

P R O D U I T O F F I C I E L D E F O R M A T I O N M I C R O S O F T

22462A

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

Volume 2

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 : X18-85342

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 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.
- 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 équipes¹. 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



¹ 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 11 : Autorisation de l'accès des utilisateurs aux ressources

Leçon 1 : Autorisation de l'accès des utilisateurs aux objets	11-3
Leçon 2 : Autorisation des utilisateurs à exécuter le code	11-12
Leçon 3 : Configuration des autorisations au niveau du schéma	11-21
Atelier pratique 11 : Autorisation de l'accès des utilisateurs aux ressources	11-28

Module 12 : Audit des environnements SQL Server

Leçon 1 : Options d'audit de l'accès aux données dans SQL	12-3
Leçon 2 : Implémentation de SQL Server Audit	12-13
Leçon 3 : Gestion de SQL Server Audit	12-27
Atelier pratique 12 : Audit des environnements SQL Server	12-32

Module 13 : Automatisation de la gestion de SQL Server 2012

Leçon 1 : Automatisation de la gestion de SQL Server	13-3
Leçon 2 : Utilisation de SQL Server Agent	13-11
Leçon 3 : Gestion des travaux de SQL Server Agent	13-20
Atelier pratique 13 : Automatisation de la gestion de SQL Server	13-27

Module 14 : Configuration de la sécurité pour SQL Server Agent

Leçon 1 : Comprendre la sécurité de SQL Server Agent	14-3
Leçon 2 : Configuration des informations d'identification	14-13
Leçon 3 : Configuration de comptes proxy	14-18
Atelier pratique 14 : Configuration de la sécurité pour SQL Server Agent	14-24

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

Leçon 1 : Configuration de la messagerie de base de données	15-3
Leçon 2 : Surveillance des erreurs SQL Server	15-12
Leçon 3 : Configuration des opérateurs, des alertes et des notifications	15-19
Atelier pratique 15 : Surveillance des travaux de l'agent SQL avec alertes et notifications	15-31

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

Leçon 1 : Vérification de l'intégrité de la base de données	16-3
Leçon 2 : Gestion des index	16-13
Leçon 3 : Automatisation de la maintenance régulière des bases de données	16-27
Atelier pratique 16 : Exécution de la maintenance régulière des bases de données	16-31

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

Leçon 1 : Capture de l'activité avec SQL Server Profiler et le générateur de profils des événements étendus	17-3
Leçon 2 : Amélioration des performances avec l'Assistant Paramétrage du moteur de base de données	17-17
Leçon 3 : Utilisation des options de suivi	17-26
Atelier pratique 17 : Trace de l'accès à SQL Server 2012	17-38

Module 18 : Surveillance de SQL Server 2012

Leçon 1 : Surveillance de l'activité	18-3
Leçon 2 : Capture et gestion des données de performances	18-15
Leçon 3 : Analyse des données de performances collectées	18-25
Atelier pratique 18 : Surveillance de SQL Server 2012	18-34

Module 19 : Gestion de plusieurs serveurs

Leçon 1 : Utilisation de plusieurs serveurs	19-3
Leçon 2 : Virtualisation de SQL Server	19-9
Leçon 3 : Déploiement et mise à niveau des applications de la couche Données	19-16
Atelier pratique 19 : Gestion de plusieurs serveurs	19-23

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

Leçon 1 : Méthodologie de résolution des problèmes SQL Server	20-3
Leçon 2 : Résolution des problèmes liés aux services	20-7
Leçon 3 : Résolution des problèmes de connexion et de connectivité	20-13
Leçon 4 : Résolution des problèmes liés aux accès concurrentiels	20-17
Atelier pratique 20 : Résolution des problèmes courants	20-25

Annexe A : Concepts principaux de la haute disponibilité et de la réPLICATION dans SQL Server

Leçon 1 : Concepts principaux de la haute disponibilité	A-3
Leçon 2 : Concepts principaux de la réPLICATION	A-12

Annexe : Corrigés des ateliers pratiques

Atelier pratique du module 11 : Autorisation de l'accès des utilisateurs aux ressources	L11-65
Atelier pratique du module 12 : Audit des environnements SQL Server	L12-69
Atelier pratique du module 13 : Automatisation de la gestion de SQL Server	L13-75
Atelier pratique du module 14 : Configuration de la sécurité pour SQL Server Agent	L14-81
Atelier pratique du module 15 : Surveillance des travaux de l'agent SQL avec alertes et notifications	L15-87
Atelier pratique du module 16 : Exécution de la maintenance régulière des bases de données	L16-95
Atelier pratique du module 17 : Trace de l'accès à SQL Server 2012	L17-101
Atelier pratique du module 18 : Surveillance de SQL Server 2012	L18-107
Atelier pratique du module 19 : Gestion de plusieurs serveurs	L19-113
Atelier pratique du module 20 : Résolution des problèmes courants	L20-119

Module 11

Autorisation de l'accès des utilisateurs aux ressources

Sommaire :

Leçon 1 : Autorisation de l'accès des utilisateurs aux objets	11-3
Leçon 2 : Autorisation des utilisateurs à exécuter le code	11-12
Leçon 3 : Configuration des autorisations au niveau du schéma	11-21
Atelier pratique 11 : Autorisation de l'accès des utilisateurs aux ressources	11-28

Vue d'ensemble du module

- Autorisation de l'accès des utilisateurs aux objets
- Autorisation des utilisateurs à exécuter le code
- Configuration des autorisations au niveau du schéma

Dans les deux modules précédents, vous avez appris comment la sécurité de Microsoft® SQL Server® est organisée et comment des jeux d'autorisations peuvent être attribués au niveau du serveur et de la base de données via des rôles serveur fixes, des rôles serveur définis par l'utilisateur, des rôles de base de données fixes, des rôles de base de données définis par l'utilisateur et des rôles d'application.

La dernière étape de l'autorisation des utilisateurs à accéder aux ressources SQL Server est l'autorisation des utilisateurs et des rôles à accéder aux objets de serveur et de base de données. Dans ce module, vous allez apprendre comment ces autorisations relatives aux objets sont gérées. Comme pour les autorisations d'accès sur les objets de base de données, SQL Server permet de déterminer les utilisateurs autorisés à exécuter le code, par exemple les procédures stockées et les fonctions.

Dans de nombreux cas, ces autorisations et celles sur les objets de base de données sont mieux configurées au niveau du schéma qu'au niveau de l'objet individuel. Les octrois d'autorisation basés sur le schéma peuvent simplifier votre architecture de sécurité. Vous examinerez l'octroi d'autorisations au niveau du schéma dans la dernière leçon de ce module.

Objectifs

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

- autoriser l'accès des utilisateurs aux objets ;
- autoriser les utilisateurs à exécuter le code ;
- configurer les autorisations au niveau du schéma.

Leçon 1

Autorisation de l'accès des utilisateurs aux objets

- Que sont les principaux ?
- Que sont les éléments sécurisables ?
- GRANT, REVOKE, DENY
- Sécurisation des tables et des vues
- Sécurité au niveau des colonnes
- WITH GRANT OPTION
- Démonstration 1A : Autorisation de l'accès des utilisateurs aux objets

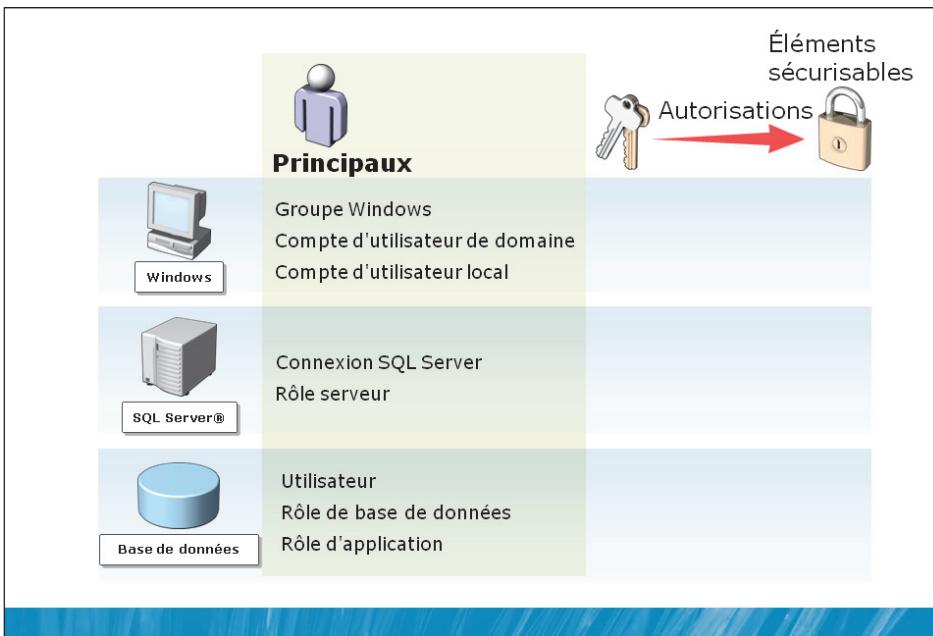
Avant de passer à la gestion des autorisations sur le code, vous devez examiner comment les autorisations sont gérées sur les objets de base de données. SQL Server dispose d'un modèle de sécurité à granularité fine qui vous permet d'accorder aux utilisateurs des autorisations minimales leur permettant d'effectuer leur travail. En particulier, des autorisations peuvent être accordées au niveau des colonnes, et pas seulement au niveau des tables et des vues. Vous apprendrez également comment vous pouvez déléguer l'octroi d'autorisations à d'autres utilisateurs.

Objectifs

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

- expliquer le rôle des principaux ;
- expliquer le rôle des éléments sécurisables ;
- utiliser les commandes GRANT, REVOKE et DENY ;
- sécuriser les tables et les vues ;
- implémenter la sécurité au niveau des colonnes ;
- déléguer la possibilité d'attribuer des autorisations à l'aide de la commande WITH GRANT OPTION.

Que sont les principaux ?



Points clés

Dans les deux modules précédents, vous avez étudié plusieurs principaux.

Les principaux sont des entités qui peuvent demander et obtenir l'accès aux ressources SQL Server. Comme les autres composants du modèle d'autorisation SQL Server, les principaux peuvent être ordonnés de façon hiérarchique. Cette diapositive résume les principaux qui ont été décrits et les place dans leurs emplacements appropriés dans la hiérarchie.

Au niveau de Windows®, les principaux comprennent des utilisateurs et des groupes. Ces utilisateurs et groupes peuvent être basés sur un domaine si le serveur fait partie d'un domaine Windows. Les comptes locaux peuvent être utilisés à partir des serveurs, que le serveur soit ou non membre d'un domaine.

Au niveau de SQL Server, des connexions peuvent être créées pour les utilisateurs autres que des utilisateurs Windows ou pour les utilisateurs qui font partie d'environnements Windows non approuvés, tels que les utilisateurs dans d'autres domaines Windows où il n'existe aucune relation d'approbation avec le domaine contenant le système SQL Server. De même, au niveau de SQL Server, les rôles serveur fixes et définis par l'utilisateur sont une forme de principal qui contient d'autres principaux.

Au niveau de la base de données, vous avez étudié les utilisateurs de base de données, les rôles de base de données fixes et définis par l'utilisateur et les rôles d'application.

Chaque principal a deux identificateurs numériques qui lui sont associés : un identificateur principal et un identificateur de sécurité (SID).

Que sont les éléments sécurisables ?

- Les éléments sécurisables sont des ressources auxquelles SQL Server contrôle l'accès
- Les éléments sécurisables sont contenus dans des étendues
 - Serveur
 - Base de données
 - Schéma

Points clés

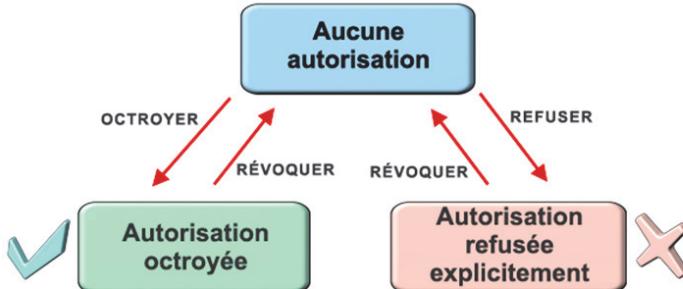
Les éléments sécurisables sont les ressources auxquelles le système d'autorisation du moteur de base de données SQL Server régule l'accès. Certains éléments sécurisables peuvent être contenus dans d'autres, de façon à créer des hiérarchies imbriquées appelées « étendues », lesquelles peuvent elles-mêmes être sécurisées. Les étendues sécurisables sont : le serveur, la base de données et le schéma.

Il est important de connaître les différentes étendues sécurisables dans SQL Server pour planifier votre modèle de sécurité.

Question : pouvez-vous suggérer une raison pour laquelle une connexion est un élément sécurisable ? Quels types d'autorisations sont nécessaires sur une connexion ?

