lorsque vous installez SQL Server, vous désignez un classement et vous sélectionnez des règles de l'ordre de tri. Le terme classement fait référence à un ensemble de règles qui détermine la manière dont les données sont comparées et triées. Les données de caractères sont triées à l'aide de règles qui définissent la séquence correcte de caractères. Vous pouvez spécifier le respect de la casse, les marques d'accentuation, les types de caractères Kana et la largeur des caractères, lors du tri des données.

Par exemple, le mot « AdventureWorks » doit-il être trié comme s'il était identique au mot « adventureworks » ? Pour un exemple légèrement plus subtil, le mot « café » doit-il être trié comme s'il était identique au mot « cafe » ?

L'effet d'un classement incorrect se fait particulièrement sentir lorsque vous utilisez des objets temporaires créés dans la base de données tempdb. Si une base de données d'application est configurée avec un classement différent sur le serveur, il convient de prendre des précautions lors de l'utilisation d'objets temporaires tels que des tables temporaires. Dans la démonstration 1A, vous verrez l'effet que les classements peuvent avoir sur la capacité à exécuter des requêtes impliquant des objets temporaires.

Modifier le classement du serveur après l'installation n'est pas chose simple. Chaque structure système qui a été triée (peut-être dans les index) doit être recréée. Toutes les bases de données système doivent être recréées.

Même s'il est possible de modifier le classement par défaut d'une base de données utilisateur après qu'elle a été créée, les colonnes et les données existantes dans la table n'en sont pas affectées. Le paramètre de classement par défaut affectera uniquement les objets nouvellement créés.

SQL Server prend en charge deux types de classements : les classements Windows et les classements SQL Server.

### Classements Windows

Lorsque vous désignez des classements Windows, le système d'exploitation définit un ensemble de règles pour le stockage et le tri des données de caractères qui s'appuient sur les règles correspondant aux paramètres régionaux Windows associés. Les règles du classement Windows spécifient l'alphabet ou la langue utilisés lorsque le tri du dictionnaire est appliqué et la page de code utilisée pour stocker les données de caractères non-Unicode.

### Classements SQL Server

Lorsque vous désignez des classements SQL Server, SQL Server met en correspondance les attributs des combinaisons communes de numéros de pages de code et d'ordres de tri qui peuvent avoir été spécifiés dans les versions antérieures de SQL Server. Les classements SQL Server contrôlent la page de code utilisée pour stocker les données non-Unicode et les règles de tri des données Unicode et non-Unicode.

Chaque classement SQL Server spécifie trois propriétés :

- l'ordre de tri à utiliser pour les types de données Unicode (comme nchar, nvarchar et nvarchar(max)) ;
- l'ordre de tri à utiliser pour les types de données de caractères non-Unicode (char, varchar et varchar(max)) ;
- la page de codes utilisée pour stocker les données de caractères non-Unicode.

Les classements SQL Server sont conservés principalement à des fins de compatibilité descendante. Lorsque vous concevez des applications, choisissez le classement Windows approprié.

SQL Server 2012 a également introduit quelques nouveaux classements qui prennent en charge des jeux de caractères supplémentaires. On les désigne sous le nom de classements SC et ils prennent en charge une beaucoup plus grande variété de caractères. Les classements SC ne sont pas pris en charge en tant que classements du serveur mais peuvent être utilisés dans des bases de données. Vous pouvez identifier un classement SC par son nom qui se termine par le suffixe \_SC.

**Question** : quelle est la différence entre un classement Respecter les accents et un classement Non-respect des accents ?

## Démonstration 1A : Utilisation des classements

Dans cette démonstration, vous allez apprendre comment les classements affectent les requêtes T-SQL

### Procédure de démonstration

1. Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre **Se conn** au serveur, tapez **Proseware**, puis cliquez sur **Se conn**. Dans le menu **Fichier** cliquez sur **Ouvrir**, sur **Projet/Solution**, naviguez jusqu'à **F:\22462A\_Labs\22462A\_03\_PRJ\22462A\_03\_PRJ.ssmssqln** puis cliquez sur **Ouvrir**.
3. Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** à partir de l'**Explorateur de solutions**.
4. Ouvrez le fichier de script **11 – Demonstration 1A.sql**.
5. Suivez les instructions contenues dans les commentaires du fichier de script.

## Leçon 2

# Installation de SQL Server

- Vue d'ensemble du processus d'installation
- Outil d'analyse de configuration système
- Contrôles post-installation
- Démonstration 2A : Utilisation de l'outil d'analyse de configuration système

Une fois les décisions prises sur la manière dont SQL Server doit être configuré, vous pouvez procéder à l'installation de SQL Server. Dans cette leçon, vous verrez les phases de l'exécution du programme d'installation, puis vous verrez comment SQL Server vérifie la compatibilité du système à l'aide d'un outil appelé Outil d'analyse de configuration système.

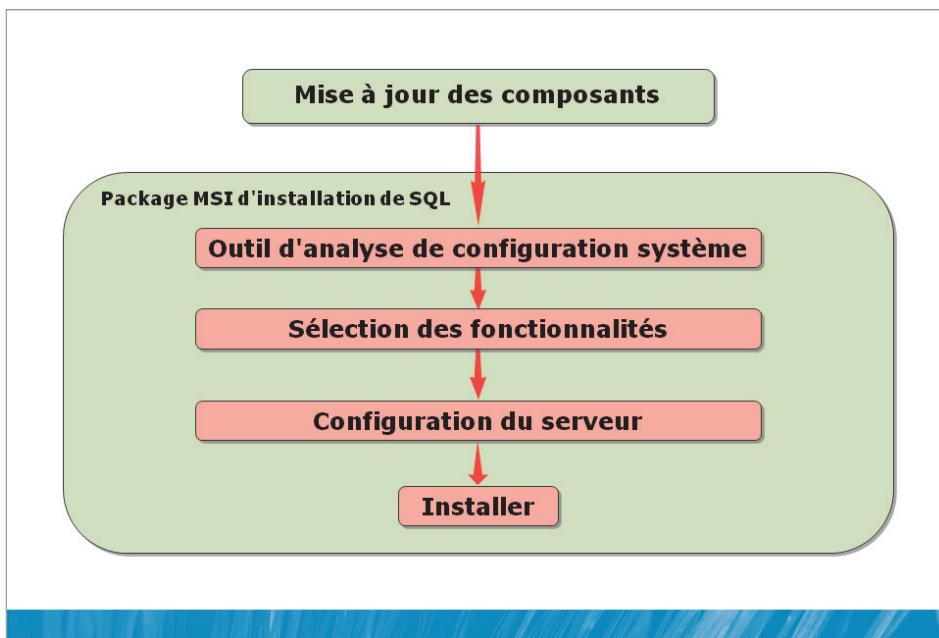
Dans le cas de la plupart des utilisateurs, le programme d'installation signalera que tout a été installé comme prévu. Pour les rares situations où cela n'aboutit pas, vous apprendrez également comment réaliser les contrôles et la résolution des problèmes post-installation.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- décrire les phases du processus d'installation de SQL Server ;
- expliquer le rôle de l'outil d'analyse de configuration système ;
- réaliser les contrôles post-installation.

## Vue d'ensemble du processus d'installation



### Points clés

Vous pouvez installer les composants requis par SQL Server 2012 en exécutant le programme d'installation de SQL Server 2012. Les exceptions à cette règle sont StreamInsight et Master Data Services, qui doivent être installées séparément. StreamInsight et Master Data Services sont livrés avec des programmes d'installation distincts.

Le processus d'installation principal de SQL Server 2012 est composé de deux phases essentielles : la mise à jour des composants et le package MSI d'installation de SQL. Comprendre le processus d'installation vous aidera à planifier et à effectuer les installations de SQL Server 2012.

### Mise à jour des composants

Au cours de la phase d'installation de mise à jour des composants, le programme d'installation de SQL Server 2012 recherche les composants ci-dessous :

- Windows Installer 4.5 ;
- .NET Framework ;
- SQL Server Native Client (sqlIncli.msi) ;
- Windows PowerShell ;
- autres fichiers de prise en charge de l'installation de SQL Server.

### Package MSI d'installation de SQL

Après la phase de mise à jour des composants, il convient d'utiliser Windows Installer pour installer le package MSI d'installation de SQL. Au cours de cette phase, le programme d'installation effectue les tâches suivantes :

- il détermine le type d'installation (instance par défaut ou instance nommée) ;
- il analyse l'ordinateur à l'aide de l'outil d'analyse de configuration système ;
- il détermine les fonctionnalités à installer (y compris les mises à jour automatiques si nécessaire) et exécute le programme d'installation approprié.

Le champ ID d'instance contrôle le nom des répertoires pour lesquels les fonctionnalités seront installées. Par défaut, c'est le nom de l'instance mais il peut être modifié. La modification d'une instance nommée ne change pas automatiquement le champ ID d'instance si vous avez déjà tapé un ID d'instance.

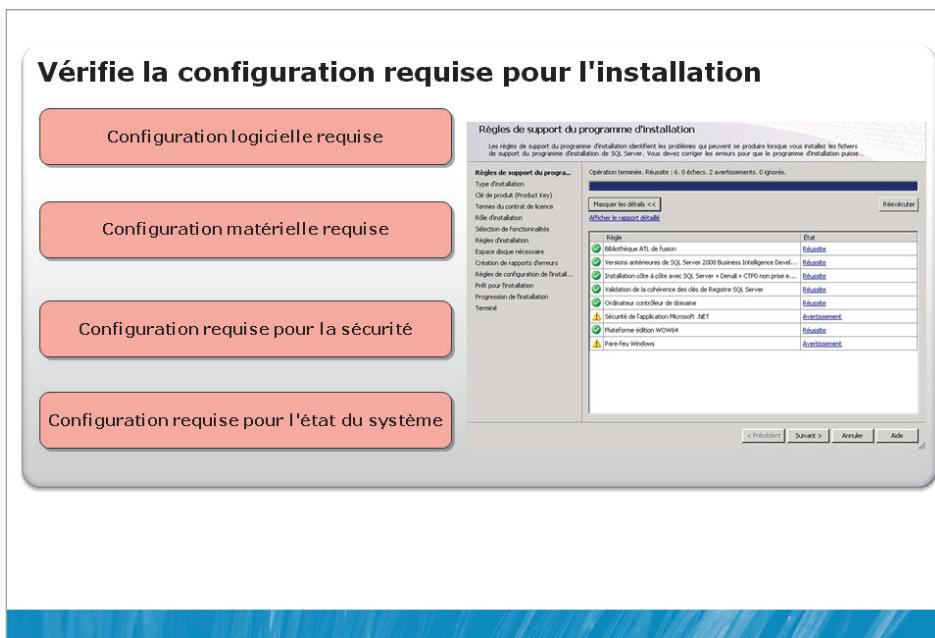
### Configuration du serveur

Le groupe BUILTIN\Administrateurs n'obtient plus le contrôle administratif dans SQL Server, alors que c'était le cas dans les versions antérieures. Toutefois, il est souvent acceptable d'ajouter BUILTIN\Administrateurs comme un groupe doté des autorisations d'administrateur. Cela correspond à tous les administrateurs de l'ordinateur. Analysis Services est configuré de façon similaire.

Les utilisateurs demandent souvent la possibilité de bloquer l'accès de SQL Server pour les administrateurs de l'ordinateur. Même si en excluant le groupe BUILTIN\Administrateurs vous pouvez empêcher des tentatives occasionnelles d'accéder au contenu du serveur, vous ne devez pas supposer que les administrateurs de l'ordinateur ne pourront pas accéder à SQL Server s'ils y sont déterminés.

Pendant la phase de configuration du serveur, vous avez également la possibilité de définir diverses options de configuration du système. Vous pouvez par exemple choisir d'activer ou non la fonction Filestream ou choisir les dossiers par défaut à utiliser avec les bases de données utilisateur.

## Outil d'analyse de configuration système



### Points clés

Dans le cadre de l'installation de SQL Server, l'outil d'analyse de configuration système (SCC) analyse l'ordinateur sur lequel SQL Server est installé. L'outil d'analyse de la configuration système recherche les conditions qui empêchent l'installation de SQL Server. Avant que le programme d'installation ne démarre l'Assistant Installation de SQL Server, l'outil d'analyse de configuration système extrait l'état de chaque élément vérifié, compare le résultat avec les conditions requises et vous guide pour la correction des problèmes de blocage. Si un résultat était susceptible de bloquer l'installation de SQL Server mais d'en limiter les fonctionnalités, vous seriez averti.

### Configuration logicielle requise

Dans la phase de configuration logicielle requise, de nombreux aspects de la configuration logicielle sont contrôlés. On vérifie notamment si le système d'exploitation est un système d'exploitation pris en charge pour la version et l'édition de SQL Server en cours d'installation. Le niveau du service pack du système d'exploitation est également contrôlé et tout système d'exploitation non pris en charge bloquerait la poursuite de l'installation.

De plus, dans le cadre de la phase de configuration logicielle requise, le SCC vérifie la présence du service WMI (Windows Management Instrumentation). Le service WMI doit être disponible. L'échec du contrôle pour cet élément bloquerait également l'installation.

### Configuration matérielle requise

L'outil d'analyse de configuration système avertira l'utilisateur mais ne bloquera pas l'installation si le contrôle de la RAM minimum ou recommandée échoue. Les besoins en mémoire sont uniquement pour SQL Server et ne reflètent pas les besoins supplémentaires en mémoire du système d'exploitation.

L'outil d'analyse de configuration système avertira l'utilisateur mais ne bloquera pas l'installation si le contrôle de la vitesse de processeur minimum ou recommandée échoue.

### Spécifications de sécurité

L'utilisateur qui exécute le programme d'installation doit disposer de priviléges d'administrateur sur l'ordinateur sur lequel SQL Server est installé.

### Conditions requises pour l'état du système

L'installation de SQL Server ne peut pas être exécutée si les fichiers requis par l'installation sont verrouillés par d'autres services, processus ou applications. L'échec du contrôle pour cet élément bloquera l'installation.

### Rapport

Après avoir terminé ces vérifications, l'outil SCC génère un rapport qu'il est possible d'afficher et d'enregistrer. Ce rapport inclut des informations sur tout problème susceptible d'empêcher l'installation et recommande des solutions. Il inclut également des avertissements et des recommandations, telles que des correctifs ou des configurations de sécurité recommandés, pour les problèmes qui n'empêcheront pas l'installation mais qui engendreront des difficultés. Dans la plupart des scénarios, vous devez résoudre ces problèmes et exécuter à nouveau le programme d'installation plutôt que tenter de les résoudre une fois l'installation terminée.

## Contrôles post-installation

- Vérifiez que les services SQL Server installés sont en cours d'exécution
  - Le Gestionnaire de configuration SQL Server est l'emplacement approprié
- Si des erreurs se produisent pendant l'installation, les fichiers journaux sont utiles
  - Chaque installation crée un nouveau dossier de journal horodaté
  - L'emplacement est :  
%programfiles%\Microsoft SQL Server\110\Démarrage du programme d'installation\Journal
- Les détails, le récapitulatif et les journaux supplémentaires sont créés si nécessaire à chaque phase
  - Texte des règles globales
  - Mise à jour des composants
  - Action demandée par l'utilisateur

### Points clés

Une fois que SQL Server est installé, le contrôle le plus important consiste à s'assurer que tous les services SQL Server s'exécutent. Vous devez pour ce faire utiliser le nœud Services SQL Server dans le Gestionnaire de configuration SQL Server.



**Remarque** Les services SQL Server ont des noms qui peuvent différer légèrement de leur nom affiché. Vous pouvez afficher le nom réel du service en consultant sa page de propriétés.

Les administrateurs n'ont généralement pas besoin de contrôler a posteriori le contenu des fichiers journaux d'installation de SQL Server, car le programme d'installation affiche les erreurs qui se produisent et tente en ce cas d'annuler le processus d'installation de SQL Server accompli à ce point.



**Remarque** Lorsqu'une erreur se produit lors de la phase d'installation de SQL Server, l'installation de SNAC (SQL Server Native Access Client) et des composants d'installation n'est pas annulée.

Les fichiers journaux d'installation se trouvent ici :

```
%programfiles%\Microsoft SQL Server\110\Setup Bootstrap\Log
```

En général, les administrateurs doivent uniquement consulter les fichiers journaux d'installation dans deux cas :

- lorsqu'ils tentent d'établir pourquoi l'installation a échoué, si les informations fournies par le programme d'installation sont insuffisantes ;
- lorsqu'ils collaborent avec le support technique Microsoft pour résoudre un problème d'installation.

Les fichiers journaux contiennent des détails sur les trois phases essentielles. La documentation en ligne contient des informations sur le contenu de chaque section des fichiers journaux.

**Question** : si vous découvrez après l'installation que vous avez utilisé un compte de service erroné ou inadéquat pour SQL Server, quel outil utilisez-vous pour modifier le compte ?

## Démonstration 2A : Utilisation de l'outil d'analyse de configuration système

Dans cette démonstration, vous allez apprendre comment exécuter la première phase de l'installation de SQL Server, c'est-à-dire l'outil d'analyse de configuration système. Vous découvrirez les règles que l'outil SCC implémente

### Procédure de démonstration

1. Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.
2. Dans la liste **Ordinateurs virtuels** du Gestionnaire Hyper-V, cliquez avec le bouton droit sur l'ordinateur virtuel **22462A-MIA-SQL1** puis cliquez sur **Paramètres**.
3. Dans la fenêtre Paramètres de 22462A-MIA-SQL1, dans la liste **Matiériel**, développez **Contrôleur IDE 1** puis cliquez sur **Lecteur DVD**.
4. Dans le volet propriétés du **Lecteur DVD**, cliquez sur **Fichier image**, puis sur **Parcourir**.
5. Naviguez jusqu'au fichier **C:\Program Files\Microsoft Learning\1077XA\Drives\22462A-MIA-SQL1\Virtual Hard Disks\SQLFULL\_ENU.iso** puis cliquez sur **Ouvrir**.
6. Dans la fenêtre des paramètres de 22462A-MIA-SQL1, cliquez sur **OK**.
7. Dans la fenêtre Ordinateur virtuel, au sein de la fenêtre de lecture automatique (qui devrait normalement s'afficher), cliquez sur **Exécuter SETUP.EXE** et attendez que l'installation de SQL Server démarre.
8. Dans la fenêtre Centre d'installation SQL Server, cliquez sur **Outil d'analyse de configuration système** dans la liste des options disponibles sous l'onglet **Planification**.
9. Dans la fenêtre Règles de support du programme d'installation, notez la liste des règles qui ont été cochées.

10. Cliquez sur l'avertissement dans la colonne **Statut** de la règle **Sécurité de l'application Microsoft .NET**.
11. Dans la fenêtre Résultat de la vérification de la règle, lisez les détails de l'avertissement et cliquez sur **OK**.
12. Dans la fenêtre Règles de support du programme d'installation, cliquez sur **OK**.

## Leçon 3

# Mise à niveau et automatisation de l'installation

- Mise à niveau de SQL Server
- Maintenance de SQL Server
- Installation sans assistance
- Démonstration 3A : Création d'un fichier d'installation sans assistance

Il est fréquent de devoir mettre à niveau SQL Server à partir d'une version antérieure au lieu de l'installer. Dans cette leçon, vous verrez les avantages et les limites des méthodes disponibles pour effectuer des mises à niveau.

Toutes les installations de SQL Server ne sont pas réalisées individuellement par un administrateur. Les grandes entreprises ont besoin de pouvoir configurer de nombreuses instances de SQL Server et de pouvoir les installer de façon très cohérente. Des options d'installation sans assistance sont fournies pour SQL Server et peuvent être utilisées pour répondre aux besoins de ce scénario. Les installations sans assistance peuvent également être exécutées en mode silencieux. Cela permet d'effectuer l'installation de SQL Server pendant l'installation d'autres applications.

## Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- mettre à niveau SQL Server ;
- appliquer les correctifs de SQL Server, les mises à jour cumulatives et les services packs ;
- exécuter une installation sans assistance.

## Mise à niveau de SQL Server

Mise à niveau sur place	Mise à niveau côté à côté
• Plus facile, principalement automatisé	• Un contrôle plus précis sur le processus
• Données système mises à niveau	• Peut être utilisé pour effectuer la migration test
• Aucun matériel supplémentaire	• Restauration relativement simple
• Applications pointant vers les mêmes noms	• Peut utiliser le basculement/changement
<ul style="list-style-type: none"><li>Les installations côté à côté de certaines versions de SQL Server nécessitent de prendre en compte des éléments supplémentaires<ul style="list-style-type: none"><li>Les composants partagés sont mis à niveau vers la version la plus récente si le numéro de build principal des deux installations est identique (c'est-à-dire : 2008 et 2008 R2)</li></ul></li></ul>	

### Points clés

Il existe deux méthodes de base pour réaliser les mises à niveau de SQL Server. Il n'existe aucune méthode privilégiée. Chaque méthode présente des avantages et les limites et est adaptée à certaines circonstances.

#### Mises à niveau sur place

Les mises à niveau sur place ont lieu lorsque la version installée SQL Server est directement remplacée par une nouvelle version. Il s'agit d'une méthode plus simple et très automatisée de mise à niveau, mais elle est plus risquée. Si une mise à niveau échoue, il est beaucoup plus difficile de retourner à l'état de fonctionnement antérieur. Ce risque ne peut pas être ignoré pour la plupart des utilisateurs.



**Remarque** Lorsque vous étudiez le risque, vous devez considérer que ce n'est pas forcément la mise à niveau de SQL Server qui échoue. Même si la mise à niveau de SQL Server fonctionne comme prévu, mais qu'en revanche l'application ne fonctionne pas correctement sur la nouvelle version de SQL Server, la nécessité de récupérer rapidement sera tout aussi impérative.

Les mises à niveau sur place ont l'avantage de réduire les besoins en termes de ressources matérielles supplémentaires et évitent la nécessité de rediriger les applications clientes qui sont configurées pour fonctionner avec le serveur existant.

## Mises à niveau côté à côté

Les mises à niveau côté à côté sont une solution plus sécurisée, car le système d'origine reste en place et peut être rapidement remis en production au cas où un problème de mise à niveau survenait. Toutefois, les mises à niveau côté à côté demandent plus de travail et davantage de ressources matérielles.

Pour effectuer une mise à niveau côté à côté, vous aurez besoin de suffisamment de ressources matérielles pour assurer le fonctionnement du système d'origine et du nouveau système. Il y a deux risques communément associés aux mises à niveau côté à côté, liés au temps nécessaire pour copier toutes les bases de données utilisateur vers un nouvel emplacement et à l'espace requis pour stocker ces copies.

Même si la plupart des mises à niveau côté à côté sont effectuées sur des serveurs distincts, il est possible d'installer les deux versions de SQL Server sur le même serveur. Cependant, les mises à niveau côté à côté de versions du même numéro de build principal (par exemple SQL Server 2008 R2 et SQL Server 2008) sur le même serveur constituent un cas particulier. Étant donné que le numéro de version principale est identique, les versions distinctes des composants partagés ne pourront pas coexister sur le même serveur. Les composants partagés seront mis à niveau.

Toutes les versions de SQL Server ne sont pas prises en charge lors d'une installation côté à côté. Consultez la documentation en ligne pour une matrice des versions prises en charge en cas d'installation simultanée.

## Options hybrides

Il est également possible d'utiliser des éléments d'une mise à niveau sur place avec des éléments d'une mise à niveau côté à côté. Par exemple, plutôt que de copier toutes les bases de données utilisateur, après avoir installé la nouvelle version de SQL Server à côté de la version antérieure, et avoir migré tous les objets serveur tels que les connexions, vous pouvez détacher les bases de données utilisateur depuis l'instance de serveur ancienne et les rattacher à la nouvelle instance de serveur.

 **Remarque** Une fois que les bases de données utilisateur ont été jointes à une version plus récente de SQL Server, elles ne peuvent plus être rattachées à version antérieure, même si les paramètres de compatibilité de la base de données n'ont pas été mis à niveau. Il s'agit d'un risque qui doit être pris en compte lors de l'adoption d'une approche hybride.

## Maintenance de SQL Server

- SQL Server peut être configuré pour recevoir des mises à jour automatiques directement à partir de Microsoft Update
  - Il convient de prendre des précautions pour les environnements de production de grande taille
- Les commentaires de SQL Server doivent être enregistrés à l'adresse <http://connect.microsoft.com>
  - Notifications de bogue
  - Suggestions d'améliorations
- Les mises à jour de SQL Server sont publiées de plusieurs manières
  - Correctifs logiciels
  - Mises à jour cumulatives
  - Service Packs

### Points clés

Comme avec tous les logiciels, il est possible de rencontrer au cours du temps des problèmes avec SQL Server. Le groupe produits réagit très rapidement pour corriger les problèmes quand ils sont identifiés. Si vous rencontrez des problèmes dont vous souhaitez informer Microsoft, ou si vous avez des suggestions d'amélioration du produit, veuillez visiter le site <http://connect.microsoft.com>.

La façon la plus simple de maintenir SQL Server à jour consiste à activer les mises à jour automatiques du service Microsoft Update. Les grandes organisations ou celles dotées où sont mis en œuvre des processus de changement intenses doivent faire preuve de prudence en appliquant les mises à jour automatiques. Il est possible que les mises à jour doivent être appliquées à des environnements de test avant de les appliquer aux environnements de production.

Les mises à jour de SQL Server sont publiées de plusieurs manières :

- Des correctifs logiciels (également appelés correctif QFE(Quick Fix Engineering) sont publiés pour résoudre des problèmes urgents des clients. En raison des contraintes de temps, il n'est possible d'effectuer que des tests limités sur ces correctifs ; il convient donc de ne les appliquer qu'aux systèmes qui présentent le problème concerné.
- Les mises à jour cumulatives sont des versions périodiques de cumul des correctifs logiciels qui ont bénéficié en groupe de tests plus approfondis.
- Les services packs sont des versions périodiques sur lesquelles ont été pratiqués des tests de régression complets. Microsoft recommande d'appliquer les services packs à tous les systèmes après des tests organisationnels de niveau approprié.
- SQL Server 2008 R2 et versions ultérieures peuvent également avoir des services packs intégrés dans le processus d'installation pour éviter la nécessité de les installer ensuite.

## Installation sans assistance

- Peut installer SQL Server à partir de la ligne de commande

```
Setup.exe /q /ACTION=CompleteImage /INSTANCENAME=MYNEWINST  
/INSTANCEID=<MYINST> /SQLSVCAccount="<DomainName\UserName>"  
/SQLSVCPassword=""  
/SQLSysAdminAccounts="<DomainName\UserName>"  
/AGTSVCACCOUNT="NT AUTHORITY\Network Service"  
/ACCEPTSQLSERVERLICENSETERMS
```

- Peut également mettre à niveau SQL Server à partir de la ligne de commande

```
Setup.exe /q /ACTION=upgrade /INSTANCENAME=MSSQLSERVER  
/RSUPGRADEDATABASEACCOUNT="that can connect to the report server during upgrade>"  
/RSUPGRADEPASSWORD="upgrade account>" /ISSVCAccount="NT Authority\Network Service"  
/ACCEPTSQLSERVERLICENSETERMS
```

### Points clés

Dans de nombreuses organisations, les fichiers de script pour des compilations standard d'installations logicielles sont créés par un administrateur informatique et utilisés pour garantir des installations complètes sur l'ensemble de l'organisation.

Les installations sans assistance peuvent aider au déploiement de plusieurs installations identiques de SQL Server à l'échelle d'une entreprise. Les installations sans assistance peuvent également permettre la délégation de l'installation à une autre personne.

### Méthodes d'installation sans assistance

L'une des options possibles pour effectuer l'installation sans assistance de SQL Server 2012 consiste à créer un fichier .ini contenant les informations d'installation requises et à exécuter setup.exe à l'invite de commande. Une autre solution consiste à passer les informations requises pour l'installation de SQL Server comme paramètres du programme setup.exe, plutôt qu'en plaçant les paramètres dans un fichier .ini. Vous pouvez choisir d'utiliser le fichier .ini ou d'utiliser les paramètres séparément, mais vous ne pouvez pas combiner les deux méthodes.

Dans les deux exemples sur la diapositive, la deuxième méthode a été utilisée. Le premier exemple montre une commande d'installation typique et le deuxième exemple montre comment il est possible d'effectuer une mise à niveau à l'aide de la même méthode.

### Switch /q

Le switch « /q » présenté dans les exemples indique le mode « silencieux » ; aucune interface utilisateur n'est fournie. Un autre switch « /qs » indique le mode « simple silencieux ». En mode simple silencieux, le programme d'installation s'exécute et affiche sa progression dans l'interface utilisateur mais n'accepte aucune entrée.

### **Création d'un fichier .ini**

Vous pouvez créer un fichier .ini pour l'installation sans assistance en utilisant n'importe quel éditeur de texte, comme le Bloc-notes. Le programme d'installation de SQL Server crée un fichier appelé ConfigurationFile.ini dans un dossier nommé en fonction de la date et l'heure d'installation, dans le dossier C:\Program Files\Microsoft SQL Server\110\setup bootstrap\log. Vous pouvez l'utiliser comme point de départ pour créer votre propre fichier .ini. Le fichier .ini est composé d'une section [Options] unique contenant plusieurs paramètres, dont chacun est lié à une fonctionnalité ou à un paramètre de configuration différents.

## Démonstration 3A : Création d'un fichier d'installation sans assistance

Dans cette démonstration, vous allez découvrir un fichier d'installation sans assistance qui peut être utilisé pour configurer SQL Server

### Procédure de démonstration

1. Si la démonstration 1A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre **Se conn au serveur**, tapez **Proseware**, puis cliquez sur **Se conn**. Dans le menu **Fichier** cliquez sur **Ouvrir**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_03\_PRJ\22462A\_03\_PRJ.ssmssln** puis cliquez sur **Ouvrir**.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** à partir de l'**Explorateur de solutions**.
2. Ouvrez le fichier **31 – ConfigurationFile.ini**.
3. Examinez le fichier de configuration, notez en particulier les valeurs des propriétés suivantes :
  - INSTANCEID
  - ACTION
  - FEATURES
  - QUIET
  - QUIETSIMPLE
  - INSTALLSHAREDDIR
  - INSTANCEDIR

- INSTANCENAME
- AGTSVCSTARTUPTYPE
- SQLCOLLATION
- SQLSVCACCOUNT
- SQLSYSADMINACCOUNTS
- TCPENABLED

## Atelier pratique 3 : Installation et configuration de SQL Server

- Exercice 1 : Vérification de la configuration requise pour l'installation
- Exercice 2 : Installation de l'instance SQL Server
- Exercice 3 : Exécution du programme et des contrôles post-installation
- Exercice difficile 4 : Configurer la mémoire du serveur (si le temps le permet)

Informations d'ouverture de session

Ordinateur virtuel	22462A-MIA-SQL1
Nom d'utilisateur	AdventureWorks\Administrateur
Mot de passe	Pa\$\$w0rd

Durée approximative : 45 minutes

### Configuration de l'atelier pratique

Pour cet atelier pratique, vous utiliserez l'environnement d'ordinateur virtuel disponible. Avant de débuter cet atelier pratique, vous devez procéder aux étapes suivantes :

1. Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
3. Dans la fenêtre Se connecter au serveur, tapez **Proseware** dans la zone de texte **Nom du serveur**.
4. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn.**
5. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**.
6. Dans la fenêtre Ouvrir un projet, sélectionnez le projet **D:\22462A\_Labs\22462A\_03\_PRJ\22462A\_03\_PRJ.ssmssln**.
7. Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Dans l'Explorateur de solutions, double-cliquez sur le fichier de requête **00 - Setup.sql**. Lorsque la fenêtre de requête s'affiche, cliquez sur **Exécuter** dans la barre d'outils.
8. Sur le système hôte, dans la liste **Ordinateurs virtuels** du Gestionnaire Hyper-V, cliquez avec le bouton droit sur l'ordinateur virtuel **22462A-MIA-SQL1**, puis cliquez sur **Paramètres**.
9. Dans la fenêtre Paramètres de 22462A-MIA-SQL1, dans la liste **Matériel**, développez **Contrôleur IDE 1**, puis cliquez sur **Lecteur DVD**.

10. Dans le volet propriétés du **Lecteur DVD**, cliquez sur **Fichier image**, puis sur **Parcourir**.
11. Naviguez jusqu'au fichier **C:\Program Files\Microsoft Learning\22462A\Drives\22462A-MIA-SQL1\Virtual Hard Disks\SQLFULL\_FRA.iso**, puis cliquez sur **Ouvrir**.
12. Dans la fenêtre des paramètres de 22462A-MIA-SQL1, cliquez sur **OK**.

### Scénario de l'atelier pratique

Le groupe de développement de la société a commandé un nouveau serveur pour le travail qu'ils doivent effectuer sur le système de Proseware. Malheureusement, le nouveau serveur n'arrivera pas avant des semaines et le groupe de développement ne peut pas attendre aussi longtemps pour démarrer le travail.

Le nouveau serveur fourni par le service d'assistance informatique a déjà deux instances de SQL Server installées. L'équipe de support a déterminé que le nouveau serveur est capable de prendre en charge une autre instance de SQL Server de manière temporaire, jusqu'à ce que le serveur du groupe de développement soit livré.

Vous devez installer la nouvelle instance de SQL Server et, si vous en avez le temps, vous devez configurer la mémoire des trois instances pour équilibrer leurs demandes et créer un nouvel alias pour l'instance que vous installez.

### Documentation

#### Configuration requise de l'instance de SQL Server

Élément	Configuration
Nom de l'instance	MKTGDEV
Fonctionnalités	Moteur de base de données uniquement (sauf Texte intégral et RéPLICATION)
Dossier des fichiers de données	D:\MKTGDEV pour les bases de données utilisateur et tempdb
Dossier de journal	L:\MKTGDEV pour les bases de données utilisateur et tempdb
Comptes de service	AdventureWorks\PWService pour tous les services
Démarrage	SQL Server et l'Agent SQL Server doivent démarrer automatiquement
Classement du serveur	SQL_Latin1_General_CI_AS
Mode d'authentification	Mixte
Utilisateur administratif	AdventureWorks\Administrateur
Prise en charge de Filestream	Désactivée

Notez que Pa\$\$w0rd est utilisé pour tous les mots de passe du cours.

**Configuration de la mémoire requise (utilisée dans l'exercice 4 uniquement)**

- 1,0 Go réservé pour le système d'exploitation.
- 40 % de la mémoire restante comme valeur maximale pour l'instance de serveur AdventureWorks.
- 30 % de la mémoire restante comme valeur maximale pour l'instance de serveur Proseware.
- 30 % de la mémoire restante comme valeur maximale pour l'instance de serveur PWDev.
- Configurez la mémoire minimale comme zéro pour toutes les instances.

## Exercice 1 : Vérification de la configuration requise pour l'installation

### Scénario

Étudiez la documentation de support qui décrit la configuration requise pour la nouvelle instance. Vous allez également créer les dossiers requis pour stocker les données et les fichiers journaux de l'instance.

Les tâches principales de cet exercice sont les suivantes :

1. Examinez la documentation avant de procéder à l'installation.
2. Créez les dossiers requis pour les fichiers de données et les fichiers journaux.

► **Tâche 1 : Examinez la documentation avant de procéder à l'installation**

- Prenez connaissance des éléments requis fournis dans la documentation de l'exercice.

► **Tâche 2 : Créez les dossiers requis pour les fichiers de données et les fichiers journaux**

- En fonction des spécifications fournies, créez les dossiers requis pour les fichiers de données et les fichiers journaux de la nouvelle instance SQL Server.

**Résultats** : À la fin de cet exercice, vous devez avoir lu les spécifications et avoir créé les deux dossiers demandés.

## Exercice 2 : Installation de l'instance SQL Server

### Scénario

Dans cet exercice, vous devez exécuter l'installation d'une autre instance de SQL Server sur le serveur existant, d'après les spécifications fournies. Au cours des étapes d'installation pour cet atelier pratique, vous avez monté un fichier ISO qui contient une image de SQL Server qui sera utilisée pour installer l'instance. Les élément prérequis ont déjà été installés car il existe déjà des instances de SQL Server.

La tâche principale de cet exercice est la suivante :

1. D'après les spécifications vues dans l'exercice 1, installez une autre instance de SQL Server.

► **Tâche 1 : D'après les spécifications vues dans l'exercice 1, installez une autre instance de SQL Server**

- Installez une autre instance de SQL Server en fonction des spécifications de l'exercice 1.



**Remarque** Dans la page configuration du serveur, vous devez configurer le nom et le mot de passe du compte de service, le type de démarrage de l'Agent SQL Server, et le classement. Dans la page Configuration du moteur de base de données, vous devez configurer le mode mixte, le mot de passe SA, le bouton Ajouter l'utilisateur actuel, l'onglet répertoires de données, et l'onglet FileStream.

**Résultats** : À la fin de cet exercice, vous devez avoir installé une nouvelle instance SQL Server.

## Exercice 3 : Exécution du programme et des contrôles post-installation

### Scénario

Vous devez vous assurer que les services de la nouvelle instance fonctionnent correctement et devez créer un nouvel alias pour l'instance que vous venez d'installer. Une fois que l'instance est créée, vous devez vous y connecter à l'aide de SQL Server Management Studio pour vérifier qu'elle fonctionne.

Les tâches principales de cet exercice sont les suivantes :

1. Vérifier que les services pour l'instance de SQL Server s'exécutent.
2. Configurer les alias 32 bits et 64 bits de la nouvelle instance.
3. Vous connecter à la nouvelle instance au moyen de SSMS.

#### ► Tâche 1 : Vérifier que les services pour l'instance de SQL Server s'exécutent

- À l'aide du Gestionnaire de configuration SQL Server, vérifier la bonne exécution des services récemment installés.
- Vérifier que le protocole des canaux nommés est activé pour la nouvelle instance.

#### ► Tâche 2 : Configurer les alias 32 bits et 64 bits de la nouvelle instance

- Configurer un alias 32 bits nommé PWDev pour la nouvelle instance en utilisant les canaux nommés.
- Configurer un alias 64 bits nommé PWDev pour la nouvelle instance en utilisant les canaux nommés.

#### ► Tâche 3 : Vous connecter à la nouvelle instance au moyen de SSMS

- Démarrer SQL Server Management Studio et vous connecter à la nouvelle instance pour vous assurer qu'elle fonctionne. Établir la connexion au moyen de l'alias PWDev.

**Résultats** : À la fin de cet exercice, vous devez avoir vérifié que les services s'exécutent, avoir créé un nouvel alias, et vous être connecté à l'aide de SSMS.

## Exercice difficile 4 : Configurer la mémoire du serveur (si le temps le permet)

### Scénario

Il y a maintenant trois instances de SQL Server installées sur le serveur de marketing chez AdventureWorks. Dans cet exercice, vous devez configurer la quantité de mémoire allouée à chaque instance.

Les tâches principales de cet exercice sont les suivantes :

1. Consultez la mémoire actuelle disponible sur le serveur.
2. Déterminez une allocation de mémoire appropriée pour chaque instance.
3. Configurez chaque instance de manière adéquate.

► **Tâche 1 : Consulter la mémoire actuelle disponible sur le serveur**

- Consulter la mémoire actuelle disponible sur le serveur.

► **Tâche 2 : Déterminer une allocation de mémoire appropriée pour chaque instance**

- Examiner les proportions de mémoire requise comme indiqué dans la documentation.
- Déterminer une allocation de mémoire appropriée pour chaque instance.

► **Tâche 3 : Configurer chaque instance de manière adéquate**

- Configurer chaque instance en fonction des valeurs que vous avez calculées.

**Résultats** : À la fin de cet exercice, vous devez avoir modifié l'allocation de mémoire de chaque instance de SQL Server.

## Contrôle des acquis et éléments à retenir

- Questions de contrôle des acquis
- Meilleures pratiques

### Questions de contrôle des acquis

1. Pourquoi le choix du classement d'un serveur est-il aussi important si vous pouvez de toute façon choisir les classements des bases de données individuelles ?
2. Tous les services SQL Server doivent-ils utiliser un seul compte de service basé sur un domaine ?

### Meilleures pratiques liées à un domaine technologique particulier dans ce module

1. Utilisez des comptes basés sur un domaine pour les comptes de service.
2. Configurez les comptes de service avec le moins possible de privilèges qui leur permette toujours de fonctionner.
3. Utilisez le Gestionnaire de configuration SQL Server pour modifier les comptes de service, car il vérifie que les autorisations et les listes de contrôle d'accès nécessaires sont correctement configurées.

# Module 4

## Utilisation des bases de données

### Sommaire :

Leçon 1 : Vue d'ensemble des bases de données SQL Server	4-3
Leçon 2 : Travailler avec des fichiers et des groupes de fichiers	4-15
Leçon 3 : Déplacer les fichiers de base de données	4-30
Atelier pratique 4 : Utilisation des bases de données	4-40

## Vue d'ensemble du module

- Vue d'ensemble des bases de données SQL Server
- Travailler avec des fichiers et des groupes de fichiers
- Déplacer les fichiers de base de données

L'un des rôles les plus importants des administrateurs de base de données qui travaillent avec Microsoft® SQL Server® est la gestion des bases de données. Il est important de savoir comment les données sont stockées dans les bases de données, créer des bases de données, et comment déplacer des bases de données dans un serveur ou entre des serveurs.

Lorsque les bases de données sont volumineuses, il est nécessaire d'allouer leurs données sur des volumes différents, plutôt que d'inscrire les données sur un seul gros disque. Cette allocation des données est configurée au moyen de groupes de fichiers et est utilisée pour satisfaire les besoins continus de performances et de gestion des bases de données.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- décrire le rôle et la structure des bases de données SQL Server.
- travailler avec des fichiers et des groupes de fichiers ;
- déplacer des bases de données dans des serveurs et entre serveurs ;

## Leçon 1

# Vue d'ensemble des bases de données SQL Server

- Mode de stockage des données dans SQL Server
- Détermination de l'emplacement et du nombre des fichiers
- Vérification de la capacité suffisante du fichier
- Bases de données système fournies avec SQL Server
- Vue d'ensemble de la base de données tempdb
- Démonstration 1A : Utilisation de la base de données tempdb

Avant de créer des bases de données, vous devez apprendre comment les données y sont stockées, découvrir les différents types de fichiers que SQL Server peut utiliser, où les fichiers doivent être placés, et apprendre également comment planifier la croissance à venir du fichier. Cela garantit que vous configurerez vos bases de données de façon optimale.

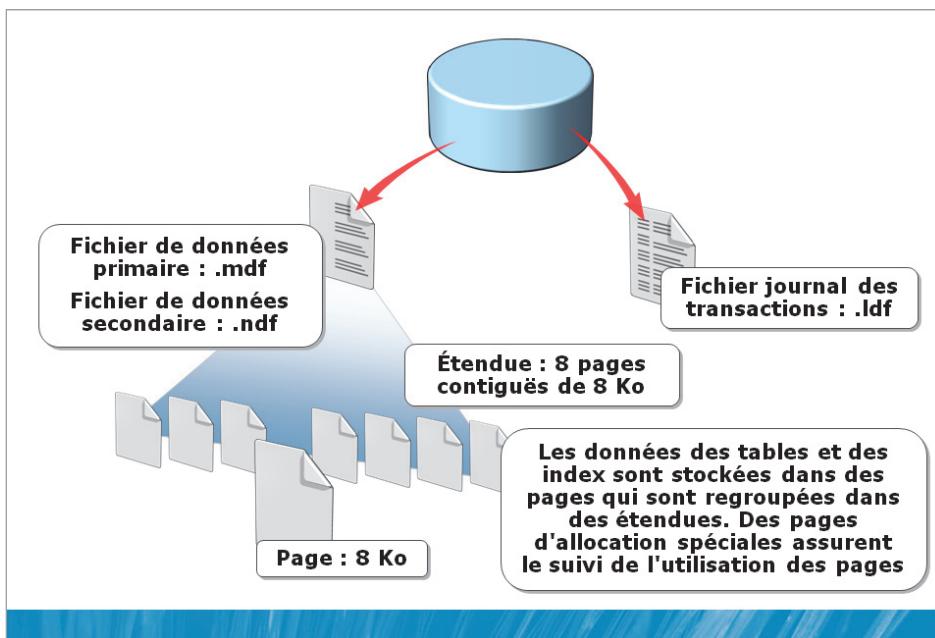
Dans cette leçon, vous allez également découvrir les bases de données système fournies avec SQL Server. Une base de données système en particulier, tempdb, a des exigences de configuration très spécifiques car ses performances sont susceptibles d'affecter les performances de toutes les applications qui utilisent le serveur.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- décrire le mode de stockage des données dans SQL Server ;
- expliquer les différents types de fichiers que SQL Server peut utiliser ;
- déterminer l'emplacement approprié et le nombre de fichiers appropriés pour les bases de données SQL Server ;
- vérifier la capacité suffisante du fichier et laisser de la place pour sa croissance ;
- expliquer le rôle de chaque base de données système fournie avec SQL Server ;
- configurer tempdb.

## Mode de stockage des données dans SQL Server



### Points clés

Les données stockées dans les bases de données SQL Server sont contenues dans un ensemble de fichiers alloués pour la base de données à utiliser. Il existe trois types de fichiers utilisés par SQL Server : les fichiers de données primaires, les fichiers de données secondaires, et les fichiers journaux de transactions.

#### Fichiers de données primaires

Le fichier de données primaire est le point de départ de la base de données. Chaque base de données comprend un fichier de données primaire unique. Tout en contenant des données à l'instar des autres fichiers de bases de données, fichier de données primaire contient des pointeurs vers les autres fichiers de la base de données. Les fichiers de données primaires utilisent généralement l'extension .mdf. Bien que l'utilisation de cette extension de fichier ne soit pas obligatoire, il est fortement recommandé de l'utiliser comme extension des fichiers de données primaires.

#### Fichiers de données secondaires

Les fichiers de données secondaires sont des fichiers de données supplémentaires facultatifs, définis par l'utilisateur, qui peuvent être utilisés pour répartir les données sur davantage de fichiers pour des raisons de performances et/ou de maintenance. Les fichiers secondaires peuvent être utilisés pour répartir des données sur plusieurs disques en plaçant chaque fichier sur un lecteur de disque distinct. En outre, si la taille d'une base de données excède la taille maximale autorisée pour un fichier Windows unique, vous pouvez avoir recours à des fichiers de données secondaires pour permettre à la base de données de continuer à croître. L'extension recommandée pour les fichiers de données secondaires est .ndf.

## Pages de fichier de données

Les pages d'un fichier de données SQL Server sont numérotées séquentiellement, en commençant par zéro pour la première page du fichier. Chaque fichier d'une base de données possède un numéro d'identification de fichier unique. Pour identifier de manière unique une page d'une base de données, l'ID de fichier et le numéro de page sont nécessaires. Chaque page a une taille de 8 Ko. Après la place laissée pour les informations d'en-tête nécessaires sur chaque page, il y a une zone de 8096 octets restants pour gérer les données. Les lignes de données peuvent avoir des valeurs de longueur de colonne fixe ou variable. Toutes les colonnes de longueur fixe d'une ligne de données doivent tenir sur une seule page, dans la limite de 8060 octets. Les pages de données ne contiennent des données que d'un seul objet de base de données, tel qu'une table ou un index.

## Étendues

Les groupes de 8 pages contiguës sont appelés une étendue. SQL Server utilise des étendues pour simplifier la gestion des pages de données. Il existe deux types d'étendues :

- les étendues uniformes : toutes les pages de l'extension contiennent des données d'un seul objet.
- étendues mixtes : les pages de l'extension peuvent contenir des données de différents objets.

La première allocation d'un objet se fait au niveau de la page, et provient toujours d'une étendue mixte. Si elles sont disponibles, d'autres pages de la même étendue mixte sont allouées à l'objet si nécessaire. Une fois que l'objet a grossi plus que sa première étendue, tous ses allocations suivants proviennent d'étendues uniformes.

Dans les fichiers de données primaires et secondaires, un petit nombre de pages est alloué au suivi de l'utilisation des extensions dans le fichier.

## Fichiers journaux

Les fichiers journaux contiennent toutes les informations utiles pour récupérer la base de données lorsque cela s'impose. Chaque base de données doit posséder au moins un fichier journal. L'extension recommandée pour les fichiers journaux est .ldf.

Toutes les transactions sont écrites dans le fichier journal à l'aide du mécanisme WAL (Write-Ahead Logging) pour assurer l'intégrité de la base de données en cas de défaillance et pour prendre en charge les restaurations de transactions.

Lorsque des pages de données doivent être modifiées, elles sont extraites vers la mémoire et modifiées en mémoire. Les pages de modification sont ensuite enregistrées dans le journal des transactions manière synchrone. À un moment donné ultérieur, durant un processus en arrière-plan appelé « point de contrôle », les pages de modifications sont écrites dans les fichiers de base de données. Pour cette raison, les pages qui sont contenues dans le journal des transactions sont essentielles à la possibilité par SQL Server de récupérer la base de données dans un état de validation connu. Les journaux des transactions sont présentés en détail dans des modules ultérieurs de ce cours.



**Remarque** Le fichier journal est également utilisée par d'autres fonctionnalités de SQL Server comme la réplication transactionnelle, la mise en miroir de bases de données et la capture de données modifiées. Ces rubriques avancées sont hors de portée de ce cours.

**Question** : dans quels scénarios les fichiers de données secondaires seraient-ils utiles ?

## Détermination de l'emplacement et du nombre des fichiers

- Isolez les fichiers journaux et les fichiers de données au niveau du disque physique
- Utilisez les niveaux RAID appropriés
- Déterminez le nombre et l'emplacement des fichiers de données en fonction des critères de performances et de maintenance
  - Utilisez des fichiers supplémentaires pour répartir les données sur d'autres piles ou emplacements de stockage
  - Utilisez plusieurs petits fichiers de données lorsqu'une maintenance simplifiée est nécessaire
  - Utilisez les fichiers de données en tant qu'unités de sauvegarde et de restauration
- Déterminez les spécifications des fichiers journaux
  - Utilisez un seul fichier journal dans la plupart des situations puisque les fichiers journaux sont écrits de manière séquentielle

### Points clés

Il est important d'isoler le journal et les fichiers de données, aussi bien pour des raisons de performances que de récupération. Cet isolement doit exister au niveau du disque physique.

### Modèles d'Accès

Les modèles d'accès du journal et des fichiers de données sont très différents. L'accès aux informations sur les fichiers journaux consiste principalement en écritures synchrones séquentielles sur les fichiers journaux et en des accès disque aléatoires occasionnels. L'accès aux informations sur les fichiers de données est principalement un accès de disque aléatoire asynchrone aux fichiers de données de la base de données. Un seul système physique de lecteur a tendance à ne pas donner de bons temps de réponse lorsque l'on combine ces types d'accès aux données.

### Récupération

Alors que les sous-systèmes de disque font des progrès en fiabilité, il existe toujours des risques d'erreur. Si un fichier de données SQL Server est perdu, la base de données peut être restaurée à partir d'une sauvegarde et le journal des transactions peut être réappliqué pour récupérer la base de données à un point récent. Si un fichier journal SQL Server est perdu, la base de données peut être forcée de récupérer à partir de fichiers de données, avec la possibilité de perte ou d'incohérence des données dans la base de données. Mais si ce sont les données et les fichiers journaux sur un sous-système de disque qui sont perdues, les options de récupération impliquent généralement de restaurer la base de données à partir d'une sauvegarde antérieure et de perdre toutes les transactions effectuées depuis lors. L'isolement des données et des fichiers journaux peut permettre d'éviter les conséquences les plus graves des défaillances de sous-système du lecteur.

## Séparation logique contre séparation physique

Il est recommandé de faire la séparation au niveau physique. Avec les systèmes SAN (réseau de stockage), il est facile de configurer le stockage afin qu'il semble être séparé, mais cette séparation est uniquement logique. C'est le même stockage physique sous-jacent qui est utilisé. Il s'agit en général d'un choix de conception médiocre.

## Gestion des fichiers de données

Idéalement, tous les fichiers de données définis pour une base de données doivent avoir la même taille. Les données sont réparties uniformément sur tous les fichiers de données disponibles. Les principaux avantages de performance de ce choix découlent du fait que les fichiers répartis sur différents emplacements de stockages.

Plusieurs avantages de gestion découlent de l'allocation de plusieurs fichiers de données :

- le principal avantage de gestion de l'allocation de plusieurs fichiers de données est la possibilité de déplacer ultérieurement les fichiers et une partie des données ;
- un autre avantage gagné grâce à l'utilisation de plusieurs fichiers de données consiste en le fait que, si la base de données est restaurée séparément, le temps de récupération peut être réduit. Cela peut être utile lorsque seule une partie des données est endommagée ;
- le fractionnement d'une base de données sur plusieurs fichiers de données peut augmenter le parallélisme du canal d'E/S ;
- le dernier avantage est que, si la taille d'une base de données excède la taille maximale autorisée pour un fichier Windows unique, vous pouvez avoir recours à des fichiers de données secondaires pour permettre à la base de données de continuer à croître.

## Nombre de fichiers journaux

Contrairement à la manière dont SQL Server écrit dans les fichiers de données, le moteur de base de données SQL Server écrit uniquement dans un seul fichier journal à tout moment. Les fichiers journaux supplémentaires ne sont utilisés que lorsqu'il n'y a plus d'espace disponible dans le fichier journal en cours de utilisation.

**Question :** pourquoi est-il important de séparer les fichiers de données et les fichiers journaux au niveau physique ?

## Vérification de la capacité suffisante du fichier

- Estimez la taille des données, des fichiers journaux et de la base de données tempdb
  - Exécutez un test de charge avec l'application réelle
  - Contactez le fournisseur de l'application
- Définissez la taille sur une valeur raisonnable
  - Laissez un espace suffisant pour les nouvelles données, sans qu'il soit nécessaire d'effectuer des expansions fréquentes
  - Contrôlez l'utilisation des fichiers de données et des fichiers journaux
  - Planifiez une expansion manuelle
  - Laissez les options de croissance automatique activées pour permettre des croissances imprévues



### Points clés

Il est important de prendre en compte la planification de capacité. Pour réaliser la planification de capacité, vous devez estimer la taille maximale de la base de données, des index, du journal des transactions, et de tempdb au cours d'une période de croissance prévue. Pour la plupart des sites vous devez viser à créer des bases de données de taille suffisante pour traiter les données qu'il est prévu de stocker dans les fichiers sur une période de douze mois.

### Planification réaliste

Les tests de performances et de capacité de SQL Server sont généralement effectués par le test de charge des applications qui utiliseront réellement SQL Server, plutôt que par la tentative d'estimer les besoins.

### Croissance automatique contre croissance planifiée

SQL Server peut développer automatiquement une base de données en fonction des paramètres de croissance définis lors de la création de ses fichiers. Bien qu'il soit préférable d'activer les options de croissance automatique pour empêcher les arrêts lors des croissances non prévues, il est important d'éviter à SQL Server la nécessité de faire grossir automatiquement les fichiers. Au lieu de cela, vous devez surveiller la croissance des fichiers au fil du temps et veiller à ce qu'ils soient suffisamment grands pour plusieurs mois ou années à venir.

De nombreux administrateurs s'inquiètent du fait que les gros fichiers de base de données font augmenter dans une certaine mesure le temps nécessaire à la réalisation des sauvegardes. La taille d'une sauvegarde SQL Server n'est pas liée directement à la taille des bases de données. Seules les parties utilisées de la base de données sont sauvegardées.

Un problème important qui accompagne la croissance automatique provient du compromis lié à la taille des incrémentations de croissance. Si un grand incrément est spécifié, il peut y avoir un délai significatif avant l'exécution de l'instruction T-SQL qui a déclenché le besoin de croissance. Si un trop petit incrément est spécifié, le système de fichiers peut devenir très fragmenté et les performances de la base de données peuvent en pârir parce que les fichiers de données ont été alloués par petits segments partout dans un sous-système de disque.



**Remarque** L'option d'initialisation instantanée des fichiers (IFI) peut réduire le temps nécessaire aux opérations de croissance automatique des fichiers de données. Elle ne s'applique pas aux fichiers journaux. L'option IFI sera traitée à la leçon suivante.

### Planification de la croissance du fichier journal

Si le journal des transactions n'est pas configuré pour se développer automatiquement, il risque de manquer d'espace lorsque certains types d'activités se produisent dans la base de données. Par exemple, l'exécution d'opérations en bloc à grande échelle, telles qu'une création d'index ou une importation en bloc, peut entraîner un remplissage rapide du journal des transactions.

Outre l'augmentation de la taille du journal des transactions, il est possible que le fichier journal soit tronqué. La troncation du journal purge le fichier des transactions validées inactives et permet au moteur de base de données SQL Server de réutiliser cette partie inutilisée du journal des transactions.



**Remarque** La manière et le moment où le journal est tronqué dépendent du mode de récupération de la base de données et seront présentés ultérieurement.

**Question :** quand est-il approprié de prédéfinir une taille maximale pour la base de données et de limiter la croissance du fichier ?

## Bases de données système fournies avec SQL Server

Base de données système	Description
master	Contient toute la configuration au niveau du système
msdb	Contient la configuration de l'Agent SQL Server, y compris l'historique des travaux, des sauvegardes et des restaurations
model	Est le modèle pour les nouvelles bases de données
tempdb	Contient les données temporaires telles que les tables temporaires, les variables de table, les tables de hachage et la banque de versions de ligne
resource	Base de données en lecture seule masquée qui contient les objets système mappés sur le schéma sys des bases de données

### Points clés

Il existe cinq bases de données système créées pendant l'installation. Il s'agit des bases master, msdb, model, tempdb et resource. Ces bases de données contiennent des métadonnées et ne peuvent pas être supprimées.

#### master

La base de données master contient toutes les informations au niveau du système. Tout ce qui est défini au niveau de l'instance de serveur est en général stockée dans la base de données master.

Si la base de données MASTER est endommagée ou altérée, SQL Server ne démarrera pas. Par conséquent, il est très important de sauvegarder cette base de données de manière régulière.

#### msdb

La base de données msdb contient les informations de l'Agent SQL Server. Les travaux, les opérateurs et les alertes sont stockés dans la base de données msdb. Il est également important d'effectuer une sauvegarde de la base de données msdb régulièrement pour empêcher la perte des travaux, des planifications, de l'historique des sauvegardes, des restaurations et des plans de maintenance. Dans les versions antérieures de SQL Server, les packages SQL Server Integration Services (SSIS) étaient également généralement stockés dans la base de données msdb. Dans SQL Server 2012, ceux-ci doivent être stockés dans la base de données dédiée SSIS.

#### model

La base de données model est le modèle sur lequel toutes les bases de données utilisateur sont basées. Toutes nouvelle base de données créée utilise la base de données model comme modèle. Tous les objets créés dans la base de données model seront présents dans toutes les nouvelles bases de données créées sur l'instance de serveur. Plusieurs sites ne modifient jamais la base de données model. Notez que même si la base de données model ne semble pas excessivement importante, SQL Server ne démarrera pas si elle n'est pas présente.

### **tempdb**

La base de données tempdb contient les données temporaires. Cette base de données est tronquée ou créé à chaque démarrage de SQL Server. Il est donc inutile d'effectuer une sauvegarde de cette base de données. En fait, il n'y a aucune possibilité d'effectuer une sauvegarde de la base de données tempdb. La base de données tempdb sera présentée plus en détail dans la rubrique suivante.

### **resource**

La base de données resource est une base de données masquée en lecture seule qui contient des objets système mappés sur le schéma sys de chaque base de données. Cette base de données contient également toutes les procédures système stockées, les vues système et les fonctions système. Dans les versions de SQL Server antérieures à SQL Server 2005, ces objets étaient définis dans la base de données master.

**Question :** suggérez un exemple d'objets que vous souhaitez créer dans la base de données model afin qu'ils soient déjà présents dans toutes les nouvelles bases de données créées sur l'instance du serveur.

## Vue d'ensemble de la base de données tempdb

- La base de données tempdb contient les données temporaires pour les objets internes, le contrôle de version de ligne et les objets utilisateur
  - Est tronquée ou régénérée à chaque redémarrage de l'instance
  - Occupe l'espace en fonction de la charge de travail et des modèles d'utilisation sur l'instance de SQL Server
  - Doit être testée avec des charges de travail réelles
- Placez la base de données tempdb sur un sous-système E/S rapide et distinct pour garantir des performances optimales
  - Fractionnez la base de données tempdb en fichiers de données de taille égale par cœur (jusqu'à 8 fichiers de données au maximum)
  - Utilisez des tests pour confirmer la meilleure valeur

### Points clés

La base de données tempdb inclut les objets internes, la banque de versions de lignes et les objets utilisateur. Les performances de la base de données tempdb est critique pour les performances globales de la plupart des installations de SQL Server.

### Objets internes

Les objets internes sont des objets utilisés par SQL Server pour ses opérations. Les objets internes comprennent les tables de travail pour les opérations de curseur ou de spool, le stockage des gros objets temporaires (LOB), les fichiers de travail pour le hachage (jointure hachée, agrégation hachée) et les résultats de tri intermédiaires.



**Remarque** L'utilisation des objets internes est un concept avancé qui n'entre pas dans le cadre de ce cours.

Les transactions associées aux niveaux d'isolement des transactions de capture instantanée peuvent provoquer la conservation brève d'autres versions des lignes dans un magasin de versions de ligne spécial de tempdb. Les versions de lignes peuvent également être produites par d'autres fonctionnalités telles que les reconstructions d'index en ligne, la fonctionnalité MARS (Multiple Active Result Sets) et les déclencheurs.



**Remarque** Les niveaux d'isolation des transactions sont présentés au cours 10776A : Developing Microsoft SQL Server 2012 Databases (*en anglais*).

## Objets utilisateur

La plupart des objets résidant dans la base de données tempdb sont des objets générés par l'utilisateur. Ils comprennent les tables temporaires, les variables de table, les jeux de résultats des fonctions table à instructions multiples et d'autres ensembles de lignes temporaires.

## Taille de tempdb

Comme tempdb est utilisée à de nombreuses fins, il est difficile d'en prédire la taille à l'avance. Les tailles appropriées de tempdb pour les nouvelles installations doivent être soigneusement testées et analysées dans des scénarios réalistes.

Le manque d'espace disque dans la base de données tempdb peut entraîner d'importantes perturbations dans l'environnement de production SQL Server et peut empêcher les applications en cours d'exécution d'effectuer certaines opérations. Vous pouvez utiliser la vue de gestion dynamique sys.dm\_db\_file\_space\_usage pour surveiller l'espace disque qui est utilisé par les fichiers de la base de données tempdb. En outre, pour surveiller l'activité d'allocation et de désallocation de pages dans la base de données tempdb au niveau des sessions ou des tâches, vous pouvez utiliser les vues de gestion dynamique sys.dm\_db\_session\_space\_usage et sys.dm\_db\_task\_space\_usage.

Par défaut, la base de données tempdb croît automatiquement en fonction des besoins, puisque le paramètre MAXSIZE des fichiers est défini sur UNLIMITED. Par conséquent, la base de données tempdb peut continuer à croître jusqu'à épuisement de l'espace sur le disque où elle se trouve.

## Fichiers multiples

Pour surmonter les restrictions d'E/S mais également éviter un concept appelé conflit des verrous internes qui affecte les pages d'allocation des pages de données tempdb, il peut être bon d'augmenter le nombre de fichiers pour améliorer les performances globales du système SQL Server. Toutefois, ne créez pas trop de fichiers. En règle générale il est recommandé d'avoir 0,25-1 fichier par cœur. En règle générale, on peut dire que ce rapport doit baisser avec l'augmentation du nombre de coeurs du système, mais qu'il convient de déterminer la configuration optimale au moyen de tests réels.

**Question :** pourquoi les goulots d'étranglement de mémoire peuvent-ils entraîner une plus grande utilisation de la base de données tempdb ?

## Démonstration 1A : Utilisation de la base de données tempdb

Dans cette démonstration, vous pouvez apprendre comment la taille des fichiers de la base de données tempdb est gérée

### Procédure de démonstration

1. Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn**. Dans le menu **Fichier** cliquez sur **Ouvrir**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_04\_PRJ\22462A\_04\_PRJ.ssmssqln** puis cliquez sur **Ouvrir**.
3. Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** à partir de l'Explorateur de solutions.
4. Ouvrez le fichier de script **11 – Demonstration 1A.sql**.
5. Suivez les instructions contenues dans les commentaires du fichier de script.

## Leçon 2

# Travailler avec des fichiers et des groupes de fichiers

- Création de bases de données utilisateur
- Configuration des options de base de données
- Initialisation instantanée des fichiers
- Démonstration 2A : Création de bases de données
- Modification de bases de données
- Expansion et réduction des fichiers de base de données
- Démonstration 2B : Modification de bases de données
- Utilisation des groupes de fichiers
- Démonstration 2C : Groupes de fichiers

La création de bases de données est un domaine d'expertise essentiel pour les administrateurs de base de données qui travaillent avec SQL Server. Autant que de comprendre comment créer des bases de données, vous devez être conscient de l'impact des options d'initialisation de fichier et savoir comment modifier les bases de données existantes. La modification des bases de données impliquera principalement l'expansion de l'espace disponible, mais en de rares occasions vous devrez peut-être réduire l'espace qu'elles occupent.

Pour les bases de données volumineuses, les groupes de fichiers permettent une meilleure gestion. Tenez compte de la manière dont les groupes de fichiers sont configurés.

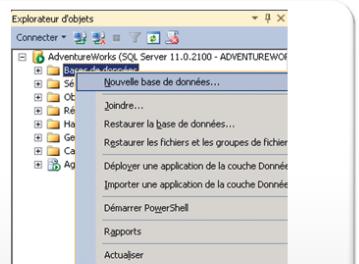
## Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- créer des bases de données utilisateur ;
- expliquer le rôle de l'initialisation instantanée des fichiers ;
- modifier des bases de données ;
- développer et réduire des bases de données ;
- utiliser des groupes de fichiers.