GRANT, REVOKE, DENY

- GRANT est utilisé pour attribuer une autorisation
- DENY est utilisé pour refuser explicitement une autorisation
 - Utilisé lorsque les autorisations héritées via l'appartenance au groupe ou au rôle
 - Doit être utilisé uniquement dans des circonstances exceptionnelles
- REVOKE permet de supprimer une instruction GRANT ou DENY



Points clés

Les autorisations sont gérées via les commandes T-SQL GRANT, DENY et REVOKE. La plupart des autorisations (mais pas toutes les autorisations) peuvent également être gérées via l'interface utilisateur graphique de SSMS.

GRANT et REVOKE

Un utilisateur qui n'a pas obtenu une autorisation ne peut pas exécuter l'action associée à l'autorisation. Par exemple, les utilisateurs ne sont pas autorisés à utiliser une commande SELECT sur les données de tables s'ils n'ont pas obtenu l'autorisation à un certain niveau. Certains autres moteurs de base de données considèrent ce scénario comme une forme implicite de refus.

La commande GRANT est utilisée pour attribuer des autorisations aux utilisateurs de base de données. La commande REVOKE permet de supprimer ces mêmes autorisations.

DENY

SQL ANSI ne fournit pas de commande DENY. Si un utilisateur n'est pas autorisé à exécuter une action, il ne peut pas l'exécuter.

La particularité des systèmes Windows est l'appartenance aux groupes. SQL ANSI n'a pas de concept de groupe. Dans SQL Server, un utilisateur Windows peut recevoir des autorisations directement, mais aussi via l'appartenance aux groupes Windows ou l'appartenance aux rôles.

La commande DENY vous permet de refuser une autorisation à un utilisateur qui a obtenu une autorisation via l'appartenance à un groupe ou à un rôle qui a des autorisations. Cela est très similaire au fonctionnement des refus dans Windows. Pour un exemple Windows, imaginez que vous pouvez décider que tous les membres du groupe Salespeople peuvent accéder à une imprimante couleur, sauf Holly (qui est membre de Salespeople). Vous accordez l'accès au groupe Salespeople et refusez l'accès à Holly.

SQL Server et DENY

SQL Server fonctionne de la même manière. Vous pouvez accorder une autorisation SELECT sur une table à chaque commercial, mais refuser à Holly l'accès à cette table.

Comme avec Windows, vous devez utiliser la commande DENY avec modération. La nécessité d'utiliser une commande DENY sur plusieurs autorisations a tendance à être considérée comme une « odeur de code » qui indique un problème potentiel avec votre conception de sécurité.

Le fait que la commande REVOKE soit utilisée non seulement pour supprimer une autorisation GRANT, mais aussi une autorisation DENY a tendance à créer la confusion chez les nouveaux utilisateurs. Cela signifie qu'un utilisateur peut bénéficier d'un accès qu'il n'avait pas avant l'émission de la commande REVOKE. Par exemple, si vous révoquez l'autorisation DENY de Holly, elle peut ensuite accéder à la table.

Question : si un utilisateur ne peut pas exécuter une action sans autorisation, pourquoi est-il nécessaire d'utiliser la commande DENY sur une autorisation ?

Sécurisation des tables et des vues

- Plusieurs autorisations relatives aux objets s'appliquent aux tables et aux vues

- SELECT
- INSERT, UPDATE, DELETE
- RÉFÉRENCES

```
USE MarketDev;
GO

GRANT SELECT ON OBJECT::Marketing.Salesperson
    TO HRApp;
GO

GRANT SELECT ON Marketing.Salesperson
    TO HRApp;
GO
```

Points clés

Les autorisations d'accès aux données qui s'appliquent aux tables et vues sont SELECT, INSERT, UPDATE, DELETE et REFERENCES.

Dans l'exemple illustré sur la diapositive, l'autorisation SELECT sur l'objet Marketing.Salesperson (qui est probablement une table ou une vue) est accordée à l'utilisateur HRApp dans la base de données MarketDev.

Composants facultatifs de GRANT

Notez que deux formes de la commande sont illustrées. Même si la terminologie complète utilise OBJECT:: comme préfixe , ce préfixe est facultatif. Dans le deuxième exemple, la même commande GRANT est illustrée sans le préfixe OBJECT:::

Il n'est pas nécessaire de spécifier le schéma de la table ou de la vue, mais cela est fortement recommandé. Si le nom du schéma n'est pas spécifié, le schéma par défaut de l'utilisateur qui accorde l'autorisation est utilisé. Si l'objet est introuvable dans le schéma par défaut de l'utilisateur, le schéma dbo est utilisé à la place.

REFERENCES

La signification des autorisations SELECT, INSERT, UPDATE et DELETE vous semble probablement évidente, mais la signification et le rôle de l'autorisation REFERENCES peuvent ne pas l'être. L'autorisation REFERENCES est nécessaire pour qu'une relation de clé étrangère puisse spécifier l'objet en tant que cible, et est uniquement requise si aucune autre autorisation n'existe sur l'objet.

Question : pourquoi une autorisation doit-elle faire référence à une table dans une référence de clé étrangère ?

Sécurité au niveau des colonnes

- Les autorisations peuvent être attribuées au niveau des colonnes
- Les autorisations sur plusieurs colonnes peuvent être attribuées dans une instruction unique
- Une option GRANT au niveau de la colonne remplace une option DENY au niveau de la table

```
GRANT SELECT ON Marketing.Salesperson
  ( salespersonID, EmailAlias)
  TO James;
GO
DENY SELECT ON Marketing.Salesperson
  TO Holly;
GO
GRANT SELECT ON Marketing.Salesperson
  ( salespersonID, FirstName, LastName)
  TO Holly;
GO
```

Points clés

Bien qu'elles ne soient pas implémentées aussi souvent que les autorisations au niveau des tables ou des vues, des autorisations au niveau des colonnes peuvent également être attribuées. Cela fournit un contrôle de sécurité encore plus précis que celui fourni en contrôlant l'accès aux tables et aux vues.

Il n'est pas nécessaire d'exécuter des instructions GRANT distinctes pour chaque colonne sur laquelle vous souhaitez attribuer des autorisations. Lorsqu'un jeu de colonnes doit être contrôlé de la même façon, une liste de colonnes peut être fournie dans une instruction GRANT unique. Dans le premier exemple illustré sur la diapositive, l'autorisation SELECT sur la table Marketing.Salesperson est accordée à James, mais l'accès à la table entière n'est pas autorisé. Seule l'autorisation sur les colonnes SalespersonID et EmailAlias est accordée.

DENY au niveau des tables et GRANT au niveau des colonnes

Il existe une anomalie dans le modèle de sécurité SQL Server dont vous devez avoir connaissance.

Une autorisation DENY au niveau des tables n'a pas la priorité par rapport à une autorisation GRANT au niveau des colonnes. Il s'agit d'une incohérence, mais elle doit être conservée pour des raisons de rétrocompatibilité. Il existe un plan pour supprimer cette incohérence à l'avenir. Évitez de l'utiliser dans une nouvelle tâche de développement.

Cette anomalie est illustrée dans le deuxième exemple de la diapositive. Holly n'est pas autorisée à utiliser une commande SELECT sur la table Salesperson, mais est ensuite autorisée à utiliser une commande SELECT sur des colonnes spécifiques de la table. Le résultat (probablement un résultat inattendu) est que Holly est toujours autorisée à utiliser une commande SELECT sur ces colonnes de la table.

WITH GRANT OPTION

- Les autorisations accordées avec la clause WITH GRANT OPTION peuvent être accordées à d'autres principaux par le bénéficiaire
- CASCADE est également utilisé pour révoquer des autorisations accordées par le bénéficiaire
 - Peut s'appliquer également à DENY

```
GRANT UPDATE ON Marketing.Salesperson
```

```
TO James  
WITH GRANT OPTION;  
GO
```

```
REVOKE UPDATE ON Marketing.Salesperson
```

```
FROM James  
CASCADE;  
GO
```

Points clés

Lorsqu'une autorisation est accordée à un principal, il est également possible de lui accorder le droit de réaffecter l'autorisation à d'autres principaux. Cet autre droit est attribué avec la clause WITH GRANT OPTION. Ce mécanisme vous permet de déléguer la responsabilité de la gestion des autorisations, mais doit être utilisé avec précaution. En général, la clause WITH GRANT OPTION doit être évitée.

Dans le premier exemple de la diapositive, James est autorisé à mettre à jour la table Marketing.Salesperson. En outre, il a le droit d'accorder cette même autorisation à d'autres utilisateurs de la base de données.

CASCADE

La difficulté de la clause WITH GRANT OPTION est lorsque vous devez utiliser une commande REVOKE ou DENY sur l'autorisation accordée à James avec WITH GRANT OPTION. Vous ne savez pas à quels autres utilisateurs James a déjà accordé l'autorisation.

Lors de la révocation ou du refus d'une autorisation qui a accordée, l'option CASCADE révoque ou refuse les autorisations que James a également accordées.

Démonstration 1A : Autorisation de l'accès des utilisateurs aux objets

Dans cette démonstration, vous allez apprendre à

- Afficher la liste complète des principaux de serveur
- Afficher la liste complète des principaux de base de données
- Accorder des autorisations sur une table
- Accorder des autorisations au niveau des colonnes

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_11_PRJ\22462A_11_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

Autorisation des utilisateurs à exécuter le code

- Sécurisation des procédures stockées
- Sécurisation des fonctions définies par l'utilisateur
- Sécurisation du code managé
- Gestion des chaînes de propriétés
- Démonstration 2A : Autorisation des utilisateurs à exécuter le code

En plus de vous fournir un contrôle sur les utilisateurs qui accèdent aux données de votre base de données ou aux objets de votre serveur, SQL Server vous permet de contrôler les utilisateurs qui peuvent exécuter votre code. Un contrôle de sécurité approprié de l'exécution du code est un aspect important de votre architecture de sécurité.

Dans cette leçon, vous apprendrez à gérer la sécurité des procédures stockées et des fonctions. Vous apprendrez également à gérer la sécurité du code qui réside dans les assemblies de code managé .NET utilisés avec l'intégration CLR SQL. Enfin, vous découvrirez comment les chaînes de propriétés affectent la relation de sécurité entre le code et les objets de base de données.

Objectifs

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

- sécuriser les procédures stockées ;
- sécuriser les fonctions définies par l'utilisateur ;
- sécuriser le code managé ;
- gérer les chaînes de propriétés.

Sécurisation des procédures stockées

- Les procédures stockées nécessitent
 - Une autorisation EXECUTE avant de pouvoir être appelées
 - Une autorisation ALTER pour modification
 - VIEW DEFINITION pour l'accès à la documentation

```

USE MarketDev ;
GO

GRANT EXECUTE ON
Reports.GetProductColors
TO Mod11User ;
GO
  
```

Points clés

Par défaut, les utilisateurs ne peuvent pas exécuter les procédures stockées que vous (ou un utilisateur) créez. Ils ont besoin de l'autorisation EXECUTE pour pouvoir exécuter une procédure stockée. Ils peuvent également avoir besoin d'autorisations pour accéder aux objets utilisés par la procédure stockée. Cette question sera abordée plus loin dans la leçon.

Dans l'exemple illustré sur la diapositive, l'utilisateur de base de données Mod11User est autorisé à exécuter la procédure stockée Reports.GetProductColors.

Gestion des procédures stockées

Deux autres autorisations sont associées à la gestion des procédures stockées :

- L'autorisation ALTER permet à un utilisateur de modifier la définition d'une procédure stockée.
- L'autorisation VIEW DEFINITION a été ajoutée dans SQL Server 2005. Dans les versions antérieures, un utilisateur avait besoin d'une autorisation ALTER sur une procédure stockée pour pouvoir afficher sa définition. Cela représentait un octroi d'autorisation inutile pour les utilisateurs impliqués dans la documentation de systèmes. L'autorisation VIEW DEFINITION a été introduite pour que cette autorisation de haut niveau ne soit plus requise lorsque seul l'accès à la documentation est nécessaire.

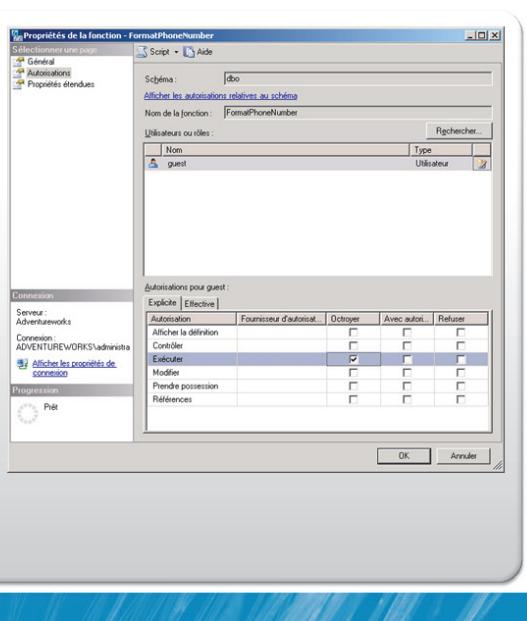


Remarque Vous ne pouvez pas utiliser SSMS pour accorder des autorisations sur les procédures stockées système. SSMS peut être utilisé pour accorder des autorisations sur d'autres procédures stockées.

Sécurisation des fonctions définies par l'utilisateur

- Les utilisateurs ont besoin de l'autorisation EXECUTE avant d'utiliser les fonctions UDF scalaires
- Les utilisateurs ont besoin de l'autorisation SELECT pour les fonctions table
- L'autorisation REFERENCES est utilisée pour les contraintes CHECK, les valeurs DEFAULT ou les colonnes calculées

```
GRANT EXECUTE ON
dbo.FormatPhoneNumber
TO public;
GO
```



Points clés

Les fonctions définies par l'utilisateur requièrent également des autorisations avant qu'elles puissent être utilisées.

- Les fonctions scalaires définies par l'utilisateur retournent une valeur unique. Les utilisateurs qui accèdent à ces fonctions requièrent une autorisation EXECUTE sur la fonction définie par l'utilisateur.
- Les fonctions table définies par l'utilisateur retournent une table de résultats au lieu d'une valeur unique. L'accès à une fonction table requiert une autorisation SELECT au lieu d'une autorisation EXECUTE, comme pour les autorisations sur une table.

Bien qu'il soit rare de mettre à jour directement une fonction table, il est possible d'attribuer des autorisations INSERT, UPDATE, DELETE sur un type de fonction table appelé fonction table inline, car ce type particulier de fonction table peut être mis à jour dans certains cas.

REFERENCES

L'autorisation REFERENCES est requise pour :

- les fonctions utilisées dans les contraintes CHECK ;
- calculer les valeurs des contraintes DEFAULT ;
- calculer les valeurs des colonnes calculées.

Rôle public et fonctions

Les fonctions fournissent souvent des fonctionnalités très élémentaires dans les systèmes, avec un risque faible. Pour cette raison, l'octroi d'autorisations sur les fonctions de base contenues dans une base de données au rôle public de la base de données est une pratique assez courante. Cela permet à un utilisateur de la base de données d'utiliser ces fonctions sans qu'il soit nécessaire d'accorder d'autres autorisations.



Remarque Bien que cette pratique soit courante pour les fonctions de base, elle est rarement utilisée (voire inutilisable) pour les procédures stockées.

Sécurisation du code managé

- Le code basé sur CLR SQL présente des spécifications d'autorisation supplémentaires en plus de celles requises pour le code T-SQL
- Les assemblies CLR sont enregistrés avec l'un des trois jeux d'autorisations suivants
 - SAFE (par défaut)
 - EXTERNAL_ACCESS
 - UNSAFE
- Les jeux d'autorisations EXTERNAL_ACCESS et UNSAFE nécessitent une configuration supplémentaire au niveau de la base de données
- Notez que UNSAFE est appelé Unrestricted dans l'interface utilisateur graphique dans SQL Server Management Studio

Points clés

Le code managé est le code .NET fourni dans les assemblies. Bien que les assemblies soient contenus dans les fichiers DLL ou EXE, seuls les assemblies contenus dans les fichiers DLL peuvent être chargés dans SQL Server via l'intégration du CLR dans SQL Server.

Une fois qu'un assembly est catalogué, les procédures, fonctions, et autres objets de code managé qu'il contient sont également catalogués. Ces objets apparaissent ensuite en tant qu'objets standard dans SQL Server et les autorisations d'objet T-SQL standard s'appliquent également. Par exemple, un utilisateur requiert l'autorisation EXECUTE sur une procédure stockée, qu'elle soit écrite en code managé ou T-SQL.

Jeux d'autorisations

Quel que soit le code .NET inclus dans un assembly, les actions qu'il est autorisé à exécuter sont déterminées par le jeu d'autorisations dans lequel il est catalogué.

Le jeu d'autorisations SAFE limite strictement les actions que l'assembly peut exécuter. Il s'agit du jeu d'autorisations par défaut.

Le jeu d'autorisations EXTERNAL_ACCESS est nécessaire pour accéder aux ressources locales et réseau, aux variables d'environnement et au registre. EXTERNAL_ACCESS est même nécessaire pour accéder à la même instance SQL Server si une connexion est établie via une interface réseau. Ce jeu d'autorisations n'est pas nécessaire pour les connexions internes directes du code managé à SQL Server, car un chemin d'accès direct distinct (appelé connexion contextuelle) est fourni pour l'accès à l'instance locale, sans utiliser une interface réseau.

Le jeu d'autorisations UNSAFE assouplit de nombreux contrôles standard sur le code et doit être évité.

Recommandations

En général, les administrateurs de base de données SQL Server trouvent acceptable l'utilisation d'assemblies SAFE ; ils demandent des explications avant d'utiliser les assemblies EXTERNAL_ACCESS et ils ont besoin de justifications particulièrement solides (ce qui devrait être rare) pour les assemblies UNSAFE. UNSAFE est le jeu d'autorisations contenant le plus de fonctions.

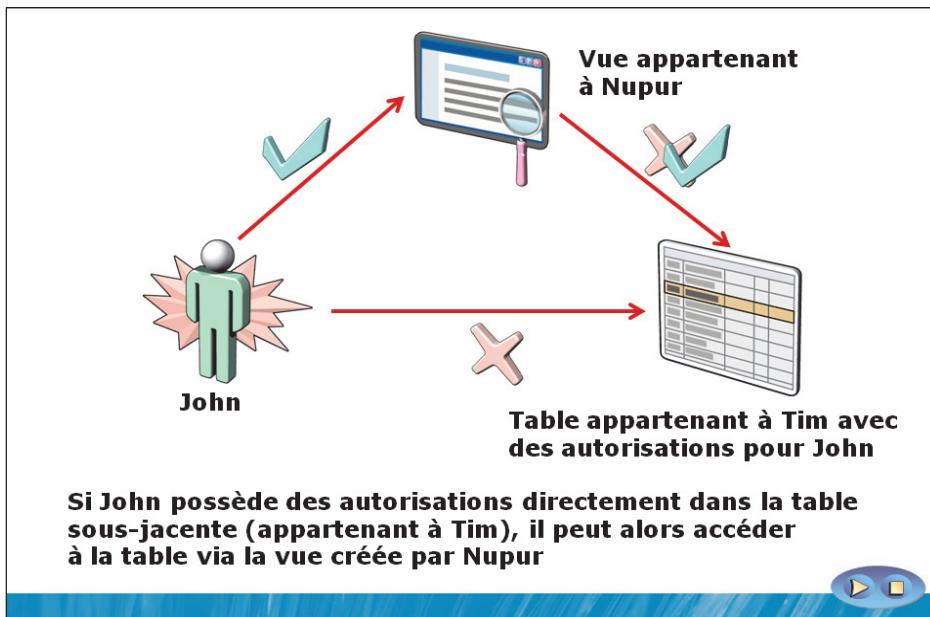
Configuration

Les jeux d'autorisations EXTERNAL_ACCESS et UNSAFE requièrent également une configuration supplémentaire. Vous ne pouvez pas spécifier qu'un jeu d'autorisations EXTERNAL_ACCESS est nécessaire lors de l'exécution de l'instruction CREATE ASSEMBLY. La base de données doit être marquée comme TRUSTWORTHY (non recommandé mais simple) ou une clé asymétrique doit être créée à partir du fichier d'assembly de la base de données master, ce qui crée une connexion qui est mappée à la clé et obtient l'autorisation EXTERNAL ACCESS ASSEMBLY sur l'assembly.

 **Remarque** Cette dernière option est clairement une rubrique avancée qui n'est pas traitée dans ce cours et qui est uniquement mentionnée par souci d'exhaustivité.

Question : quel jeu d'autorisations doit être rarement accordé ?

Gestion des chaînes de propriétés



Points clés

Tous les objets de base de données appartiennent à des propriétaires. Les objets dans l'étendue du schéma appartiennent au propriétaire du schéma et la propriété principal_id (propriétaire) pour les nouveaux objets a la valeur NULL par défaut. Un objet dont la propriété principal_id a la valeur NULL hérite sa propriété du schéma dans lequel il est contenu. Il est recommandé que tous les objets appartiennent au propriétaire de l'objet de schéma.

Lorsqu'un objet tel qu'une procédure stockée fait référence à un autre objet, une chaîne de propriétés est créée. Une chaîne de propriétés ininterrompue existe lorsque chaque objet de la chaîne a le même propriétaire. Lorsqu'une chaîne de propriété ininterrompue existe, l'accès est autorisé sur les objets sous-jacents lorsqu'il est autorisé sur les objets de niveau supérieur.

Depuis que SQL Server 2005 a introduit le concept de schémas, une idée erronée prétend que les objets SQL Server n'ont plus de propriétaires. Cela n'est pas vrai. Les objets ont toujours des propriétaires.

L'utilisation du même propriétaire pour tous les objets du schéma (qui lui-même a également un propriétaire) d'une base de données facilite la gestion des autorisations, mais il est important de comprendre que des problèmes de chaînes de propriétés peuvent toujours se produire même s'ils sont beaucoup moins courants.

Exemple de diapositive

Le chaînage des propriétés s'applique aux procédures stockées, vues et fonctions. La diapositive illustre l'application du chaînage des propriétés aux vues ou procédures stockées.

1. John ne dispose d'aucune autorisation sur la table appartenant à Nupur.
2. Nupur crée une vue qui accède à la table et autorise John à accéder à la vue. L'accès est autorisé car Nupur est le propriétaire de l'objet de niveau supérieur et de l'objet sous-jacent (c'est-à-dire la table).
3. Nupur crée ensuite une vue qui accède à une table appartenant à Tim. Même si Nupur a l'autorisation d'accéder à la table et autorise John à utiliser la vue, l'accès sera refusé à John. Cela est causé par la chaîne de propriétés rompue de l'objet de niveau supérieur à l'objet sous-jacent.
4. Mais si John obtient des autorisations directement sur la table sous-jacente appartenant à Tim, il peut ensuite accéder à la vue que Nupur a créée pour accéder à cette table.

Le problème avec l'étape 4 est que l'une des principales raisons de la création de vues ou de procédures stockées est d'éviter que les utilisateurs obtiennent des autorisations sur les objets sous-jacents.

Démonstration 2A : Autorisation des utilisateurs à exécuter le code

Dans cette démonstration, vous allez apprendre à

- Affecter une autorisation pour exécuter des procédures stockées
- Affecter des autorisations pour exécuter des fonctions

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_11_PRJ\22462A_11_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

Configuration des autorisations au niveau du schéma

- Vue d'ensemble de la séparation du schéma et de l'utilisateur
- Résolution de noms d'objets
- Octroi d'autorisations au niveau du schéma
- Démonstration 3A : Configuration des autorisations au niveau du schéma

SQL Server 2005 a introduit une modification sur la façon dont les schémas sont utilisés. Depuis cette version, les schémas sont utilisés comme conteneurs pour les objets tels que les tables, vues et procédures stockées. Les schémas peuvent s'avérer particulièrement utiles pour fournir un niveau d'organisation et de structure lorsqu'un grand nombre d'objets est contenu dans une base de données. Des autorisations de sécurité peuvent également être attribuées au niveau du schéma et non individuellement sur les objets contenus dans les schémas. Cela peut considérablement simplifier la conception de la configuration requise pour la sécurité du système.

Objectifs

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

- décrire la séparation du schéma et de l'utilisateur ;
- décrire le rôle de la résolution de noms d'objets ;
- accorder des autorisations au niveau du schéma.

Vue d'ensemble de la séparation du schéma et de l'utilisateur

- Schémas
 - Concept modifié dans SQL Server 2005
 - Plus équivalent aux utilisateurs de base de données
 - Conteneurs pour les objets de base de données
 - Crée avec CREATE SCHEMA
 - Répertorié en interrogeant la vue sys.schemas
- Les utilisateurs peuvent avoir des schémas par défaut
- Schémas intégrés
 - dbo
 - guest
 - sys
 - INFORMATION_SCHEMA

Points clés

Les schémas sont utilisés pour contenir des objets et fournir une limite de sécurité pour l'octroi d'autorisations.

Schémas

Dans SQL Server, les schémas sont essentiellement utilisés comme conteneurs d'objets, de la même manière qu'un dossier est utilisé pour contenir des fichiers au niveau du système d'exploitation. Depuis leur changement de comportement dans SQL Server 2005, les schémas peuvent être utilisés pour contenir des objets tels que les tables, procédures stockées, fonctions, types, vues, etc. Les schémas sont créés avec l'instruction CREATE SCHEMA et font partie de la convention d'affectation de noms en plusieurs parties pour les objets. Dans SQL Server, un objet est formellement référencé par :

`Server.Database.Schema.Object`

Limite de sécurité

Les schémas peuvent être utilisés pour simplifier l'octroi d'autorisations. L'octroi de l'autorisation EXECUTE sur un schéma à un utilisateur est un exemple d'application des autorisations au niveau du schéma. L'utilisateur peut ensuite exécuter toutes les procédures stockées dans le schéma. Cela simplifie l'octroi d'autorisations car il n'est pas nécessaire de configurer des autorisations individuelles sur chaque procédure stockée.

Mise à niveau d'anciennes applications

Si vous mettez à niveau des applications à partir de SQL Server 2000 et des versions antérieures, il est important de comprendre que la convention d'affectation de noms a été modifiée lorsque les schémas ont été introduits. Auparavant, les noms se présentaient sous la forme suivante :

```
Server.Database.Owner.Object
```

Les objets ont toujours des propriétaires, mais le nom du propriétaire ne fait pas partie de la convention d'affectation de noms en plusieurs parties à partir de SQL Server 2005. Lors de la mise à niveau des bases de données à partir de versions antérieures, SQL Server crée automatiquement un schéma de même nom que les propriétaires d'objets existants, de sorte que les applications qui utilisent des noms en plusieurs parties continuent à s'exécuter.