## Création de bases de données utilisateur

- Peuvent être créées dans l'interface utilisateur graphique
- Peuvent être créées avec la commande T-SQL CREATE DATABASE
- Le classement peut être spécifié



```
CREATE DATABASE Branch
ON
( NAME = Branch_dat,
  FILENAME = 'D:\Data\Branch.mdf' ,      SIZE = 100MB ,
  MAXSIZE = 500MB ,   FILEGROWTH = 20% )
LOG ON
( NAME = Branch_log,
  FILENAME = 'L:\Logs\Branch.ldf' ,    SIZE = 20MB ,
  MAXSIZE = UNLIMITED,   FILEGROWTH = 10MB );
```

### Points clés

Les bases de données peuvent être créées avec l'interface utilisateur graphique dans SSMS ou via la commande CREATE DATABASE dans T-SQL. La commande CREATE DATABASE offre davantage d'options mais l'interface graphique est plus facile à utiliser. Cette rubrique se concentre sur la commande CREATE DATABASE mais s'applique également aux options choisies dans l'interface utilisateur graphique de SSMS.

### CREATE DATABASE

Les noms de bases de données doivent être uniques dans une instance SQL Server et se conformer aux règles régissant les identificateurs. Un nom de base de données est du type « sysname » qui est défini comme nvarchar(128). Cela signifie que le nom de la base de données peut compter jusqu'à 128 caractères appartenant au jeu de caractères Unicode sur deux octets. Même si les noms de base de données peuvent être assez longs, vous constaterez que les noms longs sont assez difficiles à manipuler.

### Fichiers de données

Comme présenté précédemment dans ce module, une base de données doit avoir un seul fichier de données primaire et un fichier journal. Les clauses ON et LOG ON de la commande CREATE DATABASE spécifient les fichiers requis.

Dans l'exemple de la diapositive, la base de données nommée Branch est créée. Elle est composée de deux fichiers : un fichier de données primaire situé dans D:\Data\Branch.mdf et un fichier journal situé dans L:\Logs\Branch.ldf.



**Remarque** Chaque fichier comprend un nom de fichier logique et un chemin d'accès au fichier physique. Les noms de fichiers logiques doivent être uniques dans chaque base de données. Les opérations de SQL Server utilisent le nom logique du fichier.

Le fichier de données primaire a une taille initiale de 100 Mo et une taille maximale de 500 Mo. Sa taille augmente de 20 % à chaque croissance automatique.

Le fichier journal a une taille initiale de 20 Mo et n'a aucune limite de taille maximale. À chaque croissance automatique nécessaire, sa taille augmente d'une allocation fixe de 10 Mo.

### **Classements et valeurs par défaut**

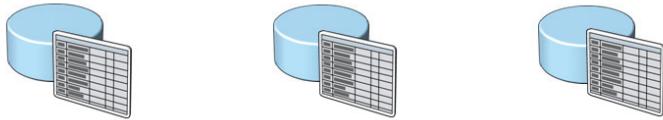
Un classement spécifique peut être affecté au niveau de la base de données si nécessaire. S'il n'est pas spécifié, le classement par défaut sera le classement spécifié pour l'instance de serveur pendant l'installation de SQL Server. Il est recommandé de conserver pour les bases de données le même classement que celui du serveur.

Bien qu'il soit possible de créer une base de données en spécifiant uniquement son nom, vous créeriez alors une base de données basée sur la base de données model et dont les fichiers de données et les fichiers journaux se trouvent dans les emplacements par défaut. Ce n'est généralement pas recommandé.

**Question :** qu'est-ce qu'un nom de fichier logique ?

## Configuration des options de base de données

**Les options de niveau base de données sont spécifiques à chaque base de données**



Option	Description
Options automatiques	Définit si certaines opérations doivent se produire automatiquement dans la base de données
Vérification de page	Définit comment la page doit être vérifiée lorsqu'elle est lue à partir du disque. Doit être définie sur CHECKSUM
Mode de récupération	Définit le mode de récupération de la base de données
Options d'état	Définit l'état de la base de données, par exemple En ligne/Hors connexion, Accès restreint ou Lecture seule

### Points clés

Chaque base de données dispose d'un ensemble d'options qui peuvent être configurées. Ces options sont spécifiques à chaque base de données et n'affectent pas les autres. Toutes les options de base de données sont issues de la base de données model lors de la création et peuvent être modifiées à l'aide de la clause SET de l'instruction ALTER DATABASE ou via la page de propriétés de chaque base de données dans SSMS.

### Catégories d'options

Il existe plusieurs catégories d'options de base de données :

- Options automatiques - Contrôlent certains comportements automatiques. De manière générale, les fonctions Fermeture automatique et Réduction automatique doivent être désactivées sur la plupart des systèmes mais les fonctions Création automatique et Mettre à jour les statistiques doivent être activées.
- Options de curseur - Contrôlent le comportement du curseur et l'étendue. En général, il n'est pas recommandé d'utiliser des curseurs lors de l'utilisation de SQL Server, à l'exception de certaines applications spécifiques comme des utilitaires. Ce cours ne traite pas des curseurs ; il convient toutefois de noter que l'emploi excessif de curseurs est une raison classique des problèmes de performances.
- Options de disponibilité de la base de données - Contrôlent si la base de données est en ligne ou hors connexion, qui peut s'y connecter et si elle est en mode lecture seule.

- Options de maintenance et de récupération telles que :
  - Modèle de récupération - Les modes de récupération de base de données sont présentées dans le module 5.
  - Vérification de page - Les versions antérieures de SQL Server proposaient une option appelée détection de page endommagée (Torn Page Detection). Cette option provoquait l'écriture par SQL Server d'un petit bitmap dans une page de base de données sur chaque secteur de lecteur de disque. Chaque secteur comporte 512 octets, ce qui signifie qu'il y a 16 secteurs par page base de données (8 Ko). C'était une méthode brute mais raisonnablement efficace pour détecter une situation où seuls certains secteurs qui étaient nécessaires pour l'écriture d'une page étaient réellement écrits. Une nouvelle option de vérification CHECKSUM a été ajoutée dans SQL Server 2005. L'utilisation de cette option entraîne le calcul et l'ajout par SQL Server d'une somme de contrôle à chaque page à mesure qu'elle est écrite, ainsi qu'une nouvelle vérification de cette somme à chaque extraction d'une page sur le disque.



**Remarque** Les sommes de contrôle de page ne sont ajoutées de lors de l'écriture suivante de la page. L'activation de cette option n'entraîne pas la réécriture de chaque page de la base de données avec une somme de contrôle.

## Initialisation instantanée des fichiers

- Permet de récupérer rapidement de l'espace disque utilisé
- Ne requiert pas le remplissage préalable de cet espace avec des zéros
- Est un problème de sécurité mineur
- Est activée en affectant le droit **Effectuer les tâches de maintenance de volume** au compte de service SQL Server
- S'applique uniquement aux fichiers de données
  - Ne s'applique pas aux fichiers journaux

### Points clés

L'une des fonctions de sécurité fournies par le système d'exploitation Windows consiste en le fait que l'utilisateur ne dispose pas de l'accès brut à l'espace disque précédemment utilisé par un autre utilisateur sans que cet espace disque ne soit préalablement remplacé.

### Initialisation instantanée des fichiers

Le temps nécessaire pour remplacer l'espace alloué aux fichiers de données à utiliser avec SQL Server peut être important. Lorsqu'un fichier 200 Go est alloué, il faut réécrire 200 Go de données.

Le risque principal encouru en ne remplaçant pas l'espace disque est qu'un administrateur serait susceptible de lire le contenu précédent du disque. Si ce risque est considéré comme négligeable, SQL Server peut tirer parti de l'initialisation instantanée des fichiers (IFI) pour éviter de passer du temps à effacer les données existantes.

SQL Server peut demander à utiliser l'espace non initialisé du système d'exploitation, mais seulement s'il dispose de droits suffisants pour cela. Si le compte de service SQL Server dispose du droit Tâches de maintenance du volume d'exécution, SQL Server peut utiliser l'initialisation instantanée des fichiers. Cette option a été intégrée dans SQL Server 2005 et n'affecte que les fichiers de données.

Un autre avantage de cette option est qu'elle réduit le temps nécessaire à la restauration des bases de données car l'espace disque nécessaire à cette opération n'a plus besoin d'être remplacé avant d'être réutilisé.

## Démonstration 2A : Création de bases de données

Dans cette démonstration, vous allez apprendre à créer une base de données à l'aide de l'interface utilisateur graphique dans SSMS et de l'instruction CREATE DATABASE dans T-SQL.

### Procédure de démonstration

1. Si la démonstration 1A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, cliquez sur **Démarrer**, cliquez sur **Tous les programmes**, cliquez sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn**. Dans le menu **Fichier** cliquez sur **Ouvrir**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_04\_PRJ\22462A\_04\_PRJ.ssmssln** puis cliquez sur **Ouvrir**.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** à partir de l'Explorateur de solutions.
2. Ouvrez le fichier de script **21 – Demonstration 2A.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

## Modification de bases de données

- Les fichiers de base de données peuvent être ajoutés, supprimés, développés et réduits
- Toutes les opérations de fichier sont des opérations en ligne utilisant
  - SSMS
  - ALTER DATABASE
  - DBCC (options telles que CHECKDB et SHRINKFILE)
- La suppression d'un fichier de base de données requiert son vidage préalable
  - Nécessité de migrer les données vers d'autres fichiers

### Points clés

Lors de l'exploitation de bases de données il est parfois nécessaire de les modifier. Le besoin le plus fréquent est l'ajout d'espace physique, soit en développant des fichiers existants, soit en ajoutant de nouveaux fichiers.

### ALTER DATABASE

Il est possible de développer et d'ajouter des fichiers aussi bien via l'instruction ALTER DATABASE que via l'interface utilisateur graphique de SSMS.

Vous aurez peut-être également besoin d'utiliser l'option de suppression d'un fichier. SQL Server ne permet pas de supprimer un fichier qui est en cours d'utilisation dans la base de données. La suppression d'un fichier doit être effectuée en deux étapes. Tout d'abord, le fichier doit être vidé à l'aide de l'option EMPTYFILE de DBCC SHRINKFILE. Il peut ensuite être supprimé à l'aide de la commande ALTER DATABASE.

Dans la démonstration 2B, vous verrez des exemples de chacune de ces opérations.

## Expansion et réduction des fichiers de base de données

- Les fichiers peuvent être développés manuellement
- Les fichiers peuvent augmenter automatiquement à l'aide des options de croissance automatique
  - Exécutée uniquement lorsqu'un nouvel espace est nécessaire
  - Peut provoquer des problèmes de blocage et d'expiration de délai
  - Doit être évitée : planifiez une expansion de fichier manuelle
- La réduction de fichiers est possible
  - Utilise DBCC SHRINKDATABASE, DBCC SHRINKFILE
  - Doit être effectuée uniquement en cas d'absolue nécessité
  - Est une opération très intrusive
  - Peut produire des bases de données très fragmentées

### Points clés

Par défaut, SQL Server développe automatiquement une base de données en fonction des paramètres de croissance définis lors de la création des fichiers. Vous pouvez également développer une base de données manuellement en allouant de l'espace supplémentaire à un fichier de base de données existant ou en créant un nouveau fichier. Vous devrez peut-être accroître l'espace des données ou du journal des transactions si les fichiers existants atteignent leurs tailles maximales.

Si une base de données a déjà épuisé tout l'espace qui lui était alloué et qu'elle n'est pas configurée pour la croissance automatique d'un fichier de données, l'erreur 1105 est générée. (Le numéro d'erreur équivalent pour l'impossibilité de croissance d'un fichier journal de transactions est 9002). Cela peut notamment se produire si la fonction de croissance automatique de la base de données n'est pas activée ou s'il n'y a pas suffisamment d'espace sur le disque dur.

### Ajout de fichiers

Une possibilité pour développer la taille d'une base de données est d'ajouter des fichiers à la base de données. Vous pouvez ajouter des fichiers, soit à l'aide de l'interface utilisateur graphique de SSMS, soit avec la commande ALTER DATABASE... ADD FILE de T-SQL. Vous verrez un exemple d'ajout d'un fichier dans la démonstration 2B.

## Fichiers croissants

Lorsque vous développez une base de données, vous devez augmenter sa taille d'au moins 1 Mo.

Idéalement, toute augmentation de taille du fichier peut être bien supérieure à ce chiffre.

Des augmentations de taille de 100 Mo ou plus sont courantes.

Lors du développement d'une base de données, le nouvel espace est immédiatement mis à la disposition du fichier de données ou du fichier journal des transactions, selon le fichier qui a été développé. Lorsque vous développez une base de données, vous devez spécifier la taille maximale que le fichier est autorisé à atteindre. Cela empêche le fichier de croître jusqu'à épuisement de l'espace disque. Pour spécifier une taille de fichier maximale, utilisez le paramètre MAXSIZE de l'instruction ALTER DATABASE ou l'option Limiter la croissance du fichier (Mo) lorsque vous utilisez la boîte de dialogue Propriétés dans SQL Server Management Studio pour développer la base de données.

## Journal des transactions

Si le journal des transactions n'est pas configuré pour se développer automatiquement, il risque de manquer d'espace lorsque certains types d'activités se produisent dans la base de données. Outre l'augmentation de la taille du journal des transactions, le fichier journal peut être tronqué. La troncation du journal purge le fichier des transactions validées inactives et permet au moteur de base de données SQL Server de réutiliser cette partie inutilisée du journal des transactions. S'il existe des transactions actives, le journal peut ne pas pouvoir être tronqué et son développement peut être la seule option disponible.

## Réduction d'une base de données

Chaque fichier d'une base de données peut être réduit par la suppression des pages inutilisées. Bien que le moteur de base de données réutilise l'espace efficacement, il arrive qu'un fichier n'ait plus besoin d'être aussi volumineux qu'il ne l'était. Réduire le fichier peut devenir nécessaire par la suite, mais la réduction de taille doit être considérée comme une option rarement utilisée. Cette réduction concerne aussi bien les fichiers de données que les journaux des transactions. Les fichiers de base de données peuvent être réduits manuellement, par groupe ou individuellement, ou la base de données peut être configurée de façon à être réduite automatiquement à intervalles définis.

## Méthodes de réduction

Vous pouvez réduire une base de données ou des fichiers de base de données spécifiques à l'aide des commandes DBCC SHRINKDATABASE et DBCC SHRINKFILE. La commande DBCC SHRINKFILE est préférable car elle fournit beaucoup plus de contrôle sur l'opération que DBCC SHRINKDATABASE.



**Remarque** La réduction d'un fichier implique généralement le déplacement de pages, ce qui peut prendre du temps.

Le fait de réduire régulièrement des fichiers est susceptible de provoquer leur croissance à nouveau. Par conséquent, même si SQL Server offre une option de réduction automatique des bases de données, celle-ci ne doit être utilisée que dans de rares situations. En effet, dans le cas de la plupart des bases de données, l'activation de cette option provoque des problèmes importants de fragmentation sur le sous-système de disque. Il est recommandé de n'exécuter une réduction de tout type qu'en cas d'absolue nécessité.

## TRUNCATE ONLY

Une autre option proposée est l'option TRUNCATE\_ONLY de DBCC SHRINKFILE qui libère l'espace libre à la fin du fichier dans le système d'exploitation, mais n'effectue pas de déplacement de page dans le fichier. Le fichier de données est réduit seulement jusqu'à la dernière étendue allouée. Cette option n'est généralement pas aussi efficace pour réduire le fichier, mais elle cause moins de fragmentation et est beaucoup plus rapide.

**Question :** quelles sont les deux stratégies de base pour développer une base de données manuellement ?

## Démonstration 2B : Modification de bases de données

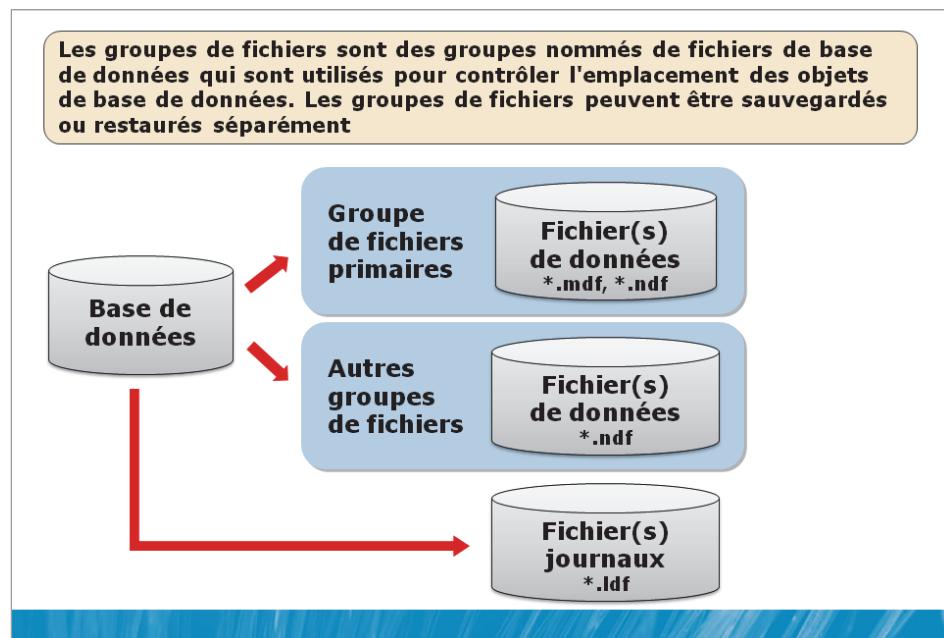
Dans cette démonstration, vous allez apprendre à

- Ajouter un fichier à une base de données
- Développer un fichier de base de données
- Réduire un fichier de base de données
- Supprimer un fichier de base de données

### Procédure de démonstration

1. Si la démonstration 2A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, cliquez sur **Démarrer**, cliquez sur **Tous les programmes**, cliquez sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn**. Dans le menu **Fichier** cliquez sur **Ouvrir**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_04\_PRJ\22462A\_04\_PRJ.ssmssln** puis cliquez sur **Ouvrir**.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** à partir de l'Explorateur de solutions.
  - Ouvrez le fichier de script **21 – Demonstration 2A.sql** et suivez les instructions qu'il contient.
2. Ouvrez le fichier de script **22 – Demonstration 2B.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

## Utilisation des groupes de fichiers



### Points clés

Les groupes de fichiers sont des collections de fichiers de données nommées, qui permettent de simplifier le placement des données et les tâches administratives, telles que les opérations de sauvegarde et de restauration. L'utilisation de fichiers et de groupes de fichiers peut améliorer les performances de la base de données car cela permet de créer une base de données sur plusieurs disques, plusieurs contrôleurs de disque ou des systèmes RAID.

### Emplacement des fichiers

Les fichiers et groupes de fichiers permettent le placement des données car une table ou un index peut être créé dans un groupe de fichiers spécifique. Lors de l'utilisation d'une instruction telle que CREATE TABLE, vous pouvez fournir un paramètre pour spécifier le groupe de fichiers dans lequel l'objet doit être créé. Cela peut améliorer les performances car toutes les opérations d'entrée/sortie de tables spécifiques peuvent être dirigées vers des disques spécifiques. Par exemple, une table fortement utilisée peut être placée dans un fichier unique d'un groupe de fichiers situé sur un disque, tandis que les autres tables moins consultées de la base de données peuvent être placées dans d'autres fichiers d'un autre groupe de fichiers situé sur d'autres disques.

**Remarque** L'emplacement des fichiers est un sujet avancé. Si cette question n'est pas examinée et testée avec soin, il est possible de réduire les performances du système en créant de nouveaux goulots d'étranglement en raison du forçage de la charge des E/S vers des fichiers et groupes de fichiers particuliers.

## Remplissage proportionnel

Dans un groupe de fichiers, le remplissage des fichiers s'effectue selon un mode proportionnel. Lorsque les données sont écrites dans le groupe de fichiers, le moteur de base de données SQL Server les écrit en les répartissant entre les fichiers du groupe proportionnellement à l'espace disponible dans chaque fichier, au lieu de remplir un premier fichier. Dès que tous les fichiers d'un groupe de fichiers sont remplis, le moteur de base de données développe automatiquement la taille d'un fichier à la fois, selon le mécanisme de tourniquet, afin d'y insérer plus de données, sous réserve que la fonction de croissance automatique de la base de données soit activée.

## Groupes de fichiers et autres fonctionnalités

Les groupes de fichiers peuvent également être utilisés en association avec les tables et les index partitionnés. Ces sujets n'entrent pas dans le cadre de ce cours.

Un groupe de fichiers est désigné comme groupe de fichiers par défaut. Lorsque les objets sont créés sans indication d'un groupe de fichiers, le groupe de fichiers par défaut est utilisé. À moins que vous ne modifiez le groupe de fichiers par défaut, le groupe de fichiers principal est le groupe par défaut.

Les groupes de fichiers peuvent également être créés en tant que groupes de fichiers en lecture seule. Les groupes de fichiers en lecture seule vous apportent les avantages suivants :

- ils peuvent être compressés (à l'aide de la compression NTFS) ;
- pendant la récupération, vous n'avez pas besoin d'appliquer les journaux pour récupérer un groupe de fichiers en lecture seule ;
- ils assurent la protection des données contre les modifications accidentelles ;
- les données d'archive peuvent être enregistrées sur des dispositifs de stockage moins onéreux.

**Question** : quel est l'avantage de stocker des données d'archive sur des groupes de fichiers distincts ?

## Démonstration 2C : Groupes de fichiers

Dans cette démonstration, vous allez apprendre à

- Créer une base de données avec plusieurs groupes de fichiers
- Créer des objets dans des groupes de fichiers

### Procédure de démonstration

1. Si la démonstration 2A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn.** Dans le menu **Fichier** cliquez sur **Ouvrir**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_04\_PRJ\22462A\_04\_PRJ.ssmssqln** puis cliquez sur **Ouvrir**.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** à partir de l'Explorateur de solutions.
  - Ouvrez et exécutez le fichier de script **21 - Demonstration 2A.sql**.
2. Ouvrez le fichier de script **23 – Demonstration 2C.sql**.

## Leçon 3

# Déplacer les fichiers de base de données

- Vue d'ensemble du détachement et de l'attachement
- Déplacement des fichiers de base de données utilisateur
- Démonstration 3A : Détachement et attachement
- Déplacement des fichiers de base de données système
- Copie de bases de données
- Démonstration 3B : Déplacement et reconfiguration de la base de données tempdb

Il est souvent nécessaire de déplacer des bases de données en déplaçant les fichiers. Les bases de données utilisateur et les bases de données système peuvent être déplacées, mais chacune requiert une approche légèrement différente. Dans cette leçon, vous apprendrez comment déplacer les deux types de base de données.

Il y a des précautions particulières à prendre lors du déplacement des bases de données système : si ces opérations ne sont pas exécutées correctement, cela peut causer l'arrêt du fonctionnement de SQL Server.

Vous pouvez également être amené à copier une base de données. SSMS fournit un assistant Copie de base de données à cet effet. Vous en apprendrez davantage sur les fonctionnalités de cet assistant.

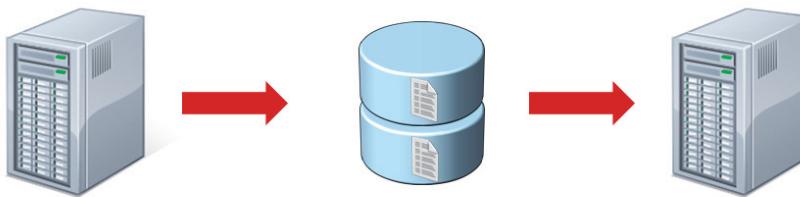
## Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- décrire la fonctionnalité détacher et attacher ;
- déplacer des fichiers de base de données utilisateur ;
- déplacer des fichiers de base de données système ;
- expliquer les fonctions de l'assistant Copie de base de données.

## Vue d'ensemble du détachement et de l'attachement

- Le détachement d'une base de données déconnecte la base de données de l'instance
  - Les fichiers de données et les fichiers journaux sont conservés intacts
  - Les fichiers détachés peuvent être rattachés sur la même instance ou sur une autre instance
- Utilisez le détachement et l'attachement pour déplacer des bases de données vers d'autres instances
  - Le détachement et l'attachement sont également utiles dans les situations de récupération d'urgence



### Points clés

Les bases de données peuvent être détachée à l'aide de SSMS ou de sp\_detach\_db. Détacher une base de données ne supprime pas les données des fichiers ni ne supprime pas les fichiers. Seules les entrées de métadonnées pour la base de données sont supprimées. La base de données détachée n'apparaît plus dans la liste des bases de données affichée dans SSMS ou par la vue système sys.databases. Les fichiers de données détachés peuvent être déplacés ou copiés vers une autre instance et y être rattachés.

### UPDATE STATISTICS

SQL Server maintient un jeu de statistiques sur la distribution des données dans les tables et les index. Dans le cadre du détachement, vous disposez d'une option pour effectuer une opération UPDATE STATISTICS sur les statistiques de table et d'index avant de réaliser l'action. Bien qu'elle puisse être utile si la base de données doit être rattachée comme base de données en lecture seule, en général il vaut mieux ne pas l'utiliser lors du détachement d'une base de données.

### Bases de données détachables

Toutes les bases de données ne peuvent pas être détachées. Les bases de données répliquées, en miroir ou dans un état suspect ne peuvent pas être détachées.

 **Remarque** Les bases de données répliquées et mises en miroir constituent des sujets avancés qui n'entrent pas dans le cadre de ce cours.

L'un des problèmes les plus courants qui empêchent de détacher une base de données est qu'il existe, au moment de la tentative, des connexions ouvertes vers la base de données. Toutes les connexions doivent être supprimées avant de détacher la base de données. SQL Server Management Studio offre la possibilité de forcer la suppression des connexions lors de cette opération.

## Attacher des bases de données

SSMS offre une option pour attacher des bases de données. Les bases de données peuvent également être attachées à l'aide de la commande CREATE DATABASE... FOR ATTACH.



**Remarque** Vous trouverez de nombreuses références pour l'utilisation des commandes sp\_attach\_db et sp\_attach\_single\_file\_db. Il s'agit d'une syntaxe plus ancienne remplacée par l'option FOR ATTACH de la commande CREATE DATABASE. Notez également qu'il n'y a aucun remplacement équivalent de la procédure sp\_detach\_db.

L'un des problèmes courants lorsque des bases de données sont rattachées est que les utilisateurs de la base de données peuvent devenir « orphelins ». Vous verrez comment traiter ce problème dans un module ultérieur.

**Question** : pourquoi la mise à jour de toutes les statistiques ne constitue-t-elle pas une bonne option avant le détachement d'une base de données ?

## Déplacement des fichiers de base de données utilisateur

- Les fichiers de données et les fichiers journaux peuvent être déplacés dans l'instance
  - Peut être effectué à l'aide du détachement et de l'attachement
  - Il est préférable d'utiliser l'instruction ALTER DATABASE
  - Utilisez le détachement et l'attachement pour effectuer un déplacement entre des instances
- Le nom logique des fichiers de base de données est nécessaire
  - Permet d'identifier les fichiers à déplacer
- La base de données doit être définie en mode hors connexion
- Les fichiers doivent être déplacés manuellement dans le système de fichiers

### Points clés

Le déplacement des fichiers d'une base de données requiert la mise hors connexion de la base de données. Le déplacement d'une base de données doit être effectué au cours d'une période de maintenance.

### Options du déplacement de bases de données

Dans SSMS, la seule possibilité pour déplacer des bases de données est de détacher la base de données, de déplacer les fichiers de données vers leur nouvel emplacement, puis de les rattacher à l'instance. Bien qu'il soit possible d'utiliser les opérations de détachement et d'attachement pour déplacer des fichiers de base de données utilisateur, il est également possible et même préférable d'utiliser l'instruction ALTER DATABASE, à moins que la base de données ne soit déplacée vers une autre instance de serveur.

Le processus de déplacement des fichiers de base de données utilisateur nécessite l'utilisation des noms logiques de ces fichiers. Vous pouvez voir les noms logiques des fichiers existants en exécutant le code suivant :

```
SELECT * FROM sys.database_files;
```

L'utilisation des opérations détacher et attacher pour le déplacement de fichiers de base de données entre instances sera présentée dans la démonstration suivante.

**Question :** pourquoi l'instruction ALTER DATABASE est préférée au détachement et à l'attachement de la base de données ?

## Démonstration 3A : Détachement et attachement

Dans cette démonstration, vous allez apprendre à utiliser le détachement et l'attachement avec les fichiers de base de données

### Procédure de démonstration

1. Si la démonstration 1A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn.** Dans le menu **Fichier** cliquez sur **Ouvrir**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_04\_PRJ\22462A\_04\_PRJ.ssmssqln** puis cliquez sur **Ouvrir**.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** à partir de l'Explorateur de solutions.
2. Ouvrez le fichier de script **31 – Demonstration 3A.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

## Déplacement des fichiers de base de données système

- Toutes les bases de données système à l'exception de la base de données **ressource** peuvent être déplacées vers de nouveaux emplacements
- Déplacement de la base de données master
  - Les paramètres de démarrage des services SQL Server doivent être modifiés
  - Les fichiers doivent être déplacés manuellement pendant l'arrêt de l'instance
- Déplacement d'autres bases de données système
  - Utilisez ALTER DATABASE comme avec les bases de données utilisateur
- Soyez prudent lorsque vous effectuez ces opérations
  - Une configuration incorrecte peut empêcher SQL Server de démarrer

### Points clés

Toutes les bases de données système peuvent être déplacées vers de nouveaux emplacements afin d'équilibrer la charge des E/S, à l'exception de la base de données resource. Le processus de déplacement de la base de données master est différent du processus appliqué aux autres bases de données.

### Déplacement de la base de données master

Les étapes du déplacement de la base de données master sont les suivantes :

1. Ouvrez SQL Server Configuration Manager.
2. Dans le noeud services de SQL Server, cliquez avec le bouton droit sur l'instance de SQL Server, sélectionnez Propriétés et cliquez sur l'onglet Paramètres de démarrage.
3. Modifiez les valeurs des paramètres de démarrage pour indiquer l'emplacement prévu des fichiers de la base de données master (paramètre -d) et du journal (paramètre -l).
4. Arrêtez l'instance SQL Server.
5. Déplacez les fichiers master.mdf et mastlog.ldf vers leur nouvel emplacement.
6. Redémarrez l'instance de SQL Server.

### Déplacement d'autres bases de données système

Les autres bases de données système (sauf la base de données resource qui ne peut pas être déplacée) sont déplacées en suivant les étapes suivantes :

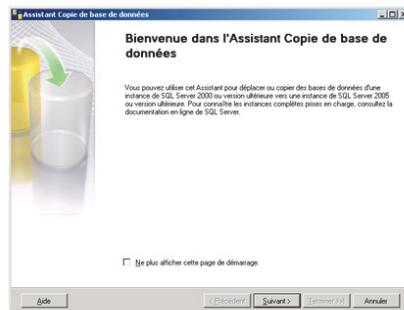
1. Pour chaque fichier à déplacer, exécutez ALTER DATABASE... MODIFY FILE comme pour les bases de données utilisateur.
2. Arrêtez l'instance SQL Server.
3. Déplacez fichiers vers le nouvel emplacement.
4. Redémarrez l'instance de SQL Server.

Dans la démonstration suivante vous verrez comment déplacer la base de données tempdb et comment augmenter le nombre de fichiers qu'elle utilise.

**Question** : quel est le plus gros problème lors de l'exécution de ces tâches ?

## Copie de bases de données

- Il existe différentes méthodes pour copier des bases de données
  - Sauvegarde et restauration
  - Détachement et attachement
  - Assistant Copie de base de données
- Assistant Copie de base de données
  - Peut être utilisé pour déplacer ou copier des bases de données entre des instances
  - Peut copier les objets supplémentaires dont la base de données dépend, tels que les connexions, les travaux, les plans de maintenance, les messages d'erreur définis par l'utilisateur et les objets partagés à partir de la base de données **master**
  - Peut être planifié par un travail de l'Agent SQL



### Points clés

Il existe plusieurs méthodes pour copier des bases de données vers d'autres emplacements, notamment des autres instances :

- vous pouvez détacher une base de données, copier les fichiers vers les nouveaux emplacements et rattacher les fichiers de base de données d'origine ainsi que les nouveaux ;
- vous pouvez sauvegarder une base de données puis la restaurer avec le même nom (sur une autre instance de serveur) ou avec un autre nom (dans la même instance de serveur) ;
- vous pouvez utiliser l'assistant Copie de base de données.

Le principal inconvénient lié au détachement/attachement et à la sauvegarde/restauration est que seule la base de données est copiée et que l'administrateur de bases de données doit gérer tous les objets dépendants, tels que les connexions, les travaux, les messages d'erreur définis par l'utilisateur, etc.

Restaurer une base de données sur une autre instance présente l'avantage que des sauvegardes doivent de toute façon être créées régulièrement, et qu'il est donc facile de les restaurer sans affecter le système source. La réalisation de la restauration est en elle-même un bon test de la stratégie de sauvegarde en cours. De plus la base de données source reste en ligne durant toute l'opération. Restaurer une base de données présente également le problème lié à la récupération des objets dépendants.

### Assistant Copie de base de données

L'assistant Copie de base de données est une bonne solution pour contourner cette limitation. Il fournit un assistant convivial qui permet de déplacer ou copier la base de données avec tous les objets qui en dépendent, sans nécessiter de script supplémentaire. En outre il est possible de planifier la copie.

L'assistant fournit deux méthodes pour copier ou déplacer la base de données. Il peut être configuré pour utiliser les opérations détachement/attachement (l'option la plus rapide), mais cela présente l'inconvénient de devoir déconnecter la base de données source pendant l'exécution des opérations de détachement/copie/attachement. La deuxième méthode consiste à utiliser les méthodes de la bibliothèque de programmation SQL Server Management Objects (SMO) pour créer les objets et transférer les données ; cette méthode est plus lente mais présente l'avantage que la base de données source est maintenue en ligne lors de la copie.

L'exécution de l'assistant Copie de base de données requiert des privilèges sysadmin sur les deux instances ainsi qu'une connexion réseau.

**Question :** quel peut être le principal inconvénient lié à la copie d'une base de données en détachant et attachant la base de données ?

## Démonstration 3B : Déplacement et reconfiguration de la base de données tempdb

Dans cette démonstration, vous allez apprendre à déplacer et reconfigurer la base de données tempdb pour disposer de plusieurs fichiers

### Procédure de démonstration

1. Si la démonstration 1A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, ouvrez le projet de scripts du serveur **22462A\_04\_PRJ\_SQL** dans SQL Server Management Studio.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** à partir de l'Explorateur de solutions.
2. Ouvrez le fichier de script **32 – Demonstration 3B.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

## Atelier pratique 4 : Utilisation des bases de données

- Exercice 1 : Modifier la configuration de la base de données tempdb
- Exercice 2 : Créer la base de données RateTracking
- Exercice 3 : Attacher la base de données OldProspects
- Exercice difficile 4 : Ajouter plusieurs fichiers à la base de données tempdb (si le temps le permet)

Informations d'ouverture de session

Ordinateur virtuel	22462A-MIA-SQL1
Nom d'utilisateur	AdventureWorks\Administrateur
Mot de passe	Pa\$\$w0rd

**Durée approximative : 45 minutes**

### Configuration de l'atelier pratique

Pour cet atelier pratique, vous utiliserez l'environnement d'ordinateur virtuel disponible. Avant de débuter cet atelier pratique, vous devez procéder aux étapes suivantes :

1. Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
3. Dans la fenêtre Se connecter au serveur, tapez **Proseware** dans la zone de texte **Nom du serveur**.
4. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Connecter**.
5. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**.
6. Dans la fenêtre Ouvrir un projet, sélectionnez le projet **D:\22462A\_Labs\22462A\_04\_PRJ\22462A\_04\_PRJ.ssmssln**.
7. Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Dans l'Explorateur de solutions, double-cliquez sur le fichier de requête **00 - Setup.sql**. Lorsque la fenêtre de requête s'affiche, cliquez sur **Exécuter** dans la barre d'outils.

## Scénario de l'atelier pratique

Maintenant que l'instance de Proseware de SQL Server a été installée et configurée sur le serveur, plusieurs configurations de base de données supplémentaires doivent être exécutées. En tant qu'administrateur de base de données, vous devez exécuter ces modifications de configuration.

Vous devez créer une nouvelle base de données sur le serveur, selon les spécifications d'un fournisseur d'application. Un client vous a envoyé une base de données qui doit être installée sur l'instance Proseware. Au lieu de vous envoyer une sauvegarde, il vous a envoyé une base de données détachée et un fichier journal. Vous devez relier la base de données à l'instance Proseware.

Un consultant a également fourni des recommandations concernant la configuration de tempdb que vous devez examiner et implémenter le cas échéant.

### Documentation

#### Spécification de la taille de tempdb (pour l'exercice 1)

Fichier	Taille (en Mo)
Données	30
Journal	10

#### Spécifications de la base de données RateTracking

Élément	Configuration requise
DatabaseName	RateTracking
Fichier de données primaire	Nom logique = RateTracking_dat Nom de fichier = RateTracking.mdf Dossier = D:\MKTG Taille initiale = 10 Mo Taille de fichier maximale = 100 Mo Croissance automatique = 10 Mo Groupe de fichiers = PRIMARY
Fichier journal	Nom logique = RateTracking_log Nom de fichier = RateTracking.ldf Dossier = L:\MKTG Taille initiale = 20 Mo Taille de fichier maximale = unlimited Croissance automatique = 20 Mo Groupe de fichiers = Non applicable

(suite)

Élément	Configuration requise
Fichier de données secondaire n°1	Nom logique = RateTracking_dat_1 Nom de fichier = RateTracking_1.ndf Dossier = D:\MKTG Taille initiale = 20 Mo Taille de fichier maximale = 100 Mo Croissance automatique = 10 Mo Groupe de fichiers = USERDATA
Fichier de données secondaire n°2	Nom logique = RateTracking_dat_2 Nom de fichier = RateTracking_2.ndf Dossier = D:\MKTG Taille initiale = 20 Mo Taille de fichier maximale = 100 Mo Croissance automatique = 10 Mo Groupe de fichiers = USERDATA
Fichier de données secondaire n°3	Nom logique = RateTracking_dat_3 Nom de fichier = RateTracking_3.ndf Dossier = D:\MKTG Taille initiale = 200 Mo Taille de fichier maximale = 500 Mo Croissance automatique = 50 Mo Groupe de fichiers = ARCHIVE
Fichier de données secondaire n°4	Nom logique = RateTracking_dat_4 Nom de fichier = RateTracking_4.ndf Dossier = D:\MKTG Taille initiale = 200 Mo Taille de fichier maximale = 500 Mo Croissance automatique = 50 Mo Groupe de fichiers = ARCHIVE
Groupe de fichiers par défaut	USERDATA

**Spécifications de tempdb imposées par le consultant (pour l'exercice 4)**

1. Déplacer le fichier de données primaire tempdb dans le dossier D:\MKTG.
2. Déplacer le fichier journal tempdb dans le dossier L:\MKTG.

3. Ajouter trois fichiers supplémentaires à tempdb conformément au tableau suivant :

Nom de fichier	Configuration requise
Fichier de données secondaire n°1	Nom logique = tempdev2 Nom de fichier = tempdb_file2.ndf Dossier = D:\MKTG Taille initiale = 20 Mo Taille de fichier maximale = illimitée Croissance automatique = 10 Mo
Fichier de données secondaire n°2	Nom logique = tempdev3 Nom de fichier = tempdb_file3.ndf Dossier = D:\MKTG Taille initiale = 20 Mo Taille de fichier maximale = illimitée Croissance automatique = 10 Mo
Fichier de données secondaire n°3	Nom logique = tempdev4 Nom de fichier = tempdb_file4.ndf Dossier = D:\MKTG Taille initiale = 20 Mo Taille de fichier maximale = illimitée Croissance automatique = 10 Mo

## Exercice 1 : Modifier la configuration de la base de données tempdb

### Scénario

Vous allez modifier la configuration actuelle de la base de données tempdb.

Les tâches principales de cet exercice sont les suivantes :

1. Ajuster la taille de tempdb.
2. Vérifier que la taille de tempdb est toujours correcte après un redémarrage.

#### ► Tâche 1 : Modifiez la taille de tempdb

- Prendre connaissance des éléments de configuration requis pour la taille de tempdb fournis dans la documentation.
- Ajuster la taille de tempdb d'après la configuration requise indiquée dans la documentation.

#### ► Tâche 2 : Vérifier que la taille de tempdb est toujours correcte après un redémarrage

- Redémarrer le serveur Proseware à l'aide du Gestionnaire de configuration SQL Server.
- Vérifier que la taille de tempdb est toujours correcte.

**Résultats :** À la fin de cet exercice, vous devez avoir examiné et configuré la base de données tempdb.

## Exercice 2 : Créer la base de données RateTracking

### Scénario

Vous allez créer une base de données nommée RateTracking conformément à un ensemble de spécifications fourni.

Les tâches principales de cet exercice sont les suivantes :

1. Créer la base de données.
2. Créer les fichiers et groupes de fichiers requis.
3. Changer le groupe de fichiers par défaut de la base de données.

#### ► Tâche 1 : Créer la base de données

- Prenez connaissance des éléments requis fournis dans la documentation associée de l'exercice.
- Créer une base de données RateTracking d'après les spécifications.

#### ► Tâche 2 : Créer les fichiers et groupes de fichiers requis

- Prendre connaissance des spécifications indiquées dans la documentation relativement aux fichiers et groupes de fichiers requis.
- Créer les fichiers et groupes de fichiers requis.

#### ► Tâche 3 : Changer le groupe de fichiers par défaut de la base de données

- Prenez connaissance des éléments requis fournis dans la documentation associée pour le groupe de fichiers par défaut.
- Modifier le groupe de fichiers par défaut.

**Résultats** : À la fin de cet exercice, vous devez avoir créé une base de données RateTracking avec plusieurs groupes de fichiers.

## Exercice 3 : Attacher la base de données OldProspects

### Scénario

Un client vous a envoyé une base de données qui doit être installée sur l'instance Proseware. Au lieu de vous envoyer une sauvegarde, il vous a envoyé une base de données détachée et un fichier journal. Vous devez relier la base de données à l'instance Proseware.

Les tâches principales de cet exercice sont les suivantes :

1. Copier les fichiers de la base de données.
2. Attacher la base de données à l'instance MKTG.

#### ► Tâche 1 : Copier les fichiers de la base de données

- Copier les fichiers vers de nouveaux dossiers conformément au tableau ci-dessous :

Nom de fichier	Dossier source	Dossier de destination
OldProspects.mdf	D:\22462A_Labs\22462A_04_PRJ	D:\MKTG
OldProspects.ldf	D:\22462A_Labs\22462A_04_PRJ	L:\MKTG

#### ► Tâche 2 : Attacher la base de données à l'instance de MKTG

- Attacher la base de données OldProspects à l'instance MKTG.

**Résultats** : À la fin de cet exercice, vous devez avoir attaché une base de données à un serveur de client.

## Exercice difficile 4 : Ajouter plusieurs fichiers à la base de données tempdb (si le temps le permet)

### Scénario

Un consultant a également fourni des recommandations concernant la configuration de tempdb que vous devez examiner et implémenter.

Les tâches principales de cet exercice sont les suivantes :

1. Consulter les spécifications du fichier tempdb.
2. Déplacer les fichiers existants.
3. Ajouter de nouveaux fichiers.
4. Redémarrer le serveur et vérifier les emplacements de fichiers.

#### ► Tâche 1 : Consulter les spécifications du fichier tempdb

- Dans la documentation associée, vérifiez les éléments requis pour tempdb dans la section Consultant.

#### ► Tâche 2 : Déplacer les fichiers existants

- Déplacer les données et les fichiers journaux de tempdb vers les emplacements demandés dans la documentation.

#### ► Tâche 3 : Ajouter de nouveaux fichiers

- Ajouter les fichiers tempdb supplémentaires comme demandé dans la documentation.

#### ► Tâche 4 : Redémarrer le serveur et vérifier les emplacements de fichiers

- Redémarrer le serveur Proseware.
- Afficher les propriétés de la base de données tempdb et vérifier la correspondance entre la liste des fichiers et les spécifications.

**Résultats :** À la fin de cet exercice, vous devez être reconfiguré tempdb conformément aux spécifications.

## Contrôle des acquis du module et éléments à retenir

- Questions de contrôle des acquis
- Meilleures pratiques

### Questions de contrôle des acquis

1. Pourquoi est-il généralement suffisant d'avoir un fichier journal dans une base de données ?
2. Pourquoi aucune donnée à l'exception des données temporaires ne doit être stockée dans la base de données système tempdb ?
3. Quelles opérations peuvent être effectuées en ligne sur les fichiers de base de données ?

### Meilleures pratiques liées à un domaine technologique particulier dans ce module

1. Planifiez et testez attentivement votre disposition de fichiers.
2. Séparez physiquement les fichiers de données et les fichiers journaux.
3. Conservez les fichiers de données d'une base de données à la même taille.
4. Créez la base de données dans une taille appropriée qui ne doit pas être développée trop souvent.
5. Réduisez les fichiers uniquement en cas d'absolue nécessité.

# Module 5

## Description des modes de récupération SQL Server 2012

### Sommaire :

Leçon 1 : Stratégies de sauvegarde	5-3
Leçon 2 : Description de la journalisation des transactions SQL Server	5-12
Leçon 3 : Planification d'une stratégie de sauvegarde SQL Server	5-24
Atelier pratique 5 : Description des modes de récupération SQL Server	5-34

## Vue d'ensemble du module

- Stratégies de sauvegarde
- Description de la journalisation des transactions SQL Server
- Planification d'une stratégie de sauvegarde SQL Server

L'un des aspects les plus importants du rôle d'administrateur de base de données est qu'il doit s'assurer que les données d'organisation sont sauvegardées de manière fiable, afin de pouvoir les récupérer si (ou quand) une erreur se produit.

Bien que le secteur de l'informatique ait conscience depuis des décennies du besoin en stratégies de sauvegarde fiables et ait étudié le sujet en profondeur, il est toujours question ici ou là de pertes de données catastrophiques. De plus, même si les stratégies mises en œuvre atteignent le but pour lesquels elles ont été conçues, leur résultat manque régulièrement à satisfaire les impératifs opérationnels des organisations.

Dans ce module, vous examinerez comment concevoir une stratégie alignée sur les besoins de l'entreprise et vous verrez comment les fonctionnalités de journalisation de Microsoft® SQL Server® peuvent vous aider à obtenir des résultats adaptés.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- décrire les concepts importants qui entourent les stratégies de sauvegarde ;
- expliquer les fonctions de journalisation des transactions dans le moteur de base de données SQL Server ;
- planifier une stratégie de sauvegarde SQL Server.

## Leçon 1

# Stratégies de sauvegarde

- Discussion : Expérience passée en matière de stratégies de sauvegarde
- Définition d'une stratégie de sauvegarde appropriée
- Choix du support de sauvegarde approprié
- Définition d'une stratégie de conservation et de test pour les sauvegardes

Cette leçon est la première d'une série qui vous permettra d'acquérir les connaissances nécessaires à la planification et à la mise en œuvre d'une stratégie de sauvegarde appropriée pour vos données d'entreprise stockées sur SQL Server.

Après avoir discuté de vos expériences respectives en matière de stratégies de sauvegarde, vous apprendrez quels sont critères principaux qui doivent en faire partie. Vous apprendrez également combien de temps doivent être conservées les sauvegardes, comment les tester et sur quels types de supports il convient de les conserver.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- déterminer une stratégie de sauvegarde appropriée ;
- choisir un support de sauvegarde approprié ;
- concevoir une stratégie de rétention des sauvegardes.

## Discussion : Expérience passée en matière de stratégies de sauvegarde

- Quels types de sauvegarde avez-vous utilisés ?
- À quelle fréquence effectuez-vous des sauvegardes ?
- Qui est chargé de planifier et d'exécuter une stratégie de sauvegarde ?
- À quelle fréquence vos sauvegardes sont-elles testées ?
- Utilisez-vous des outils tiers pour les sauvegardes ?
- Quel type de support de sauvegarde utilisez-vous ?



### Points clés

Dans cette discussion, les membres de la classe partageront leurs expériences en matière de stratégie de sauvegarde. En particulier, considérez les questions suivantes (mais ne vous y limitez pas) :

- Quels types de sauvegarde avez-vous utilisés ?
- À quelle fréquence effectuez-vous des sauvegardes ?
- Qui est chargé de planifier et d'exécuter une stratégie de sauvegarde ?
- À quelle fréquence vos sauvegardes sont-elles testées ?
- Utilisez-vous des outils tiers pour les sauvegardes ?
- Quel type de support de sauvegarde utilisez-vous ?

## Définition d'une stratégie de sauvegarde appropriée

- Différents types de sauvegarde peuvent être combinés
  - Sauvegardes basées sur des données (sauvegardes complètes, différentielles, de copie, de groupe de fichiers, de fichier)
  - Sauvegardes basées sur des journaux (sauvegardes de journal, de la fin du journal)
- Les niveaux de sécurité doivent être déterminés
  - Combien de temps peut durer la récupération ? (RTO)
  - Quelle quantité de données est-il acceptable de perdre ? (RPO)
  - Est-il possible de récupérer les données d'autres sources ?
- La stratégie de sauvegarde doit être adaptée aux besoins
  - Types et fréquence des sauvegardes
  - Support de sauvegarde à utiliser
  - Période de conservation des sauvegardes et des supports
  - Stratégie de test des sauvegardes

### Points clés

SQL Server fournit divers types de sauvegarde. Aucun type de sauvegarde n'est généralement suffisant en soi pour les besoins d'une organisation lors de l'élaboration d'une stratégie de sauvegarde. En général, une combinaison des types de sauvegarde est requise pour obtenir des résultats pertinents. Plus loin dans ce module et dans le module suivant vous en apprendrez davantage sur les types de sauvegardes assurés par SQL Server.

### Critères principaux

Lorsque vous concevez une stratégie de sauvegarde, il existe toujours un compromis entre le niveau de sécurité qui est garanti et le coût de la solution. Si vous demandez à n'importe quelle entreprise quelle quantité de données elle peut se permettre de perdre, on vous répondra certainement qu'elle ne peut s'autoriser aucune perte de données, sous quelques circonstances que ce soit. Toutefois, même si zéro perte est un objectif admirable, il n'est ni abordable ni réaliste. Pour cette raison, il convient d'établir deux objectifs lors de l'élaboration d'une stratégie de sauvegarde : un objectif de temps de récupération (RTO) et un objectif de point de récupération (RPO). Une partie de la stratégie peut également impliquer la récupération des données depuis d'autres emplacements où en sont stockées des copies.

### Objectif de temps de récupération

Il y a peu d'intérêt à disposer de données entièrement récupérables si le temps nécessaire pour ce faire est trop long. Une stratégie de sauvegarde doit avoir un objectif de temps de récupération (RTO).

Par exemple, considérez les spécifications de sauvegarde d'une grande banque en ligne. Si la banque ne pouvait pas accéder aux données de ses systèmes, combien de temps la situation serait-elle tenable ?

Imaginez à présent que la banque a fait des copies complètes de toutes ses données, mais que la restauration complète de ces données prendrait deux semaines. Quel impact sur la banque aurait une interruption de deux semaines de ses services ? La question la plus importante est celle-ci : quelle est la durée maximale d'une interruption de l'accès aux données avant que la banque ne cesse d'être viable en tant que telle ?

Ce qu'il faut comprendre avec la notion de RTO, c'est qu'un plan qui implique une récupération rapide des données associée avec une perte faible est beaucoup plus avantageux pour l'entreprise qu'un plan qui réduit la perte de données mais prend beaucoup plus longtemps à être exécuté. Un autre problème important réside dans le fait que la restauration des données peut parfois impliquer de retrouver le bon support de sauvegarde, de trouver une personne dotée des autorisations suffisantes pour pratiquer la restauration, de trouver la documentation sur la restauration, etc.

### **Objectif de point de récupération**

Une fois que le système a été récupéré, avec un peu de chance en temps voulu, la question importante qui suit est : quelle quantité de données a été perdue ? Elle est représentée par l'objectif de point de récupération (RPO).

Par exemple, alors que dans une PME on pourrait décider que restaurer la sauvegarde de la nuit précédente (impliquant la perte des données d'une journée de travail) est un compromis de risque acceptable, on pourrait avoir un point de vue très différent dans une grande entreprise. Il est courant dans les grandes sociétés de viser une perte nulle des données validées. Cela signifie que le travail qui a été validé dans la base de données doit être récupéré, mais qu'il peut être acceptable de perdre le travail qui était en cours au moment de la défaillance.

### **Mappage sur la stratégie de l'entreprise**

L'aspect le plus important de l'élaboration d'une stratégie de sauvegarde est que celle-ci doit être conçue en fonction des besoins et de la stratégie de l'entreprise. La stratégie de sauvegarde doit également être communiquée aux parties prenantes appropriées dans l'organisation. Il est important de s'assurer de la prise en compte des attentes des utilisateurs de l'entreprise, en accord avec la stratégie convenue.

Les organisations déploient souvent un grand nombre de bases de données. Le RPO et le RTO peuvent être différents pour chaque base de données. Cela signifie que les administrateurs de base de données doivent souvent utiliser des stratégies de sauvegarde différentes en fonction des bases de données qu'ils gèrent. La plupart des grandes entreprises disposent d'une méthode de classement des bases de données et des applications en termes d'importance vis-à-vis des fonctions-clés de l'organisation.

Les besoins de l'entreprise déterminent tous les aspects de la stratégie de sauvegarde, notamment la fréquence à laquelle les sauvegardes doivent se produire, la quantité de données à sauvegarder chaque fois, le type de support sur lesquels seront conservées les sauvegardes ainsi que les plans d'archivage de support.

**Question :** quels seraient les critères RPO et RTO probables pour les bases de données les plus importantes de votre organisation ?

## Choix du support de sauvegarde approprié

- Une sauvegarde unique est appelée un jeu de sauvegarde
- Les jeux de sauvegarde sont écrits sur des supports de sauvegarde composés d'une ou plusieurs unités de sauvegarde
- Les supports de sauvegarde peuvent contenir plusieurs jeux de sauvegarde
- Les unités de sauvegarde peuvent être physiques ou logiques
  - Les unités physiques sont des fichiers disque
- Certains utilisateurs peuvent effectuer des sauvegardes

### Points clés

Un jeu de sauvegarde contient le produit d'une opération de sauvegarde réussie unique effectuée par SQL Server. Un ou plusieurs jeux de sauvegarde sont écrits sur un support de sauvegarde qui est représenté par un fichier au niveau du système d'exploitation ou du périphérique.

Il est important de réaliser que, comme un fichier unique (ou jeu de sauvegardes) peut contenir plusieurs sauvegardes, il faut, au moment de pratiquer une restauration, veiller à restaurer la bonne sauvegarde extraite du fichier.

### Unités de sauvegarde physiques

SQL Server prend en charge la création de sauvegardes sur des fichiers disque. Les chemins d'accès de fichier UNC sont pris en charge pour les emplacements de disque. Ainsi, il est possible d'écrire les sauvegardes dans des fichiers partagés du réseau.

Les versions antérieures prenaient en charge l'écriture directe des sauvegardes sur des bandes, mais cette option est aujourd'hui déconseillée et ne devrait pas être utilisée pour le développement. Il est généralement recommandé d'écrire d'abord une sauvegarde sur disque, puis de copier si nécessaire le disque sur une bande.

Les sauvegardes que crée SQL Server sont encodées dans un format appelé Microsoft Tape Format (MTF) qui est un format commun utilisé par d'autres produits Microsoft en plus de SQL Server. Cela signifie que les sauvegardes de SQL Server peuvent être combinées avec d'autres sauvegardes, telles que les sauvegardes du système d'exploitation, sur les mêmes supports.

 **Remarque** Les sauvegardes compressées de SQL Server ne peuvent pas partager leur support avec d'autres types de sauvegarde.

## Unités de sauvegarde logiques

Il est possible d'indiquer un nom de fichier du système d'exploitation dans lequel la sauvegarde devra être écrite, directement dans la Commande BACKUP ou dans l'interface utilisateur graphique de SSMS. Toutefois, SQL Server permet un certain degré d'indirection par la création d'unités de sauvegarde logique. Une unité de sauvegarde logique est un nom défini par l'utilisateur facultatif qui fait référence à un emplacement de sortie spécifique.

En utilisant des unités de sauvegarde logiques, une application peut être conçue pour envoyer systématiquement et automatiquement les sauvegardes vers l'unité de sauvegarde logique et non vers un emplacement physique spécifique.

Par exemple, une RH peut être conçue pour créer des sauvegardes sur une unité de sauvegarde logique nommée HRBackupDevice. Un administrateur de base de données pourra ensuite déterminer à quel endroit les sauvegardes doivent être physiquement envoyées. L'administrateur pourra ensuite déterminer le nom d'un fichier qui devra contenir les sauvegardes de l'application RH. Il n'y a dans ce cas aucune modification à apporter à l'application RH pour s'adapter au changement d'emplacement du fichier de sauvegarde, car l'application enverra toujours la sauvegarde sur la même unité logique de sauvegarde.

## Mise en miroir de sauvegarde et agrégation par bande

Une sauvegarde unique peut cibler plusieurs unités de sauvegarde. Un seul support de sauvegarde peut prendre en charge jusqu'à 64 unités. Si plusieurs unités de sauvegarde sont utilisées, les sauvegardes peuvent être mises en miroir ou agrégées par bandes.

Avec une sauvegarde en miroir (disponible uniquement dans la version Enterprise edition), les mêmes données de sauvegarde sont écrites simultanément dans chaque unité de sauvegarde. Cette option assure la redondance de l'unité de sauvegarde physique. Une seule des unités suffit lors d'un processus de restauration.



**Remarque** Bien que la sauvegarde en miroir assure une certaine tolérance de panne par rapport au support après que la sauvegarde a été achevée, les sauvegardes en miroir sont en réalité très peu tolérantes aux pannes pendant le processus de sauvegarde. Si SQL Server ne peut pas écrire dans l'un des périphériques en miroir, l'intégralité de la sauvegarde échoue.

L'agrégation par bandes des sauvegardes provoque l'écriture sur un ensemble d'unités de sauvegarde d'une sauvegarde unique. Chaque unité de sauvegarde reçoit seulement une partie de la sauvegarde. Toutes les unités de sauvegarde doivent être présentes lors d'une restauration des données.

## Privilèges requis

La sauvegarde d'une base de données SQL Server requiert certaines autorisations qui sont présentées dans les modules 10 et 11.

**Question :** quelle est l'utilité d'agréger par bandes des sauvegardes sur plusieurs unités de sauvegarde d'un disque ?

## Définition d'une stratégie de conservation et de test pour les sauvegardes

- La planification de la conservation des sauvegardes doit faire partie de la stratégie
  - Doit faire partie du plan de test pour garantir l'exactitude
- Plusieurs éléments à prendre en compte
  - Combinaison de sauvegardes requises pour une récupération de base de données
  - Critères d'archivage
  - Synchronisation avec les vérifications de base de données
  - Emplacement de stockage sécurisé disponible
  - Matériel nécessaire pour restaurer les sauvegardes
  - Exhaustivité des sauvegardes

### Points clés

Une stratégie de sauvegarde doit comprendre des plans pour la conservation des sauvegardes et pour l'emplacement où seront conservés les sauvegardes ou leur support.

Il est trop fréquent que les entreprises effectuent des sauvegardes régulières mais, quand le moment de pratiquer une restauration est venu, celle-ci est impossible. La plupart de ces problèmes peuvent être résolus par un bon plan de rétention et de test.

Dans les paragraphes qui suivent, des exemples présentent les problèmes les plus fréquemment rencontrés et des mesures qui permettent de les éviter.

### Copies des sauvegardes insuffisantes

Quand la nécessité de restaurer les sauvegardes se fera sentir, votre entreprise dépendra de la qualité de celles-ci. Plus vous aurez de copies de vos sauvegardes, plus les supports contenant toutes les données nécessaires seront nombreux et plus vous aurez de chance de récupérer.

On considère généralement que la faute la plus grave consiste à créer une sauvegarde par-dessus la sauvegarde la plus récente. Si le système est défaillant pendant la sauvegarde, vous aurez alors perdu à la fois vos données et votre sauvegarde.

Pour bien illustrer ce propos, un administrateur de base de données A a récemment publié une demande d'aide sur le forum Microsoft SQL Server. Sa dernière sauvegarde datait d'un an, et il avait malencontreusement exécuté une opération de restauration au lieu d'une opération de sauvegarde, en utilisant son vieux fichier de l'année passée. Arrivé là, personne ne pouvait grand-chose pour l'aider.

Stratégie de prévention : plusieurs copies des sauvegardes.

### **Données suffisantes dans les sauvegardes**

La société A réalisait des sauvegardes régulières, mais aucun test de récupération n'avait jamais été effectué ; la première fois qu'une vraie récupération a été tentée, il a été découvert que parmi les fichiers qui devaient être sauvegardés, tous ne l'étaient pas.

Stratégie de prévention : reconstruction régulière des données à partir de tests de récupération sur sauvegarde.

### **Sauvegardes illisibles**

La société B réalisait des sauvegardes régulières mais ne les testait pas. Lors d'une tentative de récupération, aucune des sauvegardes n'était lisible. Cela est généralement causé par des défaillances matérielles mais cela peut aussi être dû au stockage inadapté du support.

Stratégie de prévention : test de récupération sur sauvegarde régulier.

### **Matériel non disponible**

La société C a acheté un lecteur de bande spécial pour effectuer ses sauvegardes. Une fois le moment venu de restaurer les sauvegardes, cet appareil spécial ne fonctionnait plus, et aucun autre périphérique de l'entreprise ne pouvait les lire même si elles étaient parfaitement valides.

Stratégie de prévention : test de récupération sur sauvegarde régulier.

### **Matériel ancien**

La société D réalisait des sauvegardes régulières et les conservait pendant une période de rétention correcte. Le moment venu de restaurer les sauvegardes, l'entreprise ne possédait plus aucun équipement capable de réaliser l'opération.

Stratégie de prévention : tests de récupération sur sauvegarde réguliers combinés à la récupération et à la sauvegarde sur des périphériques actuels.

### **Matériel mal aligné**

La société E effectuait des sauvegardes régulières et avait même testé la possibilité de réaliser des restaurations à partir des sauvegardes. Toutefois, comme ils testaient les restaurations sur le même lecteur qui réalisait les sauvegardes, ils ne sont pas rendus compte que le périphérique était mal aligné et que c'était le seul appareil capable de lire ces sauvegardes. Au moment où une restauration était nécessaire, le périphérique sur lequel étaient réalisées les sauvegardes était en panne.

Stratégie de prévention : test de récupération sur sauvegarde régulier sur un système séparé et une unité physique distincte.

### **Considérations d'ordre général**

Lorsqu'une stratégie de sauvegarde implique la réalisation de plusieurs types de sauvegardes, il est important de réfléchir à la combinaison des sauvegardes qui sera nécessaire lors d'une restauration.

Les entreprises doivent remplir des conditions légales ou de conformité en ce qui concerne la rétention des sauvegardes. Dans la plupart des cas, les sauvegardes complètes des bases de données sont conservées pendant une période plus longue que les autres types de sauvegarde.

Dans le cadre de la maintenance des bases de données, il est crucial de vérifier la cohérence des bases de données en utilisant DBCC CHECKDB. Ce point est abordé plus loin dans le cours.

De même que vous devez décider combien de temps conserver vos sauvegardes, vous devez décider à quel endroit les conserver. Une partie du concept de RTO consiste à envisager combien de temps il faut pour se procurer le support de sauvegarde physique s'il doit être restauré.

Vous devez également vérifier que les sauvegardes sont complètes. Tous les fichiers qui sont nécessaires pour récupérer le système (fichiers externes du système d'exploitation compris) sont-ils sauvegardés ?

## Leçon 2

# Description de la journalisation des transactions SQL Server

- Vue d'ensemble des journaux de transaction SQL Server
- Structure du fichier journal des transactions
- Utilisation des modes de récupération
- Planification des capacités pour les journaux de transaction
- Utilisation des options de point de contrôle
- Démonstration 2A : Journaux et récupération complète

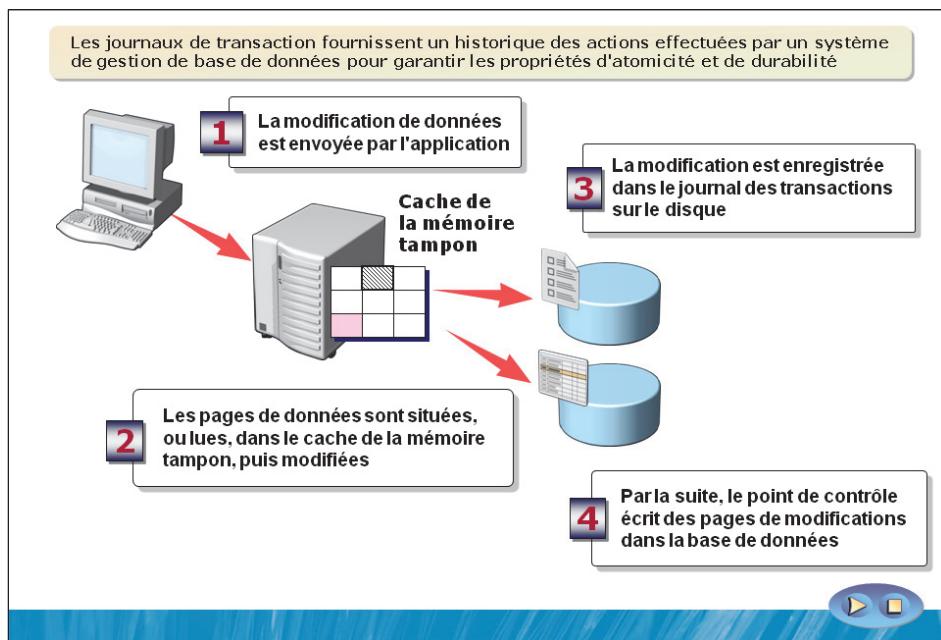
Le journal des transactions est la clé de la cohérence du moteur de base de données SQL Server. Il existe de nombreuses configurations du serveur qui peuvent affecter le fonctionnement du journal des transactions. Il est important que vous appreniez comment le journal des transactions fonctionne et comment configurer le journal des transactions pour qu'il réponde aux impératifs de récupération de votre organisation. La première configuration importante que vous devez prendre en compte est le mode de récupération sélectionné pour chaque base de données. La deuxième configuration critique est l'espace nécessaire pour l'activité du journal des transactions. Faire le bon choix pour ces deux configurations augmentera considérablement les chances que votre système réponde aux exigences de la récupération de base de données.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- expliquer le fonctionnement du journal des transactions SQL Server ;
- configurer les modes de récupération de base de données ;
- mettre en œuvre une planification de capacité pour les journaux des transactions ;
- configurer les options de point de contrôle.