Un schéma par défaut peut être attribué à chaque utilisateur, qui est utilisé lorsqu'un utilisateur fait référence à un objet sans spécifier un nom de schéma.

Schémas intégrés

Les utilisateurs dbo et Invité ont été décrits dans le dernier module. L'utilisateur dbo a un schéma associé. Les schémas sys et INFORMATION_SCHEMA sont réservés aux objets système. Vous ne pouvez pas créer des objets dans les schémas sys et INFORMATION_SCHEMA, ni supprimer ces schémas.

Résolution de noms d'objets

- Si le nom de schéma est ignoré, les règles s'appliquent au mode de résolution des noms
 - Chaque utilisateur (y compris ceux associés aux connexions du Groupe Windows) peut être affecté à un schéma par défaut
 - Les utilisateurs sans schéma par défaut sont associés à dbo comme schéma par défaut
 - La première recherche s'effectue dans le schéma par défaut de l'utilisateur
 - Si introuvable, le schéma dbo fait également l'objet d'une recherche
- Lorsqu'ils font référence à un objet dans une instruction, les utilisateurs doivent spécifier à la fois le schéma et le nom de l'objet
 - SELECT ProductID FROM Production.Product

Points clés

Il est important d'utiliser au moins des noms en deux parties pour faire référence aux objets du code SQL Server, tels que les procédures stockées, les fonctions et les vues.

Résolution de noms d'objets

Lorsque des noms d'objets sont mentionnés dans le code, SQL Server doit déterminer les objets sous-jacents de référence. Par exemple, prenons l'instruction suivante :

```
SELECT ProductID, Name, Size FROM Product;
```

Plusieurs tables Product peuvent exister dans des schémas distincts avec la base de données. Lorsque des noms en une seule partie sont utilisés, SQL Server doit ensuite déterminer la table Product de référence.

Des schémas par défaut sont attribués à la plupart des utilisateurs, mais un schéma par défaut est attribué à certains utilisateurs. Les schémas par défaut ne sont pas attribués aux utilisateurs en fonction des certificats, mais ils s'appliquent aux utilisateurs créés à partir des connexions Windows et SQL Server standard. SQL Server 2012 a introduit la possibilité d'attribuer un schéma par défaut à un groupe Windows. Les utilisateurs à qui des schémas par défaut n'ont pas été attribués utilisent le schéma dbo comme schéma par défaut.



Remarque Les utilisateurs créés à partir de certificats est une rubrique avancée qui n'est pas traitée dans ce cours, mais est mentionnée par souci d'exhaustivité.

Recherche d'objets

Lors de la recherche d'un objet, SQL Server vérifie d'abord le schéma par défaut de l'utilisateur. Si l'objet est introuvable, SQL Server vérifie le schéma dbo pour essayer de localiser l'objet.

Il est important d'inclure les noms de schéma pour faire référence aux objets au lieu de dépendre de la résolution de noms de schéma, comme dans cette version modifiée de l'instruction précédente :

```
SELECT ProductID, Name, Size FROM Production.Product;
```

Mis à part quelques rares cas, l'utilisation de noms en plusieurs parties donne un code plus fiable qui ne dépend pas des paramètres du schéma par défaut.

Octroi d'autorisations au niveau du schéma

- Au lieu d'attribuer des autorisations individuelles sur les tables, vues et procédures stockées, les autorisations peuvent être accordées au niveau du schéma
 - Applicable à tous les objets appropriés dans le schéma
 - Plus facile à gérer

```
USE MarketDev;
GO
GRANT EXECUTE
    ON SCHEMA::Marketing
    TO Mod11User;
GO
GRANT SELECT
    ON SCHEMA::DirectMarketing
    TO Mod11User;
GO
```

Points clés

Au lieu d'attribuer des autorisations individuelles sur les tables, vues et procédures stockées, les autorisations peuvent être accordées au niveau du schéma.

Exemple de diapositive

Dans le premier exemple de la diapositive, l'autorisation EXECUTE sur le schéma Marketing est accordée à Mod11User. Cela signifie que Mod11User peut ensuite exécuter toutes les procédures stockées et fonctions scalaires du schéma.

Dans le deuxième exemple de la diapositive, l'autorisation SELECT sur le schéma DirectMarketing est accordée à Mod11User. Cela signifie que Mod11User peut ensuite sélectionner toutes les tables, vues et fonctions table du schéma.

Question : pourquoi l'octroi d'autorisations au niveau du schéma est-il plus facile à gérer ?

Démonstration 3A : Configuration des autorisations au niveau du schéma

Dans cette démonstration, vous allez apprendre à

- Révoquer des autorisations sur une procédure stockée
- Affecter l'autorisation EXECUTE au niveau du schéma
- Affecter une autorisation SELECT au niveau du schéma
- Explorer les autorisations couvrantes ou implicites

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_11_PRJ\22462A_11_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** 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.

Question : l'utilisateur utilise une commande EXECUTE au niveau du schéma et une commande DENY au niveau de la procédure. Leur exécution doit-elle être autorisée ?

Atelier pratique 11 : Autorisation de l'accès des utilisateurs aux ressources

- Exercice 1 : Attribuer des autorisations au niveau du schéma
- Exercice 2 : Attribuer des autorisations au niveau de l'objet
- Exercice difficile 3 : Tester les autorisations
(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_11_PRJ\22462A_11_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 et les avez assignés aux rôles appropriés. Vous devez maintenant accorder des autorisations aux utilisateurs et rôles de base de données afin que les utilisateurs puissent accéder aux ressources dont ils ont besoin dans la base de données MarketDev, selon la configuration de sécurité fournie.

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

- 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
- Le rôle serveur suivant a été attribué :
 - Jeff Hay → sysadmin
- Les membres de rôles de base de données fixes suivants ont été attribués :
 - AdventureWorks\ITSupport → db_owner
 - DBMonitorApp → db_datareader
- Les membres de rôles de base de données définis par l'utilisateur suivants ont été attribués :
 - AdventureWorks\SalesPeople → SalesTeam
 - AdventureWorks\CreditManagement → SalesTeam
 - AdventureWorks\CorporateManagers → SalesTeam
 - AdventureWorks\Darren.Parker → SalesManagers

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 autorisations au niveau du schéma

Scénario

Une liste d'exigences de sécurité vous a été fournie. Certaines de ces exigences peuvent être remplies en utilisant les autorisations attribuées au niveau du schéma. Même s'il est facile d'accorder des autorisations substantielles au niveau du schéma, veillez à accorder uniquement les autorisations requises.

Les tâches principales de cet exercice sont les suivantes :

1. Examiner la configuration de sécurité requise qui a été mise à jour à partir du module précédent.
2. Attribuer les autorisations requises.

► **Tâche 1 : Examiner la configuration de sécurité requise qui a été mise à jour à partir du module précédent**

- Prenez connaissance des éléments requis fournis dans la documentation de l'exercice.
- Déterminez les autorisations qui doivent être attribuées au niveau du schéma.

► **Tâche 2 : Attribuer les autorisations requises**

- Attribuez les autorisations requises au niveau du schéma.

Résultats : À la fin de cet exercice, vous devez avoir attribué les autorisations requises au niveau du schéma.

Exercice 2 : Attribuer des autorisations au niveau de l'objet

Scénario

Une liste d'exigences de sécurité vous a été fournie. Certaines de ces exigences doivent être remplies à l'aide des autorisations attribuées au niveau de l'objet. Dans cet exercice, vous allez attribuer les autorisations requises au niveau de l'objet.

Les tâches principales de cet exercice sont les suivantes :

1. Examiner la configuration requise pour la sécurité.
2. Attribuer les autorisations requises.

► Tâche 1 : Examiner la configuration requise pour la sécurité

- Prenez connaissance des éléments requis fournis dans la documentation de l'exercice.
- Déterminez les autorisations qui doivent être attribuées au niveau de l'objet. Cela comprend les autorisations sur les tables, vues, procédures stockées et fonctions si nécessaire.

► Tâche 2 : Attribuer les autorisations requises

- Attribuez les autorisations requises au niveau de l'objet.

Résultats : À la fin de cet exercice, vous devez avoir attribué les autorisations requises au niveau de l'objet.

Exercice difficile 3 : Tester les autorisations (si le temps le permet)

Scénario

Vous devez tester certaines de vos octrois d'autorisation. En particulier, vous devez tester que les commerciaux peuvent sélectionner des lignes dans la table Marketing.CampaignBalance. Mais vous devez également tester que la stagiaire April Reagan ne peut pas sélectionner des lignes dans cette table même si elle est membre du groupe SalesPeople.

La tâche principale de cet exercice est la suivante :

1. Concevoir et exécuter un test.

► Tâche 1 : Concevoir et exécuter un test

- Concevez et exécutez un test pour montrer que Darcy Jayne peut sélectionner des lignes dans la table Marketing.CampaignBalance.
- Concevez et exécutez un test pour montrer qu'April Reagan ne peut pas sélectionner des lignes dans la table Marketing.CampaignBalance.

Résultats : À la fin de cet exercice, vous devez avoir testé les autorisations requises.

Contrôle des acquis et éléments à retenir

- Questions de contrôle des acquis
- Meilleures pratiques

Questions de contrôle des acquis

1. Quelle autorisation doit être attribuée à une fonction avant qu'elle puisse être utilisée dans une contrainte CHECK ?
2. Quelle autorisation doit être attribuée à un schéma pour permettre à un utilisateur de lire les données de toutes les tables, vues et fonctions table ?

Meilleures pratiques

1. Attribuez toujours le moins possible de priviléges dont les utilisateurs ont besoin.
2. Testez le code en tant qu'utilisateur standard au lieu de le tester en tant qu'administrateur.
3. Utilisez les commandes EXECUTE AS et REVERT pour tester rapidement les autorisations utilisateur.

Module 12

Audit des environnements SQL Server

Sommaire :

Leçon 1 : Options d'audit de l'accès aux données dans SQL	12-3
Leçon 2 : Implémentation de SQL Server Audit	12-13
Leçon 3 : Gestion de SQL Server Audit	12-27
Atelier pratique 12 : Audit des environnements SQL Server	12-32

Vue d'ensemble du module

- Options d'audit de l'accès aux données dans SQL Server
- Implémentation de SQL Server Audit
- Gestion de SQL Server Audit

L'un des aspects les plus importants de la configuration de la sécurité pour les systèmes Microsoft® SQL Server® est de s'assurer que les exigences en matière d'audit et de conformité sont remplies. Les organisations devront peut-être atteindre plusieurs objectifs de conformité. Le choix de la configuration appropriée pour SQL Server est souvent un élément clé pour atteindre ces objectifs.

SQL Server 2008 a introduit la fonctionnalité SQL Server Audit dans l'édition Enterprise. (Certaines fonctionnalités d'audit sont incluses dans toutes les éditions de SQL Server 2012). Cette fonctionnalité d'audit permet d'exécuter des types d'audit qui n'étaient pas pris en charge dans les versions antérieures du produit. Dans ce module, vous découvrirez les options d'audit disponibles dans SQL Server et apprendrez à implémenter la fonctionnalité SQL Server Audit et à la gérer.

Objectifs

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

- décrire les options d'audit de l'accès aux données dans SQL Server ;
- implémenter SQL Server Audit ;
- gérer SQL Server Audit.

Leçon 1

Options d'audit de l'accès aux données dans SQL Server

- Discussion : Audit de l'accès aux données
- Utilisation du mode d'audit C2
- Option d'audit Critères communs
- Utilisation de déclencheurs pour l'audit
- Utilisation de SQL Trace pour l'audit
- Démonstration 1A : Utilisation de déclencheurs DML pour l'audit

Avant SQL Server 2008, plusieurs options d'audit étaient disponibles dans SQL Server. Ces options sont toujours disponibles dans SQL Server 2012 et peuvent faire partie de votre stratégie d'audit. En général, une seule fonctionnalité ne fournit pas tous les paramètres d'audit possibles et une combinaison de fonctionnalités doit souvent être utilisée. Dans cette leçon, vous découvrirez les options disponibles.

Objectifs

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

- décrire la nécessité d'un audit ;
- utiliser le mode d'audit C2 ;
- décrire l'option d'audit Critères communs ;
- configurer les déclencheurs pour l'audit ;
- décrire l'utilisation de SQL Trace pour l'audit ;

Discussion : Audit de l'accès aux données

- Pourquoi un audit est-il requis ?
- Quelles méthodes d'audit avez-vous utilisées ?
- Quelles sont les limites des méthodes que vous avez utilisées ?
- Quelles normes nécessitant un audit votre organisation doit-elle respecter ?



Points clés

Il est important de comprendre la nécessité d'un audit.

Question : pourquoi un audit est-il requis ?

Question : quelles méthodes d'audit avez-vous utilisées ?

Question : quelles sont les limites des méthodes que vous avez utilisées ?

Question : quelles normes nécessitant un audit votre organisation doit-elle respecter ?

Utilisation du mode d'audit C2

- Évaluation de classe C2
 - Critères U.S. Trusted Computer Systems Evaluation Criteria (TCSEC)
 - Remplacé par des critères communs
- Mode d'audit C2 SQL Server
 - Configure SQL Server pour stocker les tentatives d'accès aux instructions et aux objets
 - Enregistre à la fois les réussites et les échecs
 - Configuré via l'option de « mode d'audit C2 » de sp_configure
 - À considérer avec précaution car il produit de grandes quantités d'informations sur les événements

Points clés

C2 désigne un ensemble de stratégies de sécurité qui définissent le fonctionnement d'un système sécurisé. La certification s'applique à une installation particulière, y compris le matériel, les logiciels et l'environnement dans lequel le système s'exécute. Les produits ne deviennent pas certifiés C2. Les sites deviennent certifiés C2.

Certification C2

Le système d'exploitation Windows NT® (serveur et station de travail) figurait d'abord sur la liste des produits évalués (EPL, Evaluated Products List) de l'Agence NSA (National Security Agency) en 1995. Cela signifie qu'il a été reconnu que ces produits peuvent être configurés de manière à ce que les sites qui les utilisent deviennent certifiés. Les systèmes Windows peuvent depuis longtemps être configurés de manière conforme.

C2 fait partie d'une série d'évaluations de sécurité, impliquant des produits de sécurité de niveau A, B, C et D. Ces évaluations ont été publiées par le National Computer Security Center (NCSC) dans un document intitulé Trusted Computer System Evaluation Criteria (TCSEC). Ce document était communément appelé le « livre orange » dans le secteur.



Remarque Il est intéressant de noter que le « livre orange » faisait partie de la « série arc-en-ciel ».

La stratégie de sécurité dans C2 est appelée Contrôle d'accès discrétionnaire. Les utilisateurs du système :

- possèdent des objets ;
- ont un contrôle sur la protection des objets qu'ils possèdent ;
- sont responsables de toutes leurs actions liées à l'accès.

Selon les normes actuelles, les critères C2 sont relativement faciles à remplir pour les sites.

SQL Server et C2

SQL Server peut être configuré pour répondre aux critères C2. Une option de configuration système « mode d'audit c2 » peut être activée à l'aide de la procédure stockée système sp_configure.

Bien qu'elle soit facile à configurer, cette option est désormais rarement utilisée voire inutilisable.

L'activation de l'option d'audit C2 peut avoir un impact négatif sur les performances du serveur, car il génère de grandes quantités d'informations sur les événements.

La plupart des utilisateurs qui configurent cette option ne se rendent pas compte de ce qu'ils font et finissent (parfois plus tôt que tard) par manquer d'espace disque.

Pour des raisons pratiques, C2 a désormais été remplacé par la conformité Critères communs, qui est décrite dans la rubrique suivante.

Option d'audit Critères communs

- Critères communs
 - Ratifié comme norme internationale en 1999
 - Remplace l'évaluation C2
 - Est détenu par un groupe de plus de 20 nations
 - Est désormais la norme ISO 15408
- Option de configuration « Conformité des critères communs activée »
 - Activé via sp_configure dans l'édition Entreprise
 - Offres
 - RIP (Residual information protection)
 - Possibilité d'afficher les statistiques de connexion
 - La colonne GRANT ne remplace pas la table DENY

Points clés

Les évaluations C2 étaient basées aux États-Unis. Les Critères communs sont une norme internationale qui a été ratifiée par plus de vingt nations en 1999 et a remplacé l'évaluation C2 comme critère dans la plupart des normes.

Ils sont mis à jour régulièrement par plus de vingt pays et ont été adoptés en tant que norme ISO 15408 (International Standards Organization).

Option « conformité des critères communs activée »

SQL Server fournit une option de serveur « conformité des critères communs activée » qui peut être définie à l'aide de la procédure stockée système sp_configure. Cette option est disponible dans l'édition Enterprise pour une utilisation en production. (Elle est également disponible dans les éditions Developer et Evaluation pour une utilisation non destinée à la production). Vous devez non seulement activer l'option conformité des critères communs activée, mais aussi télécharger et exécuter un script qui achève la configuration de SQL Server pour assurer sa conformité au niveau EAL4+ des Critères communs. Vous pouvez télécharger ce script sur le site Web des Critères communs de Microsoft SQL Server.

Lorsque cette option est activée, trois changements relatifs à l'exécution de SQL Server se produisent :

Problème	Description
RIP (Residual Information Protection)	La mémoire est toujours remplacée par un modèle binaire connue avant d'être réutilisée.
Possibilité d'afficher les statistiques de connexion	L'audit des connexions est automatiquement activée.
La colonne GRANT ne remplace pas la table DENY	Modifie le comportement du système d'autorisation.

L'implémentation de RIP renforce la sécurité, mais a un impact négatif sur les performances du système.

Question : pourquoi est-il nécessaire de modifier GRANT puisque la commande GRANT au niveau de la colonne remplace la commande DENY au niveau de la table ?

Utilisation de déclencheurs pour l'audit

- Les déclencheurs peuvent fournir une partie d'une solution d'audit
 - Déclencheurs DML pour la modification de données
 - Déclencheurs de connexion pour le suivi des connexions
- Limitations
 - Impact sur les performances
 - Possibilité de désactiver les déclencheurs
 - Absence de déclencheurs SELECT
 - Problèmes d'imbrication d'un déclencheur
 - Problèmes de déclencheurs récursifs
 - Complexité relative à l'ordre d'exécution des déclencheurs

Points clés

Les déclencheurs peuvent jouer un rôle important dans l'audit. Avant SQL Server 2008, de nombreuses actions ne pouvaient être auditées que via des déclencheurs.

SQL Server 2005 SP2 a introduit les déclencheurs de connexion. Ceux-ci permettent de suivre plus de détails de connexions et de restaurer des connexions en fonction de la logique métier ou d'administration.

Toutefois, les déclencheurs ne sont pas parfaits :

- Les performances du système sont affectées par les déclencheurs. Avant SQL Server 2005, les tables virtuelles insérées et supprimées des déclencheurs étaient implémentées de la même manière que les vues sur le journal des transactions. Elles n'offraient pas de performances élevées. À partir de SQL Server 2005, la structure interne de ces tables internes a changé. Elles sont désormais basées sur une table de versions de ligne qui réside dans la base de données tempdb. Les déclencheurs qui utilisent ces tables s'exécutent beaucoup plus vite que dans les versions antérieures, mais il peut y avoir un impact important sur les performances de la base de données tempdb qui doit être pris en compte.
- Les déclencheurs peuvent être désactivés. Il s'agit d'un problème important pour l'audit.
- Les utilisateurs demandaient des déclencheurs SELECT. Ils ne souhaitaient pas seulement suivre les modifications de données. De nombreux utilisateurs dans les environnements de haute sécurité souhaitaient voir non seulement les commandes exécutées pour récupérer les données, mais aussi les données récupérées.

- Les déclencheurs ont une limite d'imbrication de trente-deux niveaux au-delà de laquelle ils ne fonctionnent pas.
- Les déclencheurs récursifs peuvent être désactivés.
- Seule une capacité de contrôle limitée de l'ordre d'exécution des déclencheurs est fournie. L'audit doit normalement être le dernier déclencheur qui s'exécute pour s'assurer qu'il capture toutes les modifications apportées par d'autres déclencheurs.

Question : d'après vous, à quoi le terme « déclencheur récursif » peut-il faire référence ?

Utilisation de SQL Trace pour l'audit

- SQL Server Profiler est utilisé pour effectuer le suivi des commandes envoyées au serveur et des erreurs retournées
 - Il peut consommer beaucoup de ressources
 - Il s'exécute en mode interactif
 - Il peut suivre l'exécution des commandes
- Trace SQL
 - Fourni comme un ensemble de procédures stockées système qui permettent la création des traces
 - Il peut être utilisé dans les applications
 - Il est relativement léger lorsqu'il est bien filtré

Points clés

De nombreux utilisateurs ont essayé d'utiliser SQL Server Profiler pour l'audit, car il permet d'effectuer le suivi des commandes envoyées à SQL Server et des erreurs retournées. SQL Server Profiler peut avoir un impact négatif sur les performances lorsqu'il est exécuté en mode interactif sur les systèmes de production.

SQL Trace est un ensemble de procédures stockées système qui sont utilisées par SQL Server Profiler. L'exécution de ces procédures pour gérer le suivi offre une méthode de suivi beaucoup plus simple, en particulier lorsque les événements sont filtrés correctement.

SQL Trace peut jouer un rôle dans l'audit. Comme il a la capacité de capturer les commandes envoyées au serveur, il peut être utilisé pour auditer ces commandes.

Depuis SQL Server 2005, SQL Trace utilise un mécanisme de suivi côté serveur qui garantit aucune perte d'événements, tant que de l'espace est disponible sur le disque et qu'aucune erreur d'écriture ne se produit. Si le disque se remplit ou si des erreurs d'écriture se produisent, le suivi s'arrête. SQL Server continue à moins que le mode d'audit c2 soit également activé. Le risque de manquer des événements doit être pris en compte lors de l'évaluation de l'utilisation de SQL Trace pour les besoins d'audit.

Démonstration 1A : Utilisation de déclencheurs DML pour l'audit

Dans cette démonstration, vous allez apprendre à utiliser les déclencheurs DML pour l'audit

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_12_PRJ\22462A_12_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.

Question : les déclencheurs DML peuvent-ils être utilisés pour auditer la lecture des données d'une table ?

Leçon 2

Implémentation de SQL Server Audit

- Présentation des événements étendus
- Présentation de SQL Server Audit
- Configuration de SQL Server Audit
- Actions et groupes d'actions d'audit
- Définition des cibles d'audit
- Création d'audits
- Création des spécifications de l'audit du serveur
- Création des spécifications de l'audit de la base de données
- Vues de gestion dynamiques et vues système associées à l'audit
- Démonstration 2A : Utilisation de SQL Server Audit

SQL Server 2008 a introduit la fonctionnalité SQL Server Audit. Elle était basée sur un nouveau moteur de gestion complète appelé Événements étendus. Dans cette leçon, vous apprendrez la terminologie de base utilisée par les événements étendus, mais aussi comment SQL Server Audit a été créé en tant que package spécifique dans les événements étendus.

La préparation de SQL Server Audit pour l'utilisation requiert la configuration de plusieurs objets. Dans cette leçon, chacun de ces objets est présenté, avec les détails relatifs à leur configuration. Enfin, vous découvrirez les vues de gestion dynamiques (DMV, Dynamic Management Views) et les vues système qui ont été introduites pour prendre en charge la fonctionnalité SQL Server Audit.

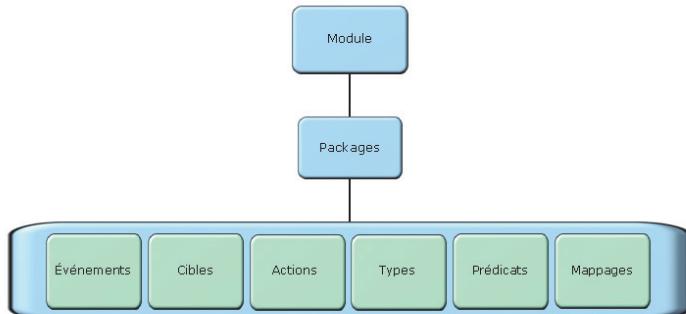
Objectifs

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

- décrire l'infrastructure des événements étendus ;
- décrire SQL Server Audit ;
- configurer SQL Server Audit ;
- détailler les rôles des actions et des groupes d'actions d'audit ;
- définir les cibles d'audit ;
- créer des audits ;
- créer les spécifications de l'audit du serveur ;
- créer les spécifications de l'audit de la base de données ;
- utiliser les vues de gestion dynamiques et les vues système associées à l'audit.

Présentation des événements étendus

- Il s'agit d'un moteur léger de gestion complète des serveurs
- Il est conçu pour être en mesure de traiter tout type d'événement
- Il peut être intégré avec le Suivi d'événements pour Windows (ETW)



Points clés

Un grand nombre d'événements se produisent dans le moteur de base de données SQL Server. Par exemple, un utilisateur peut exécuter une requête, le moteur de base de données peut demander de la mémoire supplémentaire ou les autorisations peuvent être vérifiées. SQL Server 2008 a introduit une nouvelle fonctionnalité appelée Événements étendus qui vous permet de définir les actions à exécuter lorsque des événements se produisent. Lorsque SQL Server exécute son code interne, il vérifie si un utilisateur externe a défini une action qui doit être exécutée à ce moment dans le code. Si une action est définie, un événement est déclenché et ses détails sont envoyés à un emplacement cible. Les cibles peuvent être les fichiers du système d'exploitation, les mémoires tampons en anneau basées sur la mémoire et les journaux des événements Windows.

Les événements étendus sont considérés comme un moteur léger de gestion complète, car il a un impact très limité sur les performances du moteur de base de données qu'il surveille. Ils peuvent être utilisés à de nombreuses fins pour lesquelles SQL Trace est actuellement utilisé.