## Vue d'ensemble des journaux de transaction SQL Server



### Points clés

Deux des conditions communes pour la gestion des transactions dans les systèmes de gestion de base de données sont l'atomicité et la durabilité des transactions. L'atomicité nécessite qu'une transaction soit validée entièrement ou qu'aucun travail du tout n'est validé. La durabilité nécessite qu'une fois qu'une transaction est validée, elle survivra à un redémarrage du système, même si la cause en est une panne. SQL Server utilise le journal des transactions pour vérifier l'atomicité et la durabilité des transactions.

Le journal des transactions peut être utilisé pour restaurer les transactions qui ont été partiellement effectuées, pour vérifier que les transactions ne sont pas laissées dans un état partiellement achevé. La restauration d'une transaction peut se produire en raison d'une demande d'un utilisateur ou d'une application cliente (telles que l'exécution d'une instruction ROLLBACK TRANSACTION), ou parce qu'une transaction a peut-être été partiellement exécutée au moment de la défaillance du système.

### Protocole WAL

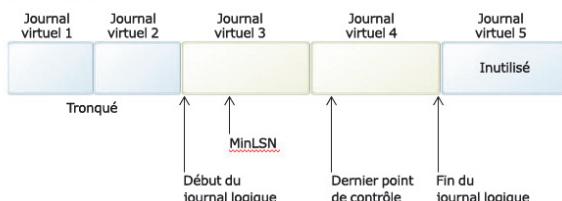
Lorsque SQL Server doit modifier les données dans une page de base de données, il vérifie si la page est présente dans le cache de la mémoire tampon. Si la page n'est pas présente, elle doit d'abord être lue dans le cache de la mémoire tampon. SQL Server modifie alors la page en mémoire et écrit les informations de restauration par progression et restauration (annulation) dans le journal des transactions. Pendant cette écriture, la page « modifiée » en mémoire est verrouillée par un mécanisme de verrouillage à court terme, ou verrou, jusqu'à ce que l'écriture dans le journal des transactions soit terminée. Ensuite, un processus de point de contrôle en arrière-plan vide les pages de modifications de la base de données. Ce processus s'appelle le protocole WAL (pour Write Ahead Logging). Tous les enregistrements sont écrits dans le journal avant que les pages de modification affectées ne soient écrites dans les fichiers de données ou que la transaction ne soit validée.

Le protocole WAL vérifie que la base de données peut toujours être définie dans un état cohérent après une défaillance. Ce processus de récupération est exposé en détail dans le module 7. Cependant le résultat du processus est que les transactions validées avant l'échec sont appliquées dans la base de données mais que les transactions qui étaient en cours au moment de la défaillance, alors que le travail était incomplet, sont annulées.

Le fait d'écrire à l'avance toutes les modifications apportées au fichier journal permet également de restaurer les transactions si nécessaire.

## Structure du fichier journal des transactions

- Des informations suffisantes sont stockées pour permettre de
  - Restaurer les transactions en cas de besoin
  - Récupérer la base de données en cas d'échec
- Le protocole WAL (Write Ahead Logging) est utilisé pour créer des entrées de journal
  - Les journaux de transactions sont écrits dans l'ordre chronologique de façon circulaire
  - La stratégie de troncation des journaux est basée sur le mode de récupération



### Points clés

SQL Server doit s'assurer qu'il dispose de suffisamment d'informations dans les journaux de transactions pour traiter les demandes de restauration des utilisateurs ou des applications, et pour récupérer la base de données. SQL Server doit également conserver des entrées du fichier journal suffisantes pour répondre aux besoins des autres fonctionnalités SQL Server qui doivent accéder au moteur de base de données, telles que la réplication et la capture de données modifiées.

### Structure du journal des transactions et fichiers journaux virtuels

Les journaux de transactions sont écrits dans l'ordre chronologique de façon circulaire. Étant donné que le journal des transactions devra peut-être s'agrandir, en interne il est divisé en un ensemble de fichiers journaux virtuels (VLF) au lieu d'être structuré comme un fichier unique. La taille et le nombre des fichiers journaux virtuels d'un fichier journal physique sont variables. Le moteur de base de données choisit la taille des fichiers journaux virtuels dynamiquement lors de la création ou du développement des fichiers journaux, en fonction de la croissance qui se produit et conformément aux valeurs présentées dans le tableau suivant :

Incrément de croissance	Nombre de VLF
Incrément de croissance <= 64 Mo	4
64 Mo > Incrément de croissance <= 1 Go	8
1 Go < Incrément de croissance	16

Lorsqu'une écriture du journal atteint la fin du journal existant, SQL Server bascule au début et poursuit l'écriture. Ce mécanisme fonctionne aussi longtemps que les données précédentes de cette section du fichier journal ont déjà été écrites dans la base de données et que cet espace a été libéré. Cela « qui libère vers le haut » du processus est sous le terme de « troncation du journal ».

Si, lors du remplacement du contenu précédent du fichier journal, le point où il devient nécessaire de remplacer des données qui doivent être conservées est atteint, SQL Server augmente la taille du fichier journal. Si le fichier journal n'a pas été configuré pour autoriser la croissance automatique ou si le volume disque qui contient le journal est plein, SQL Server ne peut accomplir la transaction et renvoie une erreur indiquant que le fichier journal est plein.

Lorsqu'il est possible d'augmenter la taille du fichier journal, SQL Server alloue de nouveaux fichiers journaux virtuels, selon l'incrément de croissance automatique qui a été choisi dans la configuration du journal.

 **Remarque** Il est impossible d'utiliser l'initialisation instantanée du fichier (IFI) avec les journaux de transactions. Cela signifie que les transactions peuvent être bloquées pendant la croissance du fichier journal.

### Troncation du fichier journal

Il existe deux manières de tronquer un journal des transactions. Les journaux des transactions sont normalement tronqués dans le cadre de leur processus de sauvegarde. La deuxième option consiste à avoir sélectionné le mode de récupération de base de données qui tronque automatiquement le contenu du fichier journal.

Chaque fois que le fichier journal est tronqué, seules les données allant jusqu'au début de la transaction active la plus ancienne peuvent être tronquées. Les entrées du journal sont classées logiquement en fonction d'une valeur appelée numéro de séquence d'enregistrement (NSE). Le point de départ de la transaction active la plus ancienne dans le fichier journal est appelé la valeur MinLSN. Toutes les entrées de journal dont la valeur de séquence d'enregistrement est supérieure à la valeur MinLSN ne peuvent pas être tronquées car elles peuvent être indispensables à une éventuelle récupération. À chaque troncation, seules les données qui vont jusqu'au début du journal virtuel qui contient la valeur minimale du point de départ de la dernière opération de point de contrôle, MinLSN, et la transaction la plus ancienne qui n'est pas encore répliquée (uniquement lors de l'utilisation de la réplication) sont tronquées.

 **Remarque** La réplication n'est pas traitée dans ce cours. Il est cependant important de savoir que la configuration et l'état des données répliquées peuvent affecter la troncation du journal des transactions.

**Question** : pourquoi les performances d'écriture dans le fichier journal sont-elles si importantes pour les performances de transaction d'une base de données ?

## Utilisation des modes de récupération

Mode de récupération	Description
<b>Simple</b>	<ul style="list-style-type: none"> <li>N'autorise pas ou ne requiert pas de sauvegardes de journal</li> <li>Tronque automatiquement le journal afin de minimiser l'espace requis</li> </ul>
<b>Complète</b>	<ul style="list-style-type: none"> <li>Nécessite des sauvegardes de journal pour faciliter la gestion</li> <li>Évite la perte de données causée par un fichier de données endommagé ou manquant</li> <li>Permet la récupération à un point spécifié dans le temps</li> </ul>
<b>Journalisée en bloc</b>	<ul style="list-style-type: none"> <li>Nécessite des sauvegardes de journal pour faciliter la gestion</li> <li>Peut améliorer les performances des opérations de copie en bloc</li> <li>Réduit l'espace du journal utilisé en utilisant un enregistrement minimal pour la plupart des opérations en bloc</li> </ul>

### Points clés

SQL Server possède trois modes de récupération de base de données. Ces 3 modes de récupération modes préserveront les données en cas d'incident. Ces modes présentent néanmoins des différences importantes qui doivent être prises en compte par l'opérateur de base de données lors de la sélection d'un mode pour sa base de données.

Le choix du mode de récupération approprié joue un rôle important dans votre stratégie de la récupération. Le mode de récupération sélectionné pour votre base de données détermine de nombreux facteurs, notamment :

- la charge de travail de la maintenance ;
- l'exposition à une perte éventuelle ;
- les types de sauvegarde disponibles.

Lors du choix d'un mode de récupération pour votre base de données, il convient de tenir compte de sa taille, de la charge de travail éventuelle liée à sa maintenance et du niveau de risque acceptable de perte de données potentielle.

### Mode de récupération simple

Le mode de récupération simple réduit les tâches d'administration du journal des transactions car celui-ci n'est pas sauvegardé. Il entraîne un risque de perte de travail assez élevé en cas d'endommagement de la base de données. Les données ne sont récupérables que jusqu'à la sauvegarde la plus récente effectuée. Par conséquent, en mode de récupération simple, les intervalles de sauvegarde doivent être suffisamment courts pour empêcher la perte d'un volume significatif de données. Toutefois, les intervalles doivent être suffisamment longs pour que la charge de sauvegarde n'affecte pas la production. L'inclusion des sauvegardes différentielles dans une stratégie de sauvegarde selon le mode de récupération simple peut réduire la charge de traitement.

Dans les versions antérieures de SQL Server, le mode de récupération simple était désigné comme « Tronquer le journal au point de contrôle ». Ce nom a été modifié pour centrer l'attention sur les options de récupération plutôt que sur le processus impliqué pour dans la mise en œuvre de l'option. À chaque occurrence d'un processus de point de contrôle, SQL Server tronque automatiquement le journal des transactions jusqu'à la fin du fichier journal virtuel qui précède le journal virtuel qui contient la valeur MinLSN. Cela signifie que le seul rôle du journal des transactions est de fournir les données des transactions actives lors de la récupération d'une base de données.

### Mode de restauration complète

Le mode de récupération complète correspond au mode de maintenance de base de données normal pour les bases de données pour lesquelles la durabilité des transactions est nécessaire. Le mode de restauration complète est le mode de récupération par défaut lors de l'installation de SQL Server.

Le mode de récupération pour les nouvelles bases de données est fondé sur le mode de récupération de la base de données model. Celui-ci peut être modifié sur les sites où l'on préfère un autre mode de récupération par défaut.

Avec le mode de restauration complète, il est obligatoire de réaliser la sauvegarde des fichiers journaux. Ce mode de récupération consigne complètement toutes les transactions et conserve les enregistrements du journal des transactions après leur sauvegarde. Le mode de récupération complète permet de récupérer une base de données jusqu'au point de défaillance, en partant du principe que la fin du journal peut être sauvegardée après la défaillance. Le mode de restauration complète prend également en charge une option de restauration de pages de données individuelles ou de restauration à l'état d'un moment particulier.

### Mode de récupération utilisant les journaux de transactions

Ce mode de récupération peut réduire les besoins de consignation dans le journal des transactions pour de nombreuses opérations en bloc. Il a été conçu uniquement comme complément au mode de récupération complète. Par exemple, lors de l'exécution de certaines opérations en bloc à grande échelle telles qu'une importation en bloc ou une création d'index, une base de données peut être basculée temporairement dans le mode de récupération utilisant les journaux de transaction. Ce basculement temporaire peut améliorer les performances en n'enregistrant que les allocations d'étendue et en réduisant la consommation d'espace de journal.

Les sauvegardes du journal des transactions sont toujours nécessaires à l'utilisation du mode de récupération utilisant les journaux de transactions. Comme le mode de récupération complète, le mode de récupération utilisant les journaux de transactions conserve les enregistrements du journal des transactions après leur sauvegarde. En revanche, il génère des sauvegardes du journal plus volumineuses et un risque accentué de perte de travail, car le mode de récupération utilisant les journaux de transactions ne prend pas en charge la récupération jusqu'à une date et heure.

 **Remarque** Un résultat surprenant est que les sauvegardes des fichiers journaux peuvent souvent être plus grandes que les journaux des transactions. Cela est dû au fait que SQL Server récupère les étendues modifiées à partir de fichiers de données tout en réalisant la sauvegarde de données journalisées de façon minimale.

**Question :** quel type d'organisation peut considérer le mode de récupération simple comme adéquat ?

## Planification des capacités pour les journaux de transaction

- Les besoins en capacité reposent sur plusieurs facteurs
  - Mode de récupération utilisé pour la base de données
  - Fréquence de sauvegarde du journal des transactions en mode de récupération complète et journalisée en bloc
  - Nombre et taille des transactions dans la base de données
- Examinez le comportement du journal pendant le test de pré déploiement

### Points clés

Le mode de récupération que vous choisissez pour une base de données aura une grande incidence sur la taille du fichier journal de la base de données.

En mode de récupération simple, SQL Server tronque le journal après chaque point de contrôle. En mode de récupération utilisant les journaux de transactions, le journal est tronqué après chaque sauvegarde de journal, pour vérifier qu'il existe une chaîne ininterrompue de fichiers journaux de sauvegarde. La troncation d'un fichier journal se produit après une sauvegarde de fichier journal. C'est une idée répandue qu'une sauvegarde complète de base de données interrompt la chaîne des sauvegardes de fichier journal, mais cela est faux.

### Détermination de la taille du fichier journal

Il est très difficile de calculer les spécifications de taille des fichiers journaux. Comme avec d'autres aspects de SQL Server, l'observation d'un test réaliste fournit les meilleures indications.

Il existe une autre idée courante qui dit que le fichier journal d'une base de données en mode de récupération simple n'augmente pas. Ce n'est pas vrai non plus. En mode de récupération simple, le journal des transactions doit être suffisamment grand pour contenir toutes les informations de la transaction active la plus ancienne. Les grandes transactions ou les transactions longues peuvent causer la nécessité d'ajouter de l'espace supplémentaire au fichier journal.

Une des questions les plus courantes posées par les utilisateurs de SQL Server sur les forums concerne la manière de tronquer un fichier journal. Par exemple, un utilisateur indique que dans son entreprise ils ont une base de données de 20 Go mais un fichier journal de 300 Go. Cette situation est caractéristique de la création de bases de données en mode de restauration complète accompagnée d'une mauvaise gestion des journaux de bases de données. En raison de ce manque de gestion, les fichiers journaux continuent tout bonnement à croître. Souvent l'utilisateur a également tenté de tronquer le journal des transactions (sans succès). Des sauvegardes du journal suivies d'une réduction auraient corrigé le problème.

## Impossibilité de tronquer un journal

D'autres fonctionnalités de SQL Server peuvent également vous empêcher de tronquer des fichiers journaux. Par exemple, la mise en miroir de bases de données, la réPLICATION transactionnelle et la capture de données modifiées peuvent affecter la capacité du moteur de base de données à tronquer les fichiers journaux.

 **Remarque** La mise en miroir de bases de données, la réPLICATION transactionnelle et la capture de données modifiées n'entrent pas dans le cadre de ce cours.

Si vous vous demandez pourquoi un journal des transactions ne peut pas être tronqué, exécuter la requête suivante peut vous aider :

```
SELECT name, log_reuse_wait_desc FROM sys.databases;
```

Les valeurs actuelles qui peuvent être retournées pour la colonne log\_reuse\_wait\_desc sont :

0 = Nothing (Rien)

1 = Checkpoint (Point de contrôle)

2 = Log Backup (Sauvegarde de journal)

3 = Active backup or restore (Sauvegarde ou restauration active)

4 = Active transaction (Transaction active)

5 = Database mirroring (Mise en miroir de bases de données)

6 = Replication (RéPLICATION)

7 = Database snapshot creation (Création d'une capture instantanée de base de données)

8 = Log Scan (Analyse du journal)

9 = Other (transient) (Autre (transitoire))

Après avoir résolu le problème affiché, effectuez une sauvegarde de fichier journal (si vous utilisez le mode de récupération complète) pour tronquer le journal, puis utilisez DBCC SHRINKFILE pour réduire sa taille.

 **Remarque** Si l'utilisation de DBCC SHRINKFILE comme indiqué ci-dessus ne permet pas de réduire la taille du fichier journal, c'est que sa partie active devait se trouver à la fin du journal à ce moment.

**Question** : quelle raison la plus courante justifierait une augmentation excessive du fichier journal ?

## Utilisation des options de point de contrôle

- Type d'opérations de point de contrôle
  - Automatique
  - Indirect
  - Manuel
  - Interne
- Instruction CHECKPOINT
  - Permet de définir la durée de récupération cible

### Points clés

SQL Server a quatre types d'opération de point de contrôle :

- L'option Automatique est activée automatiquement en arrière-plan pour respecter la limite de durée supérieure suggérée par l'option de configuration du serveur Intervalle de récupération. Les points de contrôle automatiques s'exécutent jusqu'à la fin. Les points de contrôle automatiques peuvent être accélérés en fonction du nombre d'écritures en attente et si le moteur de base de données détecte une augmentation de la latence d'écriture supérieure à 20 millisecondes.
- L'option Indirect est activée en arrière-plan pour respecter le délai de récupération cible spécifié pour une base de données spécifique. Le temps de récupération cible par défaut est 0, ce qui entraîne l'application des paramètres de point de contrôle à la base de données. Si vous avez utilisé ALTER DATABASE pour définir TARGET\_RECOVERY\_TIME à une valeur supérieure à zéro, cette valeur est utilisée au lieu de l'intervalle de récupération spécifié pour l'instance de serveur.

- L'option Manuel est activée lorsque vous exécutez une commande CHECKPOINT Transact-SQL. Le point de contrôle manuel intervient dans la base de données à laquelle vous êtes actuellement connecté. Par défaut, les points de contrôle manuels s'exécutent jusqu'à la fin. Le paramètre facultatif de durée du point de contrôle indique une durée requise, en secondes, pour l'achèvement du point de contrôle.
- L'option Interne est activée par différentes opérations du serveur telles que la création d'une capture instantanée de base de données ou de sauvegarde pour garantir que les images de disque correspondent à l'état actuel du journal.

Vous pouvez contrôler le temps cible d'une opération de point de contrôle en exécutant l'instruction CHECKPOINT, comme suit :

```
CHECKPOINT 5;
```

## Démonstration 2A : Journaux et récupération complète

Dans cette démonstration, vous allez découvrir le fonctionnement de la troncation de journal en mode de récupération complète

### Procédure de démonstration

1. Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn.** Dans le menu **Fichier** cliquez sur **Ouvrir**, sur **Projet/Solution**, naviguez jusqu'à **F:\22462A\_Labs\22462A\_05\_PRJ\22462A\_05\_PRJ.ssmssqln** puis cliquez sur **Ouvrir**.
3. Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** à partir de l'Explorateur de solutions.
4. Ouvrez le fichier de script **21 – Démonstration 2A.sql**.
5. Suivez les instructions contenues dans les commentaires du fichier de script.

## Leçon 3

# Planification d'une stratégie de sauvegarde SQL Server

- Vue d'ensemble des types de sauvegarde Microsoft SQL Server
- Stratégies de sauvegarde complète de base de données
- Stratégies de sauvegarde du journal des transactions
- Stratégies de sauvegarde différentielle
- Discussion : Répondre aux besoins de récupération de l'entreprise

Maintenant que vous avez acquis des connaissances sur les journaux des transactions de SQL Server et sur les modes de récupération de base de données, il est temps de déterminer quels sont les types de sauvegardes disponibles avec SQL Server.

En plus d'apprendre des généralités sur tous les types de sauvegarde disponibles, il est important de connaître de manière plus approfondie les trois types de sauvegarde les plus courants. Ces types sont : la sauvegarde complète de base de données, la sauvegarde différentielle et la sauvegarde du journal des transactions.

Pour organiser une stratégie de sauvegarde efficace vous devez conformer votre combinaison de types de sauvegarde aux besoins de récupération de votre entreprise. Il est préférable dans la plupart des organisations d'utiliser une combinaison de types de sauvegarde plutôt que de compter sur un type unique.

## Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- détailler les types de sauvegardes disponibles avec Microsoft SQL Server ;
- décrire le mode de sauvegarde complète de bases de données ;
- décrire la sauvegarde différentielle ;
- décrire la sauvegarde des journaux des transactions.

## Vue d'ensemble des types de sauvegarde Microsoft SQL Server

Types de sauvegardes	Description
Complète	Tous les fichiers de données et une partie active du journal des transactions
Différentielle	Parties de la base de données modifiées depuis la dernière sauvegarde complète
Partielle	Groupe de fichiers primaire, groupes de fichiers en lecture/écriture et groupes de fichiers spécifiés en lecture seule
Journal des transactions	Modifications de bases de données enregistrées dans les journaux
Sauvegarde de la fin du journal	Sauvegarde de la fin du journal effectuée juste avant une opération de restauration
Fichier/Groupe de fichiers	Fichiers ou groupes de fichiers spécifiés
Copie seule	Base de données ou journal (sans impact sur la séquence de sauvegarde)

### Points clés

Avant de mener une exploration plus détaillée, il est important de connaître tous les types de sauvegarde qui sont disponibles dans SQL Server. Tous les types de sauvegarde ne sont disponibles pour tous les modes de récupération de base de données. Par exemple, les sauvegardes du journal des transactions ne peuvent pas être réalisées pour une base de données qui est en mode de récupération simple.

### Sauvegardes complètes

Une sauvegarde complète de base de données inclut les fichiers de données et la partie active du journal des transactions. La première étape de la sauvegarde est l'exécution d'une opération CHECKPOINT. La partie active du journal des transactions comporte toutes les informations de la transaction active la plus ancienne. Une sauvegarde complète représente la base de données au moment où a été réalisée la phase de lecture de la sauvegarde, et fait office de référence en cas de défaillance du système. Les sauvegardes complètes ne tronquent pas le journal des transactions.

### Sauvegardes différentielles

Une sauvegarde différentielle sert à n'enregistrer que les données qui ont été modifiées depuis la dernière sauvegarde complète. Les sauvegardes différentielles sont basées sur le contenu du fichier de données plutôt que sur le contenu du fichier journal et contiennent les étendues qui ont été modifiées depuis la dernière sauvegarde complète de la base de données. Les sauvegardes différentielles sont généralement plus rapides à restaurer que les sauvegardes du journal des transactions, mais elles proposent moins d'options. Par exemple, la récupération dans le temps n'est pas disponible à moins que des sauvegardes différentielles ne soient combinées avec des sauvegardes de fichier journal.

## Sauvegardes partielles

Une sauvegarde partielle est similaire à une sauvegarde complète, mais la première ne contient pas tous les groupes de fichiers. Les sauvegardes partielles contiennent toutes les données du groupe de fichiers primaire, de chaque groupe de fichiers en lecture/écriture, ainsi que de tout fichier en lecture seule. La sauvegarde partielle d'une base de données en lecture seule contient uniquement le groupe de fichiers primaire.

 **Remarque** L'utilisation des sauvegardes partielles est un sujet avancé qui n'entre pas dans le cadre de ce cours.

## Sauvegardes des journaux des transactions

Les sauvegardes du journal des transactions enregistrent toutes les modifications de la base de données en sauvegardant les enregistrements du journal des transactions. Les sauvegardes du journal des transactions permettent la récupération dans le temps et elles ont une taille généralement beaucoup plus faible que les sauvegardes complètes de base de données. Comme les sauvegardes du journal des transactions sont plus petites, vous pouvez les exécuter beaucoup plus fréquemment. Une fois le journal des transactions sauvegardé, les enregistrements de journal qui ont été sauvegardés et qui ne sont pas dans la partie active du journal sont tronqués. Les sauvegardes du journal des transactions ne sont pas disponibles en mode de récupération simple.

## Sauvegardes de la fin du journal

Une sauvegarde du journal des transactions qui est effectuée juste avant une opération de restauration est nommée « sauvegarde de la fin de journal ». En général, les sauvegardes de la fin du journal sont effectuées après une défaillance de disque qui affecte uniquement les fichiers de données. Depuis la version SQL Server 2005, SQL Server vous demande de réaliser une sauvegarde de la fin du journal avant de vous permettre de restaurer une base de données, afin de vous protéger contre une perte de données accidentelle.

Il est souvent possible de réaliser des sauvegardes de la fin du journal même si les fichiers de données de la base de données ne sont plus accessibles.

## Sauvegardes de fichiers ou de groupes de fichiers

Si l'exécution d'une sauvegarde complète de bases de données très volumineuses ne convient pas, vous pouvez effectuer des sauvegardes de fichiers ou de groupes de fichiers.

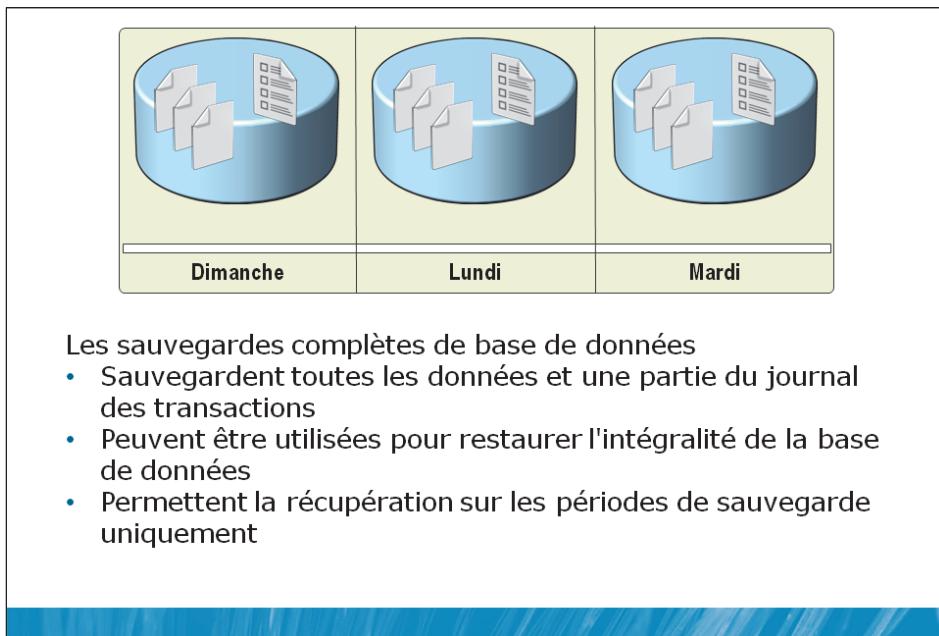
 **Remarque** L'utilisation des sauvegardes de fichiers et de groupes de fichiers est un sujet avancé qui n'entre pas dans le cadre de ce cours.

## Sauvegardes de copie seule

SQL Server 2005 et les versions ultérieures prennent en charge la création de sauvegardes de copie uniquement. Contrairement aux autres sauvegardes, les sauvegardes de copie uniquement n'ont aucun impact sur les procédures globales de sauvegarde et de restauration de la base de données. Les sauvegardes de copie uniquement peuvent être utilisées pour créer une copie de la sauvegarde afin de la mettre en lieu sûr. Elles sont également utiles lors de l'exécution d'opérations de restauration en ligne. Tous les modes de récupération prennent en charge les sauvegardes de données de copie seule.

**Question :** quel type de base de données tirerait profit de sauvegardes partielles ?

## Stratégies de sauvegarde complète de base de données



Les sauvegardes complètes de base de données

- Sauvegardent toutes les données et une partie du journal des transactions
- Peuvent être utilisées pour restaurer l'intégralité de la base de données
- Permettent la récupération sur les périodes de sauvegarde uniquement

### Points clés

Une stratégie de sauvegarde complète de la base de données implique des sauvegardes complètes et régulières afin de conserver la base de données. En cas de échec, la base de données peut être restaurée dans l'état de la dernière sauvegarde complète.

### Sauvegardes complètes de bases de données

Une sauvegarde complète de base de données consiste à sauvegarder la base de données entière ainsi que la partie du journal des transactions qui couvre les modifications survenues lors de la lecture des pages de données.

Les sauvegardes complètes de base de données représentent une copie de la base de données à la fin de la phase de lecture des données, et non au début. Il est possible d'effectuer des sauvegardes alors que le système est en cours d'utilisation. À la fin de la sauvegarde, SQL Server écrit dans la sauvegarde les entrées du journal des transactions qui couvrent la période d'exécution de la sauvegarde.

### Scénarios courants

Pour une petite base de données dont la sauvegarde peut s'effectuer rapidement, il est recommandé d'utiliser des sauvegardes complètes. Cependant, au fur et à mesure que la taille d'une base de données augmente, les sauvegardes complètes prennent de plus en plus de temps et d'espace de stockage.

Par conséquent, pour les bases de données volumineuses, il est conseillé de compléter les sauvegardes complètes par d'autres formes de sauvegardes.

En mode de récupération simple, après chaque sauvegarde, la base de données est exposée à des pertes de travail potentielles en cas de sinistre. Le risque de perte de travail augmente après chaque mise à jour, et ce, jusqu'à la sauvegarde complète suivante, après laquelle le risque de perte de travail redévient nul et un nouveau cycle de risque de perte de travail commence.

Parmi les scénarios éventuellement adaptés à une stratégie de sauvegarde complète de la base de données, peuvent être cités :

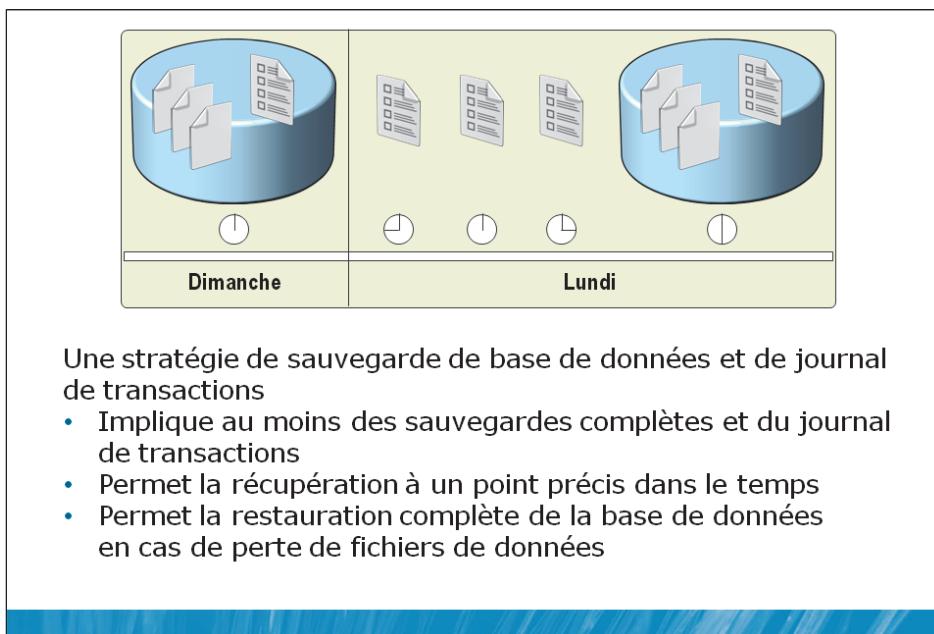
- les systèmes de test ;
- les entrepôts de données où les données peuvent être récupérées à partir d'un système source et où les données de l'entrepôt de données ne changent pas régulièrement ;
- les systèmes où les données peuvent être récupérées à partir d'autres sources.

### **Exemple de la diapositive**

Dans l'exemple de la diapositive, une sauvegarde complète est effectuée le dimanche, le lundi et le mardi. Cela signifie que pendant la journée du lundi, il est possible de perdre jusqu'à une journée complète de données, jusqu'à l'exécution de la sauvegarde. La journée du mardi est tout aussi exposée. Après la sauvegarde du mardi, le risque augmente chaque jour jusqu'au dimanche où une nouvelle sauvegarde est réalisée.

**Question** : une base de données de petite taille convient-elle pour une stratégie de sauvegarde complète de base de données ?

## Stratégies de sauvegarde du journal des transactions



Une stratégie de sauvegarde de base de données et de journal de transactions

- Implique au moins des sauvegardes complètes et du journal de transactions
- Permet la récupération à un point précis dans le temps
- Permet la restauration complète de la base de données en cas de perte de fichiers de données

### Points clés

Une stratégie de sauvegarde qui implique des sauvegardes du journal des transactions doit être combinée avec une stratégie de sauvegarde de base de données complète ou avec une stratégie qui combine l'utilisation des sauvegardes complètes et différentielles.

### Sauvegardes des journaux des transactions

Les sauvegardes du journal des transactions conservent toutes les données depuis la dernière sauvegarde du journal. Au lieu de la lecture des pages de la base de données, les sauvegardes du journal des transactions sont basées sur la lecture des données du journal des transactions. Une stratégie de sauvegarde basée sur les sauvegardes du journal des transactions est adaptée aux bases de données fréquemment modifiées.

Lorsqu'il est nécessaire de récupérer une base de données, la dernière sauvegarde complète de base de données doit être restaurée ainsi que la sauvegarde différentielle la plus récente (s'il en existe une). Après que la base de données a été restaurée, les journaux des transactions qui ont été sauvegardés depuis ce moment sont également restaurés, dans l'ordre. Étant donné que la restauration fonctionne sur une base transactionnelle, il est possible de restaurer une base de données à un moment précis des transactions journalisées dans la sauvegarde de fichier journal.

En plus de fournir des fonctions qui vous permettent de restaurer les transactions qui ont été sauvegardées, la sauvegarde du journal des transactions tronque celui-ci. Cela permet de réutiliser les VLF dans le journal des transactions. Si vous ne sauvegardez pas le journal suffisamment souvent, les fichiers journaux peuvent se remplir complètement.

### Exemple de la diapositive

Dans l'exemple présenté sur la diapositive, les sauvegardes complètes de base de données effectuées la nuit sont complétées par des sauvegardes des journaux des transactions effectuées régulièrement pendant la journée. Si le système tombe en panne, la récupération peut être effectuée à l'heure de la dernière sauvegarde du journal des transactions. Si, en revanche, seuls les fichiers de données de la base de données sont défaillants et qu'une sauvegarde de la fin du journal a pu être réalisée, aucune des données validées ne sera perdue.

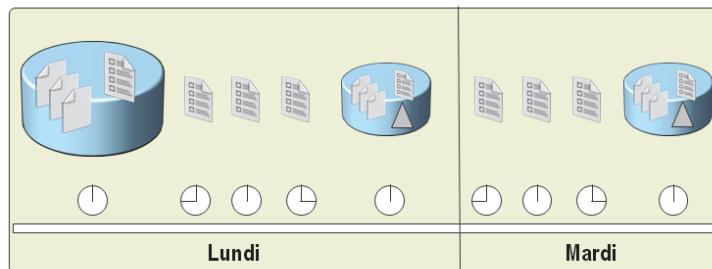
### Combinaisons de différents types de sauvegarde

Les sauvegardes du journal des transactions sont généralement beaucoup plus petites que les autres sauvegardes, notamment si elles sont réalisées régulièrement. La perte de données peut être réduite au maximum par une stratégie de sauvegarde basée sur les sauvegardes du journal des transactions en association avec d'autres types de sauvegarde.

Comme les sauvegardes des fichiers journaux prennent en général plus longtemps à restaurer que d'autres types de sauvegarde, il est conseillé de combiner la sauvegarde des journaux de transactions avec des sauvegardes différentielles régulières. Lors d'une récupération, seules les sauvegardes du journal des transactions qui ont été effectuées après la dernière sauvegarde différentielle doivent être restaurées.

**Question :** d'après l'exemple de la diapositive, quelles transactions seront contenues dans la première sauvegarde du journal le lundi matin ?

## Stratégies de sauvegarde différentielle



### Une stratégie de sauvegarde différentielle

- Implique l'exécution de sauvegardes de base de données complètes et différentielles
- Inclut les sauvegardes différentielles contenant uniquement des données modifiées
- S'avère utile uniquement si un sous-ensemble d'une base de données est modifié plus fréquemment que le reste de la base de données

### Points clés

Les sauvegardes différentielles constituent un bon moyen de réduction des risques de perte de travail et de réduction de la surcharge liée à la maintenance. Lorsque la proportion d'une base de données modifiée entre les intervalles de sauvegarde est beaucoup plus petite que la taille entière de la base de données, il peut être utile d'envisager une stratégie de sauvegarde différentielle. Mais si votre base de données est très petite, les sauvegardes différentielles peuvent ne pas vous permettre de gagner beaucoup de temps.

### Sauvegardes différentielles

SQL Server maintient une carte des étendues modifiées depuis la dernière sauvegarde complète. Dans une sauvegarde différentielle, SQL Server sauvegarde les étendues qui ont été modifiées. Il est toutefois important de se rendre compte que SQL Server n'efface pas cette carte des étendues modifiées une fois la sauvegarde différentielle effectuée. La carte n'est effacée que lors de la sauvegarde complète suivante. Cela signifie qu'une deuxième sauvegarde différentielle effectuée sur une base de données comprendra toutes les modifications apportées depuis la dernière sauvegarde complète, et pas simplement les modifications apportées depuis la dernière sauvegarde différentielle.

### Scénarios utiles pour les sauvegardes différentielles

Étant donné qu'elles n'enregistrent que les données qui ont été modifiées depuis la dernière sauvegarde complète, les sauvegardes différentielles sont généralement beaucoup plus rapides et occupent moins d'espace disque que les sauvegardes du journal des transactions pour la même période.

Les sauvegardes différentielles sont particulièrement utiles dans les cas où un sous-ensemble de base de données est modifié plus fréquemment que le reste de la base de données. En pareil cas, les sauvegardes différentielles vous permettent d'effectuer des sauvegardes régulières sans les contraintes liées aux sauvegardes complètes.

### Exemple de la diapositive

Dans l'exemple de la diapositive, une sauvegarde complète de la base de données est effectuée à minuit le dimanche soir (tôt le lundi matin). Des sauvegardes différentielles sont ensuite effectuées à minuit les autres soirs de la semaine. La sauvegarde différentielle effectuée le lundi soir comprendrait toutes les données modifiées au cours de la journée du lundi. La sauvegarde différentielle effectuée mardi soir comprendrait toutes les données modifiées dans les journées de lundi et mardi. La sauvegarde différentielle effectuée le vendredi soir comprendrait toutes les données qui ont été modifiées lundi, mardi, mercredi, jeudi, et vendredi. Cela signifie que les sauvegardes différentielles peuvent considérablement augmenter de taille entre chaque intervalle de sauvegarde complète.

### Combinaisons de sauvegardes

Les sauvegardes différentielles doivent être combinées avec d'autres formes de sauvegarde. Comme une sauvegarde différentielle enregistre toutes les données modifiées depuis la dernière sauvegarde complète, il est impossible de réaliser une sauvegarde différentielle si une sauvegarde complète n'a pas été réalisée.

Un autre point important à prendre en considération est que, lorsqu'une restauration est nécessaire, plusieurs sauvegardes doivent être restaurées pour remettre le système en ligne et non une sauvegarde unique. Cela augmente le risque de sécurité d'une organisation et il convient de le prendre en compte lors de la planification d'une stratégie de sauvegarde.

Les sauvegardes différentielles peuvent également être utilisées en association avec les sauvegardes complètes et les sauvegardes du journal des transactions.

**Question :** comment pouvez-vous estimer la taille d'une sauvegarde différentielle ?

## Discussion : Répondre aux besoins de récupération de l'entreprise

- Comment concevriez-vous une stratégie de sauvegarde d'une base de données prenant en charge un magasin en ligne ?



**Question :** imaginez-vous devoir élaborer une stratégie de récupération pour le site d'un magasin en ligne. Avant de pouvoir concevoir la stratégie, quelles questions devez-vous poser ?

## Atelier pratique 5 : Description des modes de récupération SQL Server

- Exercice 1 : Planifier une stratégie de sauvegarde
- Exercice 2 : Configurer les modes de récupération
- Exercice difficile 3 : Examiner les modes et les stratégies de récupération (si le temps le permet)

Informations d'ouverture de session

Ordinateur virtuel	22462A-MIA-SQL1
Nom d'utilisateur	AdventureWorks\Administrateur
Mot de passe	Pa\$\$w0rd

**Durée approximative : 45 minutes**

### Configuration de l'atelier pratique

Pour cet atelier pratique, vous utiliserez l'environnement d'ordinateur virtuel disponible. Avant de débuter cet atelier pratique, vous devez procéder aux étapes suivantes :

1. Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
3. Dans la fenêtre Se connecter au serveur, tapez **Proseware** dans la zone de texte **Nom du serveur**.
4. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn.**
5. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**.
6. Dans la fenêtre Ouvrir un projet, sélectionnez le projet **D:\22462A\_Labs\22462A\_05\_PRJ\22462A\_05\_PRJ.ssmssln**.
7. Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Dans l'Explorateur de solutions, double-cliquez sur le fichier de requête **00 - Setup.sql**. Lorsque la fenêtre de requête s'affiche, cliquez sur **Exécuter** dans la barre d'outils.

## Scénario de l'atelier pratique

Vous devez implémenter une stratégie de récupération de base de données. La division commerciale de Proseworks Inc. vous a indiqué les besoins de disponibilité des bases de données sur la nouvelle instance SQL Server de Proseware. Vous devez planifier comment répondre à ces impératifs puis comment mettre en œuvre votre stratégie.

Si vous avez le temps, il existe une autre problème que votre gestionnaire souhaite vous confier. Une autre instance de SQL Server est installée pour prendre en charge les opérations du service client. Il se pourrait que les bases de données existantes sur l'instance de serveur ne soient pas configurées correctement et que leurs stratégies de sauvegarde ne soient pas valides d'après les exigences de RPO et de RTO. Dans cet exercice, vous devez examiner les modes de récupération de base de données et les stratégies de sauvegarde pour les bases de données de l'instance CustomerService et fournir des modifications recommandées.

### Documentation

#### Exigences de continuité des bases de données d'entreprise sur l'instance de serveur Proseware (pour les exercices 1 et 2)

##### Objectifs de temps de récupération

1. La base de données MarketDev ne doit jamais être indisponible pendant plus de huit heures.
2. La base de données Recherche ne doit jamais être indisponible pendant plus de deux heures.

##### Objectifs de point de restauration

1. Lorsque la base de données MarketDev est récupérée à partir d'une défaillance, il ne doit pas y avoir plus de 30 minutes de transactions de perdues.
2. Lorsque la base de données Recherche est récupérée à partir d'une défaillance, toutes les transactions qui étaient terminées à la fin de la journée ouvrable précédente doivent être récupérées.

##### Caractéristiques prévues

Caractéristique	Valeur estimée
Taille de la base de données MarketDev	20 Go
Taille de la base de données Recherche	200 Mo
Débit de sauvegarde total	100 Mo par minute
Débit total de restauration	80 Mo par minute
Taux moyen de modification de la base de données MarketDev pendant les heures d'ouverture	1 Go par heure
Taux moyen de modification de la base de données Recherche pendant les heures d'ouverture	10 Mo par heure
Pourcentage de la base de données MarketDev modifié chaque jour (moyenne)	1.2%

(suite)

Caractéristique	Valeur estimée
Pourcentage de la base de données Recherche modifié chaque jour (moyenne)	80%
Heures d'ouverture (aucune sauvegarde complète de la base de données autorisée pendant ces heures)	8 h à 18 h

### **Exigences de continuité des bases de données d'entreprise sur l'instance de serveur CustomerService (pour l'exercice 3)**

#### **Objectifs de temps de récupération**

1. La base de données CreditControl ne doit jamais être indisponible pendant plus de deux heures.
2. La base de données PotentialIssue ne doit jamais être indisponible pendant plus d'une heure.

#### **Objectifs de point de restauration**

1. Lorsque la base de données de CreditControl est récupérée à partir d'une défaillance, il ne doit pas y avoir plus de cinq minutes de transactions de perdues.
2. Lorsque la base de données PotentialIssue est récupérée à partir d'une défaillance, il ne doit pas y avoir plus de 30 minutes de transactions de perdues.

#### **Caractéristiques prévues**

Caractéristique	Valeur estimée
Taille de la base de données CreditControl	20 Go
Taille de la base de données PotentialIssue (au début de chaque semaine, l'archivage étant terminé)	200 Mo
Débit de sauvegarde total	100 Mo par minute
Débit total de restauration	80 Mo par minute
Taux moyen de modification de la base de données CreditControl pendant les heures d'ouverture	500 Mo par heure
Taux moyen de modification de la base de données PotentialIssue (constant toute la semaine, 24 heures par jour)	10 Mo par heure
Pourcentage de la base de données CreditControl modifié chaque jour (moyenne)	60%
Pourcentage de la base de données PotentialIssue modifié chaque jour (moyenne)	50%
Heures d'ouverture (aucune activité de la base de données autorisée pendant ces heures)	8 h à 19 h

**Stratégie de sauvegarde existante pour la base de données CreditControl**Mode de récupération : **Complète**

Type de sauvegarde	Planification
• Complète	• Samedi à 6 h du matin, mercredi à 6 h du matin
• Différentielle	• Dimanche à 22 h, lundi à 22 h, mardi à 22 h, jeudi à 22 h, vendredi à 22 h
• Journalisation	• Toutes les 60 minutes à l'heure pile

**Stratégie de sauvegarde existante pour la base de données PotentialIssue**Mode de récupération : **Complète**

Type de sauvegarde	Planification
• Complète	• Dimanche à 22 h
• Journalisation	• Toutes les 15 minutes, début 10 minutes après l'heure pile

## Exercice 1 : Planifier une stratégie de sauvegarde

### Scénario

Vous devez planifier une stratégie de sauvegarde pour les deux bases de données de l'instance Proseware. On vous a fourni les RPO (objectifs de point de récupération) et les RTO (objectifs de temps de récupération) pour les deux bases de données, dans le cadre d'une décision de continuité de l'activité.

Les tâches principales de cet exercice sont les suivantes :

1. Étudier les besoins de l'activité.
2. Déterminer une stratégie de sauvegarde appropriée pour chaque base de données.

#### ► Tâche 1 : Étudier les besoins de l'activité

- Examiner les impératifs de l'activité fournis dans la documentation de l'exercice.

#### ► Tâche 2 : Déterminer une stratégie de sauvegarde appropriée pour chaque base de données

- Déterminez une stratégie de sauvegarde appropriée pour chaque base de données.
- Pour la base de données MarketDev :
  - Quel mode de récupération faut-il utiliser ?
  - Remplissez le tableau suivant pour la planification de la sauvegarde :

Type de sauvegarde	Planification
•	•
•	•
•	•

- Pour la base de données de recherche :
  - Quel mode de récupération faut-il utiliser ?
  - Remplissez le tableau suivant pour la planification de la sauvegarde :

Type de sauvegarde	Planification
•	•
•	•
•	•

**Résultats** : À la fin de cet exercice, vous devez avoir élaboré un plan de sauvegarde des deux bases de données.

## Exercice 2 : Configurer les modes de récupération

### Scénario

Vous avez examiné les modes de récupération de base de données et constaté que les modes de récupération de base de données en cours ne répondent pas aux besoins de disponibilité de l'entreprise. Dans cet exercice, vous devez définir les modes de récupération des bases de données qui ne répondent pas aux exigences.

La tâche principale de cet exercice est la suivante :

1. Examiner et ajuster les modes de récupération des bases de données.

#### ► Tâche 1 : Examiner et ajuster les modes de récupération des bases de données

- Reprenez les modes de récupération que vous avez choisis dans l'exercice 1 et vérifiez si les modes de récupération existants pour les bases de données MarketDev et Recherche correspondent à vos recommandations. Dans le cas contraire, modifiez les modes de récupération conformément à vos recommandations.

**Résultats :** À la fin de cet exercice, vous devez avoir examiné et modifié les modes de récupération de base de données conformément aux exigences.

## Exercice difficile 3 : Examiner les modes et les stratégies de récupération (si le temps le permet)

### Scénario

Une autre instance de SQL Server est installée pour prendre en charge les opérations du service client. Il se pourrait que les bases de données existantes sur l'instance de serveur ne soient pas configurées correctement et que leurs stratégies de sauvegarde ne soient pas valides d'après les exigences de RPO et de RTO. Dans cet exercice, vous devez examiner les modes de récupération de base de données et les stratégies de sauvegarde pour les bases de données de l'instance CustomerService et fournir des modifications recommandées.

Les tâches principales de cet exercice sont les suivantes :

1. Vérifier les éléments requis de RPO et de RTO pour les bases de données.
2. Examinez les modes de récupération et les stratégies de sauvegarde existants.
3. Indiquez si la stratégie sera réussie ou non.

► **Tâche 1 : Vérifier les éléments requis de RPO et de RTO pour les bases de données**

- La documentation associée comporte les détails de la continuité d'activité requise pour les bases de données. Vous devez examiner cette documentation.

► **Tâche 2 : Examiner les modes de récupération et les stratégies de sauvegarde existants**

- La documentation associée contient également des informations sur la sauvegarde des bases de données. Vous devez examiner cette documentation.

► **Tâche 3 : Indiquer si la stratégie sera réussie ou non**

- Vous devez évaluer si la stratégie de sauvegarde et la configuration actuelle du mode de récupération sont en mesure de répondre aux exigences de continuité de l'activité. Dans le cas contraire, expliquez pourquoi cela ne fonctionnera pas.

**Résultats** : À la fin de cet exercice, vous devez avoir évalué la stratégie.

## Contrôle des acquis du module et éléments à retenir

- Questions de contrôle des acquis
- Meilleures pratiques

### Questions de contrôle des acquis

1. Quelles sont les fonctionnalités uniques des restaurations de journal de transaction ?
2. Quand une stratégie de sauvegarde complète de base de données peut-elle être adéquate ?
3. Qu'est-ce qui peut empêcher la troncation du journal des transactions ?

### Meilleures pratiques

1. Planifiez votre stratégie de sauvegarde avec précaution.
2. Planifiez la stratégie de sauvegarde conjointement avec les besoins de l'entreprise.
3. Choisissez le mode de récupération de base de données le plus approprié.
4. Planifiez la taille des journaux de transaction en fonction de leur fréquence de sauvegarde.
5. Utilisez les sauvegardes différentielles pour accélérer la récupération.



# Module 6

## Sauvegarde des bases de données SQL Server 2012

### Sommaire :

Leçon 1 : Sauvegarde des bases de données et des journaux de transactions	6-3
Leçon 2 : Gestion des sauvegardes de base de données	6-15
Leçon 3 : Utilisation des options de sauvegarde	6-22
Atelier pratique 6 : Sauvegarde des bases de données SQL Server	6-28

## Vue d'ensemble du module

- Sauvegarde des bases de données et des journaux de transactions
- Gestion des sauvegardes de base de données
- Utilisation des options de sauvegarde

Garantir la fiabilité des sauvegardes des données d'entreprise est l'un des rôles les plus importants pour les administrateurs de base de données. Vous avez appris que Microsoft® SQL Server® fournit de nombreux types de sauvegardes. Dans ce module, vous allez explorer la plupart de ces types de sauvegarde de manière détaillée et apprendre à implémenter les sauvegardes.

Indépendamment de l'apprentissage à effectuer des sauvegardes complètes de base de données, des sauvegardes différentielles de base de données et des sauvegardes de journaux de transactions, vous verrez également comment appliquer des options qui affectent la façon dont les sauvegardes s'exécutent. L'automatisation et la planification des sauvegardes seront traités dans des modules ultérieurs.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- sauvegarder les bases de données et les journaux de transactions ;
- gérer les sauvegardes de base de données ;
- utiliser les options d'enregistrement avancées.

## Leçon 1

# Sauvegarde des bases de données et des journaux de transactions

- Exécution d'une sauvegarde complète de base de données
- Utilisation de jeux de sauvegarde
- Utilisation de la compression de sauvegarde
- Exécution des sauvegardes différentielles
- Exécution des sauvegardes des journaux des transactions
- Démonstration 1A : Sauvegarde des bases de données

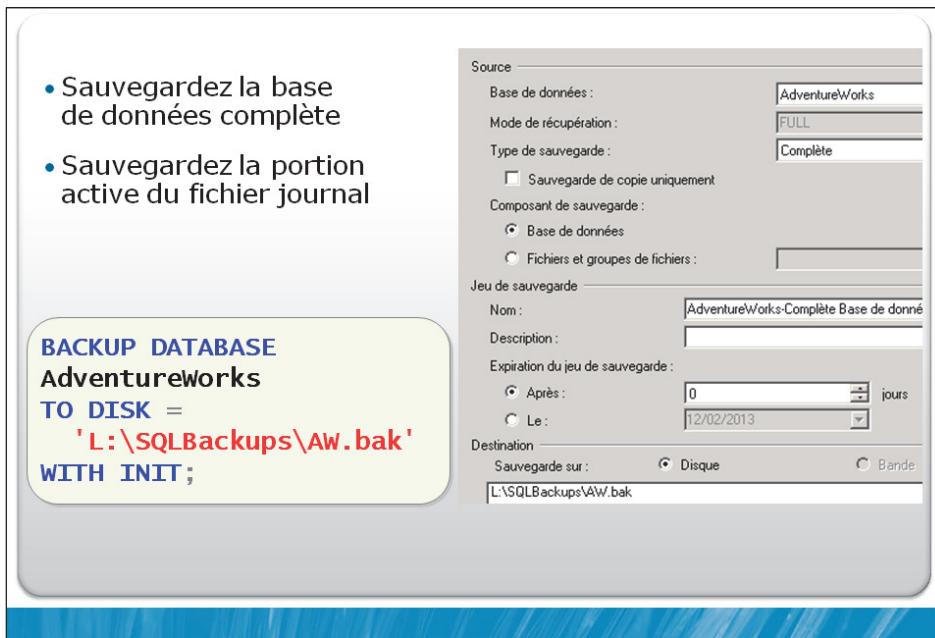
Dans le module précédent, vous venez de découvrir comment planifier une stratégie de sauvegarde pour un système SQL Server. Cette leçon montre comment effectuer les formes les plus courantes de sauvegarde SQL Server : sauvegardes complètes de base de données, sauvegardes différentielles de base de données et sauvegardes des journaux de transactions.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- exécuter une sauvegarde complète de base de données ;
- utiliser des jeux de sauvegarde ;
- utiliser la compression de sauvegarde ;
- exécuter des sauvegardes différentielles ;
- exécuter des sauvegardes de journaux de transactions.

## Exécution d'une sauvegarde complète de base de données



### Points clés

Les sauvegardes complètes de base de données peuvent être effectuées à l'aide de la commande `BACKUP DATABASE` dans T-SQL ou à l'aide de l'interface utilisateur graphique de SSMS. Une sauvegarde complète de base de données enregistre toutes les pages de données de la base de données, puis enregistre également la partie active du journal des transactions.

### Exemple de la diapositive

Dans l'exemple de la diapositive, une sauvegarde complète de base de données a lieu pour la base de données AdventureWorks, sur un fichier de disque L:\SQLBackups\AW.bak. L'option `INIT` qui est incluse dans la commande, demande à SQL Server de créer le fichier, s'il n'existe pas, ou de remplacer le fichier s'il existe déjà.

L'option d'initialisation par défaut, `NOINIT` indique à SQL Server de créer le fichier s'il n'existe pas, et de l'ajouter au fichier s'il contient déjà une sauvegarde SQL Server. La commande `BACKUP DATABASE` inclut de nombreuses autres options. Les options les plus importantes sont expliquées tout au long de ce module.

### Expiration de la sauvegarde

Un aspect important lorsque vous effectuez une sauvegarde consiste à inclure la durée associée au contenu de la sauvegarde. La base de données peut être utilisée lorsque la sauvegarde est effectuée.

Par exemple, si une sauvegarde démarre à 22 h 00 et se termine à 01 h 00, la sauvegarde fournira-t-elle une copie de la base de données telle qu'elle était à 22 h 00, une copie de la base de données telle qu'elle était à 01 h 00 du matin ou fournira-t-elle une copie de la base de données à une heure comprise entre le début et la fin ?

Dans les versions antérieures de SQL Server, le processus de sauvegarde écrivait des pages de données sur l'unité de sauvegarde de manière séquentielle. Toutefois, si un utilisateur avait besoin de modifier une page de données, SQL Server revenait au début de la file d'attente de sauvegarde de cette page de données, et incitait l'utilisateur à attendre que la page soit écrite sur l'unité de sauvegarde. Dans ces versions de SQL Server, la sauvegarde effectuée correspondait à une copie de la base de données au démarrage de la sauvegarde.

Pour minimiser l'impact sur les utilisateurs, les versions ultérieures de SQL Server écrivent toutes les pages de données sur l'unité de sauvegarde dans l'ordre, mais utilise le journal des transactions pour suivre toutes les pages qui sont modifiées au cours de la sauvegarde. SQL Server écrit la partie pertinente du journal des transactions à la fin de la sauvegarde. Ce processus rend les sauvegardes légèrement plus volumineuses que dans les versions antérieures, en particulier si d'importantes activités de mise à jour se produisent en même temps que la sauvegarde. Ce processus modifié signifie également que la sauvegarde contient une copie de la base de données à un instant précis avant l'achèvement de la sauvegarde, et pas au démarrage de celle-ci.

**Question :** que se passe-t-il quand vous ne spécifiez ni l'option INIT ni l'option NOINIT et qu'une sauvegarde existe déjà sur l'unité de sauvegarde ?

## Utilisation de jeux de sauvegarde

- Un jeu de sauvegarde représente une sauvegarde d'un type quelconque
- Les jeux de sauvegarde sont écrits sur des supports de sauvegarde
  - Comprend une ou plusieurs unités de sauvegarde sur bande ou sur disque
  - Les sauvegardes sont agrégées par bandes sur les unités
  - Les périphériques à bande et à disque ne peuvent pas être associés
- Les unités et les supports de sauvegarde sont créés lors de leur première utilisation
- Chaque unité de sauvegarde a un en-tête contenant des métadonnées des jeux de sauvegarde
- Les supports de sauvegarde peuvent être en miroir dans l'édition Enterprise

### Points clés

Les utilisateurs sont souvent perturbés par le fait qu'un seul fichier du système d'exploitation peut contenir plusieurs sauvegardes SQL Server. L'erreur la plus courante associée à cela consiste à restaurer la première sauvegarde à partir d'un fichier, en supposant qu'il s'agit de la dernière sauvegarde du fichier.

Une sauvegarde unique est appelée « jeu de sauvegarde ». Elle est écrite sur un support de sauvegarde, qui peut lui-même contenir jusqu'à 64 unités de sauvegarde. Une unité de sauvegarde peut être un disque ou un lecteur de bande. Les lecteurs de bande doivent être attachés localement et les sauvegardes écrites sur la bande peuvent être associées aux sauvegardes Windows.

 **Remarque** Les lecteurs de bande qui sont mappés sur un réseau ne peuvent pas être utilisés directement avec la sauvegarde SQL Server. De plus, la capacité de SQL Server à effectuer une sauvegarde directement sur bande est déconseillée et sera supprimée dans une future version de SQL Server.

Les unités de disques sont les plus couramment utilisées. Si un support de sauvegarde couvre plusieurs unités de sauvegarde, les sauvegardes sont agrégées par bandes dans les unités.

 **Remarque** Aucun périphérique de parité n'est utilisé lors de l'agrégation par bande. Si deux unités de sauvegarde sont utilisées conjointement, chacune reçoit la moitié de la sauvegarde. Les deux doivent également être présentes pour la tentative de restauration de la sauvegarde.

Chaque opération de sauvegarde sur un support de sauvegarde doit être écrite sur le même nombre et les mêmes types d'unités de sauvegarde. Enterprise Edition de SQL Server prend également en charge la mise en miroir des supports de sauvegarde pour améliorer la probabilité de restaurer la sauvegarde. La même image de sauvegarde est écrite à plusieurs emplacements.

Les supports de sauvegarde et les unités de sauvegarde sont créés lors de la première tentative de sauvegarde. Les supports de sauvegarde peuvent également être nommés au moment de la création.

Les sauvegardes créées sur une seule unité qui n'est pas mise en miroir ou sur un ensemble d'unités en miroir au sein d'un support de sauvegarde sont appelées une famille de supports. Le nombre d'unités de sauvegarde utilisées pour le support de sauvegarde détermine le nombre de familles de supports d'un support de sauvegarde. Si, par exemple, un support de sauvegarde utilise deux unités de sauvegarde qui ne sont pas mises en miroir, le support de sauvegarde contiendra deux familles de supports.

### Option FORMAT

SQL Server a été conçu pour réduire le risque de perte de données par inadvertance.

Par exemple, imaginez une sauvegarde complète de base de données qui a été écrite dans deux fichiers, à l'aide de la commande :

```
BACKUP DATABASE AdventureWorks  
TO DISK = 'D:\SQLBackups\AW_1.bak',  
        DISK = 'L:\SQLBackups\AW_2.bak'  
WITH INIT;
```

Les deux fichiers disque répertoriés composent un support de sauvegarde. Les données de la sauvegarde sont réparties entre les deux fichiers. Une autre sauvegarde peut être créée ultérieurement, à l'identique avec une commande telle que :

```
BACKUP DATABASE AdventureWorks  
TO DISK = 'D:\SQLBackups\AW_1.bak',  
        DISK = 'L:\SQLBackups\AW_2.bak'  
WITH NOINIT;
```

Les données de la deuxième sauvegarde sont à nouveau agrégées par bandes dans les deux fichiers et l'en-tête du support de sauvegarde est mis à jour pour indiquer qu'il contient maintenant les deux sauvegardes.

Toutefois, si l'utilisateur tente de créer une autre sauvegarde avec une commande telle que :

```
BACKUP DATABASE AdventureWorksDW  
TO DISK = 'D:\SQLBackups\AW_1.bak';
```

SQL Server retourne une erreur. Avant que le membre du support de sauvegarde puisse être remplacé, l'option FORMAT doit être ajoutée à la clause WITH dans la commande de sauvegarde :

```
BACKUP DATABASE AdventureWorksDW  
TO DISK = 'D:\SQLBackups\AW_1.bak'  
WITH FORMAT, INIT;
```

Utilisez avec précaution l'option FORMAT pour remplacer le contenu d'un fichier de sauvegarde et diviser le support de sauvegarde. Le formatage d'un fichier de sauvegarde d'un support de sauvegarde rend inutilisable la totalité du jeu de sauvegarde.

**Question** : quel avantage présente l'agrégation par bandes des sauvegardes sur plusieurs unités de sauvegarde d'un disque ?

## Utilisation de la compression de sauvegarde

### Compression de sauvegarde

- Présent à partir de SQL Server 2008
- Compresse la taille de sauvegarde sur l'unité
- Réduit les spécifications d'E/S, augmente l'utilisation de l'UC
- Sauvegardes plus rapides mais, surtout, restaurations plus rapides

### Restrictions

- Ne peut pas coexister sur un support avec des sauvegardes non compressées
- Ne peut pas coexister sur des bandes contenant des sauvegardes NT



### Points clés

Plusieurs technologies liées à la compression ont été ajoutées à SQL Server dans la version SQL Server 2008. La compression de la sauvegarde n'altère pas les performances de l'unité centrale, mais permet de réduire grandement la taille d'une sauvegarde et d'accroître les performances optimales de sauvegarde et de restauration. La compression de la sauvegarde peut être configurée en tant qu'option de serveur ou dans le cadre d'une commande BACKUP T-SQL comme suit :

```
BACKUP DATABASE AdventureWorksDW  
TO DISK = 'D:\SQLBackups\AW_1.bak'  
WITH FORMAT, COMPRESSION;
```

### Impact sur les performances des sauvegardes compressées

Une sauvegarde compressée étant plus petite qu'une sauvegarde non compressée de la même quantité de données, la compression d'une sauvegarde réduit la quantité requise d'E/S de périphérique et diminue considérablement la durée des sauvegardes.

Tout type de compression est susceptible d'améliorer l'utilisation de l'UC et les ressources processeur supplémentaires qui sont utilisées par le processus de compression, peuvent entraver les opérations simultanées sur les systèmes dont la capacité de l'unité centrale est restreinte. La plupart des systèmes SQL Server actuels sont axés sur les E/S, plutôt que sur la limite de l'UC. L'intérêt de la réduction E/S est généralement supérieure à l'augmentation des spécifications de l'UC d'un point de vue significatif.



**Remarque** Sur les systèmes où la charge de l'UC a son importance, il est également possible de créer une session de faible priorité pour créer des sauvegardes en limitant l'utilisation de l'UC grâce au gouverneur de ressources de SQL Server. L'utilisation du gouverneur de ressources est une rubrique avancée qui n'entre pas dans le cadre de ce cours.

## Temps de récupération

Bien qu'une réduction du temps nécessaire pour effectuer des sauvegardes est bénéfique, les sauvegardes sont généralement effectuées au cours de l'utilisation du système. Toutefois, la compression profite non seulement du processus de sauvegarde, mais également du processus de restauration, et peut considérablement améliorer la capacité de réponse aux exigences RTO.

## Pourcentages de compression

Le degré de compression qui est effectuée dépend entièrement du degré de compressibilité des données de la base de données. Certaines données se compressent parfaitement, d'autres données non. Une réduction des E/S et de la taille de sauvegarde de 30 à 50 % n'est pas rare dans les systèmes professionnels courants.

 **Remarque** Lorsque la compression de la sauvegarde peut être utilisée dans une base de données chiffrée à l'aide du chiffrement transparent des données, le taux de compression est minimal. Le chiffrement transparent des données est une rubrique avancée qui n'est pas traitée dans ce cours.

## Restrictions sur la compression des sauvegardes

Les restrictions suivantes s'appliquent aux sauvegardes compressées :

- Les sauvegardes compressées et non compressées ne peuvent pas co-exister sur un support de sauvegardes.
- Les versions antérieures de SQL Server ne peuvent pas lire les sauvegardes compressées, mais les éditions inférieures du produit peuvent restaurer les sauvegardes compressées, même si ces éditions ne peuvent pas créer des sauvegardes compressées.
- Les sauvegardes Windows ne peuvent pas partager un support de sauvegarde avec les sauvegardes compressées de SQL Server.
- Le paramètre par défaut de la compression de la sauvegarde peut être défini par l'option de configuration du serveur qui représente la « valeur par défaut de compression de la sauvegarde »
- SQL Server 2008 R2 a introduit la création de sauvegardes compressées dans Standard Edition de SQL Server.

**Question** : pourquoi les temps de sauvegarde et de restauration diminuent généralement lorsque la compression de la sauvegarde est utilisée ?

## Exécution des sauvegardes différentielles

- Sauvegardez les étendues modifiées depuis la dernière sauvegarde complète de la base de données
- Stockez la partie active du journal des transactions afin de pouvoir récupérer la base de données
- Indépendant d'autres sauvegardes différentielles

Source	AdventureWorks
Base de données :	FULL
Mode de récupération :	Différentielle
Type de sauvegarde :	Complexe
	Différentielle
Composant de sauvegarde :	Journal des transactions

Remarque : vous ne pouvez pas créer une sauvegarde différentielle de base de données si aucune sauvegarde complète n'a été créée jusqu'alors

```
BACKUP DATABASE Adventureworks  
TO DISK = 'L:\SQLBackups\AW_Diff.bak'  
WITH DIFFERENTIAL, INIT;
```

### Points clés

Alors que les sauvegardes complètes de base de données sont idéales, souvent vous ne disposez pas de suffisamment de temps pour effectuer des sauvegardes complètes de base de données. Dans le cas où un pourcentage relativement faible de la base de données est modifié, par rapport à la taille globale de celle-ci, les sauvegardes différentielles sont une bonne option à prendre en compte.

### Sauvegardes différentielles

Vous pouvez effectuer une sauvegarde différentielle à l'aide de SQL Server Management Studio ou en ajoutant l'option DIFFERENTIAL à la commande BACKUP DATABASE T-SQL.

SQL Server gère une carte des extensions modifiées appelées la page bitmap différentielle. Une page est conservée pour chaque section de 4 Go de chaque fichier de données. Chaque fois qu'une sauvegarde complète de la base de données est créée, SQL Server désactive la carte. Lorsque des données sont modifiées dans les fichiers, SQL Server met à jour cette carte. Une sauvegarde différentielle enregistre toutes les extensions qui ont été modifiées depuis la dernière sauvegarde complète de la base de données, et pas seulement celles modifiées depuis la dernière sauvegarde différentielle.

 **Remarque** Vous ne pouvez pas créer une sauvegarde différentielle sauf si une sauvegarde complète de la base de données a déjà été effectuée.

Une sauvegarde différentielle enregistre également la partie active du journal des transactions, de la même manière qu'une sauvegarde complète de la base de données le ferait.

La syntaxe de la sauvegarde différentielle est identique à la syntaxe des sauvegardes complètes de base de données, à la seule différence qu'il propose également l'option DIFFERENTIAL. Toutes les autres options disponibles pour les sauvegardes complètes de base de données sont également disponibles pour les sauvegardes différentielles.

**Question :** une sauvegarde différentielle tronque-t-elle le journal des transactions ?

## Exécution des sauvegardes des journaux des transactions

- Sauvegardez uniquement le journal des transactions
- Sauvegarde le journal à partir de la dernière sauvegarde réussie du fichier journal et jusqu'à sa fin actuelle
- Tronque les enregistrements de journal inactifs sauf si des options sont spécifiées

Source \_\_\_\_\_

Base de données :	AdventureWorks
Mode de récupération :	FULL
Type de sauvegarde :	Journal des transactions
<input type="checkbox"/> Sauvegarde de copie uniquement	Complète
Composant de sauvegarde :	Différentielle
	Journal des transactions

Remarque : la base de données doit être en mode de récupération complète ou journalisée en bloc

```
BACKUP LOG Adventureworks
TO DISK = 'L:\SQLBackups\AW_Log.bak'
WITH NOINIT;
```

### Points clés

Vous pouvez effectuer une sauvegarde du journal des transactions à l'aide de SQL Server Management Studio ou en utilisant l'instruction BACKUP LOG T-SQL. Avant qu'une sauvegarde du journal des transactions puisse être effectuée, la base de données doit être en mode de récupération complète ou journalisée en bloc. En outre, une sauvegarde du journal des transactions ne peut se produire que lorsqu'une sauvegarde complète de base de données a été effectuée au préalable.

Une sauvegarde du journal des transactions n'enregistre pas de pages de données à partir de la base de données, sauf si cette dernière est définie pour le mode de récupération de journalisation en bloc. Une sauvegarde du journal des transactions recherche le MaxLSN de la dernière sauvegarde des journaux de transactions réussie, puis enregistre toutes les entrées de journal au-delà de ce point jusqu'au MaxLSN actuel. Le journal des transactions est tronqué ensuite dans la mesure du possible. La transaction active la plus longue doit être conservée, au cas où la base de données devrait être récupérée après une défaillance.

## Chaînes d'enregistrements de journaux

Pour qu'une base de données puisse être restaurée à l'aide des sauvegardes des journaux de transactions, une chaîne ininterrompue d'enregistrements de journaux doit être disponible depuis la dernière sauvegarde complète de la base de données au point de restauration souhaité. Si la chaîne est interrompue, il est uniquement possible d'effectuer une restauration jusqu'au point où la chaîne de sauvegarde a été interrompue.

Par exemple, imaginez un scénario dans lequel une base de données est créée, puis ultérieurement, une sauvegarde complète de la base de données est effectuée. À ce stade, la base de données peut être récupérée. Si le mode de récupération de la base de données est ensuite modifié en mode simple, puis modifié ultérieurement en mode complet, une interruption dans la chaîne de fichier journal se produit. Bien qu'une sauvegarde complète de la base de données précédente s'est produite, la base de données peut uniquement être récupérée au point où la dernière sauvegarde du journal des transactions a été effectuée (le cas échéant) avant la modification du mode de récupération simple.

Après un basculement du mode de récupération simple au mode complet, une sauvegarde complète de base de données doit être effectuée pour créer un point de départ pour les sauvegardes de journaux de transactions.

## Troncation de journal

Lorsque l'action par défaut est de tronquer le journal des transactions à la suite d'une sauvegarde du journal des transactions, cette troncation n'est pas effectuée si l'option COPY\_ONLY est utilisée. L'option COPY\_ONLY est traitée dans la leçon 3 de ce module.

## Démonstration 1A : Sauvegarde des bases de données

Dans cette démonstration, vous allez apprendre à

- Sauvegarder une base de données
- Utiliser la compression de la sauvegarde

### Procédure de démonstration

1. Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn**. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**, accédez à **D:\22462A\_Labs\22462A\_06\_PRJ\22462A\_06\_PRJ.ssmssqln**, puis cliquez sur **Ouvrir**.
3. Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** dans l'Explorateur de solutions.
4. Ouvrez le fichier de script **11 – Demonstration 1A.sql**.
5. Suivez les instructions contenues dans les commentaires du fichier de script.

## Leçon 2

# Gestion des sauvegardes de base de données

- Options pour garantir l'intégrité de la sauvegarde
- Affichage des informations de sauvegarde
- Démonstration 2A : Affichage de l'historique des sauvegardes

Dans le module 5, plusieurs scénarios d'exemple ont été introduits dans lesquels les utilisateurs peuvent effectuer des sauvegardes, mais pour le moment les sauvegardes ont besoin d'être restaurées et la restauration demeure impossible. La première vérification à effectuer sur une sauvegarde consiste à vérifier qu'il est accessible en lecture. L'option de vérifier une sauvegarde et d'autres options pour s'assurer de l'intégrité de la sauvegarde, est développée dans cette leçon.

Il est également important de savoir comment rechercher des informations sur les sauvegardes qui ont été effectuées. SQL Server conserve l'historique des opérations de sauvegarde dans la base de données msdb. Dans cette leçon, vous allez apprendre à interroger les tables qui contiennent ces informations et apprendre également à extraire les informations d'en-tête des unités de sauvegarde.

## Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- décrire les options permettant de garantir l'intégrité de la sauvegarde ;
- afficher les données de sauvegarde.

## Options pour garantir l'intégrité de la sauvegarde

### Supports de sauvegarde mis en miroir

- Un jeu de sauvegarde peut être mis en miroir (jusqu'au nombre de 4)
- Les miroirs nécessitent le même nombre d'unités de sauvegarde
- Prise en charge dans l'édition Enterprise uniquement

### Option de sauvegarde CHECKSUM

- Disponible pour tous les types de sauvegarde
- Génère une somme de contrôle sur le flux de données de sauvegarde
- Peut être utilisé pour vérifier la sauvegarde

### Vérification des sauvegardes

- L'option RESTORE VERIFYONLY peut être utilisée pour la vérification des sauvegardes
- Utile une fois associée à l'option CHECKSUM

## Points clés

Une grande partie des efforts peuvent être déployés dans l'exécution de sauvegardes. Cet effort peut être entièrement perdu si les sauvegardes qui sont effectuées ne sont pas utilisables au moment de leur restauration. SQL Server comprend plusieurs options pour contrecarrer toute incapacité à restaurer des sauvegardes.

### Supports de sauvegarde mis en miroir

Un support de sauvegarde mis en miroir est une copie du support de sauvegarde éventuellement créée en parallèle lors de l'opération de sauvegarde, avec la version Enterprise Edition de SQL Server.

Un support de sauvegarde mis en miroir se compose de deux à quatre miroirs ; chaque miroir contient le support de sauvegarde entier. Chaque miroir doit être configuré avec le même nombre d'unités de sauvegarde et les unités de sauvegarde doivent être du même type de périphérique.

La mise en miroir d'un support de sauvegarde augmente la disponibilité en supposant qu'il est préférable d'avoir plusieurs copies d'une sauvegarde, plutôt qu'une seule copie. Mais il est important de savoir que la mise en miroir d'un support de sauvegarde expose votre système à un niveau élevé de risque de défaillance matérielle, comme une panne des unités de sauvegarde, ce qui provoquerait l'échec de la sauvegarde complète.

Un support de sauvegarde mis en miroir est créé à l'aide de l'option MIRROR TO comme le montre la commande suivante :

```
BACKUP DATABASE AdventureWorksDW
TO DISK = 'D:\SQLBackups\AW.bak'
MIRROR TO DISK = 'L:\SQLBackups\AW_M.bak'
WITH FORMAT, INIT;
```

### Option WITH CHECKSUM

SQL Server 2005 a introduit la possibilité d'effectuer une opération de somme de contrôle sur un flux de données entier. Cette option utilise légèrement plus de ressources processeur que les sauvegardes sans le calcul d'une somme de contrôle. L'option WITH CHECKSUM valide les informations de niveau page (somme de contrôle ou page endommagée si l'un ou l'autre est présent), ainsi que la somme de contrôle pour le flux de sauvegarde. La valeur de somme de contrôle est écrite à la fin de la sauvegarde et peut être activée au cours des opérations de restauration ou pendant des opérations de vérification de sauvegarde effectuées avec la commande RESTORE VERIFYONLY.

Une somme de contrôle de sauvegarde est activée à l'aide de l'option CHECKSUM comme le montre la commande suivante :

```
BACKUP DATABASE AdventureWorksDW  
TO DISK = 'D:\SQLBackups\AW.bak'  
WITH CHECKSUM;
```



**Remarque** L'option COMPRESSION active également l'option CHECKSUM automatiquement.

### Vérification des sauvegardes

Pour la vérification des sauvegardes, il existe la commande RESTORE VERIFYONLY qui permet de contrôler la validité des sauvegardes. Elle effectue les tests ci-dessous :

- achèvement du jeu de sauvegarde ;
- lisibilité de tous les volumes ;
- intégrité des identificateurs de page (au même niveau, comme si elle était sur le point d'écrire les données) ;
- validité de la somme de contrôle (si elle est présente sur le support).
- taille d'espace disponible sur les périphériques de destination.

La valeur de la somme de contrôle peut être validée si la sauvegarde a été effectuée avec l'option WITH CHECKSUM. Sans l'option CHECKSUM lors d'une sauvegarde, les options de vérification contrôlent uniquement les métadonnées et non les données de sauvegarde réelles.

La vérification peut également être effectuée via une option figurant dans la tâche de la base de données de sauvegarde dans SSMS, et dans les plans de maintenance de SQL Server.

 **Remarque** Envisagez de vérifier les sauvegardes sur un autre système que celui où la sauvegarde a été effectuée, afin d'éviter le cas où une sauvegarde est uniquement accessible en lecture seule sur le matériel source.

 **Remarque** Vérifiez que vous créez vos sauvegardes sur des disques différents de ceux sur lesquels résident vos fichiers de bases de données. Évitez de remplacer votre sauvegarde la plus récente.

**Question** : pouvez-vous garantir la possibilité de récupération d'une base de données si une sauvegarde de cette base de données peut être vérifiée ?

## Affichage des informations de sauvegarde

- SQL Server suit toutes les activités de sauvegarde d'un ensemble de tables dans la base de données msdb
  - L'historique est accessible via T-SQL ou SSMS
- Les informations peuvent être extraites des supports de sauvegarde
  - RESTORE LABELONLY retourne des informations sur les supports de sauvegarde se trouvant sur une unité de sauvegarde spécifiée
  - RESTORE HEADERONLY retourne toutes les informations d'en-tête de sauvegarde de tous les jeux de sauvegarde d'une unité de sauvegarde spécifique
  - RESTORE FILELISTONLY retourne une liste des fichiers de données et des fichiers journaux contenus dans un jeu de sauvegarde

### Points clés

SQL Server suit toutes les activités de sauvegarde d'un ensemble de tables dans la base de données msdb :

- backupfile
- backupfilegroup
- backupmediafamily
- backupmediaset
- backupset

Ces tables peuvent être interrogées pour récupérer des informations sur les sauvegardes qui ont été effectuées. Dans la démonstration 2A, vous allez apprendre à réaliser ces requêtes.

SSMS dispose également des options permettant de récupérer les détails des opérations de sauvegarde sur les bases de données et les journaux. Dans la démonstration 2A, vous verrez également un exemple de rapport qui est lancé de SSMS et qui affiche les données de sauvegarde pertinentes.

## Suppression de l'historique de sauvegarde

L'historique de sauvegarde peut être supprimé à l'aide de procédures stockées système. Examinez la commande suivante :

```
EXEC sp_delete_backuphistory @oldest_date = '20090101';
```

Cette commande supprime tout l'historique antérieur à la date spécifiée. La date spécifiée correspond à la date la plus ancienne à conserver.

Examinez également la commande suivante :

```
EXEC sp_delete_database_backuphistory @database_name = 'Market';
```

Cette commande supprime l'historique d'une base de données nommée Market.

Si une base de données est restaurée sur un autre serveur, les informations de sauvegarde ne sont pas restaurées avec la base de données, car elles sont conservées dans la base de données msdb du système d'origine.

## Récupération des métadonnées de sauvegarde

Les informations sur un support de sauvegarde donné sont disponibles en exécutant la commande RESTORE avec les options suivantes :

Commande	Description
RESTORE LABELONLY	Retourne des informations sur les supports de sauvegarde sur une unité de sauvegarde spécifiée.
RESTORE HEADERONLY	Retourne toutes les informations d'en-tête de sauvegarde pour tous les jeux de sauvegarde sur une unité de sauvegarde particulière.
RESTORE FILELISTONLY	Retourne une liste des fichiers de données et des fichiers journaux contenus dans un jeu de sauvegarde.

**Question** : dans quels cas SQL Server n'obtient-il pas d'informations complètes sur les sauvegardes d'une base de données stockées dans msdb ?

## Démonstration 2A : Affichage de l'historique des sauvegardes

Dans cette démonstration, vous allez apprendre à

- Afficher l'historique de sauvegarde à l'aide de SSMS
- Interroger les tables d'historique de sauvegarde via T-SQL
- Utiliser la commande RESTORE HEADERONLY pour récupérer les informations de jeu de sauvegarde

### Procédure de démonstration

1. Si la démonstration 1A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn.** Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**, accédez à **D:\22462A\_Labs\22462A\_06\_PRJ\22462A\_06\_PRJ.ssmssln**, puis cliquez sur **Ouvrir**.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** dans l'Explorateur de solutions.
2. Ouvrez le fichier de script **21 – Demonstration 2A.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

## Leçon 3

# Utilisation des options de sauvegarde

- Éléments à prendre en compte en matière de sauvegarde
- Sauvegardes de copie uniquement
- Sauvegardes de la fin du journal
- Démonstration 3A : Sauvegarde de la fin du journal

Maintenant que vous venez de découvrir les options les plus courantes associées à la sauvegarde des bases de données et des journaux de transactions, il est important de tenir compte des considérations générales qui entourent la création de sauvegardes, ainsi que des options liées aux sauvegardes les moins courantes, mais utiles.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- prendre en compte les considérations relatives à la sauvegarde ;
- effectuer des sauvegardes de copie uniquement ;
- exécuter une sauvegarde de la fin du journal.

## Éléments à prendre en compte en matière de sauvegarde

- Les sauvegardes sont effectuées en ligne
  - N'empêchent pas l'accès utilisateur
  - Peuvent ralentir les autres opérations en raison de la charge d'E/S
- La base de données doit être en ligne pour les opérations de sauvegarde normale
  - Les sauvegardes des journaux de transactions sont toujours possibles sur une base de données endommagée
  - Le fichier journal doit être intact
- Intégration aux options du système d'exploitation
  - Le service SQL Writer fournit une fonctionnalité de sauvegarde via l'infrastructure du service de cliché instantané des volumes (VSS)
  - L'interface VDI permet aux éditeurs de logiciels indépendants (ISV) d'intégrer la fonctionnalité de sauvegarde et de restauration à leurs produits (couramment utilisée pour les outils de sauvegarde tiers)

### Points clés

Les sauvegardes SQL Server peuvent être créées lorsque d'autres utilisateurs utilisent le système. D'autres utilisateurs peuvent, toutefois, être concernés par la charge d'E/S placée sur le système par l'opération de sauvegarde. SQL Server place également certaines limitations sur les types de commandes qui peuvent être exécutées lorsqu'une sauvegarde est effectuée. Par exemple, la commande ALTER DATABASE ne peut pas être utilisée avec la commande ADD FILE ou REMOVE FILE, et réduire une base de données n'est pas autorisé pendant une sauvegarde.

La Commande BACKUP ne peut pas être incluse dans une transaction explicite ou implicite. Vous ne pouvez pas restaurer une sauvegarde.

Les bases de données peuvent être sauvegardées lorsqu'elles sont en ligne, mais il est toujours possible d'effectuer une sauvegarde du journal des transactions lorsque la base de données est endommagée, en supposant que le fichier journal lui-même est toujours intact. C'est la raison principale pour laquelle il est important de séparer les fichiers de données et les fichiers journaux sur un support physique distinct.

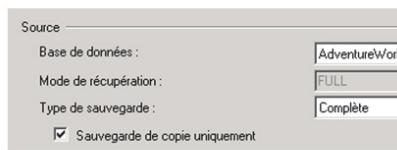
### VSS et VDI

Le service VSS (Volume Shadow Copy Service) de Windows et l'interface de programmation VDI (Virtual Device Interface) peuvent être utilisés avec SQL Server. L'utilisation principale de ces interfaces fait en sorte que les outils de sauvegarde tiers peuvent fonctionner avec SQL Server.

Sur les très grands systèmes, il est courant de créer une image de disque alors que le système est en cours de fonctionnement, ce qui explique que les sauvegardes SQL Server standard peuvent prendre du temps à s'exécuter. L'interface de programmation VDI permet à une application de figer les opérations de SQL Server momentanément lorsqu'une capture instantanée cohérente des fichiers de bases de données est créée. Cette forme de capture instantanée est souvent utilisée sur les systèmes de réplication SAN géographiquement distribués.

## Sauvegardes de copie uniquement

- Sauvegardez la base de données sans modifier l'ordre de la restauration
- Les sauvegardes des journaux de transactions de type copie uniquement ne tronquent pas le journal
- Les sauvegardes complètes de base de données de type copie uniquement n'affectent pas la base différentielle



Remarque : utilisez COPY\_ONLY pour les sauvegardes hors séquence

```
BACKUP DATABASE Adventureworks
TO DISK = 'L:\SQLBackups\Aw_Copy.bak'
WITH COPY_ONLY, INIT;
```

### Points clés

Une sauvegarde de copie uniquement est une sauvegarde SQL Server qui est indépendante de la séquence des sauvegardes SQL Server classiques. Normalement, une sauvegarde modifie la base de données et affecte la restauration des sauvegardes ultérieures.

Cependant, il peut être utile d'effectuer une sauvegarde à une fin précise sans affecter les procédures globales de sauvegarde et de restauration de la base de données.

Les sauvegardes de copie uniquement peuvent être effectuées soit dans la base de données, soit dans le journal des transactions. La restauration d'une sauvegarde complète de type copie uniquement est identique à la restauration de toute sauvegarde complète.

**Question :** pouvez-vous suggérer un scénario dans lequel vous pourriez utiliser une sauvegarde de copie uniquement ?

## Sauvegardes de la fin du journal

- Utilisés pour capturer la fin du journal avant de démarrer une séquence de restauration
  - Effectue une sauvegarde de journal normale
- Options
  - NORECOVERY lorsque des opérations de restauration suivront (base de données définie dans l'état RECOVERING)
  - CONTINUE\_AFTER\_ERROR lorsque des fichiers de données sont manquants ou endommagés mais que les fichiers journaux sont intacts

### Points clés

Dans SQL Server 2005 et les versions ultérieures, vous devez effectuer une sauvegarde de la fin du journal avant de commencer à effectuer la restauration sur une base de données existante. Cette configuration requise permet d'éviter un scénario classique de perte de données d'une base de données qui n'a pas été entièrement sauvegardée avant qu'une restauration de base de données ne se produise. La configuration vérifie que, par défaut, toutes les transactions doivent avoir fait l'objet d'au-moins une sauvegarde, avant de pouvoir être remplacées. Une sauvegarde de la fin du journal empêche la perte de données et préserve la continuité de la séquence de journaux de transactions consécutifs.

### WITH NORECOVERY

L'option WITH NORECOVERY est généralement appliquée pour restaurer les opérations.

Elle est détaillée dans le module 7. Toutefois, les utilisateurs sont souvent surpris de constater que la commande permettant de créer une sauvegarde de la fin du journal contient également une option WITH NORECOVERY. Cette option sauvegarde le journal des transactions, ensuite modifie immédiatement la base de données à un état de récupération. La raison la plus courante d'utiliser cette option est liée à l'envoi des journaux, lorsqu'un serveur modifie les rôles d'un serveur principal vers un serveur secondaire.

 **Remarque** l'envoi des journaux est une option avancée qui n'est pas traitée dans ce cours.

## Sauvegardes de la fin du journal

Si vous récupérez une base de données jusqu'au point de défaillance, la sauvegarde de la fin du journal est souvent la dernière sauvegarde pertinente du plan de récupération. Il s'agit d'un type standard de sauvegarde des journaux de transactions. Si vous ne pouvez pas effectuer la sauvegarde de la fin du journal, vous pouvez uniquement récupérer une base de données jusqu'à la fin de la dernière sauvegarde ayant été créée avant la défaillance.

Tous les scénarios de restauration ne nécessitent pas une sauvegarde de la fin du journal. Vous n'avez pas besoin de disposer d'une sauvegarde de la fin du journal si le point de récupération est contenu dans une sauvegarde de journal antérieure ou si vous déplacez ou remplacez (par écrasement) la base de données et ne souhaitez pas la restaurer à un point donné après la sauvegarde la plus récente.

Utilisez l'option CONTINUE\_AFTER\_ERROR uniquement si vous sauvegardez la fin d'une base de données endommagée.

Si vous ne pouvez pas sauvegarder la fin du journal à l'aide de l'option NO\_TRUNCATE lorsque la base de données est endommagée, vous pouvez essayer d'effectuer une sauvegarde de la fin du journal en spécifiant CONTINUE\_AFTER\_ERROR au lieu de NO\_TRUNCATE.

L'option NO\_TRUNCATE est disponible et est équivalente à l'utilisation conjointe des options COPY\_ONLY et CONTINUE\_AFTER\_ERROR. Elle doit être utilisée pour les bases de données endommagées. Elle fait en sorte que le moteur de base de données tente d'effectuer une sauvegarde, quel que soit l'état de la base de données. Cela signifie qu'une sauvegarde, effectuée lors de l'utilisation de l'option NO\_TRUNCATE, peut comporter des métadonnées incomplètes. Sans l'option de base de données NO\_TRUNCATE, la base de données doit être en ligne.

**Question :** quel est le plus grand avantage à pouvoir effectuer des sauvegardes de la fin du journal même si les fichiers de données sont endommagés ?

## Démonstration 3A : Sauvegarde de la fin du journal

Dans cette démonstration, vous allez voir comment sauvegarder la fin du journal d'une base de données endommagée

### Procédure de démonstration

1. Si la démonstration 1A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn.** Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**, accédez à **D:\22462A\_Labs\22462A\_06\_PRJ\22462A\_06\_PRJ.ssmssln**, puis cliquez sur **Ouvrir**.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** dans l'Explorateur de solutions.
2. Ouvrez le fichier de script **31 – Demonstration 3A.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

## Atelier pratique 6 : Sauvegarde des bases de données SQL Server

- Exercice 1 : Déterminer la compression de sauvegarde
- Exercice 2 : Sauvegarde des journaux des transactions
- Exercice 3 : Sauvegarde différentielle
- Exercice 4 : Sauvegarde de copie uniquement
- Exercice difficile 5 : Sauvegarde partielle (seulement si le temps le permet)

Informations d'ouverture de session

Ordinateur virtuel	22462A-MIA-SQL1
Nom d'utilisateur	AdventureWorks\Administrateur
Mot de passe	Pa\$\$w0rd

**Durée approximative : 45 minutes**

### Configuration de l'atelier pratique

Pour cet atelier pratique, vous utiliserez l'environnement d'ordinateur virtuel disponible.

Avant de commencer cet atelier pratique, vous devez procéder aux étapes suivantes :

1. Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
3. Dans la fenêtre Se connecter au serveur, tapez **Proseware** dans la zone de texte **Nom du serveur**.
4. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn**.
5. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**.
6. Dans la fenêtre Ouvrir un projet, ouvrez le projet **D:\22462A\_Labs\22462A\_06\_PRJ\22462A\_06\_PRJ.ssmssln**.
7. Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Dans l'Explorateur de solutions, double-cliquez sur le fichier de requête **00 - Setup.sql**. Lorsque la fenêtre de requête s'affiche, cliquez sur **Exécuter** dans la barre d'outils.

### Scénario de l'atelier pratique

Vous avez examiné et mis à jour les modes de récupération. En tant qu'administrateur de base de données, vous devez implémenter une stratégie de sauvegarde de base de données. Vous disposez des détails de la stratégie de sauvegarde requise pour plusieurs bases de données sur une instance de SQL Server. Vous devez effectuer les sauvegardes nécessaires.

## Exercice 1 : Déterminer la compression de sauvegarde

### Scénario

La taille des sauvegardes de base de données a augmenté avec le nombre de commandes passées. Vous souhaitez analyser la quantité d'espace qui peut être stockée à l'aide de la compression de la sauvegarde. Dans cet exercice, vous devez étudier l'efficacité de la compression de sauvegarde en sauvegardant la base de données MarketDev sur l'instance de Proseware. Vous allez effectuer une sauvegarde complète de base de données avec la compression de sauvegarde désactivée. Vous allez effectuer une autre sauvegarde complète de la base de données avec la compression de sauvegarde activée. Vous allez comparer les fichiers de sauvegarde pour identifier les économies d'espace possibles.

Les tâches principales de cet exercice sont les suivantes :

1. Créer une sauvegarde de base de données sans compression.
2. Créer une sauvegarde de base de données avec compression.
3. Comparer les tailles de fichier créées.

► **Tâche 1 : Créer une sauvegarde de base de données sans compression**

- À l'aide de l'Explorateur Windows, créez un nouveau dossier L:\SQLBackups.
- Effectuez une sauvegarde complète de la base de données MarketDev avec la fonction de compression désactivée dans le fichier L:\SQLBackups\MarketDev\_Full\_Uncompressed.BAK.

► **Tâche 2 : Créer une sauvegarde de base de données avec compression**

- Effectuez une sauvegarde complète de la base de données MarketDev avec la fonction de compression activée dans le fichier L:\SQLBackups\MarketDev\_Full\_Compressed.BAK.

► **Tâche 3 : Comparer les tailles de fichier créées**

- Calculez l'économie d'espace disponible par compression comme suit :

```
SpaceSavings=(Uncompressed size-Compressed size)*100/Uncompressed size
```

**Résultats :** À la fin de cet exercice, vous êtes en mesure de calculer l'espace économisé à l'aide de la compression de la sauvegarde de la base de données MarketDev.

## Exercice 2 : Sauvegarde des journaux des transactions

### Scénario

Une partie de la gestion actuelle de la base de données MarketDev repose sur une série de sauvegardes des journaux de transactions pour permettre la récupération limitée dans le temps. Dans cet exercice, vous devez sauvegarder le journal des transactions.

Les tâches principales de cet exercice sont les suivantes :

1. Exécuter un script pour introduire la charge de travail dans la base de données MarketDev.
2. Sauvegarder le journal des transactions dans la base de données MarketDev.

► **Tâche 1 : Exécuter un script pour introduire la charge de travail dans la base de données MarketDev**

- Ouvrez et exécutez le fichier de script 61 - Workload File.sql dans l'Explorateur de solutions.

► **Tâche 2 : Sauvegarder le journal des transactions dans la base de données MarketDev**

- Sauvegardez le journal des transactions de la base de données MarketDev dans le fichier L:\SQLBackups\MarketDev\_Log\_Compressed.BAK. Utilisez la fonction de compression de la sauvegarde pendant l'opération de sauvegarde.

**Résultats** : À la fin de cet exercice, vous êtes en mesure d'effectuer une sauvegarde de journaux de transactions.

## Exercice 3 : Sauvegarde différentielle

### Scénario

Il existe un problème lié aux volumes de données dans la base de données MarketDev qui seront trop importants au point que les sauvegardes complètes quotidiennes ne seront pas possibles. Dans cet exercice, vous devez effectuer une sauvegarde différentielle pour gérer la taille des sauvegardes de base de données.

Les tâches principales de cet exercice sont les suivantes :

1. Exécuter un script pour introduire la charge de travail dans la base de données MarketDev.
2. Créer une sauvegarde différentielle de la base de données MarketDev.

► **Tâche 1 : Exécuter un script pour introduire la charge de travail dans la base de données MarketDev**

- Ouvrez et exécutez le fichier de script 61 - Workload File.sql dans l'Explorateur de solutions.

► **Tâche 2 : Créer une sauvegarde différentielle de la base de données MarketDev**

- Créez une sauvegarde différentielle de la base de données MarketDev dans le fichier L:\SQLBackups\MarketDev\_Differential\_Compressed.BAK. Utilisez la fonction de compression de la sauvegarde pendant l'opération de sauvegarde.
- À l'aide de l'Explorateur Windows, notez la taille de la sauvegarde différentielle comparée à la sauvegarde complète.

► **Tâche 3 : Exécuter un script pour introduire la charge de travail dans la base de données MarketDev**

- Ouvrez et exécutez le fichier de script 61 - Workload File.sql dans l'Explorateur de solutions.

► **Tâche 4 : Ajouter une sauvegarde différentielle dans le fichier de sauvegarde différentielle précédent**

- Ajoutez une sauvegarde différentielle de la base de données MarketDev dans le fichier L:\SQLBackups\MarketDev\_Differential\_Compressed.BAK.
- À l'aide de l'Explorateur Windows, notez que la taille de la sauvegarde différentielle a augmenté. Le fichier contient à présent deux sauvegardes.

**Résultats** : À la fin de cet exercice, vous devez avoir effectué deux sauvegardes différentielles.

## Exercice 4 : Sauvegarde de copie uniquement

### Scénario

Une équipe a régulièrement besoin d'une copie temporaire de la base de données MarketDev. Il est important que les sauvegardes ne gênent pas la stratégie de sauvegarde qui est utilisée. Dans cet exercice, vous devez effectuer une sauvegarde de copie uniquement et vérifier la sauvegarde.

La tâche principale de cet exercice est la suivante :

1. Créer une sauvegarde de copie uniquement de la base de données MarketDev, en optant pour la vérification de la sauvegarde.

### ► Tâche 1 : Créer une sauvegarde de copie uniquement de la base de données MarketDev, en optant pour la vérification de la sauvegarde

- Créez une sauvegarde de copie uniquement de la base de données MarketDev dans le fichier L:\SQLBackups\MarketDev\_Copy\_Compressed.BAK. Veillez à :
  - vérifier la sauvegarde lors de sa création ;
  - utiliser la compression de sauvegarde ;
  - choisir de créer un nouveau support de sauvegarde nommé MarketDev Copy Backup ;
  - pour la description du support de sauvegarde, utiliser le support MarketDev Copy Backup pour l'équipe d'intégration.

**Résultats** : À la fin de cet exercice, vous devez avoir effectué une sauvegarde de copie uniquement.

## Exercice difficile 5 : Sauvegarde partielle (seulement si le temps le permet)

### Scénario

Sur l'instance de Proseware, il existe une base de données appelée RateTracking qui comporte deux groupes de fichiers. Le groupe de fichiers ARCHIVE est défini en lecture seule et le groupe de fichiers par défaut USERDATA ainsi que le groupe de fichiers PRIMARY sont en mode lecture-écriture. Dans cet exercice, vous devez sauvegarder les groupes de fichiers en lecture-écriture, en utilisant les commandes T-SQL.

La tâche principale de cet exercice est la suivante :

1. Effectuer une sauvegarde des groupes de fichiers en lecture-écriture sur la base de données RateTracking.

► **Tâche 1 : Effectuer une sauvegarde des groupes de fichiers en lecture-écriture sur la base de données RateTracking**

- Effectuez une sauvegarde des groupes de fichiers en lecture-écriture (USERDATA et PRIMARY) sur la base de données RateTracking. Entrez la sauvegarde dans le fichier L:\SQLBackups\RateTracking\_ReadWrite.BAK. Utilisez les options CHECKSUM et INIT.

**Résultats** : À la fin de cet exercice, vous devez avoir effectué une sauvegarde partielle.

## Contrôle des acquis et éléments à retenir

- Questions de contrôle des acquis
- Meilleures pratiques

### Questions de contrôle des acquis

1. Quels types de sauvegarde peuvent être effectués en mode de récupération simple ?
2. Comment les informations de sauvegarde peuvent-elles être lues ?

### Meilleures pratiques

1. Envisagez d'utiliser CHECKSUM pour créer une somme de contrôle sur vos fichiers de sauvegarde.
2. Utilisez la compression de la sauvegarde pour augmenter les performances de sauvegarde et de restauration, ainsi que l'espace de stockage sécurisé.
3. Envisagez de mettre en miroir vos sauvegardes pour renforcer la sécurité.
4. Vérifiez si la sauvegarde différentielle peut accélérer votre processus de restauration en mode de récupération complète.
5. Utilisez COPY\_ONLY pour les sauvegardes hors séquence.

# Module 7

## Restauration des bases de données SQL Server 2012

### Sommaire :

Leçon 1 : Présentation du processus de restauration	7-4
Leçon 2 : Restauration des bases de données	7-10
Leçon 3 : Utilisation de la récupération limitée dans le temps	7-21
Leçon 4 : Restauration des bases de données système et des fichiers individuels	7-29
Atelier pratique 7 : Restauration des bases de données SQL Server 2012	7-37

## Vue d'ensemble du module

- Présentation du processus de restauration
- Restauration des bases de données
- Utilisation de la récupération limitée dans le temps
- Restauration des bases de données système et des fichiers individuels

Dans le module précédent, vous avez découvert comment créer des sauvegardes des bases de données Microsoft® SQL Server® 2012. Une stratégie de sauvegarde peut impliquer différents types de sauvegarde. Cela signifie qu'il est important que vous compreniez le processus requis lors de la restauration de bases de données. Souvent, lorsque des restaurations de bases de données s'imposent, cela indique l'urgence d'une situation. Malheureusement, les administrateurs de base de données font souvent des erreurs de jugement une fois placés dans ces situations de récupération d'urgence. La première règle à respecter dans une situation défavorable est toujours de « ne pas causer de dégâts supplémentaires ». Dans les situations urgentes, il est plus important que jamais d'avoir un plan clairement établi sur la manière de procéder. Une bonne compréhension du processus requis et un plan approprié peuvent vous permettre d'éviter d'aggraver la situation.

Les restaurations de bases de données ne sont pas toutes liées à des défaillances du système. Dans la plupart des cas de défaillance du système, il est nécessaire de rétablir le système à un état le plus proche possible de celui précédent la défaillance. Certaines défaillances sont liées à des erreurs humaines. Dans ces cas, vous pouvez choisir de récupérer le système à un point antérieur à la défaillance. Les fonctionnalités de récupération limitée dans le temps de SQL Server 2012 peuvent vous y aider.

La probabilité que les bases de données utilisateur soient plus affectées que les bases de données système dans le cas d'une défaillance du système est plus grande car les bases de données utilisateur sont généralement beaucoup plus volumineuses que les bases de données système. Toutefois, les bases de données système peuvent elles aussi être affectées par des défaillances, et il convient de faire preuve d'un soin particulier lors de leur récupération. Vous devez notamment comprendre la façon dont chaque base de données système doit être récupérée car le processus de récupération n'est pas identique pour toutes.

## Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- comprendre le processus de restauration ;
- restaurer des bases de données ;
- utiliser la récupération limitée dans le temps ;
- restaurer des bases de données système et des fichiers individuels.

## Leçon 1

# Présentation du processus de restauration

- Types de restauration
- Préparation en vue de la restauration des sauvegardes
- Discussion : Détermination des sauvegardes requises pour la restauration

Vous avez découvert qu'il est possible de créer de nombreux types de sauvegarde avec SQL Server 2012. De même, différents types de processus de restauration peuvent s'avérer nécessaires.

Précédemment, nous vous avons indiqué que lorsque la récupération d'une base de données s'impose, il nécessaire d'adopter une procédure appropriée afin d'éviter tout dommage supplémentaire. Une fois que l'étape préliminaire de tentative de création d'une sauvegarde de la fin du journal a été effectuée, la décision la plus importante qui doit être prise consiste à déterminer quelles sauvegardes de base de données doivent être restaurées et dans quel ordre.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- décrire les différents types de restauration ;
- déterminer quelles sauvegardes restaurer et dans quel ordre.

## Types de restauration

### Types de restauration

- Restauration complète de la base de données avec le mode de récupération simple
- Restauration complète de la base de données en mode de récupération complète
- Restauration des bases de données système
- Restauration des fichiers endommagés uniquement
- Options de restauration avancées, notamment la restauration en ligne, fragmentaire et de pages

### Points clés

Restaurer une base de données dans SQL Server 2012 est un processus en deux étapes. La première étape concerne la restauration des pages de données à partir d'une ou de plusieurs sauvegardes. Une fois que les pages de données ont été restaurées, il se peut que la base de données soit dans un état incohérent. Pour remédier à cette situation, lors de la deuxième étape du processus de restauration, les détails disponibles du journal des transactions sont utilisés pour récupérer la base de données. La disponibilité des scénarios de restauration pour la base de données dépend du modèle de récupération de la base de données et de l'édition de SQL Server.

### **Restauration complète de la base de données avec le mode de récupération simple**

La stratégie de restauration la plus basique pour les bases de données SQL Server consiste à restaurer et récupérer une sauvegarde complète de base de données. Si une sauvegarde différentielle de la base de données est disponible, il est possible de restaurer la dernière après la restauration de la sauvegarde complète de la base de données mais avant le processus de récupération de la base de données.

Dans la plupart des cas faisant appel au mode de récupération simple, aucune sauvegarde différentielle n'est exécutée, auquel cas seule la dernière sauvegarde complète de base de données est restaurée, et la base de données est rétablie dans l'état dans lequel elle se trouvait juste avant la fin de sa sauvegarde complète.

## Restauration complète de la base de données en mode de récupération complète

La stratégie de restauration la plus courante nécessite un mode de récupération complète ou journalisée en bloc et implique une restauration complète, différentielle (le cas échéant), ainsi que des sauvegardes de fichiers journaux.

Pour effectuer une restauration vers la dernière limite dans le temps disponible :

1. Essayez d'exécuter une sauvegarde de la fin du journal. La possibilité d'exécution de cette étape dépend du type de défaillance qui s'est produit.
2. Restaurez la dernière sauvegarde complète de la base de données.
3. Restaurez la sauvegarde différentielle la plus récente si une sauvegarde différentielle a été créée.
4. Restaurez toutes les sauvegardes des journaux de transactions effectuées après le point de la sauvegarde différentielle la plus récente (ou la sauvegarde complète si aucune sauvegarde différentielle n'a été créée) en respectant l'ordre dans lequel elles ont été créées. Si une sauvegarde de la fin du journal a été créée avec succès, elle est restaurée en tant que sauvegarde du journal des transactions la plus récente.

Théoriquement, l'objectif de la restauration est de récupérer la base de données vers la dernière limite dans le temps possible. Toutefois, il existe des options permettant de la restaurer vers des limites antérieures dans le temps. Ces options sont présentées dans la leçon 3 de ce module.



**Remarque** Même si des sauvegardes différentielles et des sauvegardes des journaux de transactions ont été créées, vous pouvez choisir de ne pas les appliquer si vous souhaitez rétablir l'état de la base de données vers une limite antérieure dans le temps.

## Restauration des bases de données système

La restauration des bases de données système est possible mais nécessite des processus spéciaux pour éviter que d'autres problèmes se produisent. Par exemple, si une base de données MASTER reste dans un état cohérent, SQL Server refuse de démarrer tant que cette base de données n'est pas récupérée. La récupération des bases de données système est présentée dans la leçon 4 de ce module.

## Restauration des fichiers

Si des fichiers individuels d'une base de données ont été endommagés ou perdus, la possibilité de les restaurer présente l'intérêt de réduire considérablement le temps global de récupération de la base de données. La récupération des fichiers individuels n'est prise en charge que pour les fichiers en lecture seule si vous utilisez le mode de récupération simple, mais elle est possible pour les fichiers en lecture-écriture lorsque vous utilisez une récupération journalisée en bloc ou complète. La récupération des fichiers individuels utilise un processus qui est semblable au processus de restauration complète de base de données et sera abordé à la leçon 4 de ce module.

## Restauration en ligne

La restauration en ligne implique la restauration des données pendant que la base de données est en ligne. Il s'agit de l'option par défaut pour les restaurations de fichiers, de pages ou fragmentaire. Dans SQL Server 2012, la restauration en ligne est disponible uniquement dans l'édition Enterprise.

## Restauration fragmentaire

Une restauration fragmentaire permet de restaurer et récupérer la base de données par étapes, selon des groupes de fichiers, au lieu de restaurer l'intégralité de la base de données en une seule fois. Le premier groupe de fichiers qui doit être restauré est le groupe de fichiers primaire. Dans SQL Server 2012, la restauration fragmentaire est disponible uniquement dans l'édition Enterprise.

## Restauration de pages

Il existe une autre option avancée qui permet de restaurer une page de données individuelles. Si une page de données individuelles est endommagée, les utilisateurs verront généralement une erreur 823 ou 824 lorsqu'ils exécutent une requête qui tente d'accéder à cette page. Dans ce cas, il est possible d'utiliser une restauration de pages en ligne pour tenter de la récupérer. Une fois que la restauration a débuté, si une requête d'utilisateur essaie d'accéder à la page, l'utilisateur voit s'afficher l'erreur 829, qui indique que la page est en cours de restauration (« page is restoring »). Si la restauration de pages réussit, les requêtes utilisateur qui accèdent à la page retournent à nouveau les résultats attendus. Les restaurations de pages sont prises en charge dans les modes de récupération complète et journalisée en bloc, pas dans le mode de récupération simple. Dans SQL Server 2012, la restauration de pages en ligne est disponible uniquement dans l'édition Enterprise. La restauration de pages hors connexion est disponible dans les éditions inférieures.

## Préparations en vue de la restauration des sauvegardes

- Effectuer une sauvegarde de la fin du journal si nécessaire
  - S'applique uniquement au modèle de récupération complet et journalisé en bloc
- Identifier les sauvegardes à restaurer
  - Dernière sauvegarde complète, de fichiers ou de groupes de fichiers servant de base
  - Dernière sauvegarde différentielle, le cas échéant
  - Sauvegardes des journaux si vous utilisez le mode de récupération complète et journalisée en bloc

### Points clés

Dans les situations critiques, les utilisateurs font souvent des choix inappropriés quant aux actions à effectuer. Il est important d'éviter toute action qui fera empirer la situation. Avant de restaurer une base de données, il convient de tenter une sauvegarde de la fin du journal, sauf si vous avez l'intention de remplacer l'état actuel de cette base de données. La sauvegarde de la fin du journal peut souvent être effectuée, même si les fichiers de données de la base de données ont été endommagés. Cette opération est critique quand vous devez restaurer la base de données vers la dernière limite dans le temps possible.

### Identification des sauvegardes à restaurer

La récupération d'une base de données dépend de la restauration des sauvegardes appropriées dans le bon ordre. Le processus normal pour restaurer une base de données se déroule comme suit :

1. Restaurez la dernière sauvegarde complète de la base de données comme base de travail.  
(Si les fichiers endommagés ou manquants ne sont que des fichiers individuels, vous ne pouvez restaurer que ces fichiers).
2. Si des sauvegardes différentielles ont été créées, seule la dernière est nécessaire. (Les sauvegardes différentielles enregistrent toutes les étendues de base de données qui ont été modifiées depuis la dernière sauvegarde complète de la base de données. Elles ne sont pas incrémentielles par nature).
3. Si des sauvegardes de journaux de transactions ont été créées, toutes celles depuis la dernière sauvegarde différentielle sont requises. Vous devez également inclure la sauvegarde de la fin du journal créée au début du processus de restauration, si la sauvegarde de la fin du journal a réussi.  
(Cette étape ne s'applique pas aux bases de données en mode de récupération simple).

## Discussion : Détermination des sauvegardes requises pour la restauration

- Planification de la sauvegarde
  - Sauvegardes complètes des bases de données samedi à 22h
  - Sauvegardes différentielles les lundi, mardi, jeudi, vendredi à 22h
  - Sauvegardes des journaux toutes les heures entre 9h et 18h
- Défaillance survenue jeudi à 10 h 30
- Quel processus de restauration doit être suivi ?



### Discussion

L'exemple sur la diapositive décrit la planification de la sauvegarde pour une organisation.

**Question :** si une défaillance se produit jeudi à 10 h 30, quel est le processus de restauration à mettre en place ?

## Leçon 2

# Restauration des bases de données

- Phases du processus de restauration
- Option WITH RECOVERY
- Restauration d'une base de données
- Restauration d'un journal des transactions
- Option WITH STANDBY
- Démonstration 2A : Restauration des bases de données

Il est important de comprendre les phases que SQL Server 2012 utilise lors de la restauration d'une base de données. Une fois que la décision de restaurer une base de données a été prise, vous devez savoir comment implémenter le processus de restauration. Ce processus peut impliquer à la fois la base de données et les sauvegardes des journaux de transactions. Lorsque plusieurs sauvegardes doivent être restaurées dans un même traitement, vous devez déterminer vers quelle limite la récupération de la base de données doit être effectuée. Si la récupération de la base de données a lieu trop tôt dans le processus, vous ne pourrez pas effectuer le processus de restauration dans son intégralité.

## Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- décrire les différentes phases du processus de restauration ;
- restaurer une base de données à partir d'une sauvegarde complète ou différentielle ;
- restaurer une sauvegarde du journal des transactions ;
- contrôler la récupération de la base de données à l'aide de l'option WITH RECOVERY ;
- autoriser l'accès en lecture seule à une base de données en récupération à l'aide de l'option WITH STANDBY.

## Phases du processus de restauration

- Le processus de restauration d'une base de données SQL Server 2012 se déroule en trois étapes

Phase	Description
Copie des données	Crée les fichiers et copie les données dans les fichiers
Restauration par progression	Applique des transactions validées à partir des entrées de journal restaurées
Restauration par annulation	Rétablit les transactions qui n'étaient pas validées à partir du point de récupération

- Les phases de restauration par progression et par annulation sont appelées récupération

### Points clés

La restauration d'une base de données SQL Server 2012 consiste en trois phases : copie de données, restauration par progression et restauration. La combinaison des phases de restauration par progression et de restauration est communément appelée récupération d'une base de données.

### Copie des données

La phase de copie de données est généralement la plus longue phase dans une restauration de base de données. Les fichiers de données de la base de données doivent être récupérées à partir des sauvegardes. Avant que toutes les pages de données ne soient restaurées, l'en-tête de la sauvegarde est lu et SQL Server recrée les fichiers de données et les fichiers journaux requis. Si l'initialisation instantanée des fichiers (IFI) n'a pas été activée en accordant des droits au compte de service SQL Server, la réécriture des fichiers de données peut prendre beaucoup de temps.

Une fois les fichiers de données et les fichiers journaux recréés, les fichiers de données sont restaurés à partir de la sauvegarde complète de la base de données. Les pages de données sont récupérées dans l'ordre à partir de la sauvegarde, puis elles sont écrites dans les fichiers de données.

Les fichiers journaux doivent être réinitialisés pour pouvoir être utilisés. L'IFI n'est pas utilisée pour les fichiers journaux. Ce processus peut également prendre beaucoup de temps si les fichiers journaux sont volumineux.

Si vous restaurez également une sauvegarde différentielle, SQL Server remplace les étendues des fichiers de données par celles contenues dans la sauvegarde différentielle.

### Phase de restauration par progression

Les détails du journal des transactions sont ensuite récupérés. En mode de récupération simple, ces détails ne sont récupérés qu'à partir de la sauvegarde complète de la base de données ou de la sauvegarde différentielle, si celle-ci est également restaurée. En mode de récupération complète ou journalisée en bloc, ces détails de fichiers journaux sont complétés par le contenu de toutes les sauvegardes des journaux de transactions qui ont été effectuées après les sauvegardes complète et différentielle de la base de données.

Lors de la phase de restauration par progression, SQL Server restaure dans les pages de base de données toutes les modifications contenues dans les détails des journaux de transactions, jusqu'au point de récupération. Le point de récupération est généralement le dernier instant auquel les transactions figurent dans le journal.

### Phase de restauration

Notez que les détails du journal des transactions incluront probablement les détails des transactions qui n'ont pas été validées au point de récupération, ce qui correspond généralement au moment de la défaillance. Lors de la phase de restauration, SQL Server restaure toutes les transactions non validées.

Étant donné que l'action de la phase de restauration inclut la restauration des transactions non validées et la mise en ligne de la base de données, les sauvegardes suivantes ne peuvent pas être restaurées.

Pendant la phase de restauration, l'édition Enterprise de SQL Server 2012 permet de mettre en ligne la base de données et de la rendre accessible aux utilisateurs. Cette fonctionnalité est appelée récupération rapide. Les requêtes qui tentent d'accéder à des données en cours de restauration sont bloquées jusqu'à ce que la phase de restauration soit terminée. De ce fait, il est possible que le délai de certaines transactions soit dépassé.

**Question :** pourquoi SQL Server doit rétablir et annuler les transactions uniquement dans le cas d'une restauration d'une sauvegarde complète de base de données ?

## Option WITH RECOVERY

- Une base de données doit être récupérée avant d'être mise en ligne
- L'option de restauration WITH RECOVERY (par défaut) exécute une récupération après la restauration et place la base de données en ligne
- L'option de restauration WITH NORECOVERY laisse la base de données dans un état de récupération
  - Autorise des opérations de restauration supplémentaires sur la base de données
- Le processus de restauration implique toujours
  - WITH NORECOVERY pour toutes les sauvegardes restaurées à l'exception de la dernière
  - WITH RECOVERY pour la dernière sauvegarde restaurée

### Points clés

En règle générale, une base de données ne peut pas être mise en ligne tant qu'elle n'a pas été récupérée. Toutefois, il existe une exception à cette règle : l'utilisation de l'option de récupération rapide qui a été mentionnée dans le sujet précédent. Cette option permet aux utilisateurs d'accéder à la base de données alors que la phase de restauration est en cours.

### Événements de récupération

Notez que la récupération ne se produit pas uniquement pendant l'exécution des commandes RESTORE. Si une base de données est mise hors connexion, puis remise dans un état ONLINE, une récupération s'exécute. Le même processus de récupération se produit lorsque SQL Server 2012 redémarre.



**Remarque** D'autres événements aboutissent à une récupération de base de données : le clustering ou les basculements de mise en miroir de bases de données. Le clustering avec basculement et la mise en miroir de bases de données ne sont pas abordés dans ce cours. Ces événements sont mentionnés ici par souci d'exhaustivité.

Le processus de récupération dans SQL Server est déterminant pour la maintenance de l'intégrité transactionnelle qui requiert que toutes les transactions qui ont été validées soient enregistrées dans la base de données et que toutes celles qui ne l'ont pas été soient annulées.

## Options de récupération

Chaque commande RESTORE inclut une option permettant de spécifier WITH RECOVERY ou WITH NORECOVERY. L'option WITH RECOVERY est l'action par défaut et n'a pas besoin d'être spécifiée.

Il est important de choisir l'option appropriée (WITH RECOVERY ou WITH NORECOVERY) en exécutant une commande RESTORE. Le processus est simple dans la plupart des cas. Toutes les restaurations doivent être effectuées sans récupération (WITH NORECOVERY) à l'exception de la dernière, qui doit être exécutée avec récupération (WITH RECOVERY).

Il est impossible de restaurer d'autres sauvegardes après une restauration avec l'option WITH RECOVERY. Si une sauvegarde a été effectuée avec l'option WITH RECOVERY par accident, la séquence de restauration doit être redémarrée.

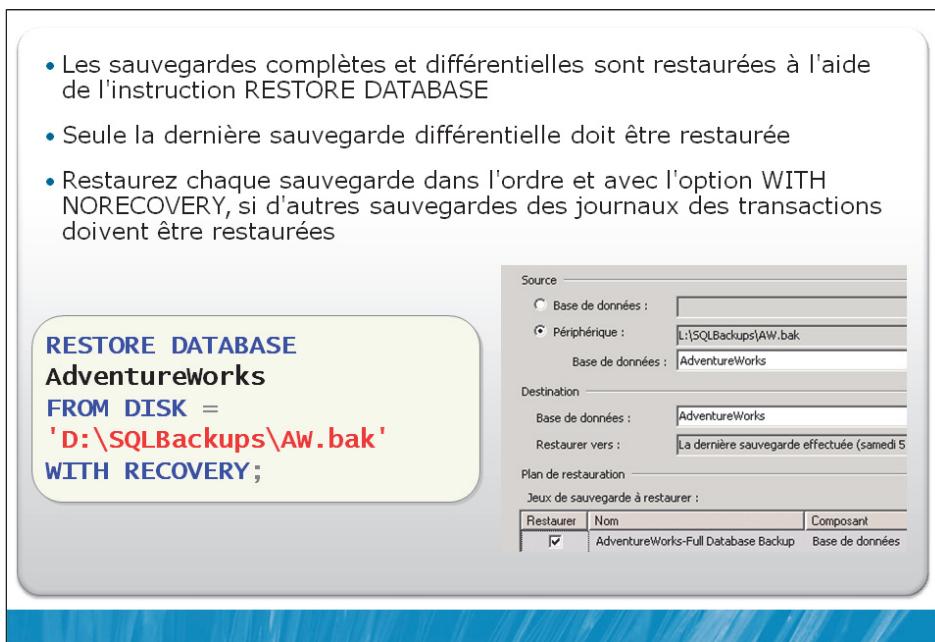
Si la dernière sauvegarde d'un jeu a également été restaurée par inadvertance avec l'option WITH NORECOVERY, il est possible de forcer la récupération de la base de données en exécutant la commande suivante :

```
RESTORE LOG databasename WITH RECOVERY;
```

**Question** : pourquoi n'est-il pas possible de restaurer des sauvegardes supplémentaires dans une base de données récupérée ?

## Restauration d'une base de données

- Les sauvegardes complètes et différentielles sont restaurées à l'aide de l'instruction RESTORE DATABASE
- Seule la dernière sauvegarde différentielle doit être restaurée
- Restaurez chaque sauvegarde dans l'ordre et avec l'option WITH NORECOVERY, si d'autres sauvegardes des journaux des transactions doivent être restaurées



### Points clés

Vous pouvez restaurer une base de données soit à partir de l'interface utilisateur graphique dans SSMS, soit en exécutant la commande RESTORE DATABASE dans T-SQL.

Si une sauvegarde de base de données est en cours de restauration, l'option WITH RECOVERY peut être utilisée car aucune sauvegarde ultérieure ne devra être restaurée. Vous pouvez aussi omettre l'option WITH RECOVERY étant donné qu'il s'agit de la valeur par défaut de la commande RESTORE DATABASE.

### Restauration différentielle

La commande pour restaurer une sauvegarde différentielle est identique à celle pour restaurer une sauvegarde complète de base de données. Les sauvegardes différentielles peuvent être ajoutées au même fichier que la sauvegarde complète de la base de données. Dans ce cas, vous devez spécifier le fichier du support de sauvegarde à restaurer.

Examinez la commande suivante :

```
RESTORE DATABASE AdventureWorks
    FROM DISK = 'D:\SQLBackups\AW.bak'
    WITH FILE = 1, NORECOVERY;

RESTORE DATABASE AdventureWorks
    FROM DISK = 'D:\SQLBackups\AW.bak'
    WITH FILE = 3,  RECOVERY;
```

Dans cet exemple, la base de données AdventureWorks est restaurée à partir du premier fichier dans le support de sauvegarde. Le support de sauvegarde est stocké dans le fichier D:\SQLBackups\AW.bak du système d'exploitation. Dans ce cas, le deuxième fichier qui se trouvait dans le support de sauvegarde correspond à la première sauvegarde différentielle qui a été effectuée sur la base de données. Le troisième fichier figurant dans le support de sauvegarde correspond à la deuxième sauvegarde différentielle qui a été effectuée sur la base de données. Étant donné que la deuxième sauvegarde différentielle est la dernière à avoir été exécutée, la seconde commande RESTORE dans l'exemple ci-dessus montre comment restaurer la dernière sauvegarde différentielle du support de sauvegarde.

## WITH REPLACE

SQL Server 2012 ne permet pas de restaurer une sauvegarde de base de données sur une base de données existante si vous n'avez pas effectué une sauvegarde de la fin du journal sur cette base de données. Si vous tentez de lancer l'opération via SQL Server Management Studio, SQL Server 2012 affiche un avertissement et tente de créer une sauvegarde de la fin du journal automatiquement. Si vous devez exécuter l'opération de restauration et que vous ne possédez pas de sauvegarde de la fin du journal, vous devez spécifier l'option WITH REPLACE.

 **Remarque** L'option WITH REPLACE doit être utilisée avec précaution car elle peut causer une perte de données.

## WITH MOVE

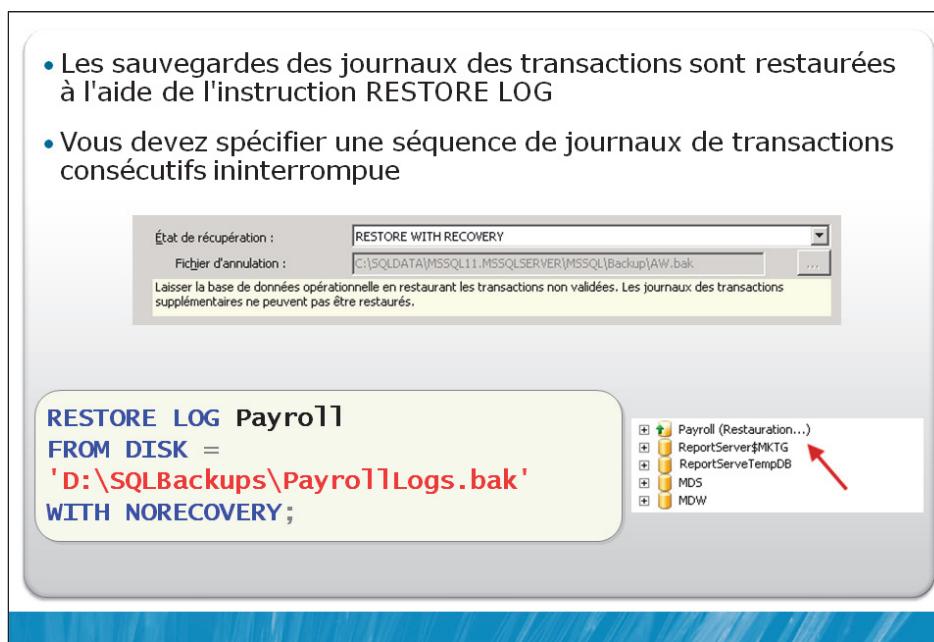
Si vous restaurez une base de données à partir d'un autre serveur, vous devrez peut-être placer les fichiers de base de données à des emplacements différents de ceux enregistrés dans la sauvegarde depuis le serveur d'origine. Vous pouvez également être amené à effectuer cette opération si vous copiez une base de données via un processus de sauvegarde et de restauration. L'option WITH MOVE permet de spécifier de nouveaux emplacements de fichiers. Examinez la commande suivante :

```
RESTORE DATABASE Spatial  
FROM DISK = 'D:\SQLBackups\Spatial.bak'  
WITH MOVE 'Spatial_Data' TO 'D:\MKTG\Spatial.mdf',  
MOVE 'Spatial_Log' TO 'L:\MKTG\Spatial.ldf';
```

Dans l'exemple ci-dessus, la base de données nommée Spatial est restaurée à partir d'un autre serveur. De même que l'emplacement source du support de sauvegarde, les nouveaux emplacements de chaque fichier de base de données ont été spécifiés dans la commande. Notez que l'option MOVE nécessite la spécification du nom de fichier logique, plutôt que le chemin d'accès au fichier physique d'origine.

## Restauration d'un journal des transactions

- Les sauvegardes des journaux des transactions sont restaurées à l'aide de l'instruction RESTORE LOG
- Vous devez spécifier une séquence de journaux de transactions consécutifs ininterrompue



### Points clés

Vous pouvez restaurer les journaux de transactions d'une base de données à l'aide de l'interface utilisateur graphique dans SSMS ou en utilisant la commande RESTORE LOG dans T-SQL. À l'exception du dernier journal, tous les fichiers journaux doivent être restaurés avec WITH NORECOVERY. Le dernier fichier journal (qui est souvent la sauvegarde de la fin du journal) est ensuite restauré avec l'option WITH RECOVERY.

### Restauration des journaux de transactions

Tous les journaux de transactions créés après la dernière sauvegarde complète ou différentielle doivent être restaurés dans l'ordre chronologique sans rupture de la chaîne de sauvegardes. S'il existe une rupture dans la chaîne des journaux de transactions, le processus de restauration échoue. Il est impossible de le poursuivre après une défaillance, et il est nécessaire de le redémarrer.

Lorsque la base de données est en mode de récupération, l'Explorateur d'objets l'indique par les mots « (Restoring...) » après le nom de la base de données, comme l'illustre l'exemple de la diapositive.

**Question :** pourquoi est-il plus rapide de restaurer les sauvegardes différentielles et les sauvegardes des journaux que de restaurer l'ensemble des sauvegardes de journaux depuis la dernière sauvegarde complète de la base de données ?

## Option WITH STANDBY

- Autorise l'accès en lecture seule à une base de données restaurée
  - Un fichier d'annulation est utilisé pour gérer les détails de la phase de restauration par annulation
- Les scénarios d'utilisation principaux sont les suivants :
  - Création d'un serveur de secours avec un accès en lecture seule aux données (copie des journaux des transactions)
  - Examen d'une base de données entre les restaurations des journaux



```
RESTORE LOG Payroll  
FROM DISK = 'D:\Backups\PyLg.bak'  
WITH STANDBY = 'D:\Backups\ULog.bak';
```

### Points clés

SQL Server 2012 permet d'afficher, à l'aide de l'option WITH STANDBY au lieu de l'option WITH NORECOVERY, le contenu d'une base de données qui n'a pas été récupérée.

D'autres sauvegardes des journaux de transactions peuvent être appliquées à une base de données qui a été restaurée avec l'option WITH STANDBY. Il existe deux raisons habituelles expliquant l'utilisation de l'option WITH STANDBY.

### WITH STANDBY et envoi de journaux

Une fonctionnalité de haute disponibilité couramment utilisée permet de gérer un serveur de secours qui peut être mis en ligne rapidement. Cette fonctionnalité est appelée envoi de journaux. L'opération de base de l'envoi de journaux consiste à automatiser le processus de sauvegarde des fichiers journaux sur un ordinateur, à copier les fichiers journaux sur un autre ordinateur, puis à les restaurer sur cet ordinateur. La base de données sur le deuxième serveur est presque complète mais inutilisable si les restaurations ont été effectuées avec l'option WITH NORECOVERY. Il est impossible de la récupérer pour une utilisation en lecture seule car il faudrait restaurer d'autres journaux de transactions ultérieurement.