Les événements étendus sont importants, car SQL Server Audit est basé sur leur infrastructure. Le moteur de gestion fourni par les événements étendus n'est pas lié à des types d'événements particuliers. Le moteur est écrit de façon à pouvoir traiter tout type d'événement.

Les configurations des événements étendus sont fournies dans des fichiers .exe ou .dll appelés « packages ». Les packages sont l'unité de déploiement et d'installation des événements étendus. Un package est un conteneur pour tous les objets qui font partie d'une configuration spécifique d'événements étendus. SQL Server Audit est un package particulier dans les événements étendus. Vous ne pouvez pas modifier sa configuration interne. Vous pouvez modifier d'autres packages.

Les événements étendus utilisent une terminologie spécifique pour décrire les objets qu'ils utilisent :

Objet	Description
Événements	Détails intéressants pendant l'exécution du code.
Cibles	Emplacements où les informations de suivi sont envoyées (les fichiers, par exemple).
Actions	Réponses qui peuvent être apportées aux événements (par exemple, un type d'action capture des plans d'exécution pour l'inclure dans le suivi).
Types	Définitions des objets avec lesquels les événements étendus s'exécutent.
Prédicats	Filtres dynamiques appliqués à la capture d'événements.
Mappages	Mappage des valeurs en chaînes. (Un exemple est le mappage des codes en descriptions).

Présentation de SQL Server Audit

- **SQL Server Audit**
 - Introduit pour la première fois dans SQL Server 2008
 - Système de suivi et de journalisation des événements basé sur des événements étendus
 - Opération complète dans l'édition Enterprise edition de SQL Server 2012
 - Opération de base dans d'autres éditions de SQL Server 2012
- **Inclut**
 - Audits
 - Serveur et spécifications d'audit de la base de données
 - Actions et groupes d'actions
 - Cibles

Points clés

SQL Server Audit a été introduit dans SQL Server 2008 pour résoudre les problèmes de conformité. L'édition Enterprise de SQL Server 2012 fournit des fonctionnalités complètes et les autres éditions fournissent des fonctionnalités de base et se limitent à la définition des spécifications de l'audit du serveur.

SQL Server Audit

Il est important de savoir que SQL Server Audit est le nom de la fonctionnalité et le nom de l'un des objets qui font partie de la fonctionnalité.

Un audit est une définition de l'emplacement où sont envoyés les résultats du processus d'audit. Cela peut sembler illogique au départ, étant donné que le nom fait penser à une action que vous exécutez, pas à un emplacement pour les résultats de l'action. Un audit est créé au niveau de l'instance et plusieurs audits peuvent être créés par instance.

Les résultats d'un audit sont envoyés à une cible.



Remarque Le terme « cible » a la même signification pour SQL Server Audit que pour les événements étendus.

Spécifications d'audit

Les spécifications de l'audit du serveur et de la base de données déterminent les actions à auditer. Il existe des ensembles prédéfinis d'actions appelés « groupes d'actions ». L'utilisation de ces groupes d'actions évite la configuration d'un grand nombre d'actions d'audit individuelles.

Configuration de SQL Server Audit

La configuration de SQL Server Audit est un processus

- Créez un audit et définissez la cible
- Créez une spécification d'audit (serveur ou base de données)
- Activez l'audit et la spécification d'audit
- Consultez les événements de sortie

Points clés

La configuration de SQL Server Audit est un processus en plusieurs étapes :

Étape	Description
Création d'un audit	Détermine le traitement des résultats. Par exemple, lorsque vous configurez un audit, vous décidez des actions à exécuter si l'espace disque est insuffisant. Vous décidez également de la durée pendant laquelle SQL Server peut mettre en mémoire tampon les résultats d'audit avant de les écrire dans la cible.
Définition de la cible	Détermine l'emplacement où la sortie est envoyée.
Création d'une spécification d'audit	Détermine les actions à auditer. Ces actions peuvent être au niveau du serveur ou de la base de données.
Activation de l'audit et de la spécification d'audit	Est l'étape d'activation des objets. (Les audits sont créés dans un état désactivé et les spécifications d'audit sont créées dans un état désactivé par défaut).
Consultez les événements de sortie	Désigne l'extraction des détails de sortie à partir des audits.

Plusieurs options existent pour la lecture des événements de sortie après leur capture :

- Les observateurs d'événements ou de fichiers journaux Windows permettent la lecture des détails du journal d'événements.
- La fonction sys.fn_get_audit_file retourne la sortie basée sur des fichiers sous la forme d'une table qui peut être interrogée dans T-SQL.

Actions et groupes d'actions d'audit

Les actions d'audit constituent d'autres tâches réalisables lorsque des événements se produisent. Les groupes d'actions sont des ensembles d'événements prédéfinis qui peuvent être utilisés au lieu de définir des événements individuels

- Catégories d'actions
 - Serveur
 - Base de données
 - Audit
- Les modifications de l'état d'un audit de serveur sont toujours auditées
- Groupes d'actions
 - Plusieurs groupes d'actions prédéfinies pour chaque catégorie d'audit sont fournis
 - Simplifier la configuration et la gestion des audits

Points clés

Les actions sont les événements qui se produisent et présentent un intérêt pour l'audit. Elles peuvent se produire à trois niveaux : serveur, base de données et audit.

Groupes d'actions

Pour éviter la nécessité de plusieurs actions individuelles, des groupes d'actions sont fournis. Cela simplifie la configuration et la gestion des audits dans la mesure où il n'est pas nécessaire de configurer un grand nombre d'actions individuelles pour l'audit.

Voici des exemples de groupes d'actions :

- BACKUP_RESTORE_GROUP
- DATABASE_MIRRORING_LOGIN_GROUP
- DATABASE_OBJECT_ACCESS_GROUP
- DBCC_GROUP
- FAILED_LOGIN_GROUP
- LOGIN_CHANGE_PASSWORD_GROUP

Notez que toute modification de l'état d'un audit est toujours auditee. Ce paramètre ne peut pas être désactivé.

SQL Server 2012 a introduit un nouveau groupe appelé USER_DEFINED_AUDIT_GROUP. Les applications peuvent provoquer l'écriture des événements d'audit dans ce groupe en appelant la procédure stockée système sp_audit_write.

Question : pourquoi n'existe-t-il pas d'option de désactivation de l'audit des changements d'audit ?

Définition des cibles d'audit

- Les résultats d'un audit sont envoyés à une cible
 - Fichier
 - Journal des événements des applications Windows
 - Journal des événements de sécurité Windows
- Les résultats doivent être réexaminés et archivés périodiquement
- Sécurité des cibles d'audit
 - Soyez prudent lors de l'utilisation du journal des applications, car un utilisateur authentifié peut le lire
 - L'écriture dans le journal des événements de sécurité requiert que le compte de service SQL Server soit ajouté à une stratégie « Générer des audits de sécurité »

Points clés

Les audits peuvent être envoyés à trois cibles dans la version actuelle.

- Les résultats peuvent être envoyés à un fichier. La sortie de fichier offre des performances optimales et constitue l'option la plus simple à configurer.
- Les résultats peuvent être envoyés au journal des événements des applications Windows. Évitez d'envoyer trop d'informations à ce journal, car les administrateurs réseau n'aiment pas les applications qui écrivent beaucoup trop de contenu dans les journaux d'événements.

 **Remarque** Soyez prudent lors de l'utilisation du journal des événements des applications Windows comme cible de sortie pour les informations sensibles, car un utilisateur authentifié peut lire son contenu.

- Les résultats peuvent être envoyés au journal des événements de sécurité Windows. Le journal des événements de sécurité est une option de sortie sécurisée, mais requiert l'ajout du compte de service SQL Server à la stratégie Générer des audits de sécurité avant qu'il puisse être utilisé.

 **Remarque** Il est important que les administrateurs SQL Server aient accès au contenu de l'audit, mais il convient de déterminer si l'utilisation du journal des événements de sécurité est appropriée.

Question : pourquoi de nombreux administrateurs de base de données SQL Server ont-ils du mal à utiliser les entrées d'audit du journal des événements de sécurité Windows ?

Création d'audits

- La création d'un audit nécessite plusieurs configurations

Configuration	Commentaire
Nom d'audit	Nom de l'audit
Délai de file d'attente (en millisecondes)	Durée avant le traitement des actions d'audit
Arrêter le serveur en cas d'échec de l'audit	Indique que SQL Server ne peut pas continuer si l'audit ne fonctionne pas
Destination de l'audit	Cible d'audit
Nombre maximal de fichiers de substitution	Nombre maximal de fichiers à conserver (uniquement pour les fichiers)
Taille de fichier maximale (Mo)	Taille maximale de chaque fichier d'audit
Réserver l'espace disque	Indique si l'espace disque pour les fichiers de contrôle doit être réservé à l'avance
Nombre maximal de fichiers	Traite le nombre de fichiers d'audit

Points clés

Lorsque vous créez un audit, vous prenez des décisions sur la façon dont SQL Server traite les résultats envoyés à la cible d'audit. Les audits peuvent être créés à l'aide de l'interface utilisateur graphique de SSMS ou via la commande CREATE SERVER AUDIT de T-SQL.

Options de configuration de l'audit

Le nom d'un audit est souvent lié aux détails du contenu de l'audit ou à la date et l'heure de création de l'audit ou une combinaison des deux.

Après avoir configuré le nom, la configuration d'un délai de file d'attente est particulièrement importante. Le délai de file d'attente indique (en millisecondes) la durée pendant laquelle SQL Server peut mettre en mémoire tampon les résultats d'audit avant de les envoyer à la cible.



Remarque La valeur choisie pour le délai de file d'attente est un compromis entre la sécurité et les performances. En cas de défaillance du serveur, les résultats contenus dans la mémoire tampon qui ne sont pas encore envoyés à la cible peuvent être perdus. Une valeur de zéro pour le délai de file d'attente provoque des écritures synchrones lorsque des événements se produisent. Cela limite le risque de perte d'événements en cas de défaillance, mais peut avoir un impact important sur les performances.

Pour un audit efficace de la production, l'option d'arrêt du serveur en cas d'échec de l'audit doit être sélectionnée. SQL Server 2012 a introduit une nouvelle option pour faire échouer l'opération qui a déclenché l'audit, au lieu d'arrêter l'intégralité de l'instance du serveur.



Remarque Si l'option d'arrêt est sélectionnée, il se peut que SQL Server ne s'initialise pas si l'audit ne peut pas s'exécuter. Dans la leçon suivante, vous découvrirez comment gérer cette situation.

Vous devez choisir une cible pour la sortie de l'audit. Les cibles d'audit disponibles ont été présentées dans une rubrique précédente.

 **Remarque** Sur Windows® XP, le journal des événements de sécurité n'est pas disponible comme destination.

 **Remarque** Chaque audit peut être la cible d'une spécification de l'audit du serveur et d'une spécification de l'audit de la base de données au maximum.

Question : pourquoi est-il recommandé de sélectionner l'option d'arrêt du serveur en cas d'échec de l'audit ?

Création des spécifications de l'audit du serveur

- Définissez les actions qui doivent être auditées et l'audit auquel les résultats doivent être envoyés
- Peut être configuré dans l'interface utilisateur graphique ou T-SQL

```
CREATE SERVER AUDIT SPECIFICATION FailedLoginSpec
FOR SERVER AUDIT Audit-20121222-171544
ADD (FAILED_LOGIN_GROUP);
```

Points clés

La création d'une spécification de l'audit du serveur peut être effectuée à l'aide de l'interface utilisateur graphique ou de T-SQL. Les spécifications de l'audit du serveur sont créées dans un état désactivé par défaut. Les objets d'audit, y compris les spécifications d'audit, sont généralement désactivés jusqu'à ce que tous les objets d'audit aient été créés.

Spécification de l'audit du serveur

Une spécification de l'audit du serveur détaille les actions à auditer. Vous pouvez choisir des groupes d'actions ou des actions et objets individuels. Dans l'exemple illustré sur la diapositive, une spécification de l'audit du serveur est créée à l'aide de l'instruction CREATE SERVER AUDIT SPECIFICATION.

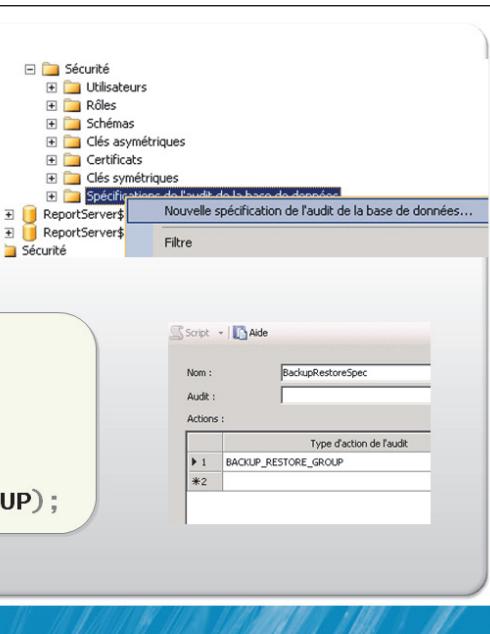
La configuration de la même spécification à l'aide de l'interface utilisateur graphique est également illustrée.

Le nom de la spécification est FailedLoginSpec et les données collectées à partir de la spécification sont envoyées à l'audit Audit-20121222-171544. Le groupe d'actions à auditer est FAILED_LOGIN_GROUP.

Question : pourquoi l'activation de la journalisation des échecs de connexion présente-t-elle des risques potentiels pour la disponibilité ?

Création des spécifications de l'audit de la base de données

- Définissez les actions qui doivent être auditées et l'audit auquel les résultats doivent être envoyés
- Peut être configuré dans l'interface utilisateur graphique ou T-SQL



Points clés

La création d'une spécification de l'audit de la base de données peut également être effectuée à l'aide de l'interface utilisateur graphique ou de T-SQL. Les spécifications de l'audit de la base de données sont également créées dans un état désactivé par défaut. Elles peuvent être créées uniquement dans l'édition Enterprise.

Spécification de l'audit de la base de données

Une spécification de l'audit du serveur détaille les actions à auditer. Vous pouvez choisir des groupes d'actions ou des actions et objets individuels.

Dans l'exemple illustré sur la diapositive, une spécification de l'audit de la base de données est créée à l'aide de l'instruction CREATE DATABASE AUDIT SPECIFICATION. La configuration de la même spécification à l'aide de l'interface utilisateur graphique est également illustrée.

Le nom de la spécification est BackupRestoreSpec et les données collectées à partir de la spécification sont envoyées à l'audit Audit-20121222-171544. Le groupe d'actions à auditer est BACKUP_RESTORE_GROUP.

Vues de gestion dynamiques et vues système associées à l'audit

- SQL Server fournit un ensemble de vues de gestion dynamiques et de vues système pour la gestion de SQL Server Audit

Vues de gestion dynamiques associées à l'audit

sys.dm_server_audit_status
sys.dm_audit_actions
sys.dm_audit_class_type_map

Vues système associées à l'audit

sys.server_audits
sys.server_file_audits
sys.server_audit_specifications
sys.server_audit_specification_details
sys.database_audit_specifications
sys.database_audit_specification_details

Points clés

SQL Server fournit plusieurs vues de gestion dynamiques et vues système qui peuvent vous aider à gérer SQL Server Audit.

Les vues de gestion dynamiques et les vues système suivantes sont disponibles :

Vue de gestion dynamique/Vue	Description
sys.dm_server_audit_status	Retourne une ligne pour chaque audit du serveur qui indique l'état actuel de l'audit.
sys.dm_audit_actions	Retourne une ligne pour chaque action d'audit qui peut être signalée dans le journal d'audit et chaque groupe d'actions qui peut être configuré dans le cadre d'un audit.
sys.dm_audit_class_type_map	Retourne une table qui mappe les types de classe aux descriptions de classe.
sys.server_audits	Contient une ligne pour chaque audit SQL Server dans une instance du serveur.
sys.server_file_audits	Contient des informations étendues sur le type d'audit de fichier dans un audit SQL Server.
sys.server_audit_specifications	Contient des informations sur les spécifications de l'audit du serveur dans un audit SQL Server.

(suite)

Vue de gestion dynamique/Vue	Description
sys.server_audit_specification_details	Contient des informations sur les détails (actions) des spécifications de l'audit du serveur dans un audit SQL Server.
Sys.database_audit_specifications	Contient des informations sur les spécifications de l'audit de la base de données dans un audit SQL Server.
Sys.database_audit_specification_details	Contient des informations sur les spécifications de l'audit de la base de données dans un audit SQL Server.

Plusieurs vues système sont utilisées dans les démonstrations à venir.

Démonstration 2A : Utilisation de SQL Server Audit

Dans cette démonstration, vous allez apprendre à

- Créer un audit SQL Server Audit et définir sa cible
- Créer et activer une spécification d'audit de base de données
- Créer un événement auditable et l'afficher dans l'Observateur d'événements Windows

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_12_PRJ\22462A_12_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.

Question : quelles sont les trois cibles d'événement possibles pour SQL Server Audit ?

Leçon 3

Gestion de SQL Server Audit

- Récupération des audits
- Utilisation de la structure d'enregistrement de l'audit
- Problèmes potentiels de SQL Server Audit
- Démonstration 3A : Affichage de la sortie d'un audit basé sur des fichiers

Il est important de pouvoir récupérer les résultats des audits et de comprendre quelques aspects de la gestion continue de SQL Server Audit. En particulier, vous examinerez les problèmes liés à la migration des bases de données entre des serveurs et apprendrez à redémarrer les serveurs si SQL Server refuse de démarrer en raison d'un échec de l'audit.

Objectifs

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

- récupérer les audits ;
- utiliser la structure d'enregistrement de l'audit ;
- identifier les problèmes potentiels d'audit de SQL Server.

Récupération des audits

- Les audits de journal des événements peuvent être récupérés à l'aide des visionneuses de journal fournies par le système d'exploitation
- Les audits basés sur des fichiers peuvent être récupérés et interrogés à l'aide de la fonction sys.fn_get_audit_file

```
SELECT * FROM sys.fn_get_audit_file(
    'J:\SQLAudits\Audit\LoginLogoutLog\*',
    NULL,
    NULL);
```

Points clés

Aucune configuration particulière n'est requise pour afficher les audits envoyés aux journaux d'événements. SSMS fournit un lecteur de journal pour ces cibles.

Récupération de la sortie du fichier

Pour les journaux envoyés à un fichier, SQL Server fournit une fonction qui retourne le contenu des journaux basés sur des fichiers sous la forme d'une table que vous pouvez interroger avec T-SQL.



Remarque Le nom de fichier que vous fournissez au paramètre FILEPATH lors de la création d'un audit du serveur est en réalité le nom d'un dossier.

Le dossier contenant les journaux d'audit comporte souvent plusieurs fichiers d'audit. La fonction sys.fn_get_audit_file est utilisée pour récupérer ces fichiers. Elle accepte trois paramètres : file_pattern, initial_file_name et audit_record_offset. Le paramètre file_pattern fourni peut être dans l'un des trois formats :

Format	Description
<chemin d'accès>*	Collecte tous les fichiers d'audit dans l'emplacement spécifié.
<chemin d'accès>\LoginsAudit_{GUID}	Collecte tous les fichiers d'audit portant la paire nom/GUID spécifiée.
<chemin d'accès>\LoginsAudit_{GUID}_00_29384.sqlaudit	Collecte un fichier d'audit spécifique.

Utilisation de la structure d'enregistrement de l'audit

Chaque ligne écrite dans la cible est appelée un enregistrement d'audit

- Les actions ne remplissent pas toutes les colonnes
- 4 000 caractères de donnée maximum pour les champs de caractère dans les enregistrements d'audit
 - Plusieurs enregistrements peuvent être requis pour une action
 - Tous les autres champs sont dupliqués dans chaque ligne
 - La colonne sequence_no est incrémentée de chaque ligne dans un enregistrement d'audit de plusieurs lignes

Points clés

La structure d'enregistrement de l'audit est détaillée dans la documentation en ligne sous la rubrique « sys.fn_get_audit_file (Transact-SQL) ».

Les enregistrements d'audit doivent pouvoir être stockés dans les journaux d'événements système ainsi que dans les fichiers. Par conséquent, le format d'enregistrement est limité en taille par les règles associées ces systèmes de journalisation des événements. Les champs de caractère sont divisés en blocs de 4000 caractères et les blocs sont répartis sur plusieurs entrées.

Cela signifie qu'un événement unique peut générer plusieurs entrées d'audit. Une colonne sequence_no est fournie pour indiquer l'ordre de plusieurs entrées de ligne.

Problèmes potentiels de SQL Server Audit

- Déplacement de bases de données entre serveurs
 - Peut provoquer des spécifications d'audit orphelines semblables aux SID incompatibles pour les utilisateurs
 - Se produit en attachant (ou en restaurant) une base de données avec une spécification d'audit GUID qui n'existe pas sur le serveur
 - Résoudre les incompatibilités via CREATE SERVER AUDIT
 - Si la base de données est déplacée vers une édition de SQL Server qui ne prend pas en charge l'audit, la liaison a lieu mais l'audit est ignoré
- Serveurs en miroir
 - Doit respecter les conditions GUID d'audit sur les deux partenaires de mise en miroir
- Performances
 - L'impact des écritures d'audit doit être pris en compte
 - Un échec lors du lancement de l'audit peut entraîner l'échec du démarrage du serveur

Points clés

Il existe plusieurs problèmes potentiels à prendre en compte avec SQL Server Audit.

- Chaque audit est identifié par un GUID. Lorsqu'une base de données est restaurée ou reliée à un serveur, une tentative de mise en correspondance du GUI de la base de données avec celui de l'audit du serveur est effectuée. En l'absence de correspondance, l'audit ne s'exécute pas tant que la situation n'est pas corrigée en exécutant la commande CREATE SERVER AUDIT pour définir le GUID approprié.
- Si les bases de données sont reliées à des éditions de SQL Server qui ne prennent pas en charge le même niveau de fonction d'audit, la liaison a lieu mais l'audit est ignoré.
- Les serveurs en miroir présentent un problème similaire d'incompatibilité de GUID. Le partenaire de mise en miroir doit disposer d'un audit du serveur avec le même GUID. Pour ce faire, utilisez la commande CREATE SERVER AUDIT et fournissez la valeur GUID pour qu'elle corresponde à la valeur du serveur principal.
- En général, l'impact sur les performances des écritures d'audit doit être pris en compte. Si l'espace disque se remplit, SQL Server ne peut pas démarrer. Dans ce cas, vous devrez peut-être forcer l'entrée au serveur en effectuant un démarrage en mode mono-utilisateur et en utilisant le paramètre de démarrage -f.

Question : pourquoi les audits sont-ils identifiés par un GUID et un nom ?

Démonstration 3A : Affichage de la sortie d'un audit basé sur des fichiers

Dans cette démonstration, vous découvrirez comment afficher les résultats d'un audit basé sur des fichiers

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_12_PRJ\22462A_12_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.
Question : pourquoi le journal d'audit contient-il deux entrées ?

Atelier pratique 12 : Audit des environnements SQL Server

- Exercice 1 : Déterminer la configuration d'audit et créer l'audit
- Exercice 2 : Créer les spécifications de l'audit du serveur
- Exercice 3 : Créer les spécifications de l'audit de la base de données
- Exercice difficile 4 : Tester les fonctionnalités d'audit (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_12_PRJ\22462A_12_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 autorisé les utilisateurs à accéder à l'instance Proseware. Le service de la conformité vous a fourni les détails des critères d'audit pour l'instance du serveur Proseware et la base de données MarketDev. Les critères d'audit incluent la nécessité d'auditer les activités par rapport aux tables de la base de données MarketDev contenant des informations sensibles. Dans cet atelier pratique, vous implémenterez une stratégie pour activer l'audit approprié.

Si vous avez suffisamment de temps, vous devez tester la stratégie d'audit et écrire une requête pour récupérer les enregistrements d'audit.

Documentation

Critères d'audit du service de la conformité

1. Tous les enregistrements d'audit doivent être écrits dans le dossier C:\Audit\JournalAudit.
2. Les enregistrements d'audit doivent être écrits aussi vite que possible. Toutefois, une tolérance de deux secondes des enregistrements d'audit est la perte maximale autorisée en cas d'échec.
3. L'instance du serveur ne doit pas continuer à s'exécuter si l'audit n'est pas effectué.
4. Le nom de l'audit doit être Audit de la conformité de Proseware.
5. Il n'existe aucune limite au nombre de fichiers d'audit pouvant être créés ; toutefois, la taille de chaque fichier d'audit doit être limitée à 1 Go.
6. Au niveau du serveur, les éléments suivants doivent être audités :
 - échecs des tentatives de connexion ;
 - modifications de l'appartenance des rôles serveur ;
 - modifications des connexions du serveur (principaux) ;
 - modifications des mots de passe.
7. Au niveau de la base de données MarketDev, les éléments suivants doivent être audités :
 - modifications de l'appartenance des rôles de base de données ;
 - sauvegardes et restaurations de la base de données ;
 - modifications des autorisations dans la base de données ;
 - modifications des utilisateurs de base de données (principaux) ;
 - modification de la propriété des bases de données ;
 - mises à jour de la table Marketing.CampaignBalance ;
 - exécutions de la procédure stockée Marketing.MoveCampaignBalance.

Exercice 1 : Déterminer la configuration d'audit et créer l'audit

Scénario

Vous devez déterminer la configuration d'un audit du serveur, selon les exigences de sécurité de l'entreprise. Dans cet exercice, vous allez créer l'audit du serveur requis.

Les tâches principales de cet exercice sont les suivantes :

- Examiner la configuration requise.
- Créer l'audit du serveur.

► Tâche 1 : Examiner la configuration requise

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

► Tâche 2 : Créer l'audit du serveur

- Déterminez la configuration de l'audit du serveur requis.
- Créez l'audit du serveur à l'aide de SQL Server Management Studio.
- Activez l'audit du serveur.

Résultats : À la fin de cet exercice, vous devez avoir créé l'audit du serveur requis.

Exercice 2 : Créer les spécifications de l'audit du serveur

Scénario

Vous devez déterminer les exigences de l'entreprise qui peuvent être remplis via les spécifications de l'audit du serveur. Vous devrez ensuite déterminer les spécifications requises de l'audit du serveur et les créer.