**Remarque** L'envoi de journaux est un sujet avancé qui n'entre pas dans le cadre de ce cours. Toutefois, une présentation de cette notion est fournie dans l'annexe A.

L'option WITH STANDBY a été conçue pour vous aider dans cette situation, en créant une version de récupération modifiée qui copie dans un fichier du système d'exploitation les transactions qui auraient été supprimées par la phase de restauration. Lorsque l'opération suivante de restauration du journal des transactions est requise, SQL Server réapplique automatiquement au journal les transactions depuis ce fichier avant de poursuivre la restauration du journal.

### **WITH STANDBY pour l'inspection**

Imaginez qu'une erreur utilisateur ait provoqué la suppression accidentelle de certaines données. Vous risquez de ne pas savoir à quel moment cette erreur s'est produite. Lorsque vous restaurerez la base de données, il se peut alors que vous ne sachiez pas quel fichier journal contient la suppression. Vous pouvez utiliser l'option WITH STANDBY sur chaque restauration de fichier journal et examiner l'état de la base de données après chaque opération de restauration.

**Question :** quelle raison peut expliquer l'octroi d'un accès en lecture seule à une base de données ?

**Question :** quelle pourrait être la limite de l'option WITH STANDBY lorsqu'elle est utilisée pour autoriser la création de rapports sur la deuxième base de données de secours dans un environnement d'envoi de journaux ?

## Démonstration 2A : Restauration des bases de données

Dans cette démonstration, vous allez apprendre à restaurer des bases de données à partir d'un mode de récupération complète

### Procédure de démonstration

1. Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn**. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_07\_PRJ\22462A\_07\_PRJ.ssmssln**, puis cliquez sur **Ouvrir**.
3. Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** dans l'Explorateur de solutions. Remarque : le script d'installation de ce module permet de générer une erreur en cas de fichiers manquants ; ce comportement est normal.
4. Ouvrez le fichier de script **21 – Demonstration 2A.sql**.
5. Suivez les instructions contenues dans les commentaires du fichier de script.

## Leçon 3

# Utilisation de la récupération limitée dans le temps

- Vue d'ensemble de la récupération limitée dans le temps
- Option STOPAT
- Discussion : Synchronisation de la récupération de plusieurs bases de données
- Option STOPATMARK
- Démonstration 3A : Utilisation de l'option STOPATMARK

Dans la leçon précédente, vous avez découvert comment récupérer une base de données vers la dernière limite dans le temps possible. Toutefois, il existe des cas où la base de données doit être récupérée vers une limite antérieure dans le temps. Vous avez appris que vous pouviez arrêter le processus de restauration après la restauration d'une des sauvegardes, puis lancer la récupération de la base de données. Alors que l'arrêt du processus de restauration après une restauration fournit un niveau de contrôle ordinaire sur le point de récupération, SQL Server 2012 fournit des options supplémentaires qui offrent un contrôle plus précis du point de récupération.

## Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- décrire la récupération dans le temps ;
- implémenter l'option de restauration Stop at time ;
- expliquer les complexités englobant la synchronisation de plusieurs bases de données ;
- implémenter l'option de restauration Stop at mark.

## Vue d'ensemble de la récupération limitée dans le temps

- Autorise la récupération d'une base de données jusqu'à un point dans le temps arbitraire qui figure dans les sauvegardes des journaux des transactions
- Un point dans le temps peut être défini comme suit
  - Valeur datetime fournie
  - Marque définie via une transaction nommée
- La base de données doit être en mode de récupération complète
- Les journaux peuvent contenir des sections BULK\_LOGGED
  - Si le point dans le temps de la restauration se situe au cours d'une période d'opérations journalisées de façon minimale, la restauration échoue

### Points clés

La récupération d'une base de données vers la limite dans le temps la plus récente est l'option la plus couramment demandée. Toutefois, la restauration vers une limite antérieure dans le temps peut également s'avérer nécessaire.

SQL Server 2012 permet d'arrêter la restauration d'une base de données au niveau d'une limite spécifiée dans le temps, puis de lancer la récupération. Cette limite dans le temps peut être définie de deux façons différentes. Vous pouvez spécifier une valeur datetime qui indique le moment exact du point de récupération. N'oubliez pas que les temps de traitement ont tendance à être approximatifs et que les systèmes informatiques peuvent exécuter une grande quantité de tâches sur une courte période. Dans ce cas, la valeur datetime risque de ne pas être suffisamment précise pour spécifier un point de récupération.

L'autre solution fournie par SQL Server 2012 consiste à spécifier une transaction nommée (appelée marque du journal des transactions) comme point de récupération.

Pour que l'une ou l'autre de ces options fonctionne, il faut que la base de données soit en mode de récupération complète ou journalisée en bloc. SQL Server peut uniquement interrompre le traitement à des points de la chaîne du journal des transactions lorsque la base de données est en mode de récupération complète. Si une base de données passe du mode de récupération complète au mode de récupération journalisée en bloc pour traiter les transactions en bloc, puis repasse en mode de récupération complète, le point de récupération ne peut pas être le moment auquel la base de données était en mode de récupération journalisée en bloc. Si vous essayez de spécifier un point de récupération pendant lequel la base de données était en mode de récupération journalisée en bloc, la restauration échoue et une erreur est retournée.

**Question :** quelle autre option de restauration peut être utile si la limite dans le temps n'est pas connue avec précision et qu'aucun indicateur n'a été défini ?

## Option STOPAT

- Peut être utilisé avec SSMS ou T-SQL
- Fournit STOPAT avec RECOVERY dans le cadre des instructions RESTORE de la séquence
  - Nul besoin de savoir dans quelle sauvegarde de journaux des transactions se trouve le point dans le temps
  - Si le point dans le temps est postérieur à la date et heure incluses dans la sauvegarde, un message d'avertissement s'affiche et la base de données ne sera pas récupérée à l'issue de la restauration
  - Si le point dans le temps est antérieur à la date et heure incluses dans la sauvegarde, l'instruction RESTORE échoue
  - Si le point dans le temps est fourni dans les délais de la sauvegarde, la base de données est récupérée jusqu'à cette date/heure

### Points clés

L'option STOPAT permet de spécifier un point de récupération basé sur une valeur datetime. Étant donné que l'administrateur de base de données risque de ne pas savoir à l'avance quelle sauvegarde du journal des transactions contient l'heure à laquelle la récupération doit se produire, la syntaxe de la commande RESTORE LOG permet de spécifier l'option RECOVERY pour chaque commande de restauration du journal dans la séquence.

Prenez la séquence de restauration suivante :

```
RESTORE DATABASE database_name FROM full_backup  
WITH NORECOVERY;  
  
RESTORE DATABASE database_name FROM differential_backup  
WITH NORECOVERY;  
  
RESTORE LOG database_name FROM first_log_backup  
WITH STOPAT = time, RECOVERY;  
  
... (les sauvegardes de journal supplémentaires peuvent être restaurées ici)  
  
RESTORE LOG database_name FROM final_log_backup  
WITH STOPAT = time, RECOVERY;
```

Notez que l'option RECOVERY est spécifiée sur chaque commande RESTORE LOG, pas simplement sur la dernière commande.

Le comportement des options STOPAT et RECOVERY est le suivant :

- Si l'heure spécifiée est antérieure à la première heure dans la sauvegarde du journal des transactions, la commande de restauration échoue et retourne une erreur.
- Si l'heure spécifiée est comprise dans la période couverte par la sauvegarde du journal des transactions, la commande de restauration récupère la base de données à ce moment précis.
- Si l'heure spécifiée est postérieure à la dernière figurant dans la sauvegarde du journal des transactions, la commande de restauration restaure les journaux, envoie un message d'avertissement, et la base de données n'est pas récupérée afin que d'autres sauvegardes du journal des transactions puissent être appliquées.

Avec ce comportement décrit ci-dessus, la base de données est récupérée jusqu'au point demandé, même lorsque les options STOPAT et RECOVERY sont spécifiées avec chaque restauration, tant que le point demandé n'est pas antérieur à la séquence de restauration.

**Question** : pourquoi pouvez-vous être amené à récupérer une base de données jusqu'à une limite spécifique dans le temps ?

## Discussion : Synchronisation de la récupération de plusieurs bases de données

- Une application peut utiliser des données se trouvant dans plusieurs bases de données, y compris des données dans plusieurs instances de SQL Server
  - Utilisez-vous des applications de bases de données multiples ?
  - Quels problèmes peuvent se produire lorsque les bases de données doivent être restaurées ?
  - Pourquoi la restauration jusqu'à une limite dans le temps peut-elle s'avérer insuffisante ?



### Discussion

Une application peut utiliser des données se trouvant dans plusieurs bases de données, y compris des données dans plusieurs instances de SQL Server.

**Question :** utilisez-vous des applications de bases de données multiples ?

**Question :** quels problèmes peuvent se produire lorsque les bases de données doivent être restaurées ?

**Question :** pourquoi la restauration jusqu'à une limite dans le temps peut-elle s'avérer insuffisante ?

## Option STOPATMARK

- Ne peut être effectuée qu'avec T-SQL
- Transactions marquées avec
  - BEGIN TRAN <nom> WITH MARK <description>
- La restauration présente deux options associées
  - STOPATMARK avance jusqu'à la marque et inclut la transaction marquée dans la restauration par progression
  - STOPBEFOREMARK avance jusqu'à la marque et exclut la transaction marquée à partir de la restauration par progression
- Si l'indicateur n'est pas présente dans la sauvegarde du journal des transactions, la sauvegarde est restaurée, mais la base de données n'est pas récupérée

### Points clés

Si un contrôle plus précis du point de récupération est requis, l'option STOPATMARK peut s'avérer utile. L'interface utilisateur graphique dans SSMS ne comporte aucune option permettant de traiter les marques du journal des transactions pendant les opérations de restauration. Ce processus doit être exécuté via T-SQL.

### Marquage d'une transaction

Si vous savez à l'avance que vous devrez peut-être effectuer une récupération vers la limite d'une opération spécifique, vous pouvez placer une marque dans le journal des transactions pour enregistrer cet emplacement précis. Examinez la commande suivante :

```
BEGIN TRAN UpdPrc WITH MARK 'Start of nightly update process';
```

Avec cette commande, une transaction est débutée mais le nom UpdPrc lui est également attribué, et une marque de transaction portant le même nom que la transaction est créée. La valeur indiquée après la clause WITH MARK n'est qu'une description ; elle n'est pas utilisée dans le traitement de la marque de transaction.

Si vous ne connaissez pas le nom d'une transaction qui a été marquée, vous pouvez interroger la table dbo.logmarkhistory dans la base de données msdb.

## Options relatives au marquage

L'option STOPATMARK est similaire à l'option STOPAT de la commande RESTORE. SQL Server s'arrête à la marque de transaction nommée et inclut la transaction nommée dans la phase de restauration par progression.

Si vous souhaitez exclure la transaction (c'est-à-dire restaurer tout jusqu'au début de la transaction nommée), vous pouvez spécifier l'option STOPBEFOREMARK à la place.

Si la marque de transaction est introuvable dans la sauvegarde du journal des transactions qui est en cours de restauration, la restauration s'exécute et la base de données n'est pas récupérée afin que les autres sauvegardes des journaux de transactions puissent être restaurées.

## Applications multi-bases de données

Le principal usage de la fonction Stop at Mark est de restaurer un ensemble complet de bases de données dans un état de cohérence réciproque à une certaine date/heure antérieure. Si vous devez effectuer une sauvegarde de plusieurs bases de données afin qu'elles puissent être récupérées à un point cohérent, envisagez de marquer tous les journaux de transactions avant de lancer les sauvegardes.

**Question :** pourquoi l'option STOPAT peut-elle ne pas constituer un bon choix pour synchroniser la restauration de plusieurs bases de données et pourquoi l'option STOPATMARK pourrait-elle être préférable ?

## Démonstration 3A : Utilisation de l'option STOPATMARK

Dans cette démonstration, vous allez apprendre à restaurer une base de données jusqu'à une marque dans le journal des transactions

### Procédure de démonstration

1. Si la démonstration 2A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn.** Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_07\_PRJ\22462A\_07\_PRJ.ssmssqln**, puis cliquez sur **Ouvrir**.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** dans l'Explorateur de solutions. Remarque : le script d'installation de ce module permet de générer une erreur en cas de fichiers manquants ; ce comportement est normal.
2. Ouvrez et exécutez le fichier de script **31 – Demonstration 3A.sql** dans l'Explorateur de solutions.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

## Leçon 4

# Restauration des bases de données système et des fichiers individuels

- Récupération des bases de données système
- Restauration de la base de données MASTER
- Restauration d'un fichier ou d'un groupe de fichiers à partir d'une sauvegarde
- Démonstration 4A : Restauration d'un fichier

Les bases de données utilisateur sont restaurées beaucoup plus régulièrement que les bases de données système, car elles sont généralement beaucoup plus volumineuses et, par conséquent, plus exposées aux défaillances étant donné qu'elles sont souvent réparties sur plusieurs périphériques. Toutefois, les défaillances peuvent affecter les bases de données système, qui devront peut-être être restaurées. Restaurer une base de données système n'est pas identique à la restauration des bases de données utilisateur.

La base de données MASTER est la base de données la plus critique pour un système SQL Server, et sa récupération implique davantage d'étapes que la récupération d'autres bases de données.

Dans certaines situations, au lieu d'une restauration complète de base de données, seule la restauration d'un fichier unique ou d'un groupe de fichiers peut s'avérer nécessaire, ce qui accélère le processus global de restauration.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- récupérer les bases de données système ;
- restaurer la base de données MASTER ;
- restaurer un fichier ou groupe de fichiers à partir d'une sauvegarde.

## Récupération des bases de données système

Base de données système	Description
master	Sauvegarde requise : Oui Mode de récupération : Simple Restauration en mode mono-utilisateur
model	Sauvegarde requise : Oui Mode de récupération : Configurable par l'utilisateur Restauration en utilisant l'indicateur de trace -T3608
msdb	Sauvegarde requise : Oui Mode de récupération : Simple (par défaut) Restauration comme n'importe quelle autre base de données utilisateur
tempdb/resource	Aucune sauvegarde ne peut être effectuée tempdb est créé au cours du démarrage d'une instance Restaurer la ressource en utilisant la restauration de fichiers ou l'installation

### Points clés

Le processus de récupération n'est pas identique pour toutes les bases de données système. Pour chaque base de données système, les exigences en matière de récupération sont spécifiques.

#### master

La base de données MASTER contient toutes les configurations de niveau système. SQL Server requiert la base de données MASTER pour qu'une instance de SQL Server puisse s'exécuter totalement.

Une procédure spéciale doit être suivie pour restaurer la base de données MASTER. Le processus de restauration de cette base de données est présenté dans le sujet suivant.

#### model

La base de données model constitue le modèle de toutes les bases de données créées sur l'instance de SQL Server. Lorsque la base de données model est endommagée, l'instance de SQL Server ne peut pas démarrer. Cela signifie qu'il est impossible d'utiliser une commande de restauration normale pour récupérer la base de données model si elle est endommagée. Si la base de données model est endommagée, l'instance doit être démarrée en spécifiant l'indicateur de trace -T3608 comme paramètre de ligne de commande. Cet indicateur de trace démarre uniquement la base de données MASTER. Une fois que SQL Server est en cours d'exécution, vous pouvez restaurer la base de données model à l'aide de la commande RESTORE DATABASE normale.

#### msdb

La base de données msdb est utilisée par SQL Server Agent pour planifier les alertes et les travaux et pour enregistrer les détails des opérateurs. Elle contient également des tables d'historique, telles que celles qui enregistrent les détails des opérations de sauvegarde et de restauration. Si la base de données msdb est endommagée, SQL Server Agent ne la démarre pas. Vous pouvez alors la restaurer comme les bases de données utilisateur, à l'aide de la commande RESTORE DATABASE. Le service SQL Server Agent peut ensuite être redémarré.

### **resource**

La base de données resource est une base de données en lecture seule qui contient les copies de tous les objets système fournis avec Microsoft SQL Server 2012. Aucune opération de sauvegarde ne peut être effectuée sur cette base de données, qui est masquée. En revanche, elle peut être endommagée par des défaillances dans les zones telles que les sous-systèmes d'E/S ou la mémoire. Si la base de données resource est endommagée, elle peut être restaurée par une restauration au niveau fichier dans Windows ou en exécutant le programme d'installation de SQL Server.

### **tempdb**

La base de données tempdb est un espace de travail permettant de gérer les ensembles de résultats temporaires ou intermédiaires. Cette base de données est recréée chaque fois qu'une instance de SQL Server est démarrée. Lorsque l'instance de serveur est arrêtée, toutes les données de tempdb sont supprimées définitivement. Aucune sauvegarde ne peut être exécutée sur la base de données tempdb, mais étant donné qu'elle est recréée à chaque redémarrage de l'instance, le redémarrage de l'instance est suffisant pour récupérer tempdb en cas d'endommagement.

## Restauration de la base de données MASTER

1. Démarrez l'instance de serveur en mode mono-utilisateur
2. Utilisez l'instruction RESTORE DATABASE pour restaurer une sauvegarde complète de base de données de la base de données master
3. SQL s'arrête et met fin au processus SQLCMD
4. Redémarrez SQL Server normalement (pas un utilisateur unique)

### Points clés

La base de données MASTER est essentielle au fonctionnement de SQL Server. Étant donné que SQL Server ne démarre pas si la base de données MASTER est manquante ou endommagée, vous ne pouvez pas exécuter de commande RESTORE DATABASE standard pour la restaurer si une base de données est manquante ou endommagée.

### Récupération de la base de données MASTER

Préalablement au démarrage de ce processus, l'existence d'une version de base de données MASTER est impérative pour que l'instance de SQL Server démarre totalement. Si votre base de données MASTER est endommagée, une base de données MASTER temporaire doit être créée au préalable. Il n'est pas nécessaire que la configuration de cette base de données MASTER temporaire soit correcte, car elle ne sera utilisée que pour démarrer l'instance. La base de données MASTER correcte est ensuite restaurée à l'aide du processus décrit sur la diapositive.

Il existe trois manières d'obtenir une base de données MASTER temporaire. Vous pouvez utiliser le programme d'installation de SQL Server pour reconstruire les bases de données système soit depuis l'emplacement auquel vous avez installé SQL Server, soit en exécutant le programme d'installation qui se trouve à l'emplacement suivant : Microsoft SQL Server\110\Setup\Bootstrap\SQL11\setup.exe. (Ce chemin d'accès est approximatif et est susceptible de changer ultérieurement).



**Remarque** Le programme d'installation remplace toutes les bases de données système, qui devront toutes être restaurées ultérieurement.

L'autre méthode permettant d'obtenir une base de données MASTER temporaire consiste à utiliser la sauvegarde au niveau fichier des fichiers de base de données MASTER pour restaurer la base de données MASTER. Cette sauvegarde au niveau fichier doit avoir été effectuée lorsque la base de données MASTER n'était pas en cours d'utilisation, c'est-à-dire lorsque SQL Server n'était pas en cours d'exécution, ou à l'aide du service VSS.

 **Remarque** La copie de la base de données MASTER depuis une autre instance n'est pas prise en charge. Le service VSS ne rentre pas dans le cadre de ce cours.

Enfin, la dernière solution consiste à rechercher une base de données master.mdf dans le dossier Templates se trouvant dans le dossier MSSQL\Binn pour chaque instance.

Une fois qu'une base de données MASTER temporaire a été mise en place, utilisez la procédure suivante pour récupérer la base de données MASTER correcte :

- Démarrez l'instance de serveur en mode mono-utilisateur. (Le gestionnaire de configuration de SQL Server peut être utilisé pour démarrer SQL Server en mode mono-utilisateur à l'aide de l'option de démarrage -m).
- Utilisez une instruction RESTORE DATABASE pour restaurer une sauvegarde complète de la base de données MASTER. (En mode mono-utilisateur, il est recommandé d'entrer l'instruction RESTORE DATABASE à l'aide de l'utilitaire sqlcmd).
- Une fois la base de données MASTER restaurée, l'instance de SQL Server s'arrête et met fin à votre connexion sqlcmd.
- Supprimez le paramètre de démarrage en mode mono-utilisateur.
- Redémarrez SQL Server.

## Restauration d'un fichier ou d'un groupe de fichiers à partir d'une sauvegarde

1. Créer une sauvegarde de la fin du journal
2. Restaurer un fichier ou un groupe de fichiers endommagé à partir d'une sauvegarde récente
3. Restaurer des sauvegardes de fichiers différentielles
4. Restaurer les journaux des transactions de manière séquentielle \*
5. Récupérer la base de données

\* Restaurez les sauvegardes des journaux des transactions en ordre en commençant par le journal du fichier le plus ancien et en terminant par la fin du journal (uniquement les modes de récupération complète/journalisée en bloc)

### Points clés

Restaurer des fichiers individuels ou des groupes de fichiers à partir des sauvegardes plutôt que de restaurer les bases de données dans leur intégralité s'avère généralement une solution beaucoup plus rapide en cas d'endommagement ou si des fichiers ou groupes de fichiers sont manquants.



**Remarque** Étant donné qu'il est possible de ne sauvegarder que les fichiers ou les groupes de fichiers, il n'est pas nécessaire d'avoir effectué des sauvegardes de fichiers ou de groupes de fichiers avant de restaurer des fichiers individuels ou des groupes de fichiers. SQL Server peut extraire des fichiers de base de données spécifiques d'une sauvegarde complète ou d'une sauvegarde différentielle.

### Processus de restauration

1. Créez une sauvegarde de la fin du journal à partir du journal des transactions actif. (Si vous ne pouvez pas effectuer cette opération parce que le journal est endommagé, vous devez restaurer la base de données entièrement ou la restaurer vers une limite antérieure dans le temps).
2. Restaurez chaque fichier endommagé à partir de sa sauvegarde la plus récente.
3. Restaurez la sauvegarde différentielle de fichiers la plus récente, si elle existe, pour chaque fichier restauré.

4. Restaurez les sauvegardes des journaux de transactions dans l'ordre, en commençant par celle qui couvre les fichiers restaurés les plus anciens et en terminant par celle de la fin du journal créée à l'étape 1.
5. Récupérez la base de données.

Vous devez restaurer les sauvegardes des journaux de transactions qui ont été créées après les sauvegardes de fichiers afin de rétablir la base de données dans un état cohérent. Les sauvegardes des journaux de transactions peuvent être rapidement restaurées par progression, car seuls les changements concernant les fichiers ou groupes de fichiers restaurés sont appliqués. Les fichiers intacts ne sont pas copiés ni restaurés par progression. Cependant, le traitement de la séquence complète des sauvegardes de journaux reste nécessaire.

## Démonstration 4A : Restauration d'un fichier

Dans cette démonstration, vous allez apprendre à restaurer un fichier d'une sauvegarde complète de base de données en mode de récupération complète

### Procédure de démonstration

1. Si la démonstration 2A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn.** Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_07\_PRJ\22462A\_07\_PRJ.ssmssqln**, puis cliquez sur **Ouvrir**.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** dans l'Explorateur de solutions. Remarque : le script d'installation de ce module permet de générer une erreur en cas de fichiers manquants ; ce comportement est normal.
2. Ouvrez et exécutez le fichier de script **41 – Demonstration 4A.sql** à partir de l'Explorateur de solutions.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

## Atelier pratique 7 : Restauration des bases de données SQL Server 2012

- Exercice 1 : Déterminer une stratégie de restauration
- Exercice 2 : Restaurer la base de données
- Exercice difficile 3 : En mode veille  
(seulement si le temps le permet)

Informations d'ouverture de session

Ordinateur virtuel	22462A-MIA-SQL1
Nom d'utilisateur	AdventureWorks\Administrateur
Mot de passe	Pa\$\$w0rd

Durée approximative : 45 minutes

### Configuration de l'atelier pratique

Pour cet atelier pratique, vous utiliserez l'environnement d'ordinateur virtuel disponible.

Avant de commencer cet atelier pratique, vous devez procéder aux étapes suivantes :

1. Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
3. Dans la fenêtre Se connecter au serveur, tapez **Proseware** dans la zone de texte **Nom du serveur**.
4. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn**.
5. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**.
6. Dans la fenêtre Ouvrir un projet, sélectionnez le projet **D:\22462A\_Labs\22462A\_07\_PRJ.ssmssln**.
7. Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Dans l'Explorateur de solutions, double-cliquez sur le fichier de requête **00 - Setup.sql**. Lorsque la fenêtre de requête s'affiche, cliquez sur **Exécuter** dans la barre d'outils.



**Remarque** Le script d'installation de ce module est destiné à générer une erreur si des fichiers sont manquants. C'est normal.

### Scénario de l'atelier pratique

On vous a fourni une série de sauvegardes issues d'une base de données d'un autre serveur à partir duquel vous devez effectuer la restauration sur le serveur Proseware, Inc. avec le nom de base de données MarketYields. Le fichier de sauvegarde inclut plusieurs sauvegardes complètes et différentielles, ainsi que des sauvegardes de fichiers journaux. Vous devez identifier les sauvegardes figurant dans le fichier, déterminer celles qui doivent être restaurées et effectuer des opérations de restauration. Lorsque vous restaurez la base de données, vous devez vérifier qu'elle reste en veille semi-automatique, car d'autres sauvegardes de journaux peuvent être appliquées ultérieurement.

Si vous avez le temps, testez l'opération de veille.

## Exercice 1 : Déterminer une stratégie de restauration

### Scénario

Vous devez restaurer une sauvegarde de base de données à partir d'une autre instance de Proseware. Il vous a été fourni un fichier de sauvegarde contenant plusieurs sauvegardes complètes, différentielles et de fichiers journaux. Dans cet exercice, vous devez déterminer les sauvegardes qui sont contenues dans le fichier et les sauvegardes qui doivent être restaurées et dans quel ordre.

Les tâches principales de cet exercice sont les suivantes :

1. Examiner les sauvegardes contenues dans le fichier de sauvegarde.
2. Déterminer comment la restauration doit être effectuée.

► **Tâche 1 : Examiner les sauvegardes contenues dans le fichier de sauvegarde**

- Utilisez l'option HEADERONLY de la commande RESTORE pour identifier les sauvegardes contenues dans le fichier D:\MSSQLSERVER\MarketYields.bak.

► **Tâche 2 : Déterminer comment la restauration doit être effectuée**

- Déterminez quelles sauvegardes doivent être restaurées et dans quel ordre.

**Résultats** : À la fin de cet exercice, vous devez avoir identifié les sauvegardes qui doivent être restaurées.

## Exercice 2 : Restaurer la base de données

### Scénario

Vous avez identifié les sauvegardes à restaurer. Vous devez maintenant restaurer la base de données MarketYields sur l'instance de Proseware à partir des sauvegardes que vous avez choisies. Vous laisserez la base de données en mode veille.

La tâche principale de cet exercice est la suivante :

1. Restaurer la base de données.

### ► Tâche 1 : Restaurer la base de données

- À l'aide de SSMS, restaurez la base de données MarketYields en utilisant les sauvegardes que vous avez déterminé comme nécessaires lors de l'exercice 1. Veillez à utiliser l'option STANDBY en attribuant au fichier d'annulation le nom STANDBY L:\MKTG\Log \_Standby.bak. Vous devez déplacer le fichier mdf dans le dossier D:\MKTG et le fichier ldf dans le dossier L:\MKTG.
- Dans l'Explorateur d'objets, actualisez la liste des bases de données et vérifiez l'état de la base de données MarketYields sur l'instance de Proseware.

**Résultats** : À la fin de cet exercice, vous devez avoir restauré la base de données en mode veille.

## Exercice difficile 3 : En mode veille (seulement si le temps le permet)

### Scénario

Dans cet exercice, vous allez vérifier que le mode veille fonctionne comme prévu. Vous allez accéder à la base de données et restaurer un autre fichier journal pour vérifier que la base de données peut être restaurée.

Les tâches principales de cet exercice sont les suivantes :

1. Exécuter des requêtes sur la base de données STANDBY pour vous assurer qu'elle est accessible.
2. Restaurer un autre fichier journal, en laissant la base de données en mode veille.

► **Tâche 1 : Exécuter des requêtes sur la base de données STANDBY pour vous assurer qu'elle est accessible**

- Ouvrez une fenêtre de requête sur la base de données MarketYields sur l'instance de Proseware.
- Sélectionnez le nombre de lignes dans la table LogData.
- Fermez la fenêtre de requête.

► **Tâche 2 : Restaurer un autre fichier journal, en laissant la base de données en mode veille**

- Restaurez le fichier journal D:\MSSQLSERVER\MarketYields\_log.bak. Veillez à laisser la base de données en mode veille.
- Dans l'Explorateur d'objets, actualisez la liste des bases de données et vérifiez l'état de la base de données MarketYields sur l'instance de Proseware.

**Résultats** : À la fin de cet exercice, vous devez avoir testé la fonctionnalité STANDBY.

## Contrôle des acquis et éléments à retenir

- Questions de contrôle des acquis
- Meilleures pratiques

### Questions de contrôle des acquis

1. Quelles sont les trois phases du processus de restauration ?
2. Quelle opération est toujours exécutée avant qu'une base de données ne démarre et ne soit connectée ?

### Meilleures pratiques liées à un domaine technologique particulier dans ce module

1. N'oubliez pas de sauvegarder la fin du journal avant de démarrer une séquence de restauration.
2. Utilisez la restauration différentielle pour accélérer le processus de restauration si elle est disponible.
3. Utilisez la restauration de fichiers pour accélérer les restaurations lorsque certains fichiers de la base de données ne sont pas endommagés.
4. Effectuez des sauvegardes régulières des bases de données système master, msdb et model.
5. Créez un plan de récupération d'urgence pour votre serveur SQL et testez régulièrement la restauration des bases de données.

# Module 8

## Importation et exportation de données

### Sommaire :

Leçon 1 : Transfert des données vers/de SQL Server	8-3
Leçon 2 : Importation et exportation des données de table	8-15
Leçon 3 : Insertion de données en bloc	8-20
Atelier pratique 8 : Importation et exportation de données	8-29

## Vue d'ensemble du module

- Transfert des données vers/de SQL Server
- Importation et exportation des données de table
- Insertion de données en bloc

Bien qu'une grande proportion des données qui résident sur un système Microsoft® SQL Server® puisse être entrée directement par les utilisateurs qui exécutent des programmes d'application active, il peut aussi y avoir besoin de déplacer des données vers d'autres emplacements, en provenance et en direction de SQL Server.

SQL Server fournit un ensemble d'outils permettant de transférer des données en direction ou en provenance de SQL Server. Certains de ces outils, tels que l'utilitaire bcp et SQL Server Integration Services sont externes au moteur de base de données. D'autres outils, tels que l'instruction BULK INSERT et la fonction OPENROWSET, sont intégrés dans le moteur de base de données. Ce module vous permettra d'explorer brièvement chacun de ces outils.

Il est occasionnellement nécessaire d'importer de grandes quantités de données dans SQL Server. Bien qu'il soit possible d'utiliser les paramètres par défaut de SQL Server lors de l'importation des données, il est possible d'atteindre de meilleures performances en exerçant un contrôle sur l'usage des contraintes, des déclencheurs et des index au cours du processus d'importation.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- transférer des données vers et à partir de SQL Server ;
- importer et exporter des données de table ;
- insérer des données en bloc et optimiser le processus d'insertion en bloc.

## Leçon 1

# Transfert des données vers/de SQL Server

- Présentation du transfert de données
- Outils disponibles pour le transfert de données
- Amélioration des performances du transfert de données
- Désactivation et reconstruction des index
- Désactivation et activation des contraintes
- Démonstration 1A : Désactivation et activation des contraintes

La première étape de l'apprentissage du transfert de données vers et à partir de SQL Server consiste à se familiariser avec les processus impliqués et avec les outils fournis par SQL Server.

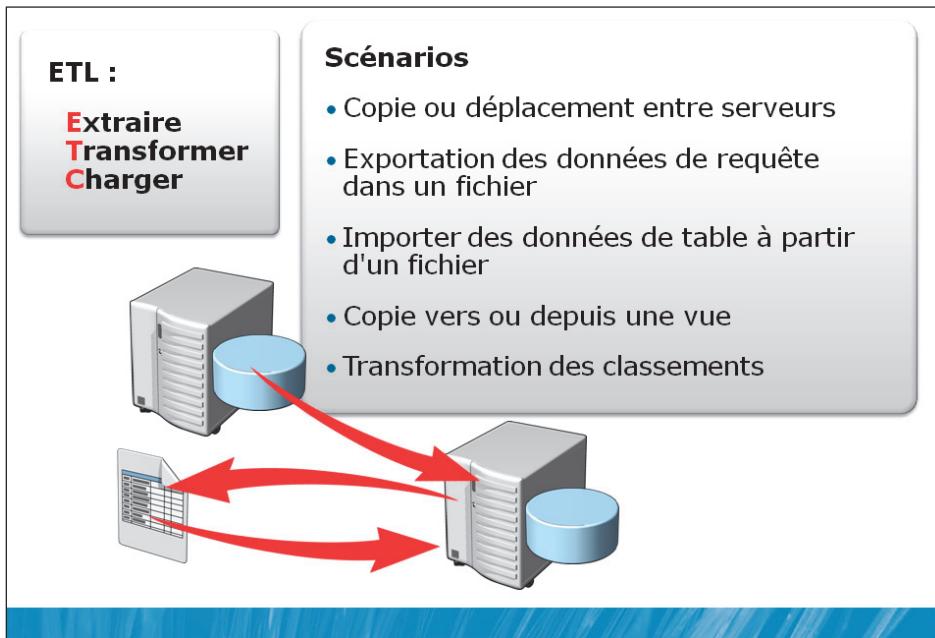
Lorsqu'il est nécessaire d'insérer de grandes quantités de données dans des tables SQL Server, il est peu probable que les paramètres par défaut des contraintes, des déclencheurs et des index produisent les meilleures performances possibles. Vous pouvez obtenir de meilleures performances en contrôlant le moment d'exécution des vérifications par les contraintes et en contrôlant le moment de l'actualisation des pages d'index d'une table.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- expliquer les concepts principaux du transfert de données ;
- décrire les outils fournis par SQL Server pour le transfert de données ;
- améliorer les performances du transfert de données ;
- supprimer et reconstruire les index ;
- désactiver et réactiver les contraintes.

## Présentation du transfert de données



### Points clés

Toutes les données ne peuvent pas être écrites ligne par ligne par les utilisateurs de la base de données. Bien souvent les données doivent être importées à partir de sources de données externes telles que d'autres serveurs de base de données, ou à partir de fichiers du système d'exploitation. Les utilisateurs demandent aussi souvent à ce que les données des tables de bases de données soient exportées vers des fichiers du système d'exploitation. Dans les modules précédents vous avez vu comment les classements peuvent provoquer des problèmes quand ils sont mal configurés. La correction du classement d'une base de données requiert souvent l'exportation et la ré-importation des données dans la base de données.

### Étapes du Transfert de données

Bien que toutes les spécifications du transfert de données ne soient pas identiques, il existe un processus standard à suivre dans le cadre de la plupart des tâches de transfert de données. Les principales étapes de ce processus sont :

- l'extraction des données à partir d'une source de données indiquée ;
- la transformation des données d'une manière adaptée au système cible ;
- le chargement des données dans le système cible.

Prises ensemble, ces trois étapes sont généralement appelées le processus ETL (Extract, Transform, Load). Les outils qui mettent en œuvre ces processus sont généralement désignés sous le nom d'outils ETL.



**Remarque** Dans certains cas, un processus ELT (Extract, Load, Transform) peut être plus adéquat. Vous pouvez par exemple décider d'effectuer la transformation des données après que les données ont été chargées dans le moteur de base de données, et non avant.

## Extraction de données

Même s'il existe d'autres options, l'extraction de données implique généralement l'exécution de requêtes sur un système source pour récupérer les données, ou l'ouverture et la lecture de fichiers du système d'exploitation. Une autre option impliquerait la requête des vues fournies par le système source.

Le processus d'extraction est accompagné de deux objectifs courants :

- éviter l'impact excessif sur le système source. Par exemple, ne pas lire les tables entières de données alors que vous ne devez lire que les lignes sélectionnées. Ne pas relire non plus continuellement les mêmes données et éviter l'exécution des instructions qui bloquent de quelque manière les utilisateurs du système source ;
- vérifier la cohérence de l'extraction de données. Par exemple, ne pas inclure de ligne du système source plusieurs fois dans la sortie de l'extraction.

## Transformation des données

La phase de transformation d'un processus ELT comporte généralement plusieurs étapes :

- les données doivent être nettoyées. Par exemple, vous devrez peut-être supprimer des données incorrectes ou fournir les valeurs par défaut des colonnes manquantes ;
- des recherches doivent être effectuées. Par exemple, les données d'entrée peuvent inclure le nom d'un client, mais la base de données peut avoir besoin d'un ID du client ;
- les données doivent être agrégées. Par exemple, les données d'entrée peuvent inclure chaque transaction survenue un jour donné, mais la base de données peut n'avoir besoin que de valeurs résumées quotidiennes ;
- les données doivent être désagrégées. Cela est souvent appelé l'allocation de données. Par exemple, les données d'entrée peuvent inclure les budgets trimestriels, mais la base de données peut avoir besoin de budgets quotidiens.

En plus de ces opérations courantes, les données peuvent devoir être restructurées d'une certaine façon. Il est courant de devoir appliquer aux données l'opérateur PIVOT ou UNPIVOT. Par exemple, les données d'entrée peuvent résider dans une table telle que la suivante :

Object	Attribute	Value
1	FirstName	David
1	Age	64
1	Gender	M
1	LastName	Pelton
2	FirstName	Erin
2	Age	47
2	Gender	F
2	LastName	Hagens

Mais il se peut que la base de données requière les mêmes données dans le format suivant :

PersonID	FirstName	Age	Gender	LastName
1	David	64	M	Pelton
2	Erin	47	F	Hagens

Cette transformation est un exemple de « pivotement » des données, où les lignes deviennent des colonnes et où les colonnes deviennent des lignes.

### Chargement des données

Une fois les données mises dans un format approprié, elles peuvent être chargées dans le système cible. Au lieu d'effectuer des opérations d'insertion ligne par ligne des données, il est possible d'utiliser des options spéciales de chargement des données en bloc. En outre, il est possible d'effectuer des modifications de configuration temporaires pour améliorer les performances de l'opération de chargement.

**Question :** quels autres types d'agrégation doivent être appliqués aux données pendant la phase de transformation ?

## Outils disponibles pour le transfert de données



### Points clés

SQL Server fournit un ensemble d'outils pour la réalisation des tâches de transfert de données. Il est important de savoir dans quelles circonstances utiliser ces outils.

#### Programme de copie en bloc (bcp)

Le programme de copie en bloc (bcp) peut être utilisé pour importer un grand nombre de nouvelles lignes à partir d'un fichier de données du système d'exploitation vers une table SQL Server, ou pour exporter des données d'une table SQL Server vers un fichier du système d'exploitation. Bien que l'utilitaire bcp puisse être utilisé avec une option queryout T-SQL, qui spécifie les lignes à exporter, l'utilisation standard du bcp n'implique pas de maîtrise du T-SQL.

#### BULK INSERT

L'instruction BULK INSERT est une commande T-SQL utilisée pour importer des données directement d'un fichier de données du système d'exploitation vers une table de base de données. L'instruction BULK INSERT diffère de plusieurs façons du programme bcp. En premier lieu, l'instruction BULK INSERT est exécutée à partir de T-SQL alors que l'utilitaire bcp est un utilitaire de ligne de commande. En outre, tandis que l'utilitaire bcp peut être utilisé aussi bien pour l'importation que pour la sortie, l'instruction BULK INSERT ne peut être utilisée que pour l'importation de données.

#### OPENROWSET (BULK)

OPENROWSET est une fonction table utilisée pour se connecter et récupérer des données dans les sources de données OLE DB. Il est nécessaire de fournir comme paramètres de la fonction OPENROWSET les détails complets de la méthode de connexion à la source de données. OPENROWSET peut être utilisé pour se connecter à d'autres types de moteur de base de données.

SQL Server offre un fournisseur OLE DB spécial appelé BULK qui peut être utilisé avec la fonction OPENROWSET. Le fournisseur BULK permet d'importer des documents entiers à partir du système de fichiers.

### Assistant Importation/Exportation

SQL Server Integration Services (SSIS) est un outil ETL fourni avec SQL Server. SSIS est en mesure de se connecter à une grande variété de sources de données et de destinations, et est capable d'effectuer des transformations complexes sur les données. SSIS assure de nombreuses tâches et transformations prêtes à l'emploi et peut également être étendu par l'emploi des composants personnalisés et des scripts .NET. SQL Server fournit également l'assistant Importation/Exportation, qui constitue une méthode simple pour créer des packages SSIS sans avoir à utiliser les outils de conception SSIS.

### Chargement en bloc XML

Le fournisseur de chargement en bloc XML peut être utilisé pour importer des données XML sous forme de flux binaire dans une instruction T-SQL. Les données peuvent être insérées directement dans une colonne d'une ligne existante d'une table de base de données.

**Question :** dans quel cas est-il préférable de choisir SSIS par rapport à bcp ?

## Amélioration des performances du transfert de données

- Désactiver les contraintes, les index et les déclencheurs
  - Il est inutile de vérifier les contraintes à chaque chargement de ligne
  - Les index n'ont pas besoin d'être mis à jour pendant l'importation
  - Il est important de vérifier les besoins de votre entreprise avant de désactiver les déclencheurs
- Réduction du verrouillage
  - Envisagez l'utilisation de TABLOCK pour accélérer l'importation
- Réduction de la journalisation
  - La base de données doit être en mode BULK\_LOGGED ou SIMPLE
  - Conditions supplémentaires liées à la structure et au verrouillage
- Réduction des conversions de données
  - Utilisez le format natif pour transférer des données entre des Serveurs SQL

### Points clés

Si les contraintes, les index et les déclencheurs sont activés sur les tables cibles des transferts de données, il convient de vérifier la valeur des données pour chaque ligne importée. Cette vérification permanente peut ralentir considérablement les transferts de données SQL Server.

### Désactivation des contraintes, des index et des déclencheurs

Plutôt que de vérifier chaque valeur importée ou de mettre à jour chaque index pour chaque ligne insérée, il existe un moyen d'atteindre de meilleures performances globales en désactivant le processus de vérification ou d'actualisation d'index jusqu'à ce que toutes les données soient chargées. Cette tâche de vérification peut alors être accomplie en une fois à la fin du processus d'importation.

Prenons par exemple le cas d'une contrainte de clé étrangère (FOREIGN KEY) utilisée pour vérifier, à chaque insertion d'une commande de client dans la base de données, que ce client existe réellement. Même s'il est possible de vérifier cette référence à chaque commande de client, envisageons que ce client présente des milliers de commandes. Au lieu de vérifier chaque valeur lors de son insertion, il est possible de vérifier la référence du client en une seule opération de recherche consécutive au processus global d'importation, afin de prendre en compte toutes les commandes qui font référence à ce client.

Seules les contraintes CHECK et FOREIGN KEY peuvent être désactivées. Le processus de désactivation et de réactivation des contraintes sera présenté ultérieurement dans cette leçon.

De même que le fait d'éviter les recherches pour les contraintes FOREIGN KEY pendant l'importation de données peut améliorer les performances, le fait d'éviter l'actualisation constante des index peut avoir le même effet. Dans de nombreux cas, la reconstruction des index une fois l'importation terminée est beaucoup plus rapide que la mise à jour des index à mesure que les lignes sont importées. Il y a une exception à cette situation, lorsque le nombre de lignes déjà dans la table est beaucoup plus grand que le nombre de lignes importées.

Les déclencheurs sont des commandes exécutées en cas de modification des données. Il est important de déterminer si le traitement réalisé par les déclencheurs serait mieux réalisé en bloc après l'importation, plutôt qu'à chaque insertion.

### Contrôle du comportement de verrouillage

Par défaut, SQL Server gère la granularité des verrous qu'il acquiert pendant l'exécution des commandes. SQL Server démarre avec un verrouillage au niveau ligne et ne tente une transmission que lorsqu'il y a un nombre significatif de lignes verrouillées dans une table. La gestion d'un grand nombre de verrous accapare des ressources qui pourraient servir à réduire le temps d'exécution des requêtes. Comme les données contenues dans les tables qui sont la cible des opérations d'importation en bloc ne sont normalement accessibles que par le processus d'importation des données, l'avantage du verrouillage au niveau de la ligne est rarement perceptible. Par conséquent, il peut être recommandé de verrouiller la table entière en utilisant un indicateur de requête TABLOCK pendant l'importation.

### Utiliser un enregistrement minimal chaque fois que c'est possible

Le fonctionnement du journal des transactions a été présenté dans le module 5. L'enregistrement minimal est une opération spéciale qui peut améliorer significativement la performance de certaines opérations telles que les importations en bloc. Tout en rendant plus rapides les opérations, l'enregistrement minimal permet d'éviter la croissance excessive du journal pendant les opérations d'importation massive.

Toutes les commandes ne peuvent pas utiliser l'enregistrement minimal. Bien qu'il ne s'agisse pas d'une liste exhaustive, les éléments ci-dessous illustrent les types de restrictions qui doivent être respectées pour appliquer l'enregistrement minimal :

- la table n'est pas répliquée ;
- le verrouillage de table est spécifié (à l'aide de TABLOCK) ;
- si la table ne possède pas d'index cluster mais un ou plusieurs index non-cluster, les pages de données sont toujours enregistrées de façon minimale. Le mode d'enregistrement des pages d'index dépend cependant du fait que la table est vide ou non ;
- si la table est vide, les pages d'index sont enregistrées de façon minimale ;
- si la table n'est pas vide, les pages d'index sont enregistrées entièrement ;
- si la table possède un index cluster et est vide, les données et les pages d'index sont enregistrées de façon minimale ;
- si une table possède un index cluster et n'est pas vide, les pages de données et les pages d'index sont toutes entièrement enregistrées, quel que soit le mode de récupération.



**Remarque** Les types d'index, y compris les index cluster et non-cluster, sont présentés au cours 10776A : Developing Microsoft SQL Server 2012 Databases (*en anglais*).

**Question** : quel serait le principal problème du journal des transactions si la journalisation complète se produisait au cours d'une opération d'importation en bloc ?

## Désactivation et reconstruction des index

- Désactivation d'un index
  - Interdit l'utilisateur d'accéder à l'index
  - Interdit l'accès aux données s'il s'agit d'un index cluster
  - Conserve la définition des index dans les métadonnées
  - Accélère l'importation de données dans les tables, car l'index n'est pas mis à jour pendant l'importation
- Activation d'un index
  - Reconstruit entièrement l'index
  - Facile à automatiser car les métadonnées sont toujours présentes
- L'activation et la désactivation d'index peuvent être utilisées au lieu de supprimer et de recréer des index

### Points clés

Avant SQL Server 2005, il était nécessaire de supprimer les index afin de les empêcher d'être mis à jour lors de l'actualisation des données de la table. Le problème de la suppression de l'index était que, lorsque vous deviez remettre l'index en place en le recréant, il vous fallait savoir avec exactitude comment il était configuré.

### Désactivation d'un index

Depuis SQL Server 2005 il existe une option pour désactiver un index. Au lieu de supprimer totalement les détails d'index de la base de données, la désactivation d'un index conserve ses métadonnées en place et empêche son actualisation. Les requêtes qui sont exécutés par des utilisateurs n'utilisent pas les index désactivés.

Le principal avantage de la désactivation d'un index par rapport à sa suppression réside dans le fait que l'index peut être remis en fonctionnement par une opération de reconstruction. Lorsque vous reconstruisez un index, vous n'avez pas besoin de connaître les détails de sa configuration. Cela simplifie beaucoup la création des scripts d'administration destinés à empêcher la mise à jour des index pendant les opérations massives d'importation ou d'actualisation ou destinés à remettre les index en service à la fin de ces opérations.

 **Remarque** Le type spécial d'index appelé index cluster correspond à la structure de la table plutôt qu'à un index distinct qui accélère la localisation des lignes dans une table. Si un index cluster est désactivé, la table devient inutilisable tant que l'index n'est pas reconstruit.

**Question** : quel est le principal avantage de la désactivation/activation des index par rapport à la suppression/reconstruction d'index pendant les importations en bloc ?

## Désactivation et activation des contraintes

- Désactivation des contraintes PRIMARY KEY et UNIQUE
  - Réalisée en désactivant l'index associé
  - Entraîne la reconstruction des index une fois activées
  - Peut engendrer des défaillances pendant la réactivation s'il existe des doublons
  - Entraîne également la désactivation des contraintes de clé étrangère associées
- Désactivation des contraintes FOREIGN KEY et CHECK
  - Peut être effectuée directement sur la contrainte
  - Entraîne la non-vérification des données existantes une fois réactivées
- Lorsque vous activez une contrainte FOREIGN KEY ou CHECK, les données existantes ne sont pas vérifiées par défaut

### Points clés

Les contraintes PRIMARY KEY définissent la ou les colonnes qui identifient de façon unique chaque ligne dans une table. Les contraintes UNIQUE vérifient qu'une ou des colonnes ne contiennent pas de valeurs dupliquées. SQL Server crée des index afin d'aider à appliquer ces contraintes.

### Désactiver les contraintes PRIMARY KEY ou UNIQUE

Pour désactiver une contrainte PRIMARY KEY ou UNIQUE, vous devez désactiver l'index qui y est associé. Cela n'est généralement utile qu'avec les contraintes non cluster PRIMARY KEY. Lorsque la contrainte est réactivée, les index associés sont reconstruits automatiquement. Si des valeurs dupliquées sont détectées lors de la reconstruction, cela cause l'échec de la réactivation de la contrainte. Par conséquent, si vous désactivez ces contraintes lors de l'importation de données, vous devez vous assurer que les données importées ne violeront pas les règles que les contraintes appliquent.



**Remarque** Si une table possède une clé primaire appliquée avec un index cluster, la désactivation de l'index associé à la contrainte empêcherait l'accès à toutes les données dans la table.

Les contraintes de clé étrangère (FOREIGN KEY) servent à garantir que les entités d'une table auxquelles font référence d'autres entités d'une autre table existent réellement. Par exemple, un fournisseur doit exister avant qu'il soit possible d'entrer un bon de commande à son nom. Les contraintes FOREIGN KEY utilisent des contraintes PRIMARY KEY ou UNIQUE lors de la vérification des références. Si vous désactivez la contrainte PRIMARY KEY ou UNIQUE à laquelle fait référence la contrainte FOREIGN KEY, celle-ci sera automatiquement désactivée. Toutefois, lors de la réactivation de la contrainte PRIMARY KEY ou UNIQUE, les références FOREIGN KEY qui utilisent ces contraintes ne seront pas automatiquement réactivées.

## Désactivation des contraintes FOREIGN KEY et CHECK

Les contraintes CHECK servent à limiter les valeurs qui peuvent être contenues dans une colonne, ou la relation entre les valeurs de plusieurs colonnes dans une table.

 **Remarque** Les contraintes sont présentées dans le cours 10776A : Developing Microsoft SQL Server 2012 Databases (*en anglais*).

Les contraintes FOREIGN KEY et CHECK peuvent être désactivées ou activées à l'aide des options CHECK et NOCHECK de l'instruction ALTER TABLE. Par exemple, considérons l'exemple de code suivant qui permet de désactiver la contrainte CHECK nommée SalaryCap sur une table appelée Person.Salary :

```
ALTER TABLE Person.Salary NOCHECK CONSTRAINT SalaryCap;
```

Le code suivant est utilisé pour réactiver la contrainte :

```
ALTER TABLE Person.Salary CHECK CONSTRAINT SalaryCap;
```

**Question** : pourquoi le référencement des contraintes de clé étrangère est-il désactivé lorsque les contraintes PRIMARY KEY ou UNIQUE sont désactivées ?

## Démonstration 1A : Désactivation et activation des contraintes

Dans cette démonstration, vous allez apprendre à désactiver et réactiver une contrainte de validation



### Procédure de démonstration

1. Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn**. Dans le menu **Fichier** cliquez sur **Ouvrir**, puis sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_08\_PRJ\22462A\_08\_PRJ.ssmssln**, puis cliquez sur **Ouvrir**.
3. Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** dans l'Explorateur de solutions.
4. Ouvrez le fichier de script **11 – Demonstration 1A.sql**.
5. Suivez les instructions contenues dans les commentaires du fichier de script.

## Leçon 2

# Importation et exportation des données de table

- Vue d'ensemble de SQL Server Integration Services
- Démonstration 2A : Utilisation de SSIS
- Assistant Importation/Exportation de SQL Server
- Démonstration 2B : Utilisation de l'Assistant Importation/Exportation

SQL Server Integration Services (SSIS) est un outil ETL puissant qui peut être utilisé conjointement avec SQL Server. SSIS est capable d'effectuer des transformations complexes sur des données issues d'une ou de plusieurs sources et de les charger vers une ou plusieurs destinations.

Bien que SSIS soit un outil très complexe et polyvalent, sa configuration pour les processus simples d'importation et d'exportation a été grandement facilitée par l'assistant Importation/Exportation. Cet assistant présente à l'utilisateur des boîtes de dialogue de configuration simples et crée un package SSIS en fonction des sélections qu'il a effectuées.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- décrire SQL Server Integration Services ;
- utiliser l'assistant Importation/Exportation de SQL Server.

## Vue d'ensemble de SQL Server Integration Services

- SSIS est une infrastructure puissante qui permet de développer des solutions ETL
- Les packages SSIS contiennent les éléments suivants
  - Sources de données et destinations
  - Contrôle et flux de données
  - Transformations à effectuer
- Les packages SSIS peuvent être exécutés avec
  - Les utilitaires de ligne de commande dtexec et dtexecui
  - Les travaux de l'Agent SQL Server
- Les packages SSIS sont développés avec
  - SQL Server Business Intelligence Development Studio (BIDS)
  - Assistant Importation/Exportation

### Points clés

SQL Server Integration Services (SSIS) est un outil ETL fourni avec SQL Server dans les éditions Standard et supérieures. SSIS permet la définition de flux et de transformations de données complexes. L'objectif principal de SSIS est de créer des packages réutilisables et facilement déployables qui effectuent des transferts de données.

### Packages

Le résultat de l'élaboration d'un jeu de tâches et de transformations dans SSIS s'appelle un package. Les packages sont également l'unité de déploiement de SSIS et contiennent un certain nombre d'objets : sources de données, destinations de données, flux de contrôle et flux de données.

Chaque package a un flux de contrôle qui contient un ensemble de tâches qui doivent être exécutées. Le flux de travail qui doit être suivi lors de l'exécution des tâches est défini par un ensemble de contraintes de priorité. Si les données doivent être déplacées à l'intérieur d'un package SSIS, une tâche de flux de données est ajoutée au jeu de tâches du flux de contrôle. Chaque tâche du flux de données est configurée individuellement afin de spécifier de quel endroit doivent provenir les données, comment elles doivent être transformées et où elles doivent être envoyées.

L'un des objectifs de SSIS consiste à exécuter toutes les étapes de transformation des données du processus ETL en une seule opération, sans créer de copie intermédiaire des données avant de les transformer.

Les packages sont créés à l'aide de SQL Server Data Tools (SSDT). L'assistant Importation/Exportation de SQL Server crée aussi les packages SSIS sans que les utilisateurs n'aient à se servir de SSDT.

**Question :** dans quels cas est-il utile d'employer SSIS plutôt que d'autres options de transfert de données ?

## Démonstration 2A : Utilisation de SSIS

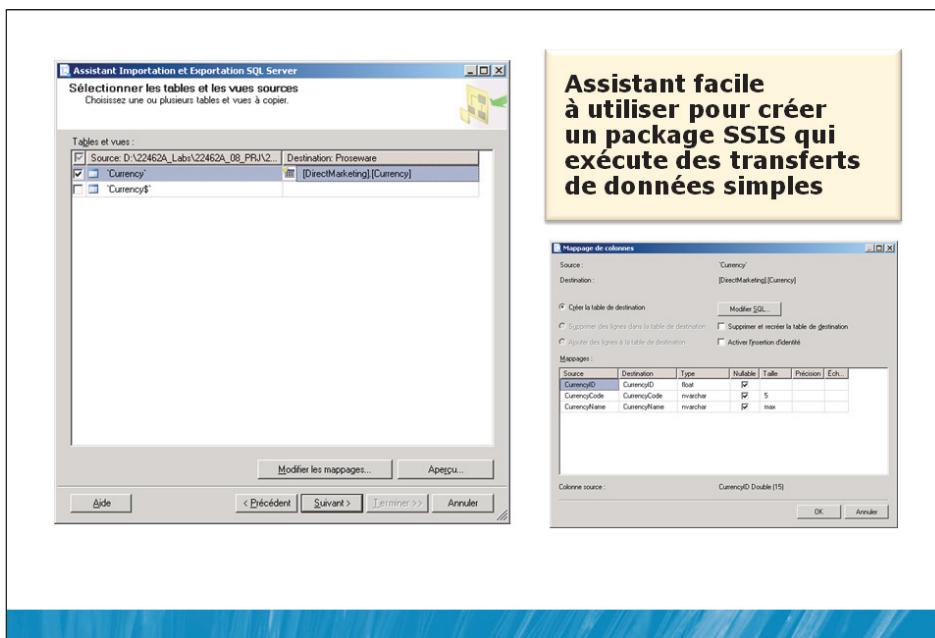
Dans cette démonstration vous allez apprendre à utiliser SQL Server Integration Services pour exporter des données d'une table vers un fichier plat



### Procédure de démonstration

1. Si la démonstration 1A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, ouvrez le projet de scripts du serveur **22462A\_08\_PRJ\_SQL** dans SQL Server Management Studio.
  - Ouvrez et exécutez le fichier de script **00 – Setup.sql** dans l'Explorateur de solutions.
2. Ouvrez le fichier de script **21 – Démonstration 2A.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

## Assistant Importation/Exportation de SQL Server



### Points clés

L'assistant Importation et exportation SQL Server peut copier des données vers et depuis n'importe quelle source de données pour laquelle un fournisseur de données .NET Framework managé ou un fournisseur OLE DB natif est disponible.

L'assistant présente quelques restrictions mais peut être utilisé avec SQL Server, des fichiers plats, des fichiers Microsoft Office Access® et Microsoft Office Excel® et de nombreux autres moteurs de bases de données.

Bien qu'il tire parti de SQL Server Integration Services, l'Assistant Importation et exportation SQL Server procure des fonctions de transformation minimales. À l'exception de la définition du nom, du type de données et des propriétés de type de données des colonnes des nouveaux fichiers et tables de destination, l'Assistant Importation et exportation SQL Server ne prend en charge aucune transformation de niveau colonne.

**Question :** si des transformations supplémentaires sont nécessaires en plus de ce qui est disponible avec l'Assistant Importation/Exportation, comment peuvent-elles être créées ?

## Démonstration 2B : Utilisation de l'Assistant Importation/Exportation

Dans cette démonstration, vous allez apprendre à utiliser l'Assistant Importation/Exportation pour exporter des données dans un fichier CSV



### Procédure de démonstration

1. Si la démonstration 1A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn.** Dans le menu **Fichier** cliquez sur **Ouvrir**, puis sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_08\_PRJ\22462A\_08\_PRJ.ssmssln**, puis cliquez sur **Ouvrir**.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** dans l'Explorateur de solutions.
2. Ouvrez le fichier de script **22 – Demonstration 2B.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

## Leçon 3

# Insertion de données en bloc

- Utilitaire bcp
- Démonstration 3A : Utilisation de bcp
- Instruction BULK INSERT
- Démonstration 3B : Utilisation de BULK INSERT
- Fonction OPENROWSET
- Démonstration 3C : Utilisation de OPENROWSET

Il existe trois options courantes qui sont utilisées pour traiter l'insertion de données en bloc avec SQL Server. L'utilitaire bcp peut être utilisé à partir de la ligne de commande pour exécuter aussi bien des opérations d'entrée que de sortie.

BULK INSERT est une commande T-SQL qui peut être utilisée pour importer des données en bloc et OPENROWSET est une fonction table qui récupère des données de diverses sources et les retourne sous forme d'une table qui peut être interrogée. OPENROWSET peut être utilisé conjointement avec une instruction SELECT... CATCH pour insérer des données en bloc.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- utiliser l'utilitaire bcp ;
- utiliser l'instruction BULK INSERT ;
- utiliser la fonction OPENROWSET.

## Utilitaire bcp

- Outil en ligne de commande qui sert à importer et exporter des données
- Utilise un fichier de format lors du transfert entre des instances SQL

### Création d'un format de fichier

```
bcp Adv.sales.Currency format nul -T -c -x -f cur.xml
```

### Exportation des données dans un fichier

```
bcp Adv.sales.Currency out cur.dat -T -c
```

### Importation de données à l'aide d'un fichier de format

```
bcp tempdb.sales.Currency2 in cur.dat -T -f cur.xml
```

## Points clés

L'utilitaire bcp sert à copier en bloc des données entre une instance de Microsoft SQL Server 2012 et un fichier de données dans un format spécifié par l'utilisateur. L'utilitaire bcp peut être utilisé pour importer un grand nombre de nouvelles lignes dans des tables SQL Server ou pour exporter des données de tables dans des fichiers de données. Sauf lorsqu'il est utilisé avec l'option queryout, l'utilitaire ne nécessite aucune connaissance de T-SQL.

## Fichiers de format

Pour importer des données dans une table, vous devez soit utiliser un fichier de format créé pour cette table, soit fournir ces informations à bcp de manière interactive. Deux types de fichiers de format sont pris en charge. Les versions actuelles de SQL Server utilisent des fichiers de format XML mais peuvent utiliser des fichiers de format texte plus ancien.

Vous pouvez utiliser l'utilitaire bcp pour créer un fichier de format exploitable par celui-ci. Dans le premier exemple de la diapositive, l'utilitaire bcp sert à créer un fichier de format selon la disposition en colonne de la table Adv.Sales.Currency.

Les paramètres clés spécifiés ont la signification suivante :

Paramètre	Description
format	Créer un format de fichier.
-T	La sécurité intégrée est utilisée pour se connecter au serveur.
-c	Le type de données caractère est utilisé pour l'exportation. Le type de données caractère assure la meilleure compatibilité entre différents types de systèmes. Une autre option –n utilise le format natif de SQL Server, qui est un format plus compact mais qui ne peut servir qu'à l'importation/exportation vers d'autres systèmes SQL Server.
-f	Nom du fichier de format.
-x	Le fichier de format doit être créé comme fichier XML.

Dans cet exemple, l'utilitaire bcp se connecterait à l'instance par défaut sur le serveur local. S'il est nécessaire de se connecter à une autre instance ou à un autre serveur, le paramètre – S permet de fournir un nom de serveur ou un nom de serveur et un nom d'instance.

### Exportation de données

Dans le deuxième exemple de la diapositive, l'utilitaire bcp sert à exporter le contenu actuel de la table Adv.Sales.Currency vers le fichier Cur.dat. Le seul paramètre qui diffère de l'exemple précédent est le paramètre « out » qui sert à indiquer le nom du fichier de sortie.

### Importation de données

Dans le troisième exemple de la diapositive, l'utilitaire bcp sert à importer le contenu d'un fichier Cur.dat vers la table tempdb.Sales.Currency2. Cet exemple illustre deux usages de paramètres intéressants. Le paramètre « in » sert à désigner le fichier qui doit être lu. Le paramètre « - f » sert à indiquer le fichier de format Cur.xml afin que SQL Server comprenne le format du fichier d'entrée.

**Question** : comment améliorer la vitesse d'importation d'une opération bcp ?

## Démonstration 3A : Utilisation de bcp

Dans cette démonstration, vous allez apprendre à importer un fichier à l'aide de l'utilitaire bcp



### Procédure de démonstration

1. Si la démonstration 1A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn.** Dans le menu **Fichier** cliquez sur **Ouvrir**, puis sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_08\_PRJ\22462A\_08\_PRJ.ssmssln**, puis cliquez sur **Ouvrir**.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** dans l'Explorateur de solutions.
2. Ouvrez le fichier de script **31 – Demonstration 3A.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

## Instruction BULK INSERT

- Fournit des options similaires à bcp
- S'exécute dans le processus SQL Server
- Options CHECK\_CONSTRAINTS et FIRE\_TRIGGERS associées
- Peut être exécutée dans une transaction définie par l'utilisateur

```
BULK INSERT Adventureworks.Sales.OrderDetail
    FROM 'f:\orders\neworders.txt'
    WITH
    (
        FIELDTERMINATOR = '|',
        ROWTERMINATOR = '|\\n'
    );
GO
```

### Points clés

BULK INSERT charge les données d'un fichier de données dans une table. Cette fonctionnalité est semblable à l'option « in » de la commande bcp ; toutefois, le fichier de données est lu par le processus SQL Server et non par un utilitaire externe. L'instruction BULK INSERT est exécutée dans un lot T-SQL. Étant donné que les fichiers de données sont ouverts par un processus SQL Server, les données ne sont pas copiées entre un processus client et les processus SQL Server. Par comparaison, l'utilitaire bcp s'exécute dans un processus distinct qui génère une plus forte charge sur le serveur lorsqu'il est exécuté sur le même système.

### Contraintes et déclencheurs

L'instruction BULK INSERT offre les options CHECK\_CONSTRAINTS et FIRE\_TRIGGERS qui peuvent être utilisées pour indiquer à SQL Server de vérifier les contraintes et les déclencheurs. Contrairement à l'utilitaire bcp, l'instruction BULK INSERT ne vérifie pas par défaut les contraintes CHECK et FOREIGN KEY, ni n'exécute les déclencheurs sur la table cible pendant les opérations d'importation.

Également à la différence de bcp, BULK INSERT peut être exécutée à partir d'une transaction définie par l'utilisateur, ce qui permet de regrouper BULK INSERT avec d'autres opérations dans une transaction unique. Il convient toutefois de prendre des précautions pour veiller à ce que la taille des lots de données qui sont importés dans une transaction unique n'est pas excessive ou pour éviter une croissance trop importante du fichier journal, même quand la base de données est en mode de récupération simple.

**Question :** quelle est la différence entre l'instruction BULK INSERT et l'utilitaire bcp ?

## Démonstration 3B : Utilisation de BULK INSERT

Dans cette démonstration, vous allez apprendre à importer un fichier à l'aide de la commande BULK INSERT

### Procédure de démonstration

1. Si la démonstration 1A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn.** Dans le menu **Fichier** cliquez sur **Ouvrir**, puis sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_08\_PRJ\22462A\_08\_PRJ.ssmssln**, puis cliquez sur **Ouvrir**.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** dans l'Explorateur de solutions.
2. Ouvrez le fichier de script **32 – Demonstration 3B.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

**Question :** pourquoi le premier message indique-t-il 199 et les messages suivants 200 ?

## Fonction OPENROWSET

- Permet d'accéder aux données à distance via une connexion à une source de données distante à l'aide d'un fournisseur OLE DB
- Offre un fournisseur BULK intégré pour les importations en bloc à partir de fichiers
  - Peut être utilisé dans une clause FROM avec la fonctionnalité intégrale
  - Peut utiliser INSERT .. SELECT pour insérer des données
  - Propose des indicateurs de table pour l'opération de contrôle

```
SELECT *
  FROM OPENROWSET(
    BULK 'c:\mssql\export.csv',
    FORMATFILE = 'c:\mssql\format.fmt',
    FIRSTROW = 2) AS a;
```

```
GO
```

### Points clés

OPENROWSET est utilisable pour accéder aux données à l'aide d'un fournisseur OLE DB. Pour que les fournisseurs OLE DB soient utilisables dans OPENROWSET, il faut activer une option de configuration du système « Utilisation de requêtes distribuées ad hoc », ainsi que définir explicitement une entrée de registre pour le fournisseur OLE DB, appelée DisallowAdhocAccess, à la valeur 0. Cette clé de Registre se trouve généralement ici : HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\MSSQLServer\Fournisseurs\MSDASQL.

Lorsque ces options ne sont pas définies, le comportement par défaut n'autorise pas l'accès ad hoc requis par la fonction OPENROWSET lorsque vous utilisez les fournisseurs externes OLE DB.

### Fournisseur BULK

Depuis SQL Server 2005, il existe un fournisseur OLE DB spécial appelé BULK qui vous permet de spécifier les fichiers du système d'exploitation qui seront retournés. Vous pouvez utiliser avec ce fournisseur les mêmes fichiers de format qu'avec bcp et BULK INSERT.

Outre l'importation des lignes de données classiques, le fournisseur BULK offre trois options spéciales qui lui permettent de lire le contenu entier du fichier dans une seule colonne d'une table. Ces options spéciales sont les suivantes :

Option	Description
SINGLE_CLOB	Lit un fichier texte entier codé sur un seul octet comme une seule valeur du type varchar(max).
SINGLE_NCLOB	Lit un fichier texte entier codé sur deux octets comme une seule valeur du type nvarchar(max).
SINGLE_BLOB	Lit un fichier binaire entier comme une seule valeur de type varbinary(max).

Prenons comme exemple l'extrait de code suivant :

```
INSERT INTO Sales.Documents(FileName, FileType, Document)
SELECT 'JanuarySales.txt' AS FileName,
      '.txt' AS FileType,
      *
FROM OPENROWSET(BULK N'K:\JanuarySales.txt', SINGLE_BLOB) AS Document;
```

Dans cet exemple de code, le fichier K:\JanuarySales.txt est inséré dans la colonne Document de la table Sales.Documents, ainsi que son nom et son type de fichier.

Par rapport à bcp, OPENROWSET comporte deux avantages : elle peut être utilisée dans une requête avec une clause WHERE (pour filtrer les lignes chargées), et elle peut être utilisée dans une instruction SELECT qui n'est pas nécessairement associée à une instruction INSERT.

**Question** : dans quel cas est-il préférable d'utiliser OPENROWSET au lieu de bcp ou BULK INSERT ?

## Démonstration 3C : Utilisation de OPENROWSET

Dans cette démonstration, vous allez apprendre à importer un fichier avec l'instruction OPENROWSET

### Procédure de démonstration

1. Si la démonstration 1A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn**. Dans le menu **Fichier** cliquez sur **Ouvrir**, puis sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_08\_PRJ\22462A\_08\_PRJ.ssmssln**, puis cliquez sur **Ouvrir**.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** dans l'Explorateur de solutions.
2. Ouvrez le fichier de script **33 – Demonstration 3C.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

## Atelier pratique 8 : Importation et exportation de données

- Exercice 1 : Importer la feuille de calcul Excel
- Exercice 2 : Importer le fichier CSV
- Exercice 3 : Créer et tester un package d'extraction
- Exercice difficile 4 : Comparer les performances de chargement (si le temps le permet)

Informations d'ouverture de session

Ordinateur virtuel	22462A-MIA-SQL1
Nom d'utilisateur	AdventureWorks\Administrateur
Mot de passe	Pa\$\$w0rd

Durée approximative : 45 minutes



### Configuration de l'atelier pratique

Pour cet atelier pratique, vous utiliserez l'environnement d'ordinateur virtuel disponible. Avant de commencer cet atelier pratique, vous devez procéder aux étapes suivantes :

1. Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
3. Dans la fenêtre Se connecter au serveur, tapez **Proseware** dans la zone de texte **Nom du serveur**.
4. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn.**
5. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**.
6. Dans la fenêtre Ouvrir un projet, sélectionnez le projet **D:\22462A\_Labs\22462A\_08\_PRJ\22462A\_08\_PRJ.ssmssqln**.
7. Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Dans l'Explorateur de solutions, double-cliquez sur le fichier de requête **00 - Setup.sql**. Lorsque la fenêtre de requête s'affiche, cliquez sur **Exécuter** dans la barre d'outils.

### Scénario de l'atelier pratique

Proseware reçoit régulièrement des mises à jour de devises et de taux de change d'un fournisseur externe. Un de ces fichiers est fourni sous la forme d'une feuille de calcul Excel, l'autre fichier est disponible en tant que fichier texte délimité par des virgules. Vous devez importer ces deux fichiers dans des tables qui seront utilisées par l'équipe de marketing direct de Proseware.

Périodiquement, l'équipe marketing a besoin d'une liste de prospects qui n'ont pas été contactés au cours du mois précédent. Vous devez créer et tester un package qui permet d'extraire ces informations pour eux dans un fichier.

Vous êtes préoccupé par la performance de l'importation du fichier de taux de change et vous envisagez de désactiver les contraintes et les index sur la table du taux de change pendant l'importation. Si vous en avez le temps, vous allez en tester la différence sur les performances de l'importation.

### Documentation

#### Exercice 1

Élément	Description
Table de sortie	DirectMarketing.Currency
Colonnes de sortie	int CurrencyID non null CurrencyCode nvarchar(3) non null CurrencyName nvarchar(50) non null
Fichier d'entrée	D:\22462A_Labs\22462A_08_PRJ\22462A_08_PRJ\Currency.xls

## Exercice 1 : Importer la feuille de calcul Excel

### Scénario

Vous devez charger un fichier de codes et de noms de devises à partir d'une feuille de calcul Excel. Dans cet exercice, vous allez utiliser l'assistant Importation pour effectuer le chargement des données.

La tâche principale de cet exercice est la suivante :

1. Importer les données à l'aide de l'assistant Importation.

#### ► Tâche 1 : Importer les données à l'aide de l'assistant Importation

- Importez la feuille de calcul Currency.xls dans une table de la base de données MarketDev appelée DirectMarketing.Currency. Si la table existe déjà, supprimez-la d'abord. Reportez-vous à la documentation pour connaître l'emplacement du fichier et le format de la table de sortie.
- Interrogez la table DirectMarketing.Currency pour vérifier que les données ont été chargées.

**Résultats :** À la fin de cet exercice, vous devez avoir importé la table DirectMarketing.Currency.

## Exercice 2 : Importer le fichier CSV

### Scénario

Vous avez également reçu un fichier CSV contenant des taux de change. Vous devez importer ces taux de change dans la table existante DirectMarketing.ExchangeRate. La table doit être tronquée avant de charger les données.

La tâche principale de cet exercice est la suivante :

1. Importer le fichier CSV.



**Remarque** Veillez à enregistrer la durée que prend la commande pour s'exécuter.

Avertissement : elle peut prendre quelques minutes. Exploitez ce temps en préparant l'exercice suivant.

### ► Tâche 1 : Importer le fichier CSV

- Truncez la table DirectMarketing.ExchangeRate.
- Examinez le fichier de format ExchangeRates.xml dans l'Explorateur de solutions.
- À l'aide de la commande T-SQL BULK INSERT, importez le fichier ExchangeRates.csv dans la table DirectMarketing.ExchangeRate. Le fichier ExchangeRates.csv se trouve à l'emplacement suivant : D:\22462A\_Labs\22462A\_08\_PRJ\22462A\_08\_PRJ\ExchangeRates.csv.
- Utilisez ExchangeRates.xml comme fichier de format et une taille de lot de 10 000, et utiliser l'option pour ignorer la première ligne car elle contient des en-têtes.

**Résultats** : À la fin de cet exercice, vous devez avoir importé la table ExchangeRate à l'aide de l'instruction T-SQL BULK INSERT.

## Exercice 3 : Créer et tester un package d'extraction

### Scénario

Périodiquement, l'équipe marketing a besoin d'une liste de prospects qui n'ont pas été contactés au cours du mois précédent. Vous devez créer et tester un package qui permet d'extraire ces informations pour eux dans un fichier.

La tâche principale de cet exercice est la suivante :

1. Créer et tester un package d'extraction.

#### ► Tâche 1 : Créer et tester un package d'extraction

- Utilisez l'assistant Exportation, exporter la table Marketing.Prospect vers un fichier texte à l'emplacement suivant : D:\MKTG\ProspectsToContact.csv. Les noms de colonnes doivent être contenus dans la première ligne. La requête d'extraction doit être telle qu'indiquée dans l'extrait de code ci-dessous :

```
SELECT ProspectID, FirstName, LastName, CellPhoneNumber,  
WorkPhoneNumber, EmailAddress, LatestContact  
FROM Marketing.Prospect  
WHERE LatestContact < DATEADD(MONTH, -1, SYSDATETIME())  
OR LatestContact IS NULL  
ORDER BY ProspectID;
```

 **Remarque** Enregistrez le package SSIS créé par l'assistant Exportation dans SQL Server à l'emplacement racine du package.

**Résultats** : À la fin de cet exercice, vous devez avoir créé et testé un package d'extraction.

## Exercice difficile 4 : Comparer les performances de chargement (si le temps le permet)

### Scénario

Vous êtes préoccupé par la performance de l'importation du fichier de taux de change et vous envisagez de désactiver les contraintes et les index sur la table du taux de change pendant l'importation. Si le temps le permet, vous allez tester la différence dans les performances des opérations d'importation.

La tâche principale de cet exercice est la suivante :

1. Ré-exécuter le chargement, les index étant désactivés.

#### ► Tâche 1 : Ré-exécuter le chargement, les index étant désactivés

- Modifiez le script de l'exercice 2 pour désactiver tous les index non cluster de la table DirectMarketing.ExchangeRate avant le chargement des données et pour reconstruire les index après que le chargement est terminé.
- Exécutez votre script modifié et comparer la durée qu'il a prise à la valeur consignée depuis l'exercice 2.

**Résultats** : À la fin de cet exercice, vous devez avoir comparé les performances de chargement obtenues avec les index activés et désactivés.

## Contrôle des acquis et éléments à retenir

- Questions de contrôle des acquis
- Meilleures pratiques

### Questions de contrôle des acquis

1. Dans quel cas devez-vous utiliser SSIS à la place d'autres utilitaires de transfert de données ?
2. Pourquoi les opérations enregistrées de façon minimale sont-elles plus rapides que les opérations enregistrées entièrement ?

### Meilleures pratiques liées à un domaine technologique particulier dans ce module

1. Choisir l'outil approprié pour les importations en bloc.
2. Utiliser SSIS pour les transformations complexes.
3. Utiliser le programme bcp ou la commande BULK INSERT pour les importations et les exportations rapides.
4. Utiliser OPENROWSET lorsque les données doivent être filtrées avant d'être insérées.
5. Effectuer une journalisation minimale pour accélérer l'importation des données.



# Module 9

## Authentification et autorisation des utilisateurs

### Sommaire :

Leçon 1 : Authentification des connexions à SQL Server	9-3
Leçon 2 : Autorisation des connexions pour accéder aux bases de données	9-13
Leçon 3 : Autorisation sur plusieurs serveurs	9-23
Atelier pratique 9 : Authentification et autorisation des utilisateurs	9-31

## Vue d'ensemble du module

- Authentification des connexions à SQL Server
- Autorisation des connexions pour accéder aux bases de données
- Autorisation sur plusieurs serveurs

La sécurisation de Microsoft® SQL Server® peut être vue comme une série d'étapes impliquant quatre domaines : la plateforme, l'authentification, les objets (y compris les données) et les applications qui accèdent au système. Une sécurité bien planifiée est importante pour protéger les informations sensibles de votre organisation.

Dans ce module, vous allez découvrir comment sécuriser SQL Server. Vous apprendrez comment SQL Server authentifie les utilisateurs et les autorise à accéder aux bases de données. Toutes les ressources ne résident pas sur le même serveur. Dans ce module, vous verrez également comment est configurée l'autorisation distribuée, c'est-à-dire les situations qui impliquent plusieurs systèmes SQL Server.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- décrire comment SQL Server authentifie les connexions ;
- décrire comment les connexions sont autorisées à accéder aux bases de données ;
- expliquer les conditions requises par les autorisations sur plusieurs serveurs.

## Leçon 1

# Authentification des connexions à SQL Server

- Vue d'ensemble de la sécurité SQL Server
- Options d'authentification SQL Server
- Gestion des connexions Windows
- Gestion des connexions et des stratégies SQL Server
- Démonstration 1A : Authentification des connexions et des jetons de connexion

La première couche de sécurité dans SQL Server consiste en l'authentification des utilisateurs. SQL Server vérifie l'identité de l'utilisateur avant d'examiner n'importe quel autre paramètre de sécurité.

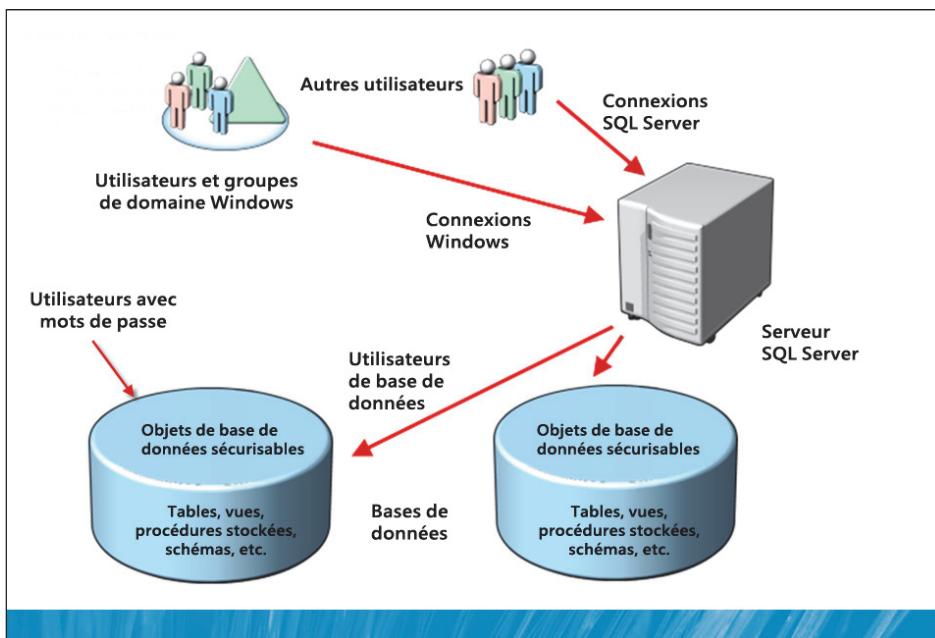
L'authentification est le processus d'identification des utilisateurs et de vérification de leur identité. Dans cette leçon, vous découvrirez les options disponibles pour l'authentification des utilisateurs, les différents types de connexions au serveur ainsi que la façon dont les stratégies sont gérées par rapport aux connexions SQL Server.

## Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- décrire la sécurité de SQL Server ;
- mettre en œuvre les options d'authentification de SQL Server ;
- autoriser l'accès à SQL Server pour des utilisateurs et les groupes Windows® ;
- autoriser l'accès à SQL Server pour d'autres utilisateurs.

## Vue d'ensemble de la sécurité SQL Server



### Points clés

Un utilisateur peut être autorisé à se connecter à SQL Server de quatre manières différentes :

- il peut s'agir d'un utilisateur que SQL Server gère directement. Ces utilisateurs désignent des connexions SQL Server ;
- il peut s'agir d'un utilisateur que Windows authentifié et SQL Server a été configuré pour autoriser l'utilisateur à se connecter. Ces utilisateurs désignent des connexions Windows ;
- il peut s'agir d'un utilisateur Windows qui est membre d'un groupe Windows. SQL Server a été configuré pour permettre aux membres du groupe de se connecter. Ces utilisateurs désignent également des connexions Windows ;
- il peut s'agir d'un utilisateur de base de données à qui un mot de passe a été affecté directement, lorsque l'utilisateur n'est pas associé à une connexion.



**Remarque** Il est également possible que les connexions soient créées à partir de certificats et de clés, mais ceci est un sujet avancé qui n'entre pas dans le cadre de ce cours.

## Connexions et utilisateurs de base de données

La distinction entre les connexions et les utilisateurs de base de données est importante. Le terme « connexion » est appliqué à un principal (ou processus) qui a été autorisé à accéder à SQL Server via l'une des trois méthodes indiquées ci-dessous.

 **Remarque** Avoir accès au serveur n'indique pas en soi qu'une connexion est autorisée à accéder à n'importe quelle base de données du serveur.

Les connexions peuvent être autorisées à accéder à une ou plusieurs bases de données. Il peut exister un mappage entre une connexion (connexion Windows ou connexion SQL Server) et « un utilisateur de base de données » dans une base de données particulière. Les utilisateurs de base de données possèdent souvent le même nom que les connexions auxquelles ils sont mappés, mais les noms peuvent être différents. Une connexion peut même être mappée à des noms d'utilisateur différents dans chaque base de données.

 **Remarque** Il est recommandé de conserver les mêmes noms pour la connexion et l'utilisateur de base de données.

## Autorisations

Même une fois qu'une connexion est autorisée à accéder à une base de données par la création d'un utilisateur de base de données, celui-ci devra se voir accorder des autorisations dans la base de données avant de pouvoir accéder aux objets sécurisables tels que les tables, les vues, les fonctions et les procédures stockées.

 **Remarque** Certaines connexions disposent d'autorisations pour toutes les bases de données car elles se sont vu attribuer des rôles au niveau serveur, par exemple le rôle sysadmin, mais ce sont des exceptions à la règle générale.

Il existe deux façons pour accorder aux utilisateurs de base de données l'autorisation d'accéder aux objets sécurisables. Ils peuvent se voir accorder des autorisations sur les objets. Sinon, il est possible de créer des rôles dans la base de données. Les rôles disposent des autorisations sur les objets sécurisables et les utilisateurs de base de données sont ajoutés en tant que membres de ces rôles. Les utilisateurs de base de données héritent toutes les autorisations associées à tous les rôles dont ils sont membres, ainsi que de toutes les autorisations qui leur ont été affectées directement.

**Question** : outre les utilisateurs Windows, quels autres types d'utilisateurs peuvent être amenés à se connecter à SQL Server ?

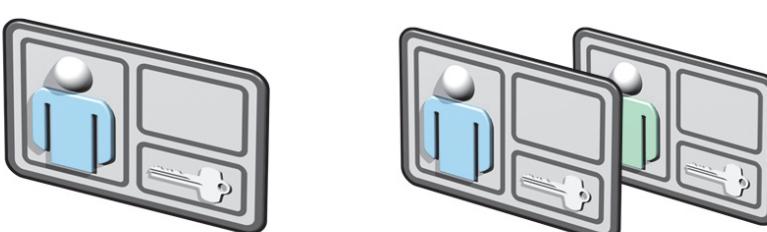
## Options d'authentification SQL Server

**Authentification = processus de vérification d'une identité**

**SQL Server peut être configuré selon deux modes**

**Authentification Windows**

**Authentification SQL Server et Windows**



The diagram illustrates the two modes of SQL Server authentication. On the left, a single user icon is shown in a window labeled 'Authentification Windows'. On the right, two windows are shown: one for 'Authentification Windows' showing a user icon and a key icon, and another for 'Authentification SQL Server et Windows' showing a user icon and a key icon.

### Points clés

L'authentification est le processus de vérification de la validité d'une identité. Si l'identité correspond à un utilisateur, celui-ci est-il bien celui qu'il prétend ? Cela peut se réaliser de deux façons élémentaires :

- SQL Server fait confiance au système d'exploitation Windows pour vérifier l'identité d'une connexion Windows. Windows peut utiliser différentes méthodes pour valider l'identité d'un utilisateur, comme la vérification d'un mot de passe, des contrôles biométriques (par exemple, l'empreinte digitale) ou des certificats. Windows peut même utiliser une combinaison de ces méthodes. Dans SQL Server, les connexions Windows sont souvent désignées sous le nom de « connexions approuvées » ;
- SQL Server peut vérifier directement l'identité d'une connexion SQL Server en vérifiant le mot de passe associé à cette connexion.

**Question :** si vousappelez votre banque au téléphone, comment vérifient-ils votre identité avant de vous communiquer les informations sur votre compte ?

### Configuration du serveur

SQL Server peut être configuré selon deux modes :

- le mode d'authentification Windows ;
- le mode d'authentification Windows et SQL Server.

Le mode d'authentification Windows était auparavant appelé le mode intégré. Dans le mode d'authentification Windows, seuls les utilisateurs que le système d'exploitation Windows a authentifiés sont autorisés à se connecter au serveur.

Dans le mode d'authentification Windows et SQL Server, aussi bien les utilisateurs qui ont été authentifiés par le système d'exploitation Windows et les utilisateurs directement authentifiés par SQL Server sont autorisés à se connecter au serveur. Ce mode est souvent appelé le mode mixte.

Le choix entre ces deux modes s'effectue au niveau de l'instance de SQL Server pendant l'installation de SQL Server et peut facilement être modifié. L'instance doit être redémarrée une fois la configuration modifiée.

 **Remarque** La configuration peut être créée à l'aide des outils d'interface utilisateur graphique dans SSMS, mais il s'agit en fait d'une seule clé de registre qui est modifiée. Cette clé de registre peut être définie via une stratégie de groupe dans Windows.

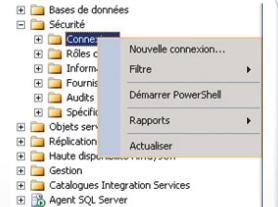
Si le mode d'authentification Windows et SQL Server est activé, une connexion SQL Server appelée « sa » est alors active. Il est important de définir un mot de passe approprié (et complexe) pour la connexion « sa » et de le stocker quelque part. Lorsque SQL Server est installé comme authentification Windows uniquement, le compte « sa » est désactivé par défaut. Si SQL Server est installé en mode mixte, le compte « sa » est activé par défaut. Modifier le mode d'authentification du serveur ne modifie pas l'état activé/désactivé de la connexion « sa ».

### Protocoles d'authentification

L'authentification Windows est généralement effectuée via le protocole Kerberos. Le protocole Kerberos est pris en charge par SQL Server sur le protocole réseau TCP/IP, les canaux nommés et les protocoles réseau de mémoire partagée. (La prise en charge de Kerberos sur les canaux nommés et la mémoire partagée a été introduite dans SQL Server 2008. SQL Server 2005 prenait en charge Kerberos sur le protocole réseau TCP/IP).

## Gestion des connexions Windows

- Créer des connexions Windows avec
  - Option Explorateur d'objets dans SSMS
  - Instruction T-SQL CREATE LOGIN
- Supprimer avec DROP LOGIN
  - Retourne une erreur si l'utilisateur est déjà connecté



```
CREATE LOGIN [Adventureworks\Student]
```

```
FROM WINDOWS
```

```
WITH DEFAULT_DATABASE=[tempdb],  
      DEFAULT_LANGUAGE=[us_english];
```

```
GO
```

```
CREATE LOGIN [Adventureworks\salespeople]
```

```
FROM WINDOWS;
```

```
GO
```

### Points clés

Les connexions peuvent être créées directement à l'aide de code T-SQL ou via l'interface utilisateur graphique fournie dans SSMS. Bien que l'interface utilisateur graphique soit utilisable, la création de connexions est une opération courante et, s'il faut en créer beaucoup, vous trouverez l'utilisation d'un script bien plus rapide et plus facile à répéter.

### CREATE LOGIN

Les connexions peuvent être créées de deux manières. Pour créer une connexion dans SSMS, développez le nœud sécurité au niveau de l'instance de serveur et cliquez avec le bouton droit sur Connexions, puis choisissez Nouvelle connexion.

Vous pouvez sinon créer des connexions au moyen de l'instruction CREATE LOGIN, comme illustré dans l'exemple de la diapositive.



**Remarque** Les noms d'utilisateur Windows et de groupe doivent être placés entre crochets comme illustré dans l'exemple de la diapositive. La raison principale en est qu'ils contiennent une barre oblique inverse.

Les connexions Windows peuvent être créées pour des utilisateurs ou des groupes Windows. Le premier exemple de la diapositive présente la création d'une connexion pour un utilisateur individuel. Notez qu'une base de données et une langue par défaut sont affectées.

Lorsqu'une base de données par défaut n'est pas affectée directement, la base de données master sera affectée par SQL Server. Lorsqu'une langue par défaut n'est pas affectée directement, la langue par défaut de l'instance de serveur sera affectée par SQL Server.

## Groupes Windows

Le deuxième exemple de la diapositive concerne un groupe Windows. Les membres du groupe AdventureWorks\Salespeople seront autorisés à se connecter au serveur sans devoir disposer de connexions individuelles.

 **Remarque** Les utilisateurs et les groupes Windows peuvent faire référence à des utilisateurs ou à des groupes locaux ou de domaine.

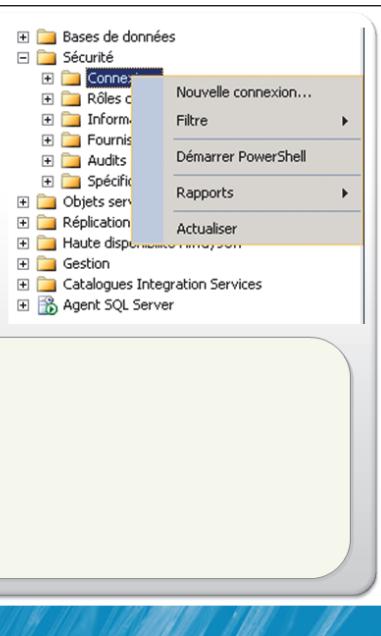
## Suppression des connexions

Les connexions sont supprimées à l'aide de l'instruction DROP LOGIN. Notez que vous ne pouvez pas supprimer une connexion si l'utilisateur est connecté. Si vous êtes amené à le faire, vous devez rechercher l'ID de session dans le moniteur d'activité de SSMS (en affichant la liste des processus) et supprimer cette session en premier lieu. Évitez de supprimer des connexions lorsque les utilisateurs de base de données qui y sont associés sont propriétaires d'objets dans les bases de données. (La propriété d'objet est présentée dans le module 11).

**Question :** dans quel cas pouvez-vous être amené à créer des connexions en fonction des groupes plutôt que des connexions en fonction des utilisateurs ?

## Gestion des connexions et des stratégies SQL Server

- Créez des connexions SQL Server avec
  - Option Explorateur d'objets dans SSMS
  - Instruction T-SQL CREATE LOGIN
- Déterminez si la stratégie de compte doit s'appliquer
- Utilisez ALTER LOGIN pour
  - Réinitialiser des mots de passe
  - Désactiver/activer les connexions



```
CREATE LOGIN SalesManager
WITH PASSWORD = 'Pa$$wOrd',
CHECK_POLICY = ON;
GO
CREATE LOGIN SalesApp
WITH PASSWORD = 'Pa$$wOrd',
CHECK_POLICY = OFF;
GO
```

### Points clés

Les connexions SQL Server sont créées pour des identités individuelles au moyen des mêmes outils qui servent à créer les connexions Windows.

### Sécurité des connexions SQL Server

L'utilisation de connexions SQL Server a été considérée pendant longtemps comme une pratique de sécurité médiocre. Cela découle de plusieurs raisons, notamment :

- l'authentification non chiffrée ;
- une lacune en matière de stratégie de compte.

Le problème du chiffrement était lié au fait que, pendant la phase d'authentification, le trafic sur le réseau n'était pas chiffré. Cela signifiait qu'un intrus qui aurait intercepté des paquets réseau aurait pu capter suffisamment d'informations pour accéder au système via une connexion SQL Server.

Le problème avec la stratégie de compte était que, même si le système SQL Server faisait partie d'un domaine ayant implémenté une stratégie de compte détaillée, celle-ci ne s'appliquait pas aux connexions SQL Server. Ces deux problèmes ont été traités dans SQL Server 2005.

## Authentification chiffrée

Le client d'accès natif (SNAC) mis à niveau de SQL Server qui était fourni avec SQL Server 2005 a été amélioré pour assurer une authentification chiffrée. Si SQL Server n'avait pas de certificat SSL (Secure Sockets Layer) installé par un administrateur, SQL Server produirait et signerait automatiquement un certificat à utiliser pour chiffrer le trafic d'authentification.

 **Remarque** L'authentification chiffrée ne s'appliquait qu'aux clients exécutant la version SQL Server 2005 de SNAC ou une version ultérieure. Si un client antérieur qui ne comprenait pas l'authentification chiffrée tentait de se connecter, SQL Server l'autorisait par défaut. Il existe une propriété configurable pour chaque protocole pris en charge dans le gestionnaire de configuration SQL Server qui peut être utilisée pour interdire l'authentification non chiffrée des clients de bas niveau en cas de problème.

## Stratégie de comptes

La mise en œuvre d'une stratégie de compte est basée sur la capacité à lire les détails de la stratégie du système d'exploitation. Windows Server 2003 et les systèmes d'exploitation ultérieurs ont introduit une API qui autorisait les applications à lire les détails de la stratégie de compte.

SQL Server est pris en charge sur certains systèmes d'exploitation (tels que Windows XP) qui ne prennent pas en charge cette API. Pour ces systèmes d'exploitation, la stratégie de compte est remplacée par un ensemble de règles de complexité des mots de passe.

L'application complète de la stratégie de compte n'est pas toujours souhaitable. Par exemple, certaines applications utilisent des connexions fixes pour se connecter au serveur. Bien souvent ces applications ne prennent pas en charge la modification régulière des mots de passe de connexion. Dans ce cas, il est courant de désactiver la stratégie de vérification de ces connexions.

## Modification du mot de passe et expiration de la connexion

Il est possible de réinitialiser les mots de passe à l'aide de l'interface utilisateur graphique de SSMS ou via l'instruction ALTER LOGIN. Si les connexions ne sont pas utilisées pendant un certain temps, elles peuvent être désactivées puis réactivées ultérieurement. Si une connexion a des chances d'être utile dans l'avenir, il est préférable de la désactiver plutôt que de la supprimer. La désactivation de la connexion s'effectue en exécutant l'instruction ALTER LOGIN, comme indiqué dans le code ci-dessous :

```
ALTER LOGIN James DISABLE;
```

**Question** : pouvez-vous nommer un type de stratégie de compte Windows ?

## Démonstration 1A : Authentification des connexions et des jetons de connexion

Dans cette démonstration, vous allez apprendre à

- Créer une connexion Windows
- Afficher la liste des connexions existantes
- Créer une connexion SQL Server avec T-SQL
- Vous connecter à SQL Server
- Vérifier les jetons de connexion disponibles
- Créer une connexion SQL Server avec l'interface graphique utilisateur
- Créer une connexion avec la stratégie désactivé
- Afficher les connexions SQL Server existantes et l'état de contrôle de leurs stratégies et expiration

### Procédure de démonstration

1. Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn.** Dans le menu **Fichier** cliquez sur **Ouvrir**, puis sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_09\_PRJ\22462A\_09\_PRJ.ssmssln**, puis cliquez sur **Ouvrir**.
3. Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** dans l'Explorateur de solutions.
4. Ouvrez le fichier de script **11 – Demonstration 1A.sql**.
5. Suivez les instructions contenues dans les commentaires du fichier de script.

## Leçon 2

# Autorisation des connexions pour accéder aux bases de données

- Vue d'ensemble des autorisations
- Octroi de droits d'accès aux bases de données
- Gestion des accès dbo et invité
- Démonstration 2A : Autorisation des connexions et des jetons d'utilisateur
- Création d'utilisateurs pour une base de données spécifique
- Démonstration 2B : Configuration des utilisateurs avec mots de passe

Une fois que l'identité d'une connexion a été vérifiée, il est nécessaire de doter cette connexion de l'accès aux bases de données dont elle a besoin. Cet accès est fourni en accordant à la connexion l'accès à une base de données, puis en la dotant des autorisations suffisantes dans la base de données pour réaliser le travail demandé.

Dans cette leçon, vous allez voir comment accorder à une connexion l'accès à une base de données, et vous découvrirez en détail des formes spéciales d'accès aux bases de données.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- décrire le processus d'autorisation de SQL Server ;
- accorder l'accès aux bases de données ;
- gérer l'accès invité et dbo ;
- configurer les utilisateurs avec des mots de passe pour les bases de données à relation contenant-contenu partielle.

## Vue d'ensemble des autorisations

- On confond souvent authentification et autorisation
- **Authentification**
  - Vérification de l'identité d'un principal (savoir qui est cette personne)
- **Autorisation**
  - Attribution d'autorisations sur un élément sécurisable à un principal (décider ce qu'un utilisateur est autorisé à effectuer)
  - Peut être implémentée en associant un principal à un rôle qui possède déjà des autorisations
  - Implémentée avec les instructions GRANT, DENY, REVOKE des autorisations sur des objets de base de données

### Points clés

Une erreur très courante dans le domaine de la sécurité consiste à confondre les concepts d'authentification et d'autorisation. L'authentification consiste à donner la preuve de qui vous êtes, et l'autorisation consiste à déterminer ce que vous avez le droit de faire.

### Terminologie formelle

Il est important de bien connaître la terminologie formelle courante relative à la sécurité telle qu'elle est présentée dans le tableau suivant :

Terme	Description
Principal	Une entité qui peut demander l'accès à une ressource.
Élément sécurisable	Une ressource qui peut être sécurisée, ou un objet dont vous pouvez contrôler l'accès.
Authentification	Le fait de vérifier que le principal est bien qui il déclare être.
Autorisation	Le fait d'accorder à un principal un accès contrôlé à un élément sécurisable. Autrement dit, cela détermine les autorisations qu'a un principal relativement à un élément sécurisable.
Rôle	Un conteneur pour les principaux qui peuvent être utilisés pour affecter des autorisations indirectement.

Un rôle est un nom fourni pour un groupe de principaux. Les rôles sont créés pour faciliter l'accord des autorisations à plusieurs principaux qui ont besoin de niveaux d'accès similaires. Les rôles dans SQL Server sont semblables aux groupes du système d'exploitation Windows.

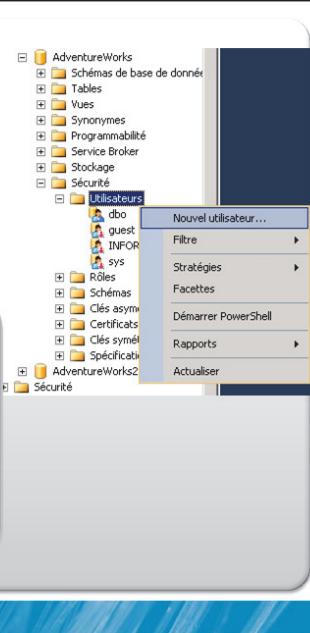
Les autorisations sont contrôlées par les commandes GRANT, REVOKE et DENY. Elles sont présentées dans le module 11.

**Question :** d'après vous, une connexion est-elle un principal ou un élément sécurisable ?

## Octroi de droits d'accès aux bases de données

- Les connexions accèdent aux bases de données via la création d'utilisateurs de base de données
- Créer des utilisateurs de base de données via
  - Option Explorateur d'objets dans SSMS
  - Instruction T-SQL CREATE LOGIN

```
CREATE USER SecureUser
FOR LOGIN SecureUser;
GO
CREATE USER Student
FOR LOGIN [Adventureworks\student];
GO
CREATE USER HRApp
FOR LOGIN HRUser;
GO
```



### Points clés

Les connexions se voient accorder l'accès aux bases de données via la création d'utilisateurs de base de données. Un utilisateur de base de données est un principal dans une base de données qui est mappée sur une connexion au serveur.

Les utilisateurs de base de données peuvent être créés via l'interface utilisateur graphique de SSMS ou via les commandes T-SQL. Pour créer un nouvel utilisateur de base de données via l'interface utilisateur graphique, développez la base de données appropriée, développez le nœud Sécurité, cliquez avec le bouton droit sur le nœud Utilisateurs et choisissez l'option Nouvel utilisateur.



**Remarque** Il y a des types particuliers d'utilisateurs de base de données qui sont créés directement à partir de certificats et ne sont pas associés à des connexions. Leur création est un sujet avancé qui n'entre pas dans le cadre de ce cours.

## CREATE USER/DROP USER

Une fois qu'une connexion existe, elle peut être liée à un nouvel utilisateur de base de données avec l'instruction CREATE USER de T-SQL. L'instruction DROP USER est utilisée pour supprimer ces utilisateurs.

Considérez les exemples présentés sur la diapositive :

- Le premier exemple crée un utilisateur de base de données nommé SecureUser pour une connexion SQL Server existante, également nommée SecureUser ;
- Le second exemple crée un utilisateur de base de données nommé Stagiaire pour une connexion Windows nommée AdventureWorks\Stagiaire ;
- Le troisième exemple crée un utilisateur de base de données HRApp pour la connexion SQL Server nommée HRUser. Notez qu'on a donné à l'utilisateur de base de données un nom différent de celui de la connexion.



**Remarque** Les noms des connexions Windows (d'utilisateurs ou de groupes) doivent être placés entre crochets.

## Gestion des accès dbo et invité

- **dbo**
  - La connexion sa et les membres du rôle sysadmin sont mappés au compte dbo, ainsi que le propriétaire de la base de données
- **Invité**
  - Le mappage à un compte d'utilisateur d'une base de données autorise les connexions sans compte d'utilisateur pour accéder à une base de données
  - Désactivé par défaut dans les bases de données utilisateur
  - Autorisation GRANT CONNECT d'accès au compte invité pour l'activer

### Points clés

Chaque base de données SQL Server comporte deux utilisateurs de base de données spécifiques : dbo et invité.

#### dbo

dbo est un utilisateur spécial qui dispose d'autorisations implicites pour effectuer toutes les activités dans la base de données. Tout membre du rôle serveur fixe sysadmin (y compris l'utilisateur « sa ») qui utilise une base de données est mappé sur l'utilisateur de base de données spécial nommé dbo à l'intérieur de chaque base de données. L'utilisateur de base de données dbo ne peut pas être supprimé et est toujours présent dans chaque base de données.

#### Propriété de la base de données

À l'instar des autres objets dans SQL Server, les bases de données également ont des propriétaires. Le propriétaire d'une base de données est également mappé à l'utilisateur dbo. Le propriétaire de la base de données peut être modifié à l'aide de l'instruction ALTER AUTHORIZATION, comme le montre le code suivant :

```
ALTER AUTHORIZATION ON DATABASE::MarketDev
    TO [ADVENTUREWORKS\Administrator];
```

Tout objet créé par un membre du rôle serveur fixe sysadmin appartient automatiquement à dbo. Les propriétaires des objets ont un accès complet à ceux-ci et n'ont pas besoin d'autorisations explicites pour pouvoir effectuer des opérations dessus.

## Invité

Le compte d'utilisateur invité autorise les connexions qui ne sont pas mappées à un utilisateur de base de données dans une base de données particulière à accéder à celle-ci. Les comptes de connexion assument l'identité de l'utilisateur invité lorsque les conditions suivantes sont satisfaites :

- la connexion peut accéder à SQL Server mais n'a pas accès à la base de données par l'intermédiaire de son propre mappage au compte d'utilisateur ;
- le compte invité a été activé.

Le compte invité peut être ajouté à une base de données pour permettre à quiconque doté d'une connexion valide d'accéder à la base de données. Le nom d'utilisateur invité est automatiquement un membre du rôle public. (Les rôles seront présentés dans le module suivant).

Un utilisateur invité fonctionne de la manière suivante :

- SQL Server vérifie si la connexion est mappée à un utilisateur de base de données dans la base de données à laquelle la connexion tente d'accéder. Si tel est le cas, SQL Server accorde à la connexion l'accès à la base de données en tant qu'utilisateur de la base de données ;
- SQL Server vérifie si un utilisateur de base de données invité est activé. Si tel est le cas, l'accès à la base de données est accordé à la connexion en tant qu'invité. Si le compte invité n'existe pas ou n'est pas activé, SQL Server refuse l'accès à la base de données.

L'utilisateur invité ne peut pas être supprimé, mais vous pouvez l'empêcher d'accéder à une base de données en exécutant la commande suivante :

```
REVOKE CONNECT FROM guest;
```

Le compte invité peut être activé en exécutant la commande suivante :

```
GRANT CONNECT TO guest;
```

 **Remarque** L'utilisateur invité est utilisé pour permettre l'accès aux bases de données master, msdb et tempdb. Il est déconseillé de tenter de révoquer l'accès invité à ces bases de données.

**Question** : quelle est l'utilité de l'utilisateur invité ?

## Démonstration 2A : Autorisation des connexions et des jetons d'utilisateur

Dans cette démonstration, vous allez apprendre à

- Créer des utilisateurs de base de données avec T-SQL
- Créer des utilisateurs de base de données avec l'interface utilisateur graphique
- Afficher les principaux des bases de données
- Afficher les jetons des utilisateurs

### Procédure de démonstration

1. Si la démonstration 1A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn.** Dans le menu **Fichier** cliquez sur **Ouvrir**, puis sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_09\_PRJ\22462A\_09\_PRJ.ssmssln**, puis cliquez sur **Ouvrir**.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** dans l'Explorateur de solutions.
  - Ouvrez et exécutez le fichier de script **11 – Demonstration 1A.sql** à partir de l'Explorateur de solutions.
2. Ouvrez le fichier de script **21 – Demonstration 2A.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

## Création d'utilisateurs pour une base de données spécifique

- Utilisateurs qui s'authentifient sur la base de données
  - Uniquement autorisés dans une base de données à relation contenant-contenu (au moins partiellement)
- Trois types courants
  - Utilisateur basé sur un utilisateur Windows sans connexion
  - Utilisateur basé sur un groupe Windows sans connexion
  - Utilisateur de base de données à relation contenant-contenu avec un mot de passe

### Points clés

Jusqu'à SQL Server 2012, pour qu'un utilisateur puisse se connecter à une base de données, il devait d'abord se voir créer une connexion à la base de données master du serveur.

Il est à présent possible de créer des utilisateurs dans une base de données sans devoir créer d'entrées pour les utilisateurs dans la base de données master. Une base de données doit être à relation contenant-contenu partielle avant de pouvoir être utilisée par les utilisateurs sans connexion. La relation contenant-contenu est une propriété d'une base de données.

Il existe trois types d'utilisateurs qui peuvent être authentifiés dans la base de données plutôt que par le serveur :

- un utilisateur Windows peut être authentifié au niveau de la base de données en fonction de ses informations d'identification d'utilisateur Windows ;
- les membres d'un groupe Windows peuvent être authentifiés au niveau de la base de données ;
- les utilisateurs de SQL Server peuvent avoir un mot de passe affecté et être authentifiés au niveau de la base de données, sans avoir besoin d'une connexion SQL Server.

Les utilisateurs qui sont authentifiés par la base de données ne peuvent pas accéder à d'autres bases de données, sauf en tant qu'utilisateurs invités.

## Démonstration 2B : Configuration des utilisateurs avec mots de passe

Dans cette démonstration, vous allez apprendre à créer des utilisateurs de base de données avec mots de passe

### Procédure de démonstration

1. Si la démonstration 2A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn**. Dans le menu **Fichier** cliquez sur **Ouvrir**, puis sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_09\_PRJ\22462A\_09\_PRJ.ssmssln**, puis cliquez sur **Ouvrir**.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** dans l'Explorateur de solutions.
  - Ouvrez et exécutez le fichier de script **21 – Demonstration 2A.sql** à partir de l'Explorateur de solutions.
2. Ouvrez le fichier de script **22 – Demonstration 2B.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

## Leçon 3

# Autorisation sur plusieurs serveurs

- Problème d'authentification mutuelle classique
- Emprunt d'identité ou délégation
- Utilisation des ID de sécurité incompatibles
- Démonstration 3A : Utilisation des SID incompatibles

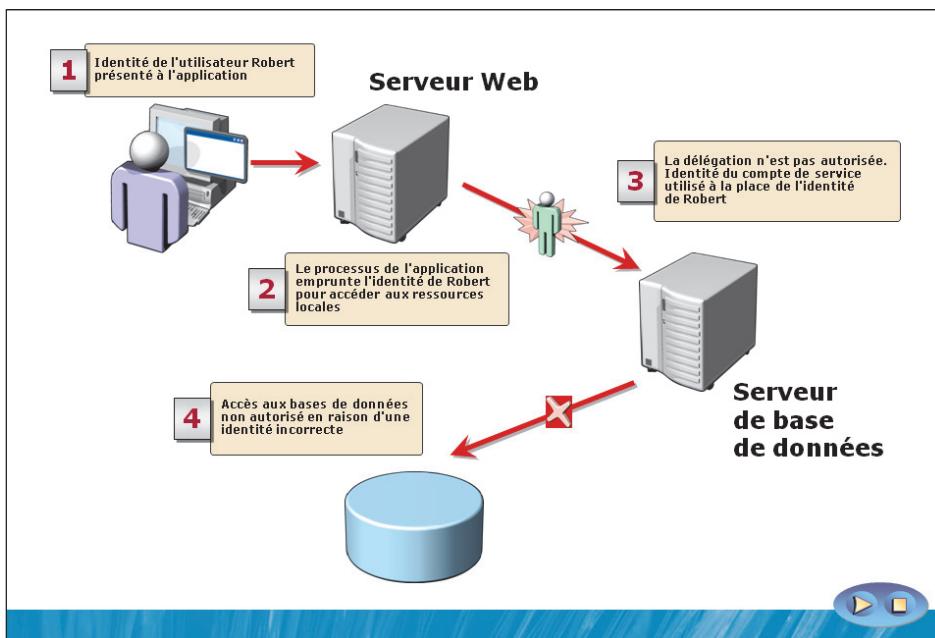
Toutes les ressources ne résident pas sur le même serveur. Il est courant de devoir accéder à des ressources qui résident sur d'autres serveurs. Deux problèmes peuvent survenir couramment lorsque l'on travaille sur plusieurs serveurs. L'un est désigné comme le problème d'authentification mutuelle, qui est lié à la différence entre l'emprunt d'identité et la délégation. Ces concepts vont vous être expliqués dans cette leçon. Le deuxième problème courant est désigné comme le problème de « SID incompatibles » qui se produit lors de la restauration ou de l'attachement d'une base de données à partir d'un autre serveur qui utilisait des connexions SQL Server. Vous verrez également dans cette leçon des méthodes pour traiter ce problème.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- décrire un problème d'authentification mutuelle classique ;
- expliquer la différence entre l'emprunt d'identité et la délégation ;
- utiliser des ID de sécurité.

## Problème d'authentification mutuelle classique



### Points clés

Il existe un problème très courant qui survient lorsque vous travaillez sur plusieurs serveurs.

Ce problème se présente comme indiqué sur la diapositive :

1. Un utilisateur lance une application. Cela peut entraîner le démarrage d'une application Web à partir d'un serveur Web d'entreprise. Cette application connecte l'utilisateur sous une autre identité Windows ou, plus fréquemment, emprunte l'identité de l'utilisateur.
2. À ce stade, l'application peut effectuer des tâches comme si l'utilisateur les réalisait directement. Le processus serveur Web peut être exécuté sous un compte à faibles priviléges, mais il remplit ses fonctions comme s'il était l'utilisateur Windows. Par exemple, l'utilisateur pourrait accéder aux fonctions de l'entreprise au sein de l'application, en fonction de son identité. Un directeur de banque pourrait accéder à la fonctionnalité « Suppression de compte bancaire » alors qu'un guichetier se verrait refuser cet accès.
3. L'application doit se connecter à une base de données pour récupérer des données de l'entreprise. Si le serveur de base de données est installé sur le même serveur que le serveur Web, l'accès au serveur de base de données s'effectue en utilisant les mêmes informations d'emprunt d'identité. Dans SQL Server, l'utilisateur aura toujours l'identité Windows appropriée. L'utilisateur aura les autorisations qui lui auront été directement affectées ainsi que les autorisations qui auront été attribuées aux rôles dont il est membre.
4. Si, en revanche, le serveur de base de données se trouve sur un autre serveur, cela pose un problème. Lorsque l'identité de l'utilisateur dans SQL Server est vérifiée, il se trouve qu'il s'agit de l'identité du service du serveur Web, comme le compte à faibles priviléges, au lieu d'être l'identité de l'utilisateur d'origine.

Il s'agit d'un problème très courant d'authentification mutuelle qui est causé par l'action par défaut du système d'exploitation Windows qui permet l'emprunt d'identité mais pas la délégation. On demande souvent aux administrateurs de base de données pourquoi cela se produit. Si ce type d'accès était autorisé par défaut entre les serveurs, cela ouvrirait une faille de sécurité importante.

Ce qui est souvent source de confusion est que l'application qui fonctionnait comme prévu dans un environnement de développement où le serveur Web et le serveur de base de données étaient tous deux sur le même serveur ne fonctionne plus une fois qu'elle est déployée dans un environnement de production où ce n'est plus le cas.

Dans la rubrique suivante, vous verrez la différence entre l'emprunt d'identité et la délégation.

**Question :** si vous avez rencontré ce problème avec un serveur Web, quel était le compte d'utilisateur qui tente souvent de se connecter à SQL Server à la place de l'utilisateur ?

## Emprunt d'identité ou délégation

- On confond souvent les deux termes
- Emprunt d'identité
  - Capacité à agir au nom d'un autre utilisateur sur l'ordinateur local
- Délégation
  - Capacité à agir au nom d'un autre utilisateur sur le réseau

### Points clés

Par défaut, lorsqu'un utilisateur se connecte au serveur Windows, un emprunt d'identité de l'utilisateur est effectué sur ce serveur. Toutefois, cela ne donne aucun droit à un processus sur le serveur d'emprunter l'identité de cet utilisateur sur le réseau. Cet emprunt d'identité sur un réseau est appelée délégation. Cela s'applique à tous les serveurs Windows qui se connectent à un SQL Server et également à un système SQL Server qui se connecte à un autre système SQL Server.

### Contraintes liées à la délégation

Pour illustrer les contraintes liées à la délégation entre deux systèmes SQL Server, prenons comme exemple le scénario suivant :

- un utilisateur se connecte à un ordinateur client qui se connecte à un serveur en train d'exécuter une instance de SQL Server, SQLSERVER1 ;
- l'utilisateur souhaite exécuter une requête distribuée impliquant une base de données située sur un autre serveur, SQLSERVER2 ;
- ce scénario, dans lequel un ordinateur se connecte à un autre ordinateur pour se connecter à un troisième ordinateur, est un exemple d'authentification mutuelle.

Chaque serveur ou ordinateur impliqué dans la délégation doit être configuré correctement.

### Conditions requises pour le client

- La connexion authentifiée Windows de l'utilisateur doit être habilitée à accéder à SQLSERVER1 et à SQLSERVER2.
- La propriété Active Directory de l'utilisateur, « Le compte est sensible et ne peut pas être délégué », ne doit pas être sélectionnée.
- L'ordinateur client doit utiliser la connectivité réseau TCP/IP ou des canaux nommés.

### Conditions requises pour le premier serveur (intermédiaire) (SQLSERVER1)

- Le serveur doit avoir un nom du principal du serveur (SPN) enregistré par l'administrateur de domaine.
  - Le compte avec lequel SQL Server s'exécute doit être approuvé pour la délégation.
  - Le serveur doit utiliser la connectivité réseau TCP/IP ou des canaux nommés.
  - Le second serveur, SQLSERVER2, doit être ajouté en tant que serveur lié. Cela peut être accompli en exécutant la procédure stockée sp\_addlinkedserver ou par des configurations réalisées dans SSMS.
- Par exemple :

```
EXEC sp_addlinkedserver 'SQLSERVER2', N'SQL Server'
```

- Les connexions au serveur lié doivent être configurées de manière à prendre en charge le mappage automatique. Pour ce faire, vous pouvez exécuter la procédure stockée sp\_addlinkedsrvlogin.
- Par exemple :

```
EXEC sp_addlinkedsrvlogin 'SQLSERVER2', 'true'
```

### Conditions requises pour le second serveur (SQLSERVER2)

- Si vous utilisez la connectivité réseau TCP/IP, le serveur doit avoir un SPN inscrit par l'administrateur de domaine.
- Le serveur doit utiliser la connectivité réseau TCP/IP ou des canaux nommés.

## Utilisation des ID de sécurité incompatibles

- Connexions et utilisateurs de base de données
  - Ils portent tous les deux un nom
  - Tous les deux ont des identificateurs de sécurité appelés SID
- Par défaut, les connexions SQL Server reçoivent de nouveaux SID une fois créées
  - Se produit même si le nom et le mot de passe sont identiques
  - SID des connexions Windows fournies par Windows
- Un problème fréquent survient lors de la restauration ou du rattachement des bases de données à partir d'une autre instance
- Deux méthodes pour résoudre le problème
  - Spécifiez le SID lors de la création de la connexion (meilleure option)
  - Corrigez le SID de connexion dans la base de données après la restauration

### Points clés

Il existe un autre problème très courant lié à l'utilisation de plusieurs serveurs ; il s'agit du problème des « SID incompatibles ».

#### SID incompatibles

Lorsqu'une connexion SQL Server est créée, un nom et un ID de sécurité (SID) lui sont alloués. Lorsqu'un utilisateur de base de données est créé pour la connexion, les détails du nom et du SID de la connexion sont entrés dans la base de données.

Si la base de données est sauvegardée puis restaurée sur un autre serveur, l'entrée d'utilisateur de base de données y est toujours présente, mais il n'existe sur le serveur aucune connexion qui y corresponde.

Les administrateurs de base de données créent souvent une nouvelle connexion et la mappent en tant qu'utilisateur dans la base de données. Ils constatent alors que cette opération échoue. Lorsque la connexion est créée, elle peut porter le même nom et avoir même le même mot de passe que la connexion d'origine sur l'autre serveur, mais par défaut SQL Server lui alloue un nouveau SID.

Il est alors impossible d'ajouter la nouvelle connexion à la base de données. Cela peut constituer une source d'agacement, car l'erreur renournée indique que l'utilisateur de la base de données existe déjà, alors que la consultation dans SSMS de la liste des mappages d'utilisateur de base de données pour cette connexion n'indique pas cette entrée. Voici un exemple intéressant d'une situation où, pour l'administration de SQL Server, il est bon de comprendre le code T-SQL plutôt que de comprendre seulement l'utilisation de l'interface utilisateur graphique dans SSMS.

## Résolution du problème des SID incompatibles

Dans les versions antérieures de SQL Server, l'option fournie pour gérer cette situation était une procédure système stockée, `sp_change_users_login`.

Depuis le service pack 2 de SQL Server 2005, une nouvelle solution est proposée :

```
ALTER USER dbuser WITH LOGIN = loginname;
```

Le problème inhérent à l'une ou l'autre de ces méthodes est qu'elles « corrigent » le SID de l'utilisateur de base de données pour qu'il corresponde au SID de la connexion. Le même problème surviendra encore, la prochaine fois que la base de données sera restaurée (ce qui arrive souvent).

## Prévention du problème

Une meilleure façon de traiter les SID incompatibles consiste à éviter que le problème ne se présente.

L'instruction `CREATE LOGIN` propose l'option `WITH SID`. Si vous spécifiez le SID du serveur d'origine lors de la création de la connexion sur le deuxième serveur, vous éviterez complètement le problème. La procédure stockée de `sp_helpprevlogin` est une autre option qui permet de faciliter l'écriture de scripts de connexions SQL Server et elle contient la valeur du SID. Les connexions créées à l'aide de scripts générés par cette procédure ne subiront pas le problème des SID incompatibles. Les détails de la procédure `sp_helpprevlogin` (ainsi que le code source de la procédure) sont disponibles sur le site [Web support.microsoft.com](http://Web.support.microsoft.com).

 **Remarque** Bien que SQL Server Integration Services inclue une tâche de transfert de connexions, cet outil désactive les connexions et leur affecte un mot de passe aléatoire.

Vous verrez un exemple du problème des SID incompatibles dans la Démonstration 3A.

## Démonstration 3A : Utilisation des SID incompatibles

Dans cette démonstration, vous allez apprendre à résoudre un problème avec des SID incompatibles

### Procédure de démonstration

1. Si la démonstration 1A ou 2A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn.** Dans le menu **Fichier** cliquez sur **Ouvrir**, puis sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_09\_PRJ\22462A\_09\_PRJ.ssmssln**, puis cliquez sur **Ouvrir**.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez les fichiers de script **00 – Setup.sql**, **11 – Demonstration 1A.sql**, et **21 – Demonstration 2A.sql** à partir de l'Explorateur de solutions.
2. Ouvrez le fichier de script **31 - Demonstration 3A.sql** et suivez les instructions contenues dans les commentaires du fichier de script.

## Atelier pratique 9 : Authentification et autorisation des utilisateurs

- Exercice 1 : Créer des connexions
- Exercice 2 : Résoudre un problème de connexion d'application
- Exercice 3 : Créer des utilisateurs de base de données
- Exercice difficile 4 : Corriger l'accès à la base de données restaurée (si le temps le permet)

Informations d'ouverture de session

Ordinateur virtuel	22462A-MIA-SQL1
Nom d'utilisateur	AdventureWorks\Administrateur
Mot de passe	Pa\$\$w0rd

Durée approximative : 45 minutes

### Configuration de l'atelier pratique

Pour cet atelier pratique, vous utiliserez l'environnement d'ordinateur virtuel disponible.

Avant de commencer cet atelier pratique, vous devez procéder aux étapes suivantes :

1. Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
3. Dans la fenêtre Se connecter au serveur, tapez **Proseware** dans la zone de texte **Nom du serveur**.
4. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Connecter**.
5. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**.
6. Dans la fenêtre Ouvrir un projet, sélectionnez le projet **D:\22462A\_Labs\22462A\_09\_PRJ\22462A\_09\_PRJ.ssmssln**.
7. Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Dans l'Explorateur de solutions, double-cliquez sur le fichier de requête **00 - Setup.sql**. Lorsque la fenêtre de requête s'affiche, cliquez sur **Exécuter** dans la barre d'outils.

## Scénario de l'atelier pratique

Vous devez configurer la sécurité de la base de données Marketing avant que l'entreprise n'ait accès au système. Vous devez configurer la sécurité afin que les utilisateurs de l'entreprise puissent se connecter à SQL Server mais ne puissent accéder qu'aux ressources auxquelles ils sont autorisés. La plupart des utilisateurs se connecteront à l'aide de leurs informations d'identification de groupe Windows. Certains utilisateurs devront néanmoins utiliser des connexions Windows individuelles. Une application requiert l'utilisation d'une connexion SQL Server.

Si vous avez le temps, la base de données LanguageDetails pose problème et vous devriez essayer de le résoudre.



**Remarque** Les modifications que vous apportez sont migrées par la suite vers l'environnement de production. Vous devez utiliser les commandes T-SQL pour implémenter les modifications nécessaires.

## Documentation

### Utilisateur Windows existant et structure de groupe

	ITSupport	SalesPeople	CreditManagement	HumanResources	CorporateManagers
David.Alexander	X				X
Jeff.Hay	X				
Palle.Petersen	X				
Terry.Adams	X				
Darren.Parker		X			X
Mike.Ray		X			
April.Reagan		X			
Jamie.Reding		X			
Darcy.Jayne		X			
Naoki.Sato		X			
Bjorn.Rettig			X		X
Don.Richardson			X		
Wendy.Kahn			X		
Neil.Black				X	X
Madeleine.Kelly				X	

## Configuration de la sécurité existante

- La connexion SQL PromoteApp a été créée.

## Configuration requise pour la sécurité

Remarque : cette liste de spécifications de sécurité s'applique à plusieurs modules. Dans le cadre du présent module, vous ne devez prendre en compte que les spécifications qui peuvent être satisfaites par les sujets abordés dans le module et dans les tâches de l'atelier pratique.

1. L'administrateur de base de données senior Jeff Hay doit disposer d'un accès et d'un contrôle complets sur l'instance de serveur Proseware.
2. Tous les membres du groupe ITSupport doivent bénéficier d'un accès et d'un contrôle complets sur la base de données MarketDev.
3. Proseware utilise une application appelée DBMonitor de Trey Research. Cette application exige une connexion SQL nommée DBMonitorApp qui requiert une capacité à lire tous les objets de la base de données MarketDev, mais pas une capacité à les mettre à jour. Elle ne requiert pas l'accès à d'autres bases de données.
4. Tous les membres du groupe CorporateManagers effectuent périodiquement une analyse FFPM (Forces, Faiblesses, Possibilités, Menaces). Pour cela ils doivent pouvoir à la fois lire et mettre à jour les lignes de la table DirectMarketing.Competitor.
5. Tous les membres du groupe SalesPeople doivent pouvoir afficher les données de toutes les tables dans le schéma DirectMarketing, sauf April Reagan qui est une stagiaire.
6. Seuls les membres du groupe ITSupport et les membres du groupe CreditManagement doivent pouvoir mettre à jour la table Marketing.CampaignBalance directement.
7. Les membres de l'entreprise appartenant au groupe SalesPeople, au groupe CreditManagement et au groupe CorporateManagers sont appelés membres de l'équipe de ventes.
8. Tous les membres de l'équipe de ventes doivent pouvoir lire les lignes de la table Marketing.CampaignBalance.
9. Tous les membres de l'équipe de ventes doivent pouvoir lire les lignes de la table DirectMarketing.Competitor.
10. Le responsable des ventes doit pouvoir lire et mettre à jour la table Marketing.SalesTerritory.
11. Tous les membres du groupe HumanResources doivent pouvoir lire et mettre à jour les lignes de la table Marketing.SalesPerson.
12. Le responsable des ventes doit pouvoir exécuter la procédure stockée Marketing.MoveCampaignBalance.
13. Tous les membres de l'équipe de ventes doivent pouvoir exécuter toutes les procédures stockées du schéma DirectMarketing.

## Exercice 1 : Créer des connexions

### Scénario

On vous a fourni les spécifications de sécurité de la base de données MarketDev. Dans cet exercice, vous devez créer des connexions Windows individuelles, des connexions de groupe Windows, des connexions SQL qui sont requises pour mettre en place les spécifications de sécurité.

Les tâches principales de cet exercice sont les suivantes :

1. Examiner la configuration requise.
2. Créer les connexions requises.

► **Tâche 1 : Examiner la configuration requise**

- Prenez connaissance des spécifications de sécurité fournies dans la documentation.

► **Tâche 2 : Créer les connexions requises**

- Créez les connexions dont vous avez déterminée qu'elles sont requises par le système. Cela inclut des connexions Windows, des connexions de groupe Windows et des connexions SQL.

**Résultats** : À la fin de cet exercice, vous aurez créé les connexions Windows et SQL requises.

## Exercice 2 : Résoudre un problème de connexion d'application

### Scénario

Le service informatique d'Adventure Works a implémenté une nouvelle application Web appelée Promote. L'application Promote requiert une connexion SQL nommée PromoteApp. La connexion SQL doit utiliser un mot de passe fixe. Cette application a fonctionné pendant un certain temps mais s'est interrompue. Il apparaît que l'application ne peut pas se connecter au serveur Proseware. Vous devez réinitialiser le mot de passe de l'utilisateur PromoteApp et désactiver la stratégie de vérification de la connexion.

La tâche principale de cet exercice est la suivante :

1. Résoudre un problème de connexion d'application.

► **Tâche 1 : Résoudre un problème de connexion d'application**

- Réinitialisez le mot de passe pour la connexion SQL PromoteApp à « Pa\$\$w0rd ».
- Désactivez la stratégie vérifiant la connexion de l'application.

**Résultats** : À la fin de cet exercice, vous aurez corrigé un problème de connexion d'application.

## Exercice 3 : Créer des utilisateurs de base de données

### Scénario

Vous avez créé les connexions requises pour le serveur Proseware conformément aux spécifications de sécurité qui vous ont été fournies. Vous devez créer des utilisateurs de base de données pour ces connexions dans la base de données MarketDev.

Les tâches principales de cet exercice sont les suivantes :

1. Examiner la configuration requise.
2. Créer les utilisateurs de base de données requis.

► **Tâche 1 : Examiner la configuration requise**

- Prenez connaissance des spécifications de sécurité fournies dans la documentation.

► **Tâche 2 : Créer les utilisateurs de base de données requis**

- Créez les utilisateurs de base de données dont vous avez déterminé qu'ils sont requis pour la base de données MarketDev.

**Résultats** : À la fin de cet exercice, vous devez avoir créé les utilisateurs de base de données requis.

## Exercice difficile 4 : Corriger l'accès à la base de données restaurée (si le temps le permet)

### Scénario

Un administrateur de base de données junior a essayé de restaurer la base de données LanguageDetails et d'autoriser l'accès à un compte de connexion SQL nommée LDUser. Il a été capable de restaurer la base de données et de créer la connexion, mais n'a pas pu créer l'utilisateur de base de données. Il soupçonne que quelque chose dans la base de données existante l'en empêche, car il peut créer et affecter d'autres connexions SQL sans problème. Vous devez restaurer la base de données LanguageDetails, créer la connexion LDUser, créer l'utilisateur de base de données LDUser et vérifier que l'utilisateur a accès à la base de données.

La tâche principale de cet exercice est la suivante :

1. Corriger l'accès à une base de données restaurée.

#### ► Tâche 1 : Corriger l'accès à une base de données restaurée

- Restaurez la base de données LanguageDetails à partir du fichier D:\22462A\_Labs\22462A\_09\_PRJ\LanguageDetails.bak sur l'instance de serveur Proseware.
- Créez une connexion LDUser avec la vérification de la stratégie désactivée et un mot de passe « Pa\$\$w0rd ».
- Corrigez l'accès à la base de données LanguageDetails pour l'utilisateur de base de données LDUser.
- Testez que la connexion LDUser peut accéder à la base de données et peut sélectionner les lignes de la table dbo.Language.

**Résultats :** À la fin de cet exercice, vous devez avoir résolu le problème.

## Contrôle des acquis et éléments à retenir

- Questions de contrôle des acquis
- Meilleures pratiques

### Questions de contrôle des acquis

1. Comment SQL Server tire-t-il parti de la stratégie de mot de passe Windows ?
2. Quelle stratégie de compte s'applique sur Windows XP ?

### Meilleures pratiques

1. Réduire le nombre de connexions SQL Server.
2. Vérifier que les dates d'expiration sont appliquées aux connexions créées temporairement.
3. Désactiver les connexions plutôt que de les supprimer s'il est fort probable qu'elles seront réutilisées.
4. Configurer la délégation Kerberos lorsqu'une identité d'utilisateur Windows doit être transmise entre les systèmes.

# Module 10

## Attribution de rôles de serveur et de base de données

### Sommaire :

Leçon 1 : Utilisation de rôles serveur	10-3
Leçon 2 : Utilisation de rôles de base de données fixes	10-12
Leçon 3 : Création de rôles de base de données définis par l'utilisateur	10-18
Atelier pratique 10 : Attribution de rôles de serveur et de base de données	10-26

## Vue d'ensemble du module

- Utilisation de rôles serveur
- Utilisation de rôles de base de données fixes
- Création de rôles de base de données définis par l'utilisateur

Une fois qu'une connexion a été authentifiée par le serveur et mappée à un utilisateur de base de données, vous devez attribuer des autorisations à la connexion ou à l'utilisateur de base de données. Les autorisations peuvent être attribuées directement à la connexion ou à l'utilisateur de base de données. Dans le module suivant, vous allez apprendre comment les autorisations sont attribuées directement. Mais même si cette opération est possible, lorsque des jeux d'autorisations s'appliquent potentiellement à plusieurs utilisateurs, les rôles peuvent être utilisés pour fournir les autorisations à la place.

SQL Server® possède un jeu de rôles fixes aux niveaux du serveur et de la base de données, et des rôles définis par l'utilisateur peuvent également être créés à ces deux niveaux. Les rôles fixes ont un jeu d'autorisations spécifique, mais les rôles définis par l'utilisateur ont un jeu d'autorisations défini par l'utilisateur qui leur est appliqué. Les autorisations associées à un rôle dont une connexion ou un utilisateur de base de données est membre sont attribuées à cette connexion ou à cet utilisateur de base de données.

### Objectifs

À la fin de ce module, vous serez à même d'effectuer les tâches suivantes :

- utiliser les rôles serveur ;
- utiliser les rôles de base de données fixes ;
- créer des rôles de base de données définis par l'utilisateur.

## Leçon 1

# Utilisation de rôles serveur

- Autorisations de portée serveur
- Autorisations de portée serveur classiques
- Vue d'ensemble des rôles serveur fixes
- Rôle serveur public
- Rôles serveur définis par l'utilisateur
- Démonstration 1A : Attribution de rôles serveur

Le premier type de rôle que vous allez étudier est le rôle serveur. Les rôles serveur ont des autorisations qui couvrent l'intégralité de l'instance du serveur. Bien que les rôles serveur soient très puissants, ils doivent être rarement utilisés.

Le rôle serveur le plus puissant est le rôle sysadmin. Soyez prudent lorsque vous attribuez des connexions à ce rôle, car ses membres disposent d'un accès complet au serveur entier.

Dans cette leçon, vous étudierez également le rôle public. Il peut être utilisé pour attribuer des autorisations au niveau de l'instance à toutes les connexions.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- décrire les autorisations de portée serveur ;
- décrire les autorisations qui peuvent généralement être attribuées au niveau du serveur ;
- expliquer l'utilisation des rôles serveur fixes ;
- expliquer la fonction du rôle serveur public ;
- configurer des rôles serveur définis par l'utilisateur.

## Autorisations de portée serveur

- Les autorisations au niveau du serveur peuvent être attribuées de trois manières
  - Rôles serveur fixe
  - Rôles serveur définis par l'utilisateur
  - Autorisations de portée serveur spécifiques
- Réduction de l'utilisation des rôles serveur fixe
  - Attribuer des autorisations plus spécifiques

```
USE master;
GO
GRANT ALTER ON LOGIN:::HRApp
TO [Adventureworks\Holly];
GO
GRANT ALTER ANY DATABASE
TO [Adventureworks\Holly];
GO
```

### Points clés

Les autorisations allouées au niveau du serveur sont très puissantes. Elles s'appliquent à l'intégralité du serveur et à ses bases de données. Pour cette raison, évitez d'allouer des autorisations de portée serveur et essayez plutôt d'attribuer des autorisations plus spécifiques.

Les rôles de portée serveur fixes sont inclus dans de nombreuses versions de SQL Server et sont surtout conservés pour la rétrocompatibilité.

### Exemples de diapositive

Sur cette diapositive, deux exemples d'octroi d'autorisations au niveau du serveur sont fournis. Le premier exemple montre comment accorder une autorisation spécifique. L'autorisation de modifier la connexion HRApp est accordée à la connexion Windows AdventureWorks\Holly.

Le second exemple montre comment utiliser l'option ANY. L'option ANY peut être appliquée à plusieurs types d'objets, y compris les objets dont l'étendue est limitée au niveau du serveur. L'utilisation de l'option ANY permet d'attribuer des autorisations sur des classes d'objet spécifiques sans qu'il soit nécessaire de créer un grand nombre d'autorisations individuelles. Notez que l'option ANY accorde à AdventureWorks\Holly la possibilité de modifier les bases de données existantes, mais elle s'applique aussi à toutes les futures bases de données créées sur le serveur.

La syntaxe complète de l'instruction GRANT est complexe. Vous trouverez des informations plus détaillées sur l'instruction GRANT dans le module 11.

## Attribution à l'aide de l'interface utilisateur graphique

Les éléments sécurisables SQL Server sont les objets qui peuvent être protégés via des autorisations. Pour accorder des autorisations à des éléments sécurisables dans l'étendue du serveur à l'aide de l'interface utilisateur graphique de SSMS :

- Utilisez l'Explorateur d'objets dans SQL Server Management Studio pour afficher les propriétés de la connexion à laquelle vous souhaitez octroyer des autorisations.
- Sous l'onglet Éléments sécurisables, ajoutez les éléments sécurisables nécessaires et définissez les autorisations requises.



**Remarque** Si aucun élément sécurisable n'apparaît sous l'onglet dans SSMS, ne considérez pas qu'aucune autorisation n'est attribuée à un élément sécurisable. La plupart du temps, l'élément sécurisable à afficher ou à modifier doit être ajouté manuellement.

## Autorisations de portée serveur classiques

- La base de données active doit être master lorsque des autorisations de portée serveur sont attribuées
- Pour afficher les attributions d'autorisations, lancez une requête sur la vue sys.server\_permissions

<b>Autorisations de portée serveur classiques</b>	
ALTER ANY DATABASE	ALTER TRACE
BACKUP DATABASE	BACKUP LOG
CONNECT SQL	CONTROL SERVER
CREATE DATABASE	SHUTDOWN
VIEW ANY DEFINITION	VIEW SERVER STATE

### Points clés

Cette diapositive présente la liste des autorisations de portée serveur qui sont généralement attribuées. Comme cela a été mentionné, très peu d'autorisations sont généralement attribuées au niveau du serveur.

Ces autorisations ne peuvent pas être attribuées sauf si la base de données actuelle est la base de données master. Sinon, une erreur est retournée.

La vue sys.server\_permissions est utilisée pour interroger les autorisations de portée serveur actuellement attribuées. Un exemple de son utilisation est illustré dans la démonstration 1A.

## Vue d'ensemble des rôles serveur fixes

Rôle	Description	Autorisation au niveau du serveur
sysadmin	Effectuer toute activité	CONTROL SERVER (avec l'option GRANT)
dbcreator	Créer et modifier des bases de données	ALTER ANY DATABASE
diskadmin	Gérer des fichiers disque	ALTER RESOURCES
serveradmin	Configurer des paramètres à l'échelle du serveur	ALTER ANY ENDPOINT, ALTER RESOURCES, ALTER SERVER STATE, ALTER SETTINGS, SHUTDOWN, VIEW SERVER STATE
securityadmin	Gérer et auditer des connexions serveur	ALTER ANY LOGIN
processadmin	Gérer des processus SQL Server	ALTER ANY CONNECTION ALTER SERVER STATE
bulkadmin	Exécuter l'instruction BULK INSERT	ADMINISTER BULK OPERATIONS
setupadmin	Configurer les serveurs liés et de réPLICATION	ALTER ANY LINKED SERVER

### Points clés

Cette diapositive contient la liste des rôles serveur fixes, une description générale de leur fonction et la liste des principales autorisations associées à chacun des rôles.

 **Remarque** La liste des autorisations répertoriées n'est pas une liste exhaustive. Pour des détails complets, consultez la documentation en ligne.

Ces rôles au niveau du serveur sont appelés des rôles serveur fixes, car vous ne pouvez pas modifier les autorisations qui leur sont associées.

### Remplissage des rôles serveur fixes

Vous pouvez ajouter des connexions dans les rôles au niveau du serveur. Un comportement qui peut être inattendu est que chaque membre d'un rôle serveur fixe peut ajouter d'autres connexions à ce même rôle.

## Rôles et autorisations

L'attribution des mêmes autorisations que celles attribuées à un rôle et l'appartenance au rôle sont deux choses différentes.

Par exemple, accorder l'autorisation CONTROL SERVER à une connexion n'est pas la même chose que définir la connexion comme un membre du rôle serveur fixe sysadmin. Les membres sysadmin sont automatiquement affectés à l'utilisateur dbo d'une base de données. Cela n'est pas le cas pour les connexions qui disposent uniquement de l'autorisation CONTROL SERVER.



**Remarque** Le rôle securityadmin est presque équivalente du point de vue fonctionnel au rôle sysadmin et doit être accordé avec précaution.

Pour modifier l'appartenance des rôles serveur fixes, utilisez la commande ALTER SERVER ROLE. Un exemple est illustré dans la démonstration 1A.

**Question** : pourquoi le rôle securityadmin a-t-il autant de pouvoir ?

## Rôle serveur public

**Public est un rôle serveur particulier dont l'étendue est définie au niveau du serveur**

- N'est pas considéré comme un rôle serveur fixe car ses autorisations peuvent être modifiées
- Par défaut, les autorisations suivantes sont attribuées
  - Autorisation VIEW ANY DATABASE
  - Autorisation CONNECT sur les points de terminaison par défaut

### Points clés

public est un rôle particulier dont l'étendue est également définie au niveau du serveur. Il n'est pas considéré comme un rôle serveur fixe, car il est possible de modifier les autorisations qui lui sont attribuées. Toutes les connexions appartiennent automatiquement au rôle public ; il s'agit donc d'un autre rôle auquel des autorisations doivent être attribuées avec précaution.

### VIEW ANY DATABASE

De nombreux outils, utilitaires et applications SQL Server partent du principe que la liste des bases de données peut être consultée, même si l'utilisateur qui la consulte n'a pas l'autorisation d'exécuter des actions sur ou dans la base de données.

Pour ce faire, le rôle public doit disposer de l'autorisation VIEW ANY DATABASE. Bien qu'il soit possible de ne plus afficher les bases de données auxquelles vous n'avez pas accès, vous devez le faire avec prudence. SSMS, en particulier, devient très lent à utiliser, car il doit ensuite vérifier vos autorisations dans chaque base de données.

Un cas d'utilisation possible sont les bases de données hébergées, où la mise à disposition d'une base de données pour un utilisateur ne doit pas permettre à cet utilisateur de consulter toutes les autres bases de données du serveur. Testez le comportement et les performances de vos applications et outils de base de données avant tout changement.

### Autorisations de connexion

Par défaut, l'autorisation CONNECT a été accordée au rôle public sur les points de terminaison de la mémoire partagée, de TCP, des canaux nommés et des protocoles VIA. Cela permet aux utilisateurs de se connecter au serveur à l'aide de l'un de ces protocoles.

## Rôles serveur définis par l'utilisateur

- Nouvelle fonctionnalité dans SQL Server 2012
- Des autorisations spécifiques peuvent être attribuées aux rôles
  - Contrôle des autorisations beaucoup plus précis qu'avec les rôles serveur fixes
  - Doit généralement être utilisé à la place des rôles serveur fixes

```
USE master;
GO
CREATE SERVER ROLE staff;
GO
```

### Points clés

La nouveauté de SQL Server 2012 est la possibilité de créer des rôles définis par l'utilisateur au niveau du serveur.

En général, vous devez éviter d'utiliser les rôles serveur fixes, car ils ont tendance à fournir aux utilisateurs d'autres autorisations que celles nécessaires pour effectuer les tâches qui leur sont attribuées. Les rôles serveur définis par l'utilisateur vous permettent de configurer un jeu spécifique d'autorisations au niveau du serveur qui sont requises pour les membres du rôle.

## Démonstration 1A : Attribution de rôles serveur

- Dans cette démonstration, vous allez apprendre à
- Afficher les rôles serveur fixes disponibles via l'interface utilisateur graphique
  - Attribuer un rôle serveur fixe via l'interface utilisateur graphique
  - Afficher les rôles serveur fixes disponibles avec T-SQL
  - Attribuer un rôle serveur fixe avec T-SQL
  - Afficher les membres des rôles serveur fixes avec T-SQL
  - Créer un rôle serveur défini par l'utilisateur avec T-SQL
  - Afficher les autorisations de serveur actuellement attribuées

### Procédure de démonstration

1. Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn.** Dans le menu **Fichier** cliquez sur **Ouvrir**, puis sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_10\_PRJ\22462A\_10\_PRJ.ssmssqln**, puis cliquez sur **Ouvrir**.
3. Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** dans l'Explorateur de solutions.
4. Ouvrez le fichier de script **11 – Demonstration 1A.sql**.
5. Suivez les instructions contenues dans les commentaires du fichier de script.

## Leçon 2

# Utilisation de rôles de base de données fixes

- Autorisations dans l'étendue de la base de données
- Vue d'ensemble des rôles de base de données fixes
- Affectation des utilisateurs à des rôles
- Propriétaire de la base de données
- Démonstration 2A : Gestion des rôles et des utilisateurs

Outre les rôles disponibles au niveau du serveur, SQL Server fournit un jeu de rôles fixes au niveau de la base de données. Ces rôles de base de données fixes sont similaires au niveau des concepts aux rôles serveur fixes, mais ils s'appliquent à l'accès aux objets de base de données ou à l'accès à la base de données proprement dite, plutôt qu'à l'accès à toutes les bases de données du serveur. Lors de la conception d'une stratégie de sécurité, après avoir déterminé tous les rôles serveur fixes requis, l'étape suivante consiste généralement à déterminer les rôles de base de données fixes requis. Dans cette leçon, vous examinerez les autorisations disponibles au niveau de la base de données, les rôles de base de données fixes qui peuvent être utilisés pour fournir des jeux d'autorisations et un rôle de base de données particulier associé au propriétaire de la base de données.

## Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- expliquer les autorisations disponibles dans l'étendue de la base de données ;
- décrire les rôles de base de données fixes ;
- affecter des utilisateurs à des rôles ;
- décrire le concept du propriétaire de la base de données.

## Autorisations dans l'étendue de la base de données

- Les autorisations au niveau de la base de données peuvent être attribuées de trois manières
  - Rôles de base de données fixes
  - Rôles de base de données définis par l'utilisateur
  - Autorisations spécifiques dans l'étendue de la base de données
- Réduction de l'utilisation des rôles de base de données fixes
  - Attribuer des autorisations plus spécifiques

```
USE Adventureworks;
GO
GRANT CREATE TABLE TO HRManager;
GO
GRANT VIEW DEFINITION TO James;
GO
```

### Points clés

Il existe trois façons d'attribuer des autorisations au niveau de la base de données :

- Les rôles de base de données fixes sont très similaires aux rôles serveur fixes, indépendamment des objets auxquels ils s'appliquent et du fait que l'étendue est définie au niveau de la base de données plutôt qu'au niveau du serveur.
- Comme pour les rôles serveur, il est possible de créer des rôles de base de données définis par l'utilisateur et d'attribuer un jeu d'autorisations défini par l'utilisateur à chaque rôle.
- Les autorisations dans l'étendue de la base de données peuvent être attribuées individuellement.

Évitez d'attribuer des rôles de base de données fixes, car ils fournissent généralement plus de fonctions que celles requises pour la plupart des utilisateurs. Attribuez des autorisations beaucoup plus spécifiques au lieu de l'appartenance au rôle de base de données fixe.

### Exemple de diapositive

Dans le premier exemple de la diapositive, l'autorisation de créer des tables dans la base de données est accordée à l'utilisateur de base de données HRManager. Il est important de comprendre que plusieurs autorisations peuvent être requises pour exécuter une action. Dans cet exemple, HRManager a également besoin de l'autorisation ALTER ANY SCHEMA ALTER pour créer des tables.

Dans le deuxième exemple de la diapositive, l'autorisation d'afficher les définitions des objets de la base de données est accordée à l'utilisateur de base de données James. Cet octroi d'autorisation est courant pour les utilisateurs qui doivent exécuter des tâches de documentation mais qui ne sont pas autorisés à modifier la conception de la base de données.

## Vue d'ensemble des rôles de base de données fixes

Rôle	Description
db_owner	Effectue toutes les opérations de configuration et de maintenance sur la base de données et peut la supprimer
db_securityadmin	Modifie l'appartenance au rôle et gère les autorisations
db_accessadmin	Ajoute ou supprime l'accès à la base de données pour les connexions
db_backupoperator	Sauvegarde la base de données
db_ddladmin	Exécute une commande DDL quelconque dans la base de données
db_datawriter	Ajoute, supprime ou modifie les données de toutes les tables utilisateur
db_datareader	Lit toutes les données de toutes les tables utilisateur
db_denydatawriter	Ne peut pas ajouter, supprimer ni modifier les données dans les tables utilisateur
db_denydatareader	Ne peut pas lire les données dans les tables utilisateur

### Points clés

La diapositive présente les rôles de base de données fixes disponibles et une description de la fonction qu'ils sont censés remplir.

Dans la pratique, les rôles de base de données fixes doivent être attribués de manière selective. Au lieu d'attribuer des rôles de base de données fixes, envisagez d'attribuer des autorisations beaucoup plus spécifiques.

Par exemple, au lieu d'attribuer le rôle db\_datareader à un utilisateur, envisagez d'attribuer l'autorisation SELECT sur les objets sur lesquels l'utilisateur doit pouvoir utiliser une commande SELECT. Cela évite les situations où des tables supplémentaires sont ajoutées à une base de données et l'utilisateur dispose déjà d'autorisations sur ces tables. Même si l'utilisateur a besoin d'une autorisation pour utiliser une commande SELECT sur tous les objets de la base de données, il est préférable d'attribuer des autorisations SELECT au niveau de la base de données plutôt que d'attribuer l'appartenance de l'utilisateur au rôle db\_datareader. En effet, il est plus facile d'examiner les octrois d'autorisation à l'aide de vues telles que sys.database\_permissions.

Comme pour les rôles serveur fixes, les rôles de base de données fixes sont fournis en grande partie pour des raisons de rétrocompatibilité.



**Remarque** Comme pour le rôle serveur fixe securityadmin, l'ajout d'utilisateurs au rôle db\_securityadmin doit être effectué avec précaution, car l'autorisation est presque équivalente du point de vue fonctionnel à l'utilisateur dbo.

## Affectation des utilisateurs à des rôles

- Les utilisateurs peuvent être affectés aux rôles
  - Utilisation de l'interface utilisateur graphique
  - Utilisation de T-SQL

Mapp...	Base de données	Utilisateur
<input checked="" type="checkbox"/>	AdventureWorks	James
<input type="checkbox"/>	master	
<input type="checkbox"/>	model	
<input type="checkbox"/>	msdb	
<input type="checkbox"/>	ReportServer	
<input type="checkbox"/>	ReportServerTempDB	
<input type="checkbox"/>	tempdb	

Compte invité activé pour : AdventureWorks

Appartenance au rôle de base de données : AdventureWorks

<input type="checkbox"/>	db_accessadmin
<input type="checkbox"/>	db_backupoperator
<input type="checkbox"/>	db_datareader
<input type="checkbox"/>	db_datawriter
<input type="checkbox"/>	db_ddladmin
<input type="checkbox"/>	db_denydatareader
<input type="checkbox"/>	db_denydatawriter
<input type="checkbox"/>	db_owner
<input type="checkbox"/>	db_securityadmin
<input checked="" type="checkbox"/>	public

### Points clés

Les utilisateurs peuvent être affectés à des rôles à l'aide de l'interface utilisateur graphique de SSMS ou de la commande T-SQL.

### Interface utilisateur graphique

Pour attribuer un rôle à un utilisateur avec SSMS, développez le serveur, le nœud Sécurité et le nœud Connexions, puis cliquez avec le bouton droit sur la connexion et cliquez sur Propriétés. Dans la fenêtre Propriétés de la connexion, cliquez sur l'onglet Mappage de l'utilisateur pour afficher la liste des bases de données auxquelles la connexion a été mappée. Lorsque vous sélectionnez chaque base de données dans le volet supérieur, la liste des rôles de base de données apparaît dans le volet inférieur. Vous pouvez ensuite sélectionner les rôles auxquels la connexion doit être affectée.

Vous pouvez également attribuer l'appartenance du point de vue du rôle. Pour attribuer ou supprimer des utilisateurs d'un rôle à l'aide de SSMS, développez la base de données appropriée, le nœud Sécurité, le nœud Rôles, le nœud Rôles de base de données, puis cliquez avec le bouton droit sur le rôle approprié et cliquez sur Propriétés. Dans l'écran Propriétés du rôle, vous pouvez ajouter et supprimer des utilisateurs de cette base de données.

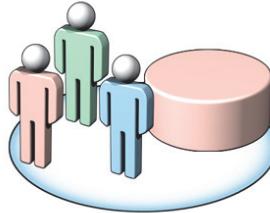
### Gestion de l'appartenance de rôle via T-SQL

L'exemple de la diapositive illustre l'utilisation de l'instruction ALTER ROLE. L'utilisateur James est ajouté au rôle de base de données fixe db\_datareader.

La clause DROP MEMBER de l'instruction ALTER ROLE supprime les membres de rôles.

**Question :** pouvez-vous citer un exemple de commande qui peut être exécutée par un membre du rôle db\_ddladmin ?

## Propriétaire de la base de données



### dbo

La connexion sa et les membres du rôle sysadmin sont mappés au compte dbo, ainsi que le propriétaire de la base de données

### Points clés

Il a été mentionné dans le dernier module qu'il existe deux utilisateurs particuliers : dbo et invité. L'utilisateur invité a été décrit dans le dernier module, avec une présentation de l'utilisateur dbo.

### dbo

Un utilisateur dbo dispose d'autorisations implicites pour réaliser toutes les activités dans la base de données. La connexion « sa » et tout membre du rôle serveur fixe sysadmin utilisant une base de données sont automatiquement mappés à l'utilisateur dbo. À l'instar des autres objets dans SQL Server, les bases de données ont des propriétaires. Le propriétaire de la base de données est également mappé à l'utilisateur dbo.



**Remarque** Les objets dans l'étendue du schéma appartiennent automatiquement au propriétaire du schéma, indépendamment de l'utilisateur qui les a créés. Les objets hors de l'étendue du schéma appartiennent automatiquement au principal de base de données qui les a créés. Pour tout principal disposant de droits db\_owner, le propriétaire est dbo.

dbo ne peut pas être supprimé et est présent dans chaque base de données. L'équivalence de dbo est particulière en ce sens qu'une fois que SQL Server détermine qu'un utilisateur est mappé à dbo, aucun autre contrôle des autorisations n'est effectué dans la base de données. Cela signifie que vous ne pouvez pas utiliser ultérieurement la commande DENY sur une autorisation dans la base de données pour un utilisateur disposant d'une équivalence dbo.

## Démonstration 2A : Gestion des rôles et des utilisateurs

Dans cette démonstration, vous allez apprendre à

- Afficher les rôles de base de données fixes disponibles via l'interface utilisateur graphique
- Attribuer un rôle de base de données fixe via l'interface utilisateur graphique
- Afficher les rôles de base de données fixes disponibles avec T-SQL
- Attribuer un rôle de base de données fixe avec T-SQL
- Afficher les membres des rôles de base de données fixes avec T-SQL

### Procédure de démonstration

1. Si la démonstration 1A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn.** Dans le menu **Fichier** cliquez sur **Ouvrir**, puis sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_10\_PRJ\22462A\_10\_PRJ.ssmssln**, puis cliquez sur **Ouvrir**.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez le fichier de script **00 – Setup.sql** dans l'Explorateur de solutions.
  - Ouvrez et exécutez le fichier de script **11 – Demonstration 1A.sql** à partir de l'Explorateur de solutions.
2. Ouvrez le fichier de script **21 – Demonstration 2A.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

## Leçon 3

# Création de rôles de base de données définis par l'utilisateur

- Utilisation des rôles de base de données définis par l'utilisateur
- Application de rôles dans des scénarios courants
- Démonstration 3A : Utilisation des rôles de base de données définis par l'utilisateur
- Définition des rôles d'application
- Démonstration 3B : Utilisation des rôles d'application

Dans les deux premières leçons de ce module, vous avez appris à utiliser les rôles serveur fixes et définis par l'utilisateur et les rôles de base de données fixes. Comme pour les rôles serveur définis par l'utilisateur, SQL Server fournit une option permettant de créer des rôles de base de données définis par l'utilisateur. Vous devez tenir compte des rôles de base de données définis par l'utilisateur au lieu des rôles de base de données fixes dans la plupart des situations. Une bonne conception des rôles de base de données définis par l'utilisateur est primordiale pour concevoir une architecture de sécurité pour votre base de données.

### Objectifs

À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- utiliser les rôles de base de données définis par l'utilisateur ;
- appliquer des rôles dans des scénarios courants ;
- définir des rôles d'application.

## Utilisation des rôles de base de données définis par l'utilisateur

- Les rôles de base de données peuvent être créés, modifiés et supprimés
  - Instruction CREATE ROLE pour créer
  - Les rôles ont des propriétaires
  - Les autorisations sont attribuées au rôle
  - Les autorisations des rôles sont héritées par les membres du rôle

```
USE MarketDev;
GO

CREATE ROLE MarketingReaders
AUTHORIZATION dbo;
GO

GRANT SELECT ON SCHEMA::Marketing
TO MarketingReaders;
GO
```

### Points clés

Les rôles de base de données définis par l'utilisateur sont créés à l'aide de l'instruction CREATE ROLE.

Dans l'exemple illustré sur la diapositive, un nouveau rôle appelé MarketingReaders est en cours de création. Notez qu'à l'instar des autres objets dans SQL Server, les rôles ont des propriétaires. Les propriétaires de rôles ont le contrôle total du rôle. Dans l'exemple illustré sur la diapositive, dbo est défini en tant que propriétaire du rôle MarketingReaders.

### Rôles et autorisations

Une fois que les rôles de base de données définis par l'utilisateur sont créés, des autorisations leur sont attribuées de la même manière que celles attribuées aux utilisateurs de base de données.

Dans l'exemple illustré sur la diapositive, l'autorisation SELECT est accordée sur le schéma Marketing aux membres du rôle MarketingReaders. À partir de ce moment, les utilisateurs ajoutés au rôle MarketingReaders ont l'autorisation d'utiliser une commande SELECT sur un objet contenu dans le schéma Marketing. (L'octroi d'autorisations sur des objets contenant des schémas est décrit dans le module suivant).

## Application de rôles dans des scénarios courants

- Scénario classique
  - Définir les utilisateurs dbo et d'autres rôles administratifs
  - Définir des groupes d'autorisations dans la base de données
  - Utiliser le rôle public pour les autorisations communes
  - Créer des rôles et leur attribuer des autorisations
  - Ajouter des utilisateurs aux rôles
- Pour une prise de décision inscrite dans le code
  - IS\_SRVROLEMEMBER, IS\_MEMBER

```
IF IS_MEMBER('BankManagers') = 0
BEGIN
    PRINT 'Operation is only for bank manager use';
    ROLLBACK;
END;
```

### Points clés

Il est important d'utiliser un processus normatif lorsque vous appliquez des rôles dans des applications métier classiques.

1. Vous devez commencer par définir les rôles d'administration au niveau du serveur ou de la base de données et déterminer un utilisateur dbo approprié. Peu d'utilisateurs doivent être équivalents à dbo.
2. Vous devez tenir compte des types d'accès dont chaque utilisateur a besoin. Cela peut impliquer de déterminer dans quelle mesure les configurations requises se chevauchent avec leur appartenance au groupe Windows. Lorsque plusieurs utilisateurs (y compris les utilisateurs de groupe) ont besoin d'autorisations communes, définissez les groupes d'autorisations en tant que rôles.
3. Si tous les utilisateurs requièrent un jeu d'autorisations, envisagez d'utiliser le rôle public dans la base de données. Par exemple, les fonctions de mise en forme sont des exemples appropriés de code qu'un utilisateur peut potentiellement être autorisé à exécuter.
4. Pour les autres groupes d'autorisation, créez des rôles appropriés et attribuez-leur des autorisations.
5. Ajoutez les utilisateurs qui ont besoin des groupes d'autorisations aux rôles qui fournissent ces autorisations.

Notez que contrairement aux groupes de domaine Windows qui sont généralement nommés après les utilisateurs qui en sont membres (par exemple, Salespeople), les rôles doivent être nommés en fonction des autorisations qu'ils fournissent. Ils s'apparentent plus aux groupes locaux Windows à cet égard.

### Test de l'appartenance au rôle dans le code T-SQL

Vous pouvez décider d'autoriser les membres d'un rôle à exécuter ou non le code T-SQL (par exemple, une procédure stockée), mais il est également possible de placer des limites dans le module de code.

Par exemple, vous pouvez décider que BankManagers et BankTellers peuvent transférer des fonds mais que BankTellers peut avoir une limite inférieure sur les montants de transfert.

La fonction IS\_SRVROLEMEMBER teste l'appartenance au rôle serveur, et la fonction IS\_MEMBER teste l'appartenance au rôle de base de données. La fonction IS\_MEMBER permet également de tester l'appartenance au groupe Windows.

Dans le fragment de code illustré dans la diapositive, une opération est annulée si l'utilisateur n'appartient pas au groupe BankManagers.

## Démonstration 3A : Utilisation des rôles de base de données définis par l'utilisateur

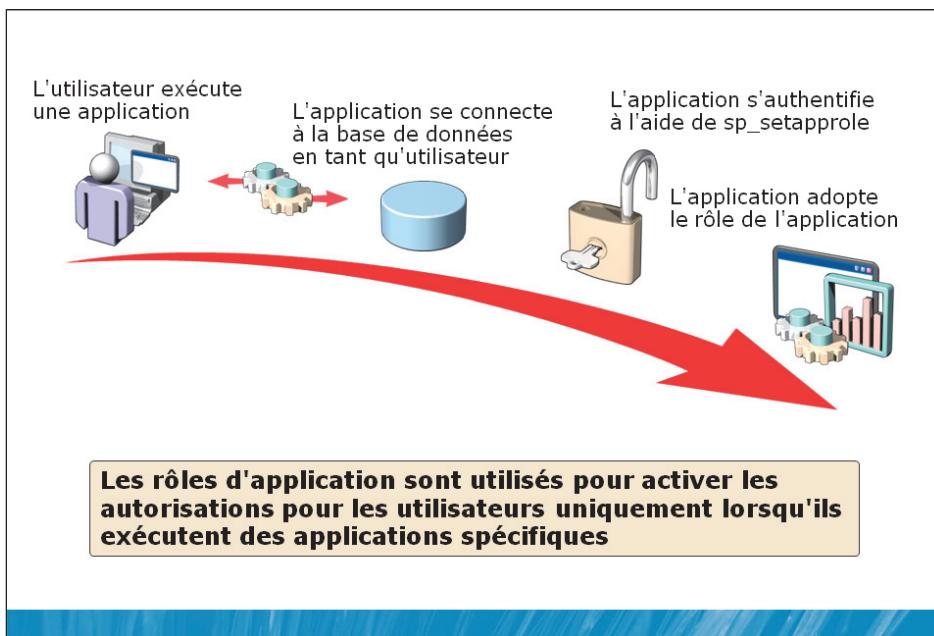
Dans cette démonstration, vous allez apprendre à

- Créer un rôle de base de données défini par l'utilisateur à l'aide de l'interface utilisateur graphique
- Créer un rôle de base de données défini par l'utilisateur avec T-SQL
- Afficher les rôles de base de données disponibles avec T-SQL

### Procédure de démonstration

1. Si la démonstration 1A ou 2A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn.** Dans le menu **Fichier** cliquez sur **Ouvrir**, puis sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_10\_PRJ\22462A\_10\_PRJ.ssmssqln**, puis cliquez sur **Ouvrir**.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez les fichiers de script **00 – Setup.sql**, **11 – Demonstration 1A.sql** et **21 – Demonstration 2A.sql** à partir de l'Explorateur de solutions.
2. Ouvrez le fichier de script **31 – Demonstration 3A.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script pour exécuter chaque lot T-SQL contenu dans le fichier.

## Définition des rôles d'application



### Points clés

Les rôles d'application sont utilisés pour activer des autorisations pour les utilisateurs uniquement lorsqu'ils exécutent des applications spécifiques.

Par exemple, supposons que vous souhaitez qu'un utilisateur puisse mettre à jour les lignes d'une table tout en utilisant une application. Mais vous ne souhaitez pas que ce même utilisateur puisse ouvrir la table dans SSMS et modifier les lignes, ou se connecter à la table à partir de Microsoft Excel.

### Rôles d'application

Les rôles d'application sont une solution à ce scénario. Ils ne contiennent pas de membres. Il s'agit d'un type particulier de rôle auquel des autorisations sont attribuées.

Un utilisateur se connecte généralement à une application à l'aide d'un compte à faibles priviléges. L'application appelle ensuite la procédure stockée système sp\_setapprole pour « activer » le rôle d'application. À ce stade, les autorisations de l'utilisateur pour la connexion existante sont remplacées par celles du rôle d'application.

**Remarque** Les autorisations du rôle d'application ne sont pas ajoutées aux autorisations de l'utilisateur. Elles les remplacent.

### Restauration à partir d'un rôle d'application

Dans les versions de SQL Server antérieures à SQL Server 2005, il n'existe pas de méthode de restauration du contexte de sécurité initial. À partir de SQL Server 2005, la procédure stockée système sp\_setapprole crée un cookie. L'application peut stocker le cookie et le renvoyer ultérieurement à la procédure stockée système sp\_unsetapprole, pour restaurer le contexte de sécurité initial.

### Mot de passe et chiffrement

Pour empêcher d'autres applications et utilisateurs d'activer le rôle d'application, un mot de passe est attribué à chaque rôle d'application. L'application doit fournir le mot de passe lorsqu'elle appelle `sp_setapprole`.

Pour éviter tout risque d'interception des paquets réseau, une option est fournie pour envoyer le mot de passe sous une forme chiffrée, afin qu'il ne soit pas visible dans les traces du réseau. Veillez à ne pas dépendre entièrement de ce comportement, car la méthode de chiffrement utilisée n'est pas très renforcée.

## Démonstration 3B : Utilisation des rôles d'application

Dans cette démonstration, vous allez apprendre à

- Créer un rôle d'application
- Transformer le contexte de sécurité en rôle d'application

### Procédure de démonstration

1. Si la démonstration 1A, 2A ou 3A n'a pas été effectuée :
  - Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
  - Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn.** Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**, naviguez jusqu'à **D:\22462A\_Labs\22462A\_10\_PRJ\22462A\_10\_PRJ.ssmssln**, puis cliquez sur **Ouvrir**.
  - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez et exécutez les fichiers de script **00 – Setup.sql**, **11 – Demonstration 1A.sql**, **21 – Demonstration 2A.sql** et **31 – Demonstration 3A.sql** à partir de l'Explorateur de solutions.
2. Ouvrez le fichier de script **32 – Demonstration 3B.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script pour exécuter chaque lot T-SQL contenu dans le fichier.

## Atelier pratique 10 : Attribution de rôles de serveur et de base de données

- Exercice 1 : Attribuer des rôles serveur
- Exercice 2 : Attribuer des rôles de base de données fixes
- Exercice 3 : Créer et attribuer des rôles de base de données définis par l'utilisateur
- Exercice difficile 4 : Vérifier les attributions de rôles (si le temps le permet)

Informations d'ouverture de session

Ordinateur virtuel	22462A-MIA-SQL1
Nom d'utilisateur	AdventureWorks\Administrateur
Mot de passe	Pa\$\$w0rd

**Durée approximative : 45 minutes**

### Configuration de l'atelier pratique

Pour cet atelier pratique, vous utiliserez l'environnement d'ordinateur virtuel disponible.

Avant de commencer cet atelier pratique, vous devez procéder aux étapes suivantes :

1. Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
3. Dans la fenêtre Se connecter au serveur, tapez **Proseware** dans la zone de texte **Nom du serveur**.
4. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn**.
5. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**.
6. Dans la fenêtre Ouvrir un projet, sélectionnez le projet **D:\22462A\_Labs\22462A\_10\_PRJ\22462A\_10\_PRJ.ssmssln**.
7. Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Dans l'Explorateur de solutions, double-cliquez sur le fichier de requête **00 – Setup.sql**. Lorsque la fenêtre de requête s'affiche, cliquez sur **Exécuter** dans la barre d'outils.

## Scénario de l'atelier pratique

Vous avez créé les connexions et les utilisateurs de base de données SQL Server. Vous devez maintenant attribuer les connexions et les utilisateurs aux rôles requis en fonction de la configuration de sécurité requise pour la base de données MarketDev. Vous devez affecter le niveau d'accès minimal qui permet à chaque utilisateur d'effectuer son travail. Cela requiert une combinaison de rôles serveur, de rôles de base de données fixes et de rôles de base de données définis par l'utilisateur.

Ne vous préoccupez pas des autorisations d'objet et de schéma, car elles seront attribuées dans le module 11. Toutefois, vous devrez tenir compte des configurations de rôle requises à ce moment.

 **Remarque** Les modifications que vous apportez sont migrées par la suite vers l'environnement de production. Vous devez utiliser les commandes T-SQL pour implémenter les modifications nécessaires.

## Documentation

### Utilisateur Windows existant et structure de groupe

	ITSupport	SalesPeople	CreditManagement	HumanResources	CorporateManagers
<b>David.Alexander</b>	X				X
<b>Jeff.Hay</b>	X				
<b>Palle.Petersen</b>	X				
<b>Terry.Adams</b>	X				
<b>Darren.Parker</b>		X			X
<b>Mike.Ray</b>		X			
<b>April.Reagan</b>		X			
<b>Jamie.Reding</b>		X			
<b>Darcy.Jayne</b>		X			
<b>Naoki.Sato</b>		X			
<b>Bjorn.Rettig</b>			X		X
<b>Don.Richardson</b>			X		
<b>Wendy.Kahn</b>			X		
<b>Neil.Black</b>				X	X
<b>Madeleine.Kelly</b>				X	

### Configuration de la sécurité existante

- Les connexions de groupe et les utilisateurs de base de données Windows suivants ont été créés :
  - AdventureWorks\ITSupport
  - AdventureWorks\SalesPeople
  - AdventureWorks\CreditManagement
  - AdventureWorks\HumanResources
  - AdventureWorks\CorporateManagers
- Les connexions et les utilisateurs de base de données Windows suivants ont été créés :
  - AdventureWorks\Jeff.Hay
  - AdventureWorks\April.Reagan
  - AdventureWorks\Darren.Parker
- Les connexions SQL suivantes ont été créées :
  - PromoteApp
  - DBMonitorApp

### Configuration requise pour la sécurité

1. L'administrateur de base de données senior Jeff Hay doit disposer d'un accès et d'un contrôle complets sur l'instance de serveur Proseware.
2. Tous les membres du groupe ITSupport doivent bénéficier d'un accès et d'un contrôle complets sur la base de données MarketDev.
3. Proseware utilise une application appelée DBMonitor de Trey Research. Cette application exige une connexion SQL nommée DBMonitorApp qui requiert une capacité à lire tous les objets de la base de données MarketDev, mais pas une capacité à les mettre à jour.
4. Tous les membres du groupe CorporateManagers effectuent périodiquement une analyse FFPM (Forces, Faiblesses, Possibilités, Menaces). Pour cela ils doivent pouvoir à la fois lire et mettre à jour les lignes de la table DirectMarketing.Competitor.
5. Tous les membres du groupe SalesPeople doivent pouvoir afficher les données de toutes les tables dans le schéma DirectMarketing, sauf April Reagan qui est une stagiaire.
6. Seuls les membres du groupe ITSupport et les membres du groupe CreditManagement doivent pouvoir mettre à jour la table Marketing.CampaignBalance directement.
7. Les membres de l'entreprise appartenant au groupe SalesPeople, au groupe CreditManagement et au groupe CorporateManagers sont appelés membres de l'équipe de ventes.
8. Tous les membres de l'équipe de ventes doivent pouvoir lire les lignes de la table Marketing.CampaignBalance.
9. Tous les membres de l'équipe de ventes doivent pouvoir lire les lignes de la table DirectMarketing.Competitor.

10. Le responsable des ventes doit pouvoir lire et mettre à jour la table Marketing.SalesTerritory.
11. Tous les membres du groupe HumanResources doivent pouvoir lire et mettre à jour les lignes de la table Marketing.SalesPerson.
12. Le responsable des ventes doit pouvoir exécuter la procédure stockée Marketing.MoveCampaignBalance.
13. Tous les membres de l'équipe de ventes doivent pouvoir exécuter toutes les procédures stockées du schéma DirectMarketing.

## Exercice 1 : Attribuer des rôles serveur

### Scénario

Vous devez implémenter les rôles serveur requis nécessaires pour prendre en charge la configuration de sécurité requise fournie.

Les tâches principales de cet exercice sont les suivantes :

1. Examiner la configuration requise.
2. Attribuer les rôles serveur requis.

► **Tâche 1 : Examiner la configuration requise**

- Prenez connaissance des spécifications de sécurité fournies dans la documentation.

► **Tâche 2 : Attribuer les rôles serveur requis**

- Attribuez les rôles serveur requis pour prendre en charge la configuration de sécurité requise pour la base de données MarketDev.

**Résultats** : À la fin de cet exercice, vous devez avoir attribué les rôles serveur requis.

## Exercice 2 : Attribuer des rôles de base de données fixes

### Scénario

Il vous a été fourni un ensemble de conditions détaillant l'accès requis par chaque connexion à la base de données MarketDev. Certaines de ces conditions peuvent être remplies par les rôles de base de données fixes, mais il est important de ne pas fournir des autorisations qui ne sont pas spécifiquement requises. Si vous pensez que des rôles de base de données définis par l'utilisateur sont nécessaires, ceux-ci seront créés dans l'exercice suivant.

Les tâches principales de cet exercice sont les suivantes :

1. Examiner la configuration requise.
2. Attribuer les rôles de base de données fixes requis.

► **Tâche 1 : Examiner la configuration requise**

- Prenez connaissance des spécifications de sécurité fournies dans la documentation.

► **Tâche 2 : Attribuer les rôles de base de données fixes requis**

- Attribuez les rôles de base de données requis pour prendre en charge la configuration de sécurité requise pour la base de données MarketDev.

**Résultats** : À la fin de cet exercice, vous avez attribué des rôles de base de données fixes selon les besoins.

## Exercice 3 : Créer et attribuer des rôles de base de données définis par l'utilisateur

### Scénario

Il vous a été fourni un ensemble de conditions détaillant l'accès requis par chaque connexion à la base de données MarketDev. Dans l'exercice 2, vous avez attribué l'appartenance au rôle de base de données fixe. D'autres configurations requises peuvent être mieux prises en charge par les rôles de base de données définis par l'utilisateur. Dans cet exercice, vous allez créer et attribuer les rôles de base de données définis par l'utilisateur requis.

Les tâches principales de cet exercice sont les suivantes :

1. Examiner la configuration requise.
2. Créer et attribuer les rôles de base de données définis par l'utilisateur requis.

#### ► Tâche 1 : Examiner la configuration requise

- Prenez connaissance des spécifications de sécurité fournies dans la documentation.

#### ► Tâche 2 : Créer et attribuer les rôles de base de données définis par l'utilisateur requis

- Créez et attribuez les rôles de base de données définis par l'utilisateur qui sont requis pour prendre en charge la configuration de sécurité requise pour la base de données MarketDev.

**Résultats** : À la fin de cet exercice, vous avez créé et attribué les rôles de base de données définis par l'utilisateur selon les besoins.

## Exercice difficile 4 : Vérifier les attributions de rôles (si le temps le permet)

### Scénario

Vous avez créé des connexions et des utilisateurs de base de données, attribué des rôles de serveur et de base de données et créé et attribué des rôles de base de données définis par l'utilisateur. Il est important de vérifier que les attributions de rôle s'exécutent comme prévu. Dans cet exercice, vous allez utiliser les vues système sys.login\_token et sys.user\_token pour vérifier les jetons disponibles pour Darren Parker.

La tâche principale de cet exercice est la suivante :

1. Vérifier les attributions de rôle pour Darren Parker.

#### ► Tâche 1 : Vérifier les attributions de rôle pour Darren Parker

- À l'aide de l'instruction EXECUTE AS, modifiez le contexte de sécurité de la connexion AdventureWorks\Darren.Parker.
- Interrogez les fonctions système sys.login\_token et sys.user\_token pour vérifier les jetons disponibles pour Darren Parker.
- Rétablissez le contexte de sécurité à l'aide de la commande REVERT.

**Résultats :** À la fin de cet exercice, vous devez avoir testé les attributions de rôle pour Darren Parker.

## Contrôle des acquis et éléments à retenir

- Questions de contrôle des acquis
- Meilleures pratiques

### Questions de contrôle des acquis

1. Est-il possible de créer des rôles de base de données dans SQL Server 2012 ?
2. Quelle fonction vous permet de déterminer dans le code T-SQL si un utilisateur est membre d'un groupe Windows ?

### Meilleures pratiques

Évitez d'accorder plus d'autorisations que nécessaire. Il est très courant que des systèmes SQL Server disposent d'autorisations excessives. Les programmes d'installation des applications présument souvent de la nécessité d'autorisations de niveau beaucoup plus élevé que nécessaire. Les utilisateurs doivent alerter les fournisseurs concernés. Mieux encore, faites en sorte que la configuration appropriée de la sécurité et des autorisations soit un critère que les fournisseurs doivent remplir.

## Module 1 : Présentation de SQL Server 2012 et de son ensemble d'outils

# Atelier pratique 1 : Présentation de SQL Server et de son ensemble d'outils

### Configuration de l'atelier pratique

Pour cet atelier pratique, vous utiliserez l'environnement d'ordinateur virtuel disponible. Avant de débuter cet atelier pratique, vous devez procéder à l'étape suivante :

- Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.

## Exercice 1 : Vérifier l'installation des composants de SQL Server

► **Tâche 1 : Vérifier que le Moteur de base de données et Reporting Services ont été installés pour l'instance MKTG**

1. Cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, sur **Outils de configuration**, puis sur **Gestionnaire de configuration SQL Server**.
2. Dans le volet gauche de la fenêtre Gestionnaire de SQL Server Configuration Manager, cliquez sur **Services SQL Server**.
3. Dans le volet droit, vérifiez que les services suivants apparaissent pour l'instance de MKTG :
  - SQL Server (MKTG)
  - SQL Full-text Filter Daemon Launcher (MKTG)
  - SQL Server Reporting Services (MKTG)
  - Agent SQL Server (MKTG)

► **Tâche 2 : Notez les services installés pour l'instance par défaut**

1. Dans le volet gauche de la fenêtre Gestionnaire de SQL Server Configuration Manager, cliquez sur **Services SQL Server**.
2. Dans le volet droit, vérifiez que les services suivants apparaissent pour l'instance par défaut :
  - SQL Server (MSSQLSERVER)
  - SQL Full-text Filter Daemon Launcher (MSSQLSERVER)
  - SQL Server Analysis Services (MSSQLSERVER)
  - Agent SQL Server (MSSQLSERVER)

► **Tâche 3 : Vérifier que tous les services nécessaires comprenant l'Agent SQL Server sont lancés et configurés pour démarrer automatiquement dans les deux instances**

1. Vérifiez que tous les services de l'instance de MKTG sont en **mode de démarrage automatique**. Ignorez le service du SQL Full-text Filter Daemon Launcher pour le moment.
2. Vérifiez que tous les services de l'instance par défaut sont en **mode de démarrage automatique**. Ignorez le service du SQL Full-text Filter Daemon Launcher pour le moment.

## Exercice 2 : Modifier les comptes de service de la nouvelle instance

### ► Tâche 1 :Modifier le compte de service du moteur de base de données MKTG

1. Dans le volet gauche de la fenêtre Gestionnaire de SQL Server Configuration Manager, cliquez sur **Services SQL Server**.
2. Dans le volet droit, cliquez avec le bouton droit sur **SQL Server (MKTG)**, puis sélectionnez **Propriétés**.
3. Dans la zone de texte **Nom du compte**, tapez **AdventureWorks\PWService**.
4. Dans la zone de texte **Mot de passe**, tapez **Pa\$\$w0rd**.
5. Dans la zone de texte **Confirmer le mot de passe**, tapez **Pa\$\$w0rd**, puis cliquez sur **OK**.
6. Dans la fenêtre **Confirmer la modification du compte**, cliquez sur **Oui**.

### ► Tâche 2 :Modifier le compte de service pour l'Agent SQL Server MKTG

1. Dans le volet gauche de la fenêtre Gestionnaire de SQL Server Configuration Manager, cliquez sur **Services SQL Server**.
2. Dans le volet droit, cliquez avec le bouton droit sur **Agent SQL Server (MKTG)**, puis sélectionnez **Propriétés**.
3. Dans la zone **Nom du compte**, tapez :
  - **AdventureWorks\PWService**.
4. Dans la zone de texte **Mot de passe**, tapez **Pa\$\$w0rd**.
5. Dans la zone de texte **Confirmer le mot de passe**, tapez **Pa\$\$w0rd**, puis cliquez sur **OK**.
6. Cliquez avec le bouton droit sur **Agent SQL Server (MKTG)** et sélectionnez **Démarrer** pour redémarrer le service.

### Exercice 3 : Activer le protocole des canaux nommés pour les deux instances

- ▶ Tâche 1 : Activer le protocole des canaux nommés pour l'instance par défaut
  1. Dans le volet gauche de la fenêtre Gestionnaire de SQL Server Configuration Manager, développez **Configuration du réseau SQL Server**, puis cliquez sur **Protocoles pour MSSQLSERVER**.
  2. Dans le volet droit, si le protocole des **Canaux nommés** n'est pas activé, cliquez avec le bouton droit sur **Canaux nommés** et sélectionnez **Activer**. Dans la fenêtre Avertissement, cliquez sur **OK**.
- ▶ Tâche 2 : Activer le protocole des canaux nommés pour l'instance de MKTG
  1. Dans le volet gauche de la fenêtre Gestionnaire de SQL Server Configuration Manager, cliquez sur **Protocoles pour MKTG**.
  2. Dans le volet droit, si le protocole des **Canaux nommés** n'est pas activé, cliquez avec le bouton droit sur **Canaux nommés** et sélectionnez **Activer**. Dans la fenêtre Avertissement, cliquez sur **OK**.
- ▶ Tâche 3 : Redémarrer les deux services Moteur de base de données
  1. Dans le volet gauche de la fenêtre Gestionnaire de SQL Server Configuration Manager, cliquez sur **Services SQL Server**.
  2. Cliquez avec le bouton droit sur **SQL Server (MSSQLSERVER)** et sélectionnez **Redémarrer**.
  3. Cliquez avec le bouton droit sur **SQL Server (MKTG)** et sélectionnez **Redémarrer**.
  4. Dans le volet gauche de la fenêtre Gestionnaire de SQL Server Configuration Manager, cliquez sur **Services SQL Server**.
  5. Dans la barre d'outils, cliquez sur l'icône **Actualiser**.

## Exercice 4 : Créer un alias pour AdvDev

### ► Tâche 1 : Créer un alias 32 bits (AdvDev) pour l'instance de MKTG

1. Dans le volet gauche de la fenêtre Gestionnaire de SQL Server Configuration Manager, développez **Configuration de SQL Native Client 11.0 (32 bits)**, puis cliquez sur **Protocoles clients**.
2. Vérifiez que le protocole **Canaux nommés** est **Activé**.
3. Dans le volet gauche, double-cliquez sur **Alias**.
4. Dans le volet gauche, cliquez avec le bouton droit sur **Alias** et sélectionnez **Nouvel alias**.
5. Dans la fenêtre **Alias-Nouveau**, dans la zone de texte **Nom de l'alias**, tapez **AdvDev**.
6. Dans la zone de liste déroulante **Protocole**, sélectionnez **Named Pipes**.
7. Dans la zone de texte **Serveur**, tapez **.\\MKTG**, puis cliquez sur **OK**.

### ► Tâche 2 : Créer un alias 64 bits (AdvDev) pour l'instance de MKTG

1. Dans le volet gauche de la fenêtre Gestionnaire de SQL Server Configuration Manager, développez **Configuration de SQL Native Client 11.0**, puis cliquez sur **Protocoles clients**.
2. Vérifiez que le protocole **Canaux nommés** est **Activé**.
3. Dans le volet gauche, double-cliquez sur **Alias**.
4. Dans le volet gauche, cliquez avec le bouton droit sur **Alias** et sélectionnez **Nouvel alias**.
5. Dans la fenêtre **Alias-Nouveau**, dans la zone de texte **Nom de l'alias**, tapez **AdvDev**.
6. Dans la zone de liste déroulante **Protocole**, sélectionnez **Named Pipes**.
7. Dans la zone de texte **Serveur**, tapez **.\\MKTG**, puis cliquez sur **OK**.

### ► Tâche 3 : Utiliser SQL Server Management Studio pour se connecter à l'alias et s'assurer qu'il fonctionne correctement

1. Cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
2. Dans la fenêtre Se connecter au serveur, vérifiez que **Type de serveur** a la valeur **Moteur de base de données**.
3. Dans la zone de texte **Nom du serveur**, tapez **AdvDev**.
4. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn.**
5. Dans l'Explorateur d'objets, sous **AdvDev**, développez **Bases de données**.



**Remarque** : les bases de données présentes incluent au moins les éléments suivants :  
Bases de données système, Instantanés de base de données, ReportServer\$MKTG  
et ReportServer\$MKTGTempDB.

6. Fermez Microsoft SQL Server Management Studio.
7. Fermez SQL Server Configuration Manager.

## Exercice difficile 5 : Vérifier que le navigateur SQL est désactivé et configurer un port TCP/IP fixe (si le temps le permet)

### ► Tâche 1 : Configurer le port TCP pour l'instance du moteur de base de données MKTG sur 51550

1. Dans la fenêtre Ordinateur virtuel, dans le menu **Démarrer**, cliquez sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, sur **Outils de configuration**, puis sur **Gestionnaire de configuration SQL Server**.
2. Dans le volet gauche de la fenêtre Gestionnaire de SQL Server Configuration Manager, développez **Configuration du réseau SQL Server**, puis cliquez sur **Protocoles pour MKTG**.
3. Cliquez avec le bouton droit sur le protocole **TCP/IP** et sélectionnez **Propriétés**.
4. Dans la fenêtre Propriétés de : TCP/IP, cliquez sur l'onglet **Adresses IP**.
5. Faites défiler vers le bas de l'écran, sous la section **IP All**, désactivez la valeur de **Ports TCP dynamiques**.
6. Pour le **Port TCP**, tapez **51550**, puis cliquez sur **OK**.
7. Dans la fenêtre d'Avertissement, cliquez sur **OK**.
8. Dans le volet gauche, cliquez sur **Services SQL Server**.
9. Cliquez avec le bouton droit sur **SQL Server (MKTG)** et sélectionnez **Redémarrer**.
10. Dans le volet gauche de la fenêtre Gestionnaire de SQL Server Configuration Manager, cliquez sur **Services SQL Server**.
11. Dans la barre d'outils, cliquez sur l'icône **Actualiser** et vérifiez que le service démarre.

### ► Tâche 2 : Désactiver le service SQLBrowser

1. Dans le volet gauche de la fenêtre Gestionnaire de SQL Server Configuration Manager, cliquez sur **Services SQL Server**.
2. Dans le volet droit, si l'état de **SQL Server Browser** est **En cours d'exécution**, cliquez avec le bouton droit sur **SQL Server Browser**, puis cliquez sur **Arrêter**.
3. Cliquez avec le bouton droit sur **SQL Server Browser**, puis cliquez sur **Propriétés**.
4. Dans la fenêtre Propriétés de SQL Server Browser, sous l'onglet **Service**, définissez **Mode de démarrage** sur **Désactivé**, puis cliquez sur **OK**.
5. Fermez le Gestionnaire de SQL Server Configuration Manager.

## Module 2 : Préparation des systèmes pour SQL Server 2012

# Atelier pratique 2 : Préparation des systèmes pour SQL Server

### Configuration de l'atelier pratique

Pour cet atelier pratique, vous utiliserez l'environnement d'ordinateur virtuel disponible. Avant de débuter cet atelier pratique, vous devez procéder aux étapes suivantes :

1. Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
3. Dans la fenêtre Se connecter au serveur, tapez **Proseware** dans la zone de texte **Nom du serveur**.
4. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn.**
5. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**.
6. Dans la fenêtre Ouvrir un projet, sélectionnez le projet **D:\22462A\_Labs\22462A\_02\_PRJ\22462A\_02\_PRJ.ssmssln**.
7. Dans l'Explorateur de solutions, double-cliquez sur le fichier de requête **00 - Setup.sql**.  
Lorsque la fenêtre de requête s'affiche, cliquez sur **Exécuter** dans la barre d'outils.

## Exercice 1 : Ajuster la configuration de la mémoire

### ► Tâche 1 : Vérifier la mémoire totale du serveur

1. Cliquez sur **Démarrer**, cliquez avec le bouton droit sur **Ordinateur**, puis cliquez sur **Propriétés**.
2. Notez la valeur de la **Mémoire installée (RAM)**.
3. Fermez la fenêtre Système.

### ► Tâche 2 : Vérifier la mémoire allouée à l'instance par défaut

1. Cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
2. Dans la fenêtre Se connecter au serveur, tapez **AdventureWorks** pour le **Nom du serveur**, puis cliquez sur **Se conn**.
3. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur l'instance de serveur **AdventureWorks**, puis cliquez sur **Propriétés**.
4. Dans le volet **Sélectionner une page**, cliquez sur **Mémoire**.
5. Notez les valeurs de la **Mémoire minimale du serveur** (en Mo) et la **Mémoire maximale du serveur** (en Mo), puis cliquez sur **Annuler**.
6. Dans **Microsoft SQL Server Management Studio (Administrateur)**, dans le menu **Fichier**, cliquez sur **Quitter**.

### ► Tâche 3 : Vérifier la mémoire allouée à l'instance MKTG

1. Cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
2. Dans la fenêtre Se connecter au serveur, tapez **Proseware** pour le **Nom du serveur**, puis cliquez sur **Se conn**.
3. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur l'instance de serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\Administrateur)**, puis cliquez sur **Propriétés**.
4. Dans le volet **Sélectionner une page**, cliquez sur **Mémoire**.
5. Notez les valeurs de la **Mémoire minimale du serveur** (en Mo) et la **Mémoire maximale du serveur** (en Mo), puis cliquez sur **Annuler**.
6. Dans **Microsoft SQL Server Management Studio (Administrateur)**, dans le menu **Fichier**, cliquez sur **Quitter**.

► **Tâche 4 : Déterminer si l'allocation de mémoire est appropriée Si ce n'est pas le cas, apportez les modifications nécessaires à la configuration de la mémoire**

1. Examinez la **Configuration de la mémoire requise** dans la document associée au manuel du stagiaire.
2. Calculez la **Mémoire maximale** pour l'instance de serveur **AdventureWorks** comme suit :
  - Exemple de calcul (les valeurs réelles dépendent de la configuration de l'ordinateur virtuel)
  - Mémoire maximale = (Mémoire serveur – 1.5 Go) \* 0,6
  - Mémoire maximale = (3,0 – 1,5) \* 0,6
  - Mémoire maximale = 0,9 (approximatif)
3. Calculez la **Mémoire maximale** pour l'instance de serveur **Proseware** comme suit :
  - Exemple de calcul (les valeurs réelles dépendent de la configuration de l'ordinateur virtuel)
  - Mémoire maximale = (Mémoire serveur – 1.5 Go) \* 0,4
  - Mémoire maximale = (3,0 – 1,5) \* 0,4
  - Mémoire maximale = 0,6 (approximatif)
4. Cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
5. Dans la fenêtre Se connecter au serveur, tapez **AdventureWorks** pour le **Nom du serveur**, puis cliquez sur **Se conn.**
6. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur l'instance de serveur **AdventureWorks**, puis cliquez sur **Propriétés**.
7. Dans le volet **Sélectionner une page**, cliquez sur **Mémoire**.
8. Définissez la **Mémoire maximale du serveur (en Mo)** sur la valeur **900**, puis cliquez sur **OK**.



**Remarque :** il aurait été plus précis d'utiliser la valeur 921 Mo au lieu de 0,9 Go, mais la valeur 900 a été préférée par souci de simplicité.

9. Dans l'Explorateur d'objets, cliquez sur **Connecter**, puis sur **Moteur de base de données**.
10. Dans la fenêtre Se connecter au serveur, tapez **Proseware** pour le **Nom du serveur**, puis cliquez sur **Se conn.**
11. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur l'instance de serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\Administrateur)**, puis cliquez sur **Propriétés**.
12. Dans le volet **Sélectionner une page**, cliquez sur **Mémoire**.
13. Définissez la **Mémoire maximale du serveur (en Mo)** sur la valeur **600**, puis cliquez sur **OK**.



**Remarque :** il aurait été plus précis d'utiliser la valeur 614 Mo au lieu de 0,6Go, mais la valeur 600 a été préférée par souci de simplicité.

## Exercice 2 : Tests de contrainte de pré-installation

### ► Tâche 1 : Configurer SQLIOSIM

1. Dans le dossier D:\22462A\_Labs\22462A\_02\_PRJ, cliquez avec le bouton droit sur **sqliosimx64.exe**, puis cliquez sur **Exécuter en tant qu'administrateur**.
2. Dans la fenêtre Microsoft SQL Server IO Simulator, prenez connaissance du contrat et cliquez sur **Yes**.
3. Dans la fenêtre d'application WinZip Self-Extractor- SQLIOS~1.EXE, cliquez sur **Unzip**.
4. Dans la fenêtre WinZip Self-Extractor, cliquez sur **OK**.
5. Dans la fenêtre d'application WinZip Self-Extractor- SQLIOS~1.EXE, cliquez sur **Close**.
6. Ouvrez l'**Explorateur Windows**, puis accédez au dossier C:\SQLIOSimX64. Notez que deux versions de SQLIOSim sont fournies : une version de ligne de commande et la version d'interface utilisateur graphique. Cliquez avec le bouton droit sur **sqliosim.exe**, puis cliquez sur **Exécuter en tant qu'administrateur**.
7. Dans la fenêtre Fichiers et configuration, cliquez sur la ligne contenant **C:\sqliosim.mdx**, puis cliquez sur **Remove**.
8. Dans la fenêtre Fichiers et configuration, cliquez sur la ligne contenant **C:\sqliosim.idx**, puis cliquez sur **Remove**.
9. Dans la fenêtre Fichiers et configuration, cliquez sur la ligne contenant **D:\sqliosim.idx**, puis cliquez sur **Remove**.
10. Dans la fenêtre Fichiers et configuration, cliquez sur la ligne contenant **L:\sqliosim.mdx**, puis cliquez sur **Remove**.
11. Dans la fenêtre Fichiers et configuration, cliquez sur la ligne contenant **D:\sqliosim.mdx**, puis dans **Size (MB)**, remplacez la valeur 4096 par **100**, remplacez également la valeur de la **Max Size** 8192 par **200** et définissez l'**Increment** sur **20**, puis cliquez sur **Apply**.
12. Sur la ligne contenant **L:\sqliosim.idx**, définissez **Size (MB)** sur **50**, **Max Size** sur **100**, **Increment** sur **10**, puis cliquez sur **Apply**.
13. Remplacez la valeur 300 de **Cycle Duration (sec)** par **60**.
14. Vérifiez que la case à cocher **Delete Files at Shutdown** est activée, puis cliquez sur **OK** pour terminer la configuration.

### ► Tâche 2 : Exécuter SQLIOSIM

1. Dans la fenêtre SQLIOSim du menu **Simulator**, cliquez sur **Start**.
2. Patientez pendant l'exécution du test qui dure environ 1 minute. Lorsque le test s'exécute, notez les informations affichées.
3. Dans la fenêtre SQLIOSim, cliquez sur **OK**.

► **Tâche 3 : Examiner les résultats de l'exécution SQLIOSIM**

1. Si des erreurs sont retournées en rouge, consultez les erreurs.
2. Recherchez le résumé final du fichier **D:\sqliosim.mdx** et notez la **Running Average IO Duration (ms)**.
3. Recherchez le résumé final du fichier **L:\sqliosim.idx** et notez la **Running Average IO Duration (ms)**.



**Remarque :** La valeur retournée pour chaque lecteur dépend de la vitesse du matériel.  
La valeur courante pour le lecteur est 15.

4. Dans la fenêtre **SQLIOSim** du menu **File**, cliquez sur **Exit**.

## Exercice difficile 3 : Vérifier des opérations d'E/S spécifiques (si le temps le permet)

### ► Tâche 1 : Installer l'utilitaire SQLIO

1. Dans le dossier D:\22462A\_Labs\22462A\_02\_PRJ, cliquez avec le bouton droit sur **SQLIO.msi**, puis cliquez sur **Installer**.
2. Dans la fenêtre Welcome to the SQLIO Setup Wizard, cliquez sur **Next**.
3. Dans la fenêtre du License Agreement, prenez connaissance du contrat, cliquez sur la case d'option **I Agree**, puis cliquez sur **Next**.
4. Dans la fenêtre Select Installation Folder, cliquez sur la case d'option **Everyone**, puis cliquez sur **Next**.
5. Dans la fenêtre Confirm Installation, cliquez sur **Next**.
6. Dans la fenêtre Installation Complete, cliquez sur **Close**.

### ► Tâche 2 : Configurer et exécuter l'utilitaire SQLIO

1. Consultez la documentation associée dans le manuel du stagiaire pour obtenir plus d'informations sur les tests SQLIO à effectuer.
2. Dans l'Explorateur Windows, accédez au dossier C:\Program Files (x86)\SQLIO.
3. Double-cliquez sur le fichier **param.txt**. Placez # au début de la ligne **c:\testfile.dat**. Supprimez # du début de la ligne de **d:\testfile.dat**.
4. Dans le menu **Fichier**, cliquez sur **Quitter**.
5. Dans la fenêtre du Bloc-notes, cliquez sur **Enregistrer**.
6. Dans l'Explorateur Windows, double-cliquez sur le fichier **C:\Program Files (x86)\SQLIO\Using SQLIO.rtf**. Notez les exemples et les descriptions des paramètres disponibles pour SQLIO.
7. Fermez le fichier Using SQLIO.rtf dans la fenêtre Wordpad.
8. Dans le menu **Démarrer**, cliquez sur **Tous les programmes**, puis sur **Accessoires**, cliquez avec le bouton droit sur **Invite de commandes**, puis cliquez sur **Exécuter en tant qu'administrateur**.
9. Tapez **CD \ program files (x86)\SQLIO** et appuyez sur Enter.
10. Tapez la commande suivante, puis appuyez sur Enter :

```
sqlio.exe -kR -s60 -fsequential -o8 -b64 -LS -Fparam.txt timeout /T 60
```

11. Analysez le résultat de la commande, notez les valeurs retournées pour les entrées/sorties par seconde et les méga-octets par seconde. Ces valeurs sont généralement référencées en termes d'E/S par seconde et de débit. Notez la latence minimale, maximale et moyenne.

12. Tapez la commande suivante, puis appuyez sur Entrée :

```
sqlio.exe -kW -s60 -frandom -o8 -b8 -LS -Fparam.txt timeout /T 60
```

13. Analysez le résultat de la commande, notez les valeurs retournées pour les entrées/sorties par seconde et les méga-octets par seconde. Ces valeurs sont généralement référencées en termes d'E/S par seconde et de débit. Notez la latence minimale, maximale et moyenne.

14. Tapez **Exit**, puis appuyez sur Entrée.

## Module 3 : Installation et configuration de SQL Server 2012

# Atelier pratique 3 : Installation et configuration de SQL Server

### Configuration de l'atelier pratique

Pour cet atelier pratique, vous utiliserez l'environnement d'ordinateur virtuel disponible. Avant de débuter cet atelier pratique, vous devez procéder aux étapes suivantes :

1. Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
3. Dans la fenêtre Se connecter au serveur, tapez **Proseware** dans la zone de texte **Nom du serveur**.
4. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn.**
5. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**.
6. Dans la fenêtre Ouvrir un projet, sélectionnez le projet **D:\22462A\_Labs\22462A\_03\_PRJ\22462A\_03\_PRJ.ssmssqln**.
7. Dans l'Explorateur de solutions, double-cliquez sur le fichier de requête **00 - Setup.sql**. Lorsque la fenêtre de requête s'affiche, cliquez sur **Exécuter** dans la barre d'outils.
8. Sur le système hôte, dans la liste **Ordinateurs virtuels** du Gestionnaire Hyper-V, cliquez avec le bouton droit sur l'ordinateur virtuel **22462A-MIA-SQL1**, puis cliquez sur **Paramètres**.
9. Dans la fenêtre Paramètres pour 22462A-MIA-SQL1, dans la liste **Matériel**, développez **Contrôleur IDE 1**, puis cliquez sur **Lecteur de DVD**.
10. Dans le volet Propriétés du **Lecteur de DVD**, cliquez sur **Fichier image**, puis sur **Parcourir**.
11. Naviguez jusqu'au fichier **C:\Programmes\Microsoft Learning\22462\Drives\22462A-MIA-SQL1\Virtual Hard Disks\SQLFULL\_FRA.iso**, puis cliquez sur **Ouvrir**.
12. Dans la fenêtre Paramètres pour 22462A-MIA-SQL1, cliquez sur **OK**.

## Exercice 1 : Vérification de la configuration requise pour l'installation

► **Tâche 1 : Examinez la documentation avant de procéder à l'installation**

- Lisez la documentation associée du manuel du stagiaire.

► **Tâche 2 : Créez les dossiers requis pour les fichiers de données et les fichiers journaux**

- À l'aide de l'Explorateur Windows, créez les dossiers suivants :
  - D:\MKTGDEV
  - L:\MKTGDEV

## Exercice 2 : Installation de l'instance SQL Server

- **Tâche 1 : D'après les spécifications vues dans l'exercice 1, installez une autre instance de SQL Server**
1. Dans la fenêtre Ordinateur virtuel, au sein de la fenêtre de lecture automatique (qui devrait normalement s'afficher), cliquez sur **Exécuter SETUP.EXE** et attendez que l'installation de SQL Server démarre. (Si la fenêtre de lecture automatique ne démarre pas ou que vous l'avez accidentellement fermée, double-cliquez sur **SETUP.EXE** à la racine du nouveau lecteur qui a été monté lors de la connexion du fichier ISO avec votre ordinateur virtuel).
  2. Dans la fenêtre Centre d'installation SQL Server, cliquez sur l'onglet **Installation**.
  3. Cliquez sur **Nouvelle installation autonome SQL Server ou ajout de fonctionnalités à une installation existante** à partir de la liste d'options disponibles et attendez que le programme d'installation de SQL Server démarre.
  4. Dans la fenêtre Règles de support du programme d'installation, cliquez sur **Afficher les détails** et notez que la liste des règles a été activée.
  5. Dans la fenêtre Règles de support du programme d'installation, cliquez sur **OK**.
  6. Vérifiez que l'option **Inclure les mises à jour du produit SQL Server** n'est pas sélectionnée et cliquez sur **Suivant**.



**Remarque** En général vous êtes encouragé à choisir cette fonctionnalité car elle permet à Microsoft d'apporter des améliorations au produit. Dans ce cas précis, elle est désélectionnée car nous sommes dans un environnement isolé sans accès réseau sur l'ordinateur virtuel.

7. Dans la fenêtre Mises à jour du produit, désactivez **Inclure les mises à jour du produit SQL Server**, puis cliquez sur **Suivant**. (Notez que l'écran suivant peut prendre un certain temps avant de s'afficher).
8. Dans la page **Règles de support du programme d'installation**, cliquez sur **Afficher les détails** (si visible) et lisez la liste des tâches à vérifier pour l'installation et vérifiez que l'état des éléments est **Réussite**. Notez les moindres avertissements répertoriés et cliquez sur **Suivant**.
9. Dans la page **Type d'installation**, assurez-vous que le bouton de l'option **Effectuer une nouvelle installation de SQL Server 2012** est sélectionné, puis cliquez sur **Suivant**.
10. Dans la fenêtre Clé de produit (Product Key), cliquez sur **Suivant**.
11. Dans la page **Termes du contrat de licence**, notez les termes du contrat de licence logiciel Microsoft et activez la case à cocher **J'accepte les termes du contrat de licence**. Cliquez sur **Suivant**.
12. Dans la page **Rôle d'installation**, vérifiez que l'option **Installation de fonctionnalités SQL Server** est sélectionnée, puis cliquez sur **Suivant**.
13. Dans la page **Sélection de fonctionnalités**, sous contrôle **Fonctionnalités de l'instance**, activez la case **Services Moteur de base de données**, puis cliquez sur **Suivant**.
14. Dans la page **Règles d'installation**, cliquez sur **Afficher les détails**, notez la liste des règles et l'état de chaque règle, puis cliquez sur **Suivant**.

15. Dans la page **Configuration de l'instance**, vérifiez que l'option **Instance nommée** est activée et tapez **MKTGDEV** dans le champ **Instance nommée**, puis cliquez sur **Suivant**.
16. Dans la page **Espace disque nécessaire**, lisez le **Résumé sur l'utilisation du disque**, puis cliquez sur **Suivant**.
17. Dans la page **Configuration du serveur**, cliquez sur le champ **Nom du compte** de la ligne **Moteur de base de données SQL Server** et sélectionnez **Parcourir...** dans la liste déroulante.
18. Dans la fenêtre de sélection de l'utilisateur, de l'ordinateur, du compte de service ou du groupe, au sein de la boîte de dialogue **Entrez le nom de l'objet à sélectionner (exemples)** : tapez **AdventureWorks\PWService**, cliquez sur **Vérifier les noms**, puis cliquez sur **OK**.
19. Dans la colonne **Mot de passe** de la ligne **Moteur de base de données SQL Server**, tapez **Pa\$\$w0rd**.
20. Dans la page **Configuration du serveur**, cliquez sur le champ **Nom du compte** de la ligne **Agent SQL Server** et sélectionnez **Parcourir...** dans la liste déroulante.
21. Dans la fenêtre de sélection de l'utilisateur, de l'ordinateur, du compte de service ou du groupe, au sein de la boîte de dialogue **Entrez le nom de l'objet à sélectionner (exemples)** : tapez **AdventureWorks\PWService**, cliquez sur **Vérifier les noms**, puis cliquez sur **OK**.
22. Dans la colonne **Mot de passe** de la ligne **Agent SQL Server**, tapez **Pa\$\$w0rd**.
23. Dans la page **Configuration du serveur**, sous l'onglet **Comptes de service**, et dans la liste déroulante **Type de démarrage** pour l'**Agent SQL Server**, sélectionnez **Automatique**.
24. Cliquez sur l'onglet **Classement**, vérifiez que l'option **French\_CI\_AS** est sélectionnée et cliquez sur **Suivant**.
25. Dans la page **Configuration du moteur de base de données**, sous l'onglet **Configuration du serveur**, dans la section **Mode d'authentification**, vérifiez que l'option **Mode mixte (authentification SQL Server et authentification Windows)** est sélectionnée.
26. Dans la zone de texte **Entrer le mot de passe**, tapez **Pa\$\$w0rd**.
27. Dans la zone de texte **Confirmer le mot de passe**, tapez **Pa\$\$w0rd**.
28. Cliquez sur **Ajouter l'utilisateur actuel**, ce qui ajoute l'utilisateur **ADVENTUREWORKS\Administrateur (Administrateur)** à la liste des administrateurs.
29. Cliquez sur l'onglet **Répertoires de données**, remplacez **Répertoire de la base de données utilisateur** par **D:\MKTGDEV**.
30. Remplacez le **Répertoire de journal de base de données utilisateur** par **L:\MKTGDEV**.
31. Remplacez le **Répertoire de base de données temporaire** par **D:\MKTGDEV**.
32. Remplacez le **Répertoire de journal de base de données temporaire** par **L:\MKTGDEV**.
33. Cliquez sur l'onglet **FILESTREAM** et vérifiez que l'option **Activer FILESTREAM pour l'accès Transact-SQL** n'est pas sélectionnée, puis cliquez sur **Suivant**.
34. Dans la page **Création de rapports d'erreurs**, vérifiez que **Envoyer des rapports d'erreurs Windows et SQL Server à Microsoft ou votre serveur de rapports d'entreprise** n'est pas activé, puis cliquez sur **Suivant**.

35. Dans la page **Règles de configuration de l'installation**, sous **Afficher les détails**, examinez la liste des règles, puis cliquez sur **Suivant**.
36. Dans la page **Prêt pour l'installation**, lisez le résumé et cliquez sur **Installer**.
37. Dans la page **Terminée**, cliquez sur **Fermer**.
38. Fermez la fenêtre Centre d'installation SQL Server.

## Exercice 3 : Exécution du programme et des contrôles post-installation

- **Tâche 1 : Vérifier que les services pour l'instance de SQL Server s'exécutent**
1. Cliquez successivement sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012, Outils de configuration**, puis sur **Gestionnaire de configuration SQL**.
  2. Dans le volet gauche de la fenêtre SQL Server Configuration Manager, cliquez sur **Services SQL Server**.
  3. Dans le volet droit, vérifiez que les services suivants sont répertoriés comme étant en cours d'exécution pour l'instance **MKTGDEV** instance :
    - SQL Server (MKTGDEV)
    - Agent SQL Server (MKTGDEV)
- **Tâche 2 : Configurer les alias 32 bits et 64 bits de la nouvelle instance**
1. Dans le volet gauche de la fenêtre SQL Server Configuration Manager, développez **Configuration du réseau SQL Server**, puis cliquez sur **Protocoles pour MKTGDEV**.
  2. Dans le volet droit, cliquez avec le bouton droit sur **Named Pipes**, puis sélectionnez **Activer**.
  3. Dans la fenêtre Avertissement, cliquez sur **OK**.
  4. Dans le volet gauche de la fenêtre SQL Server Configuration Manager, cliquez sur **Services SQL Server**.
  5. Cliquez avec le bouton droit sur **SQL Server (MKTGDEV)** et sélectionnez **Redémarrer**.
  6. Dans le volet gauche de la fenêtre SQL Server Configuration Manager, cliquez sur **Services SQL Server**.
  7. Dans la barre d'outils, cliquez sur l'icône **Actualiser**.
  8. Dans le volet gauche de la fenêtre SQL Server Configuration Manager, développez **Configuration de SQL Native Client 11.0 (32 bits)**, puis cliquez sur **Protocoles clients**.
  9. Vérifiez que le protocole **Canaux nommés** est **Activé**.
  10. Dans le volet gauche, cliquez avec le bouton droit sur **Alias** et sélectionnez **Nouvel alias**.
  11. Dans la fenêtre Alias - Nouveau, dans la zone de texte **Nom de l'alias**, tapez **PWDev**.
  12. Dans la zone de liste déroulante **Protocole**, sélectionnez **Named Pipes**.
  13. Dans la zone de texte **Serveur**, tapez **.\\MKTGDEV**, puis cliquez sur **OK**.
  14. Dans le volet gauche de la fenêtre SQL Server Configuration Manager, développez **Configuration de SQL Native Client 11.0** puis cliquez sur **Protocoles clients**.
  15. Vérifiez que le protocole **Canaux nommés** est **Activé**.
  16. Dans le volet gauche, cliquez avec le bouton droit sur **Alias** et sélectionnez **Nouvel alias**.
  17. Dans la fenêtre Alias - Nouveau, dans la zone de texte **Nom de l'alias**, tapez **PWDev**.
  18. Dans la zone de liste déroulante **Protocole**, sélectionnez **Named Pipes**.
  19. Dans la zone de texte **Serveur**, tapez **.\\MKTGDEV**, puis cliquez sur **OK**.
  20. Fermez SQL Server Configuration Manager.

► **Tâche 3 : Vous connecter à la nouvelle instance au moyen de SSMS**

1. Cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
2. Dans la fenêtre Se connecter au serveur, vérifiez que **Type de serveur** a la valeur **Moteur de base de données**.
3. Dans la zone de texte **Nom du serveur**, tapez **PWDev**.
4. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn.**
5. Dans l'Explorateur d'objets, sous **PWDev (SQL Server 11.0.2100-ADVENTUREWORKS\Administrateur)**, développez **Bases de données**.
6. Fermez Microsoft SQL Server Management Studio (Administrateur).

## Exercice difficile 4 : Configurer la mémoire du serveur (si le temps le permet)

- Tâche 1 : Consulter la mémoire actuelle disponible sur le serveur
1. Cliquez sur **Démarrer**, cliquez avec le bouton droit sur **Ordinateur**, puis cliquez sur **Propriétés**.
  2. Notez la valeur de la **Mémoire installée (RAM)**.
- Tâche 2 : Déterminer une allocation de mémoire appropriée pour chaque instance
1. Examinez la **Configuration de la mémoire requise** dans la document associée au manuel du stagiaire.
  2. Calculez la Mémoire maximale pour l'instance de serveur **AdventureWorks** comme suit :
    - Exemple de calcul (les valeurs réelles dépendent de la configuration de l'ordinateur virtuel)
    - Mémoire maximale = (Mémoire serveur – 1.0 Go) \* 0,4
    - Mémoire maximale = (3,0 – 1,0) \* 0,4
    - Mémoire maximale = 0,8 (approximatif)
  3. Calculez la Mémoire maximale pour l'instance de serveur **Proseware** comme suit :
    - Exemple de calcul (les valeurs réelles dépendent de la configuration de l'ordinateur virtuel)
    - Mémoire maximale = (Mémoire serveur – 1.0 Go) \* 0,3
    - Mémoire maximale = (3,0 – 1,0) \* 0,3
    - Mémoire maximale = 0,6 (approximatif)
  4. Calculez la Mémoire maximale pour l'instance de serveur **PWDev** comme suit :
    - Exemple de calcul (les valeurs réelles dépendent de la configuration de l'ordinateur virtuel)
    - Mémoire maximale = (Mémoire serveur – 1.0 Go) \* 0,3
    - Mémoire maximale = (3,0 – 1,0) \* 0,3
    - Mémoire maximale = 0,6 (approximatif)
- Tâche 3 : Configurer chaque instance de manière adéquate
1. Cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
  2. Dans la fenêtre Se connecter au serveur, tapez **AdventureWorks** pour le Nom du serveur, puis cliquez sur **Se conn.**
  3. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur l'instance de serveur **AdventureWorks (SQL Server 11.0.2100-ADVENTUREWORKS\Administrateur)**, puis cliquez sur **Propriétés**.
  4. Dans le volet **Sélectionner une page**, cliquez sur **Mémoire**.
  5. Définissez la **Mémoire maximale du serveur (en Mo)** sur la valeur **800**, puis cliquez sur **OK**.



**Remarque** Il aurait été préférable d'utiliser une valeur plus précise, mais la valeur 800 a été utilisée par souci de simplicité.

6. Dans l'Explorateur d'objets, cliquez sur **Connecter**, puis sur **Moteur de base de données**.
7. Dans la fenêtre Se connecter au serveur, tapez **Proseware** pour le **Nom du serveur**, puis cliquez sur **Se conn.**
8. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur l'instance de serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\Administrateur)**, puis cliquez sur **Propriétés**.
9. Dans le volet **Sélectionner une page**, cliquez sur **Mémoire**.
10. Définissez la **Mémoire maximale du serveur (en Mo)** sur la valeur **600**, puis cliquez sur **OK**.

 **Remarque** Une valeur plus précise pour 1 G0 aurait pu être utilisée, mais la valeur 600 a été préférée par souci de simplicité.

11. Dans l'Explorateur d'objets, cliquez sur **Connecter**, puis sur **Moteur de base de données**.
12. Dans la fenêtre Se connecter au serveur, tapez **PWDev** pour le **Nom du serveur**, puis cliquez sur **Se conn.**
13. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur l'instance de serveur **PWDev (SQL Server 11.0.2100-ADVENTUREWORKS\Administrateur)**, puis cliquez sur **Propriétés**.
14. Dans le volet **Sélectionner une page**, cliquez sur **Mémoire**.
15. Définissez la **Mémoire maximale du serveur (en Mo)** sur la valeur **600**, puis cliquez sur **OK**.

 **Remarque** Une valeur plus précise pour 1 G0 aurait pu être utilisée, mais la valeur 600 a été préférée par souci de simplicité.

16. Fermez Microsoft SQL Server Management Studio (Administrateur).



## Module 4 : Utilisation des bases de données

# Atelier pratique 4 : Utilisation des bases de données

### Configuration de l'atelier pratique

Pour cet atelier pratique, vous utiliserez l'environnement d'ordinateur virtuel disponible. Avant de débuter cet atelier pratique, vous devez procéder aux étapes suivantes :

1. Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
3. Dans la fenêtre Se connecter au serveur, tapez **Proseware** dans la zone de texte **Nom du serveur**.
4. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn.**.
5. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**.
6. Dans la fenêtre Ouvrir un projet, sélectionnez le projet **D:\22462A\_Labs\22462A\_04\_PRJ\22462A\_04\_PRJ.ssmssln**.
7. Dans l'Explorateur de solutions, double-cliquez sur le fichier de requête **00 - Setup.sql**.  
Lorsque la fenêtre de requête s'affiche, cliquez sur **Exécuter** dans la barre d'outils.

## Exercice 1 : Modifier la configuration de la base de données tempdb

### ► Tâche 1 : Modifiez la taille de tempdb

1. Prenez connaissance des éléments requis pour la **taille de tempdb** fournis dans la documentation associée.
2. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100 – ADVENTUREWORKS\administrateur)**, développez **Bases de données**, puis **Bases de données système**. Cliquez avec le bouton droit sur la base de données **master** et cliquez sur **Nouvelle requête**.
3. Entrez les instructions suivantes et cliquez sur **Exécuter** :

```
USE master;
GO
ALTER DATABASE tempdb
    MODIFY FILE ( NAME = tempdev, SIZE = 30MB );
GO
ALTER DATABASE tempdb
    MODIFY FILE ( NAME = templog, SIZE = 10MB );
GO
```

### ► Tâche 2 : Vérifier que la taille de tempdb est toujours correcte après un redémarrage

1. Cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, sur **Outils de configuration**, puis sur **Gestionnaire de configuration SQL Server**.
2. Dans le volet gauche de la fenêtre Gestionnaire de configuration SQL Server Configuration Manager, cliquez sur **Services SQL Server**.
3. Dans le volet droit, cliquez avec le bouton droit sur le service **SQL Server (MKTG)**, puis cliquez sur **Redémarrer**.
4. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur la base de données **tempdb**, puis cliquez sur **Propriétés**.
5. Dans la fenêtre Propriétés de la base de données – tempdb, cliquez sur **Fichiers** dans le volet **Sélectionner une page**.
6. Dans la liste **Fichiers de la base de données**, vérifiez que la Taille initiale (Mo) du fichier de données est **30** et que la taille initiale du fichier journal est **10**.

## Exercice 2 : Créer la base de données RateTracking

### ► Tâche 1 : Créer la base de données

1. Prenez connaissance des éléments requis fournis dans la documentation associée de l'exercice.
2. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100 – ADVENTUREWORKS\administrateur)**, développez **Bases de données**, puis **Bases de données système**. Cliquez avec le bouton droit sur la base de données **master** et cliquez sur **Nouvelle requête**.
3. Entrez les instructions suivantes et cliquez sur **Exécuter** :

```
USE master;
GO

CREATE DATABASE RateTracking
ON
( NAME = RateTracking_dat,
  FILENAME = 'D:\MKTG\RateTracking.mdf',
  SIZE = 10MB, MAXSIZE = 100MB,
  FILEGROWTH = 10MB
)
LOG ON
( NAME = RateTracking_log,
  FILENAME = 'L:\MKTG\RateTracking.ldf',
  SIZE = 20MB,
  MAXSIZE = UNLIMITED,
  FILEGROWTH = 20MB
);
GO
```

### ► Tâche 2 : Créer les fichiers et groupes de fichiers requis

1. Prendre connaissance des spécifications indiquées dans la documentation associée relative aux fichiers et groupes de fichiers requis.
2. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100 – ADVENTUREWORKS\administrateur)**, développez **Bases de données**, puis **Bases de données système**. Cliquez avec le bouton droit sur la base de données **master** et cliquez sur **Nouvelle requête**.
3. Entrez les instructions suivantes et cliquez sur **Exécuter** :

```
USE master;
GO
ALTER DATABASE RateTracking
ADD FILEGROUP USERDATA;
GO
ALTER DATABASE RateTracking
ADD FILE
( NAME = RateTracking_dat_1,
  FILENAME = 'D:\MKTG\RateTracking_1.ndf',
  SIZE = 20MB,
  MAXSIZE = 100MB,
  FILEGROWTH = 10MB
) TO FILEGROUP USERDATA;
GO

ALTER DATABASE RateTracking
ADD FILE
```

```
( NAME = RateTracking_dat_2,
  FILENAME = 'D:\MKTG\RateTracking_2.ndf',
  SIZE = 20MB,
  MAXSIZE = 100MB,
  FILEGROWTH = 10MB
) TO FILEGROUP USERDATA;
GO

ALTER DATABASE RateTracking
ADD FILEGROUP ARCHIVE;
GO

ALTER DATABASE RateTracking
ADD FILE
( NAME = RateTracking_dat_3,
  FILENAME = 'D:\MKTG\RateTracking_3.ndf',
  SIZE = 200MB,
  MAXSIZE = 500MB,
  FILEGROWTH = 50MB
) TO FILEGROUP ARCHIVE;
GO

ALTER DATABASE RateTracking
ADD FILE
( NAME = RateTracking_dat_4,
  FILENAME = 'D:\MKTG\RateTracking_4.ndf',
  SIZE = 200MB,
  MAXSIZE = 500MB,
  FILEGROWTH = 50MB
) TO FILEGROUP ARCHIVE;
GO
```

► **Tâche 3 : Changer le groupe de fichiers par défaut de la base de données**

1. Prenez connaissance des éléments requis fournis dans la documentation associée pour le groupe de fichiers par défaut.
2. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100 – ADVENTUREWORKS\administrateur)**, développez **Bases de données**, puis **Bases de données système**. Cliquez avec le bouton droit sur la base de données **master** et cliquez sur **Nouvelle requête**.
3. Entrez les instructions suivantes et cliquez sur **Exécuter** :

```
USE master;
GO

ALTER DATABASE RateTracking MODIFY FILEGROUP USERDATA DEFAULT;
GO
```

### Exercice 3 : Attacher la base de données OldProspects

#### ► Tâche 1 : Copier les fichiers de la base de données

- Dans l'Explorateur Windows, copiez les fichiers comme indiqué ci-dessous :

Nom de fichier	Dossier source	Dossier de destination
OldProspects.mdf	D:\22462A_Labs\22462A_04_PRJ	D:\MKTG
OldProspects.ldf	D:\22462A_Labs\22462A_04_PRJ	L:\MKTG

#### ► Tâche 2 : Attacher la base de données à l'instance de MKTG

- Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100 – ADVENTUREWORKS\administrateur)** et développez **Bases de données**.
- Cliquez avec le bouton droit sur **Bases de données**, puis cliquez sur **Joindre**.
- Cliquez sur **Ajouter** et accédez au fichier **D:\MKTG\OldProspects.mdf**, puis cliquez sur **OK**.
- Cliquez sur l'ellipse en regard de **Chemin d'accès au fichier actuel** de l'entrée de journal, puis accédez au fichier **L:\MKTG\OldProspects.ldf** et cliquez sur **OK**.
- Cliquez sur **OK** pour joindre la base de données. Notez que **OldProspects** apparaît maintenant dans la liste des Bases de données dans l'**Explorateur d'objets**.

## Exercice difficile 4 : Ajouter plusieurs fichiers à la base de données tempdb (si le temps le permet)

### ► Tâche 1 : Consulter les spécifications du fichier tempdb

- Dans la documentation associée, vérifiez les éléments requis pour tempdb dans la section Consultant.

### ► Tâche 2 : Déplacer les fichiers existants

- Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100 – ADVENTUREWORKS\administrateur)**, développez **Bases de données**, puis **Bases de données système**. Cliquez avec le bouton droit sur la base de données **master** et cliquez sur **Nouvelle requête**.

- Entrez les instructions suivantes et cliquez sur **Exécuter** :

```
USE master;
GO

ALTER DATABASE tempdb
MODIFY FILE (NAME = tempdev, FILENAME = 'D:\MKTG\tempdb.mdf');

ALTER DATABASE tempdb
MODIFY FILE (NAME = templog, FILENAME = 'L:\MKTG\templog.ldf');
GO
```

### ► Tâche 3 : Ajouter de nouveaux fichiers

- Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100 – ADVENTUREWORKS\administrateur)**, développez **Bases de données**, puis **Bases de données système**. Cliquez avec le bouton droit sur la base de données **master** et cliquez sur **Nouvelle requête**.

- Entrez les instructions suivantes et cliquez sur **Exécuter** :

```
USE master;
GO

ALTER DATABASE tempdb
ADD FILE ( NAME = N'tempdev2',
            FILENAME = N'D:\MKTG\tempdb_file2.ndf' ,
            SIZE = 20MB,
            FILEGROWTH = 10MB,
            MAXSIZE = UNLIMITED );

ALTER DATABASE tempdb
ADD FILE ( NAME = N'tempdev3',
            FILENAME = N'D:\MKTG\tempdb_file3.ndf' ,
            SIZE = 20MB,
            FILEGROWTH = 10MB,
            MAXSIZE = UNLIMITED );
ALTER DATABASE tempdb
ADD FILE ( NAME = N'tempdev4',
            FILENAME = N'D:\MKTG\tempdb_file4.ndf' ,
            SIZE = 20MB,
            FILEGROWTH = 10MB,
            MAXSIZE = UNLIMITED );
GO
```

► **Tâche 4 : Redémarrer le serveur et vérifier les emplacements de fichiers**

1. Cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, sur **Outils de configuration**, puis sur **Gestionnaire de configuration SQL Server**.
2. Dans le volet gauche de la fenêtre SQL Server Configuration Manager, cliquez sur **Services SQL Server**.
3. Dans le volet droit, cliquez avec le bouton droit sur le service **SQL Server (MKTG)**, puis cliquez sur **Redémarrer**.
4. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur la base de données **tempdb**, puis cliquez sur **Propriétés**.
5. Dans la fenêtre Propriétés de la base de données – tempdb, cliquez sur **Fichiers** dans le volet **Sélectionner une page**.
6. Assurez-vous que les fichiers répertoriés correspondent aux exigences.



## Module 5 : Description des modes de récupération SQL Server 2012

# Atelier pratique 5 : Description des modes de récupération SQL Server

### Configuration de l'atelier pratique

Pour cet atelier pratique, vous utiliserez l'environnement d'ordinateur virtuel disponible. Avant de débuter cet atelier pratique, vous devez procéder aux étapes suivantes :

1. Rétablissez les ordinateurs virtuels conformément aux instructions dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
3. Dans la fenêtre Se connecter au serveur, tapez **Proseware** dans la zone de texte **Nom du serveur**.
4. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn.**
5. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**.
6. Dans la fenêtre Ouvrir un projet, ouvrez le projet **D:\22462A\_Labs\22462A\_05\_PRJ\22462A\_05\_PRJ.ssmssqln**.
7. Dans l'Explorateur de solutions, double-cliquez sur le fichier de requête **00 - Setup.sql**.  
Lorsque la fenêtre de requête s'affiche, cliquez sur **Exécuter** dans la barre d'outils.

## Exercice 1 : Planifier une stratégie de sauvegarde

### ► Tâche 1 : Étudier les besoins de l'activité

- Prenez connaissance des besoins de l'activité professionnelle fournis dans la documentation de l'exercice.

### ► Tâche 2 : Déterminer une stratégie de sauvegarde appropriée pour chaque base de données

- Il n'y a pas qu'une seule réponse correcte. Les options proposées ci-dessous sont basées sur une stratégie d'exemple qui répond aux conditions.

Pour la base de données MarketDev : **Mode de récupération complète**

Type de sauvegarde	Planifiée
Complète	Quotidienne 7:00 pm
Journalisation	Toutes les 20 minutes, démarrage à 08h20 et reprise jusqu'à 6:00 pm

#### Remarques :

- La sauvegarde complète de la base de données doit s'effectuer en 3,4 heures environ (20 Go/100 Mo par minute). Cela signifie que vous ne pouvez pas utiliser qu'une stratégie de sauvegarde complète de la base de données car elle ne respecterait pas les conditions RPO. La sauvegarde complète de la base de données doit se terminer à 22h24, ce qui respecte la fenêtre de temps disponible pour les sauvegardes.
- Chaque sauvegarde de journal doit s'exécuter en environ 3,4 minutes (1 Go par heure / 3 sauvegardes de fichiers journaux par heure /100 Mo par minute). Cela entre dans l'intervalle de 20 minutes et répond aux exigences RPO.
- La restauration complète de la base de données doit s'effectuer en 4,3 heures environ (20 Go/80 Mo par minute). La restauration du fichier journal doit s'exécuter en 2,1 heures environ (10 heures \* 1 Go par heure /80 Mo par minute) et répond aux exigences RTO.

Pour la base de données de recherche : **Mode de récupération simple**

Type de sauvegarde	Planifiée
Complète	Quotidienne 6:30 pm

#### Remarques :

- La récupération à la dernière sauvegarde de base de données quotidienne complète est conforme aux exigences RPO.
- La sauvegarde quotidienne doit se terminer en environ 2 minutes (200 Mo/100 Mo par minute).
- La récupération complète doit se terminer en environ 2,5 minutes et se conformer à RTO (200 Mo/80 Mo par minute).

## Exercice 2 : Configurer les modes de récupération

### ► Tâche 1 : Examiner et ajuster les modes de récupération des bases de données

1. Le mode de récupération de la base de données MarketDev est simple et doit être **Complet**. Ajustez le mode à l'aide de la procédure suivante :
  1. Dans SSMS, dans l'Explorateur d'objets développez le serveur **Proseware (SQL Server 11.0.2100 – ADVENTUREWORKS\administrateur)**, cliquez avec le bouton droit sur **Bases de données**, puis cliquez sur **Actualiser**.
  2. Développez **Bases de données**, cliquez avec le bouton droit sur **MarketDev**, puis cliquez sur **Propriétés**.
  3. Dans le volet **Sélectionner une page**, cliquez sur **Options**.
  4. Dans le volet droit, sélectionnez **Complet** dans la liste déroulante **Mode de récupération**, puis cliquez sur **OK**.
2. Le mode de récupération de la base de données de recherche est correct et n'a pas besoin d'être modifié.

### Exercice difficile 3 : examiner les modes et les stratégies de récupération (si le temps le permet)

► **Tâche 1 : Vérifier les éléments requis de RPO et de RTO pour les bases de données**

- La documentation associée inclut les détails des spécifications de suivi de l'activité pour les bases de données. Vous devez examiner cette documentation.

► **Tâche 2 : Examiner les modes de récupération et les stratégies de sauvegarde existants**

- La documentation associée contient également des informations sur la sauvegarde des bases de données. Vous devez examiner cette documentation.

► **Tâche 3 : Indiquer si la stratégie sera réussie ou non**

1. Pour la base de données CreditControl, une sauvegarde complète prendrait environ 3,4 heures (20 Go/100 Mo par minute). Cela ne répondrait pas aux exigences de la fenêtre de temps, puisque la sauvegarde complète du mercredi 06h00 ne s'achèverait pas avant le début des heures de bureau.
2. Pour la base de données PotentialIssue, les sauvegardes de journaux de 15 minutes respecteraient les exigences RPO. Une restauration complète prendrait environ 24 minutes [(200 Mo + (7 jours \* 24 heures) \* 10 Mo par heure) / 80 Mo par minute)], ce qui répond aux exigences RTO. La sauvegarde complète de la base de données s'achèverait en environ 2 minutes (200 Mo/100 Mo par minute), ce qui signifie que la base de données complète s'achèverait dans la fenêtre de temps disponible. La stratégie de sauvegarde de la base de données PotentialIssue répond aux besoins de l'activité professionnelle.

## Module 6 : Sauvegarde des bases de données SQL Server

# Atelier pratique 6 : Sauvegarde des bases de données SQL Server

### Configuration de l'atelier pratique

Pour cet atelier pratique, vous utiliserez l'environnement d'ordinateur virtuel disponible.

Avant de commencer cet atelier pratique, vous devez procéder aux étapes suivantes :

1. Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
3. Dans la fenêtre Se connecter au serveur, tapez **Proseware** dans la zone de texte **Nom du serveur**.
4. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn**.
5. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**.
6. Dans la fenêtre Ouvrir un projet, ouvrez le projet **D:\22462A\_Labs\22462A\_06\_PRJ\22462A\_06\_PRJ.ssmssln**.
7. Dans l'Explorateur de solutions, double-cliquez sur le fichier de requête **00 - Setup.sql**.  
Lorsque la fenêtre de requête s'affiche, cliquez sur **Exécuter** dans la barre d'outils.

## Exercice 1 : Déterminer la compression de sauvegarde

### ► Tâche 1 : Créer une sauvegarde de base de données sans compression

1. À l'aide de l'Explorateur Windows, créez un nouveau dossier **L:\SQLBackups**.
2. Dans l'Explorateur d'objets, dans SQL Server Management Studio, développez le serveur **Proseware (SQL Server 11.0.2100 –ADVENTUREWORKS\administrateur)**, développez **Bases de données**, cliquez avec le bouton droit sur la base de données **MarketDev**, cliquez sur **Tâches**, puis cliquez sur **Sauvegarder**.
3. Dans la fenêtre Sauvegarder la base de données – MarketDev, cliquez sur **Supprimer**.
4. Dans la fenêtre Sauvegarder la base de données – MarketDev, cliquez sur **Ajouter**.
5. Dans la zone de texte **Nom de fichier**, tapez **L:\SQLBackups\MarketDev\_Full\_Uncompressed.BAK**, puis cliquez sur **OK**.
6. Dans le volet **Sélectionner une page**, cliquez sur **Options**.
7. Dans la liste déroulante **Définissez la compression de la sauvegarde**, cliquez sur **Ne pas compresser la sauvegarde**, puis cliquez sur **OK**.
8. Dans la fenêtre Microsoft SQL Server Management Studio, cliquez sur **OK**.

### ► Tâche 2 : Créer une sauvegarde de base de données avec compression

1. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur la base de données **MarketDev**, cliquez sur **Tâches**, puis cliquez sur **Sauvegarder**.
2. Dans la fenêtre Sauvegarder la base de données – MarketDev, cliquez sur **Supprimer**.
3. Dans la fenêtre Sauvegarder la base de données – MarketDev, cliquez sur **Ajouter**.
4. Dans la zone de texte **Nom de fichier**, tapez **L:\SQLBackups\MarketDev\_Full\_Compressed.BAK**, puis cliquez sur **OK**.
5. Dans le volet **Sélectionner une page**, cliquez sur **Options**.
6. Dans la liste déroulante **Définissez la compression de la sauvegarde**, cliquez sur **Compresser la sauvegarde**, puis cliquez sur **OK**.
7. Dans la fenêtre Microsoft SQL Server Management Studio, cliquez sur **OK**.

### ► Tâche 3 : Comparer les tailles de fichier créées

1. Dans l'Explorateur Windows, recherchez le dossier **L:\SQLBackups** et notez la taille des fichiers de sauvegarde compressés et décompressés.
2. Calculez l'économie d'espace disponible par compression comme suit :

```
SpaceSavings=(Uncompressed size-Compressed size)*100/Uncompressed size
```



**Remarque** Une valeur courante correspondrait à  $(233-66)*100/233 = 72\%$  d'économie.

## Exercice 2 : Sauvegarde des journaux des transactions

- Tâche 1 : Exécuter un script pour introduire la charge de travail dans la base de données MarketDev
1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **61 - Workload File.sql** pour l'ouvrir.
  2. Dans la **barre d'outils**, cliquez sur **Exécuter**.
- Tâche 2 : Sauvegarder le journal des transactions dans la base de données MarketDev
1. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur la base de données **MarketDev**, cliquez sur **Tâches**, puis cliquez sur **Sauvegarder**.
  2. Dans la fenêtre Sauvegarder la base de données – MarketDev, cliquez sur **Supprimer**.
  3. Dans la fenêtre Sauvegarder la base de données – MarketDev, cliquez sur **Ajouter**.
  4. Dans la zone de texte **Nom de fichier**, tapez **L:\SQLBackups\MarketDev\_Log\_Compressed.BAK**, puis cliquez sur **OK**.
  5. Dans la zone de liste déroulante **Type de sauvegarde**, sélectionnez **Journal des transactions**.
  6. Dans le volet **Sélectionner une page**, cliquez sur **Options**.
  7. Dans la liste déroulante **Définissez la compression de la sauvegarde**, cliquez sur **Compresser la sauvegarde**, puis cliquez sur **OK**.
  8. Dans la fenêtre Microsoft SQL Server Management Studio, cliquez sur **OK**.

### Exercice 3 : Sauvegarde différentielle

- **Tâche 1 : Exécuter un script pour introduire la charge de travail dans la base de données MarketDev**
1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **61 - Workload File.sql** pour l'ouvrir.
  2. Dans la **barre d'outils**, cliquez sur **Exécuter**.
- **Tâche 2 : Créer une sauvegarde différentielle de la base de données MarketDev**
1. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur la base de données **MarketDev**, cliquez sur **Tâches**, puis cliquez sur **Sauvegarder**.
  2. Dans la fenêtre Sauvegarder la base de données – MarketDev, cliquez sur **Supprimer**.
  3. Dans la fenêtre Sauvegarder la base de données – MarketDev, cliquez sur **Ajouter**.
  4. Dans la zone de texte **Nom de fichier**, tapez **L:\SQLBackups\MarketDev\_Differential\_Compressed.BAK**, puis cliquez sur **OK**.
  5. Dans la zone de liste déroulante **Type de sauvegarde**, sélectionnez **Différentielle**.
  6. Dans le volet **Sélectionner une page**, cliquez sur **Options**.
  7. Dans la liste déroulante **Définissez la compression de la sauvegarde**, cliquez sur **Compresser la sauvegarde**, puis cliquez sur **OK**.
  8. Dans la fenêtre Microsoft SQL Server Management Studio, cliquez sur **OK**.
  9. Dans l'Explorateur Windows, notez la taille de la sauvegarde différentielle **L:\SQLBackups\MarketDev\_Differential\_Compressed.BAK** par rapport à la sauvegarde complète **L:\SQLBackups\MarketDev\_Full\_Compressed.BAK**.



**Remarque** Une valeur courante pour la taille de la sauvegarde différentielle pourrait correspondre à 240 Ko. Une valeur courante pour la taille de sauvegarde complète serait 66 Mo.

- **Tâche 3 : Exécuter un script pour introduire la charge de travail dans la base de données MarketDev**

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **71 - Workload File.sql** pour l'ouvrir.
2. Dans la **barre d'outils**, cliquez sur **Exécuter**.

- **Tâche 4 : Ajouter une sauvegarde différentielle dans le fichier de sauvegarde différentielle précédent**

1. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur la base de données **MarketDev**, cliquez sur **Tâches**, puis cliquez sur **Sauvegarder**.
2. Vérifiez que **Description** indique **L:\SQLBackups\MarketDev\_Differential\_Compressed.BAK**.
3. Dans la zone de liste déroulante **Type de sauvegarde**, sélectionnez **Différentielle**.
4. Dans le volet **Sélectionner une page**, cliquez sur **Options**.

5. Dans l'option **Remplacer le support**, vérifiez que **Ajouter au jeu de sauvegarde existant** est sélectionné.
6. Dans la liste déroulante **Définissez la compression de la sauvegarde**, cliquez sur **Compresser la sauvegarde**, puis cliquez sur **OK**.
7. Dans la fenêtre Microsoft SQL Server Management Studio, cliquez sur **OK**.
8. À l'aide de l'Explorateur Windows, notez que la taille de la **sauvegarde différentielle** a augmenté. Le fichier contient à présent deux sauvegardes.



**Remarque** Une valeur courante pour la taille du fichier de sauvegarde pourrait correspondre à 54 Mo.

## Exercice 4 : Sauvegarde de copie uniquement

- Tâche 1 : Créer une sauvegarde de copie uniquement de la base de données MarketDev, en optant pour la vérification de la sauvegarde
1. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur la base de données **MarketDev**, cliquez sur **Tâches**, puis cliquez sur **Sauvegarder**.
  2. Dans la fenêtre Sauvegarder la base de données – MarketDev, cliquez sur **Supprimer**.
  3. Dans la fenêtre Sauvegarder la base de données – MarketDev, cliquez sur **Ajouter**.
  4. Dans la zone de texte **Nom de fichier**, tapez **L:\SQLBackups\MarketDev\_Copy\_Compressed.BAK**, puis cliquez sur **OK**.
  5. Dans la zone de liste déroulante **Type de sauvegarde**, sélectionnez **Complète**.
  6. Vérifiez que l'option **Sauvegarde de copie** uniquement est activée.
  7. Dans le volet **Sélectionner une page**, cliquez sur **Options**.
  8. Dans les options **Remplacer le support**, sélectionnez **Sauvegarder sur un nouveau support de sauvegarde et effacer tous les jeux de sauvegarde existants**.
  9. Dans la zone de texte **Nouveau nom du support de sauvegarde**, tapez **MarketDev Copy Backup**.
  10. Dans la zone de texte **Description du nouveau support de sauvegarde**, tapez **MarketDev Copy Backup for Integration Team**.
  11. Dans les options **Fiabilité**, assurez-vous que **Vérifier la sauvegarde en fin d'opération** est activé.
  12. Dans la liste déroulante **Définissez la compression de la sauvegarde**, cliquez sur **Compresser la sauvegarde**, puis cliquez sur **OK**.
  13. Dans la fenêtre Microsoft SQL Server Management Studio, cliquez sur **OK**.
  14. À l'aide de l'Explorateur Windows, notez la taille de la **sauvegarde de copie** par rapport à la **sauvegarde complète**.

## Exercice difficile 5 : Sauvegarde partielle (seulement si le temps le permet)

► Tâche 1 : Effectuer une sauvegarde des groupes de fichiers en lecture-écriture sur la base de données RateTracking

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **91 - Lab Exercise 5.sql** pour l'ouvrir.
2. Examinez le script.
3. Dans la **barre d'outils**, cliquez sur **Exécuter**.



## Module 7 : Restauration des bases de données SQL Server 2012

# Atelier pratique 7 : Restauration des bases de données SQL Server 2012

### Configuration de l'atelier pratique

Pour cet atelier pratique, vous utiliserez l'environnement d'ordinateur virtuel disponible.

Avant de commencer cet atelier pratique, vous devez procéder aux étapes suivantes :

1. Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
3. Dans la fenêtre Se connecter au serveur, tapez **Proseware** dans la zone de texte **Nom du serveur**.
4. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn**.
5. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**.
6. Dans la fenêtre Ouvrir un projet, sélectionnez le projet **D:\22462A\_Labs\22462A\_07\_PRJ\22462A\_07\_PRJ.ssmssqln**.
7. Dans l'Explorateur de solutions, double-cliquez sur le fichier de requête **00 - Setup.sql**.  
Lorsque la fenêtre de requête s'affiche, cliquez sur **Exécuter** dans la barre d'outils.



**Remarque** Le script d'installation de ce module permet de générer une erreur en cas de fichiers manquants ; ce comportement est normal.

## Exercice 1 : Déterminer une stratégie de restauration

### ► Tâche 1 : Examiner les sauvegardes contenues dans le fichier de sauvegarde

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **51 – Lab Exercise 1 Task 1.sql** pour l'ouvrir.
2. Examinez le script.
3. Dans la **barre d'outils**, cliquez sur **Exécuter**.
4. Les types de sauvegardes contenus dans le fichier sont (dans l'ordre) :
  1. Database
  2. Database
  3. Database Differential
  4. Transaction Log
  5. Database Differential
  6. Transaction Log
  7. Transaction Log

### ► Tâche 2 : Déterminer comment la restauration doit être effectuée

1. La restauration d'une base de données requiert :
  - la dernière sauvegarde de base de données complète (fichier 2) ;
  - la sauvegarde différentielle la plus récente (fichier 5) ;
  - toutes les sauvegardes de fichier journal depuis la dernière sauvegarde différentielle (fichiers 6, 7).
2. Avec NORECOVERY, spécification sur toutes les restaurations.

## Exercice 2 : Restaurer la base de données

### ► Tâche 1 : Restaurer la base de données

1. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\administrateur)**, cliquez avec le bouton droit sur **Bases de données**, puis cliquez sur **Restaurer la base de données**.
2. Dans les options **Source**, cliquez sur **Périphérique**, puis cliquez sur le bouton des points de suspension dans la partie droite de la zone de texte **Périphérique**.
3. Dans la fenêtre Sélectionner les unités de sauvegarde, cliquez sur **Ajouter**.
4. Accédez au fichier **D:\MSSQLServer\MarketYields.bak**, puis cliquez sur **OK**.
5. Dans la fenêtre Sélectionner les unités de sauvegarde, cliquez sur **OK**.
6. Dans la fenêtre Restaurer la base de données – MarketYields, dans le volet **Sélectionner une page**, cliquez sur **Fichiers**.
7. Dans les options **Restaurer les fichiers de base de données en tant que**, sélectionnez **Déplacer tous les fichiers dans le dossier**.
8. Dans le volet **Sélectionner une page**, cliquez sur **Options**.
9. Dans le menu déroulant **État de récupération**, sélectionnez **RESTORE WITH STANDBY**.
10. Dans la zone de texte **Fichier d'annulation**, tapez **L:\MKTG\Log\_Standby.bak**, puis cliquez sur **OK**.



**Remarque** Si le bouton **OK** n'est pas visible, vous devez faire défiler l'affichage vers le bas de la fenêtre ou augmenter la résolution de l'écran.

11. Dans la fenêtre Microsoft SQL Server Management Studio, cliquez sur **OK**.
12. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\administrateur)**, développez **Bases de données**, cliquez avec le bouton droit sur **Bases de données**, puis cliquez sur **Actualiser**.
13. La base de données **MarketYields** doit apparaître en mode **Mettre en veille/Lecture seule**.

### Exercice difficile 3 : En mode veille (seulement si le temps le permet)

► Tâche 1 : Exécuter des requêtes sur la base de données STANDBY pour vous assurer qu'elle est accessible

1. Dans l'Explorateur d'objets, cliquez sur le serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\administrateur)**, puis cliquez sur **Nouvelle requête**.

2. Tapez la requête suivante comme indiqué dans l'extrait de code ci-dessous :

```
SELECT * FROM MarketYields.dbo.LogData;
```

3. Dans la **barre d'outils**, cliquez sur **Exécuter**.

 **Remarque** 25 000 lignes doivent être retournées.

4. Fermez la fenêtre de requête. (Cliquez sur **Non** si vous êtes invité à enregistrer les modifications).

► Tâche 2 : Restaurer un autre fichier journal, en laissant la base de données en mode veille

1. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\administrateur)**, développez **Bases de données**, cliquez avec le bouton droit sur la base de données **MarketYields**, puis cliquez sur **Tâches**, **Restaurer** et **Journal des transactions**.

2. Dans les options **Restaurer le journal des transactions- MarketYields**, cliquez sur **À partir d'un fichier ou d'une bande**, cliquez sur le bouton **points de suspension** situé à droite de la zone de texte **À partir d'un fichier ou d'une bande**.

3. Dans la fenêtre Sélectionner les unités de sauvegarde, cliquez sur **Ajouter**.

4. Accédez au fichier **D:\MSSQLSERVER\MarketYields\_log.bak**, puis cliquez sur **OK**.

5. Dans la fenêtre Sélectionner les unités de sauvegarde, cliquez sur **OK**.

6. Dans le volet **Sélectionner une page**, cliquez sur **Options**.

7. Dans les options **État de récupération**, sélectionnez **Laisser la base de données en lecture seule**.

8. Dans la zone de texte **Fichier d'annulation**, tapez **L:\MKTG\Log\_Standby.bak**, puis cliquez sur **OK**.

9. Dans la fenêtre Microsoft SQL Server Management Studio, cliquez sur **Oui**.

10. Dans la fenêtre Microsoft SQL Server Management Studio, cliquez sur **OK**.

11. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\administrateur)**, développez **Bases de données**, cliquez avec le bouton droit sur **Bases de données**, puis cliquez sur **Actualiser**.

12. La base de données **MarketYields** doit apparaître en mode **Mettre en veille / Lecture seule**.

## Module 8 : Importation et exportation de données

# Atelier pratique 8 : Importation et exportation de données

### Configuration de l'atelier pratique

Pour cet atelier pratique, vous utiliserez l'environnement d'ordinateur virtuel disponible.

Avant de commencer cet atelier pratique, vous devez procéder aux étapes suivantes :

1. Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
3. Dans la fenêtre Se connecter au serveur, tapez **Proseware** dans la zone de texte **Nom du serveur**.
4. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn**.
5. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**.
6. Dans la fenêtre Ouvrir un projet, sélectionnez le projet **D:\22462A\_Labs\22462A\_08\_PRJ\22462A\_08\_PRJ.ssmssln**.
7. Dans l'Explorateur de solutions, double-cliquez sur le fichier de requête **00 - Setup.sql**.  
Lorsque la fenêtre de requête s'affiche, cliquez sur **Exécuter** dans la barre d'outils.

## Exercice 1 : Importer la feuille de calcul Excel

### ► Tâche 1 : Importer les données à l'aide de l'assistant Importation

1. Dans SQL Server Management Studio, dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100- ADVENTUREWORKS\administrateur)**, développez **Bases de données**, cliquez avec le bouton droit sur la base de données **MarketDev**, cliquez sur **Tâches**, puis cliquez sur **Importer des données**.
  2. Dans l'Assistant Importation et Exportation SQL Server, cliquez sur **Suivant**.
  3. Dans la fenêtre Choisir une source de données, sélectionnez **Microsoft Excel** dans la liste déroulante **Source de données**, puis cliquez sur **Parcourir**.
  4. Accédez au fichier **D:\22462A\_Labs\22462A\_08\_PRJ\22462A\_08\_PRJ\Currency.xls**, puis cliquez sur **Ouvrir**.
  5. Dans la fenêtre Choisir une source de données, vérifiez que la case à cocher **La première ligne possède des noms de colonnes** est activée, puis cliquez sur **Suivant**.
  6. Dans la fenêtre Choisir une destination, vérifiez que **Nom du serveur** est **Proseware** et que le nom de la base de données est **MarketDev**, puis cliquez sur **Suivant**.
  7. Dans la fenêtre Spécifier la copie ou l'interrogation de table, cliquez sur **Suivant**.
  8. Dans la fenêtre Sélectionner les tables et les vues sources, activez la case à cocher en regard de **Currency**, modifiez la **Destination** en **[DirectMarketing].[Currency]**, puis cliquez sur **Modifier les mappages**.
  9. Dans la fenêtre Mappages de colonnes.
    1. Remplacez le **Type** de la ligne **CurrencyID** par **int**.
    2. Désactivez la colonne **Nullable** pour chaque ligne.
    3. Remplacez la **Taille** de **CurrencyCode** par **3**.
    4. Remplacez la **Taille** de **CurrencyName** par **50**, puis cliquez sur **OK**.
  10. Dans la fenêtre Sélectionner les tables et les vues sources, cliquez sur **Suivant**.
  11. Dans la fenêtre Vérifier le mappage de type de données, cliquez sur **Suivant**.
  12. Dans la fenêtre Enregistrer et exécuter le package, cliquez sur **Suivant**.
  13. Dans la fenêtre Terminer l'Assistant, cliquez sur **Terminer**.
-  **Remarque** Un avertissement relatif à la troncation potentielle des données peut être généré. C'est normal.
14. Dans la fenêtre L'exécution s'est terminée avec succès, cliquez sur **Fermer**.
  15. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur la base de données **MarketDev**, puis cliquez sur **Nouvelle requête**.

16. Tapez le code comme indiqué dans l'extrait de code ci-dessous :

```
SELECT * FROM DirectMarketing.Currency;
```

17. Dans la **barre d'outils**, cliquez sur **Exécuter**.

 **Remarque** (105 ligne(s) affectée(s)).

## Exercice 2 : Importer le fichier CSV

### ► Tâche 1 : Importer le fichier CSV

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **61 - Lab Exercise 2.sql** pour l'ouvrir.
2. Examinez le script T-SQL.
3. Dans la **barre d'outils**, cliquez sur **Exécuter**.
4. Attendez la fin de l'exécution de la requête et enregistrer la durée de son traitement. (La durée de la requête est affichée dans l'angle inférieur droit de la **barre d'état**).



**Remarque** La durée d'exécution standard sur l'ordinateur virtuel est d'environ 3 minutes.

## Exercice 3 : Créer et tester un package d'extraction

### ► Tâche 1 : Créer et tester un package d'extraction

1. Dans SQL Server Management Studio, dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100- ADVENTUREWORKS\administrateur)**, développez **Bases de données**, cliquez avec le bouton droit sur la base de données **MarketDev**, cliquez sur **Tâches**, puis cliquez sur **Exporter les données**.
2. Dans l'Assistant Importation et Exportation SQL Server, cliquez sur **Suivant**.
3. Dans la fenêtre Choisir une source de données, cliquez sur **Suivant**.
4. Dans la fenêtre Choisir une destination, dans la liste déroulante **Destination**, choisissez **Destination de fichier plat**.

Dans la zone de texte **Nom de fichier**, tapez l'emplacement suivant :  
**D:\MKTG\ProspectsToContact.csv**.

5. Vérifiez que la case à cocher **Noms de colonne dans la première ligne de données** est activée, puis cliquez sur **Suivant**.
6. Dans la fenêtre Spécifier la copie ou l'interrogation de table, cliquez sur **Écrire une requête pour spécifier les données à transférer**, puis cliquez sur **Suivant**.
7. Dans la fenêtre Fournir une requête source, tapez le code comme indiqué dans l'extrait de code ci-dessous, puis cliquez sur **Analyser** :

```
SELECT ProspectID, FirstName, LastName, CellPhoneNumber,
WorkPhoneNumber, EmailAddress, LatestContact
FROM Marketing.Prospect
WHERE LatestContact < DATEADD(MONTH,-1,SYSDATETIME())
OR LatestContact IS NULL
ORDER BY ProspectID;
```

8. Dans la fenêtre Assistant Importation et Exportation SQL Server, vérifiez que l'analyse a réussi, puis cliquez sur **OK**.
9. Dans la fenêtre Fournir une requête source, cliquez sur **Suivant**.
10. Dans la fenêtre Configurer la destination du fichier plat, cliquez sur **Suivant**.
11. Dans la fenêtre Enregistrer et exécuter le package, vérifiez que les cases à cocher des options **Exécuter immédiatement** et **Enregistrer le package SSIS** sont activées, et que le bouton d'option **SQL Server** est sélectionnée, puis cliquez sur **Suivant**.
12. Dans la fenêtre Enregistrer le package SSIS, tapez **Extraction hebdomadaire des prospects à contacter** dans les zones de texte **Nom** et **Description**, puis cliquez sur **Suivant**.
13. Dans la fenêtre Terminer l'Assistant, cliquez sur **Terminer**.

 **Remarque** Un avertissement relatif à la troncation potentielle des données et une erreur en cas d'inexistence de la table, peut être généré. C'est normal.

14. Dans la fenêtre **L'exécution s'est terminée avec succès**, cliquez sur **Fermer**.

### Exercice difficile 4 : Comparer les performances de chargement (si le temps le permet)

- Tâche 1 : Ré-exécuter le chargement, les index étant désactivés
1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **81 - Lab Exercise 4.sql** pour l'ouvrir.
  2. Examinez le script T-SQL.
  3. Dans la **barre d'outils**, cliquez sur **Exécuter**.
  4. Attendez la fin de l'exécution de la requête et enregistrer la durée de son traitement. (La durée de la requête est affichée dans l'angle inférieur droit de la **barre d'état**).



**Remarque** La durée d'exécution standard sur l'ordinateur virtuel est d'environ 1,2 minutes.

## Module 9 : Authentification et autorisation des utilisateurs

# Atelier pratique 9 : Authentification et autorisation des utilisateurs

### Configuration de l'atelier pratique

Pour cet atelier pratique, vous utiliserez l'environnement d'ordinateur virtuel disponible.

Avant de commencer cet atelier pratique, vous devez procéder aux étapes suivantes :

1. Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
3. Dans la fenêtre Se connecter au serveur, tapez **Proseware** dans la zone de texte **Nom du serveur**.
4. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn**.
5. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**.
6. Dans la fenêtre Ouvrir un projet, sélectionnez le projet **D:\22462A\_Labs\22462A\_09\_PRJ\22462A\_09\_PRJ.ssmssln**.
7. Dans l'Explorateur de solutions, double-cliquez sur le fichier de requête **00 - Setup.sql**.  
Lorsque la fenêtre de requête s'affiche, cliquez sur **Exécuter** dans la barre d'outils.

## Exercice 1 : Créer des connexions

### ► Tâche 1 : Examiner la configuration requise

- Prenez connaissance des spécifications de sécurité fournies dans la documentation. Déterminez les connexions Windows, les connexions de groupes Windows et les connexions SQL requises.

### ► Tâche 2 : Créer les connexions requises

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **51 - Lab Exercise 1.sql** pour l'ouvrir.
2. Examinez le script T-SQL.
3. Dans la **barre d'outils**, cliquez sur **Exécuter**.

## Exercice 2 : Résoudre un problème de connexion d'application

### ► Tâche 1 : Résoudre un problème de connexion d'application

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **61 - Lab Exercise 2.sql** pour l'ouvrir.
2. Examinez le script T-SQL.
3. Dans la **barre d'outils**, cliquez sur **Exécuter**.

## Exercice 3 : Créer des utilisateurs de base de données

### ► Tâche 1 : Examiner la configuration requise

- Prenez connaissance des spécifications de sécurité fournies dans la documentation. Déterminez les connexions Windows, les connexions de groupes Windows et les connexions SQL requises.

### ► Tâche 2 : Créer les utilisateurs de base de données requis

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **71 - Lab Exercise 3.sql** pour l'ouvrir.
2. Examinez le script T-SQL.
3. Dans la **barre d'outils**, cliquez sur **Exécuter**.

### Exercice difficile 4 : Corriger l'accès à la base de données restaurée (si le temps le permet)

► Tâche 1 : Corriger l'accès à une base de données restaurée

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **81 - Lab Exercise 4.sql** pour l'ouvrir.
2. Examinez le script T-SQL.
3. Dans la **barre d'outils**, cliquez sur **Exécuter**.



## Module 10 : Attribution de rôles de serveur et de base de données

# Atelier pratique 10 : Attribution de rôles de serveur et de base de données

### Configuration de l'atelier pratique

Pour cet atelier pratique, vous utiliserez l'environnement d'ordinateur virtuel disponible.

Avant de commencer cet atelier pratique, vous devez procéder aux étapes suivantes :

1. Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A\_Labs\Revert.txt**.
2. Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
3. Dans la fenêtre Se connecter au serveur, tapez **Proseware** dans la zone de texte **Nom du serveur**.
4. Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn**.
5. Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**.
6. Dans la fenêtre Ouvrir un projet, sélectionnez le projet **D:\22462A\_Labs\22462A\_10\_PRJ\22462A\_10\_PRJ.ssmssqln**.
7. Dans l'Explorateur de solutions, double-cliquez sur le fichier de requête **00 - Setup.sql**.  
Lorsque la fenêtre de requête s'affiche, cliquez sur **Exécuter** dans la barre d'outils.

## Exercice 1 : Attribuer des rôles serveur

### ► Tâche 1 : Examiner la configuration requise

- Prenez connaissance des spécifications de sécurité fournies dans la documentation.

### ► Tâche 2 : Attribuer les rôles serveur requis

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **51 - Lab Exercise 1.sql** pour l'ouvrir.
2. Examinez le script T-SQL.
3. Dans la **barre d'outils**, cliquez sur **Exécuter**.

## Exercice 2 : Attribuer des rôles de base de données fixes

### ► Tâche 1 : Examiner la configuration requise

- Prenez connaissance des spécifications de sécurité fournies dans la documentation.

### ► Tâche 2 : Attribuer les rôles de base de données fixes requis

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **61 - Lab Exercise 2.sql** pour l'ouvrir.
2. Examinez le script T-SQL.
3. Dans la **barre d'outils**, cliquez sur **Exécuter**.

### Exercice 3 : Créer et attribuer des rôles de base de données définis par l'utilisateur

► Tâche 1 : Examiner la configuration requise

- Prenez connaissance des spécifications de sécurité fournies dans la documentation.

► Tâche 2 : Créer et attribuer les rôles de base de données définis par l'utilisateur requis

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **71 - Lab Exercise 3.sql** pour l'ouvrir.
2. Examinez le script T-SQL.
3. Dans la **barre d'outils**, cliquez sur **Exécuter**.

## Exercice difficile 4 : Vérifier les attributions de rôles (si le temps le permet)

### ► Tâche 1 : Vérifier les attributions de rôle pour Darren Parker

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **81 - Lab Exercise 4.sql** pour l'ouvrir.
2. Examinez le script T-SQL.
3. Dans la **barre d'outils**, cliquez sur **Exécuter**.