Les tâches principales de cet exercice sont les suivantes :

- Examiner la configuration requise.
- Créer les spécifications de l'audit du serveur.

► Tâche 1 : Examiner la configuration requise

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

► Tâche 2 : Créer les spécifications de l'audit du serveur

- Déterminez les spécifications de l'audit du serveur requises.
- Créez les spécifications de l'audit du serveur à l'aide de SQL Server Management Studio.
- Activez les spécifications de l'audit du serveur.

Résultats : À la fin de cet exercice, vous devez avoir créé les spécifications requises de l'audit du serveur.

Exercice 3 : Créer les spécifications de l'audit de la base de données

Scénario

Certains des critères d'audit nécessitent la création des spécifications de l'audit de la base de données. Dans cet exercice, vous déterminerez les critères d'audit qui peuvent être remplis par les spécifications de l'audit de la base de données. Vous créerez ensuite ces spécifications d'audit de la base de données.

Les tâches principales de cet exercice sont les suivantes :

- Examiner la configuration requise.
- Créer les spécifications de l'audit de la base de données.

► Tâche 1 : Examiner la configuration requise

- Examiner la configuration requise.

► Tâche 2 : Créer les spécifications de l'audit de la base de données

- Déterminez les spécifications requises de l'audit de la base de données.
- Créez les spécifications requises de l'audit de la base de données à l'aide de SQL Server Management Studio.
- Activez les spécifications de l'audit de la base de données que vous avez créées.

Résultats : À la fin de cet exercice, vous devez avoir créé les spécifications requises de l'audit de la base de données.

Exercice difficile 4 : Tester les fonctionnalités d'audit (si le temps le permet)

Scénario

Vous devez vérifier que l'audit que vous avez configurée s'exécute comme prévu. Vous exécuterez un script de charge de travail de test et examinerez la capture d'audit à l'aide de l'interface utilisateur graphique de SSMS et de T-SQL.

Les tâches principales de cet exercice sont les suivantes :

- Exécuter le script de charge de travail.
- Examiner les informations d'audit capturées.

► **Tâche 1 : Exécuter le script de charge de travail**

- Dans l'Explorateur de solutions, ouvrez et exécutez le script de charge de travail 81 – Lab Exercise 4a.sql.

► **Tâche 2 : Examiner les informations d'audit capturées**

- Examinez les informations d'audit capturées à l'aide de l'option Afficher les journaux d'audit dans SQL Server Management Studio. (Il s'agit d'une option de menu contextuel de l'audit du serveur).
- Écrivez une requête pour récupérer les informations du journal d'audit à l'aide de T-SQL.

Résultats : À la fin de cet exercice, vous devez avoir vérifié que l'audit s'exécute comme prévu.

Contrôle des acquis et éléments à retenir

- Questions de contrôle des acquis
- Meilleures pratiques

Questions de contrôle des acquis

1. Quelles sont les trois cibles pour les audits SQL Server ?
2. Lorsque la conformité des critères communs est activée dans SQL Server, quels changements se produisent dans les autorisations au niveau des colonnes ?
3. Vous souhaitez peut-être faire auditer des actions par un administrateur de base de données. Comment savez-vous si l'administrateur de base de données a arrêté l'audit pendant l'exécution des actions de conversion ?

Meilleures pratiques

1. Choisissez l'option d'arrêt de SQL Server en cas d'échec de l'audit. Il est généralement inutile de configurer l'audit et de rencontrer ensuite des situations où des événements se produisent sans être audités. Cela est particulièrement important dans les environnements hautement sécurisés.
2. Vérifiez que les audits de fichiers sont placés sur des lecteurs disposant d'une grande quantité d'espace libre et que cet espace libre est surveillé régulièrement.

Module 13

Automatisation de la gestion de SQL Server 2012

Sommaire :

Leçon 1 : Automatisation de la gestion de SQL Server	13-3
Leçon 2 : Utilisation de SQL Server Agent	13-11
Leçon 3 : Gestion des travaux de SQL Server Agent	13-20
Atelier pratique 13 : Automatisation de la gestion de SQL Server	13-27

Vue d'ensemble du module

- Automatisation de la gestion de SQL Server
- Utilisation de SQL Server Agent
- Gestion des travaux de SQL Server Agent

Les outils fournis avec Microsoft® SQL Server® facilitent l'administration par rapport à d'autres moteurs de base de données. Même lorsque les tâches sont faciles à exécuter, il est courant d'avoir à répéter plusieurs fois une tâche. Les administrateurs de base de données efficaces apprennent à automatiser les tâches répétitives. L'automatisation des tâches permet d'éviter les situations où un administrateur oublie d'exécuter une tâche au moment requis. Mais plus important peut-être, l'automatisation des tâches permet de s'assurer que les tâches se déroulent de manière cohérente, chaque fois qu'elles sont exécutées.

SQL Server Agent est le service fourni dans toutes les éditions de SQL Server 2012 (à l'exception de SQL Server Express Edition) qui gère l'automatisation des tâches. Un ensemble de tâches devant être exécutées est appelé un travail de SQL Server Agent. Il est important d'apprendre à créer et à gérer ces travaux.

Objectifs

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

- automatiser la gestion de SQL Server ;
- utiliser SQL Server Agent ;
- gérer les travaux de SQL Server Agent.

Leçon 1

Automatisation de la gestion de SQL Server

- Avantages de l'automatisation de la gestion de SQL Server
- Options disponibles pour l'automatisation de la gestion de SQL Server
- Vue d'ensemble de SQL Server Agent
- Démonstration 1A : Utilisation de SQL Server Agent

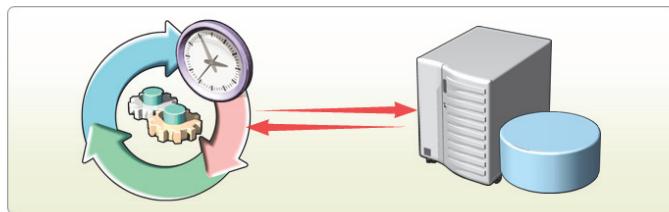
L'automatisation de la gestion de SQL Server offre de nombreux avantages. La plupart des avantages portent sur l'exécution fiable et cohérente des tâches de gestion courantes. SQL Server est une plateforme flexible qui fournit plusieurs méthodes d'automatisation de la gestion, mais l'outil le plus important pour l'automatisation de la gestion est SQL Server Agent. Tous les administrateurs de base de données qui utilisent SQL Server doivent se familiariser avec la configuration et la gestion continue de SQL Server Agent.

Objectifs

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

- expliquer les avantages de l'automatisation de la gestion de SQL Server ;
- décrire les options disponibles pour automatiser la gestion de SQL Server et l'infrastructure fournie avec SQL Server Agent ;
- décrire SQL Server Agent.

Avantages de l'automatisation de la gestion de SQL Server



- Réduction de la charge de travail administrative
 - Automatisation et planification des tâches régulières
- Gestion proactive
 - Analyse des performances
 - Identification et réponse à des problèmes potentiels

Points clés

Tous les administrateurs de base de données efficaces automatisent leurs tâches d'administration courantes. Les avantages de l'automatisation de la gestion de SQL Server sont les suivants.

Réduction de la charge de travail administrative

Malheureusement, certains administrateurs qui utilisent SQL Server, Windows® et d'autres outils perçoivent leurs rôles en termes de flux constant de tâches administratives répétitives. Par exemple, un administrateur Windows travaillant dans un service universitaire peut recevoir des demandes régulières pour créer un grand nombre de comptes d'utilisateurs. L'administrateur est content de créer chacun de ces comptes un par un, à l'aide des outils standard. Un administrateur plus efficace apprendrait à écrire un script pour créer des utilisateurs et à l'exécuter au lieu de créer manuellement les utilisateurs.

Le même type de situation se produit avec les tâches courantes dans SQL Server. Bien que ces tâches puissent être exécutées individuellement ou manuellement, les administrateurs de base de données efficaces ne le font pas. Ils automatisent toutes leurs tâches courantes et répétitives. L'automatisation supprime la charge de travail répétitive des administrateurs tout en leur permettant de gérer un plus grand nombre de systèmes ou d'exécuter des tâches plus importantes pour l'organisation.

Exécution fiable des tâches courantes

Lorsque des tâches courantes sont exécutées manuellement, il est toujours possible qu'une tâche importante soit oubliée. Par exemple, un administrateur de base de données peut oublier d'effectuer des sauvegardes de la base de données. L'automatisation permet aux administrateurs de se focaliser sur les exceptions qui se produisent pendant les tâches courantes plutôt que sur l'exécution des tâches.

Exécution cohérente des tâches courantes

Un autre problème qui peut se produire pendant l'exécution manuelle de tâches courantes est qu'elles ne sont peut-être pas exécutées de la même manière chaque fois. Imaginez une situation où un administrateur de base de données doit archiver des données d'un jeu de tables de production dans un jeu de tables d'historique chaque lundi matin. Les nouvelles tables doivent porter le même nom que les tables d'origine, avec un suffixe contenant la date actuelle.

Même si l'administrateur peut se rappeler d'exécuter cette tâche chaque lundi matin, quelle est la probabilité qu'une des erreurs suivantes se produise ?

- copier les tables incorrectes ;
- copier uniquement certaines tables ;
- oublier la date correcte lors de la création du suffixe ;
- mettre en forme de manière incorrecte la date du suffixe ;
- copier les données dans une table d'archive incorrecte.

Toute personne impliquée dans l'administration continue de systèmes vous dira que ces problèmes et d'autres problèmes se produisent de temps en temps, même lorsque les tâches sont exécutées par des administrateurs expérimentés et fiables. L'automatisation des tâches courantes contribue à ce que les tâches se déroulent de manière cohérente, chaque fois qu'elles sont exécutées.

Gestion proactive

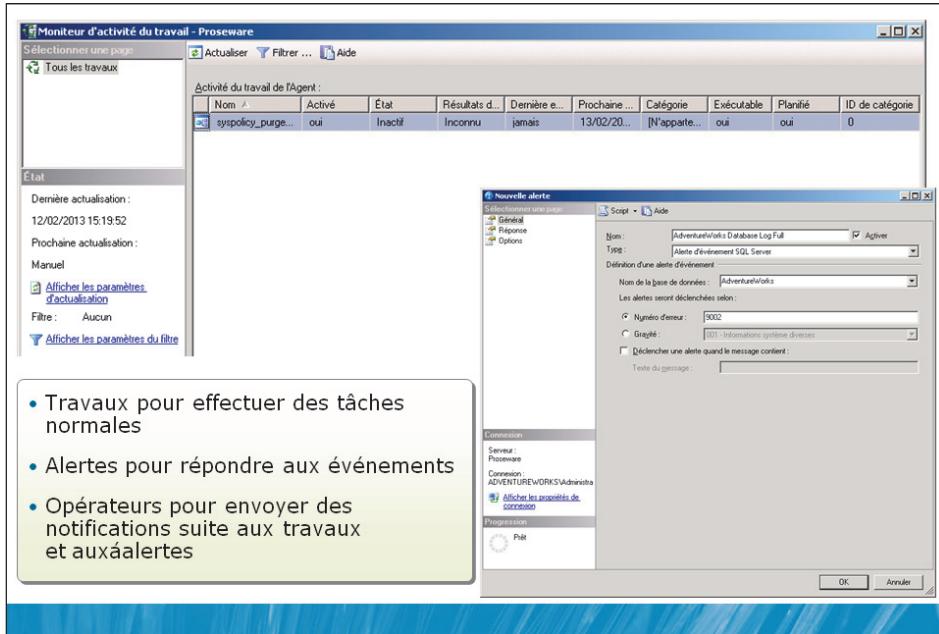
Une fois que les tâches courantes sont automatisées, il peut arriver que l'exécution courante des tâches échoue mais qu'aucun administrateur ne s'en rende compte. Par exemple, il existe de nombreux récits tragiques sur les forums de support de la communauté SQL Server provenant d'administrateurs qui ont automatisé la sauvegarde de leurs bases de données et qui ne se sont pas rendus compte que les sauvegardes avaient échoué depuis longtemps, jusqu'à ce qu'ils aient besoin des sauvegardes.

Outre l'automatisation de vos tâches de routine, vous devez veiller à créer des notifications qui vous informent en cas d'échec des tâches, même si vous ne pouvez pas imaginer une telle situation. Par exemple, vous avez peut-être créé une stratégie de sauvegarde qui crée des sauvegardes de base de données dans un dossier donné. Cela peut fonctionner de manière fiable pendant des années jusqu'à ce qu'un autre administrateur supprime ou renomme par inadvertance le dossier cible. Vous devez être informé dès que ce problème se produit pour pouvoir remédier à la situation.

Un administrateur plus proactif tentera de détecter des problèmes potentiels avant qu'ils se produisent. Par exemple, au lieu de recevoir une notification d'échec à cause d'une saturation de disque, un administrateur peut planifier des contrôles réguliers de l'espace disque disponible et s'assurer de recevoir une notification lorsque l'espace libre disponible devient trop faible. SQL Server fournit des alertes sur les états du système et les conditions de performances.

Question : quelles tâches doivent être automatisées sur les systèmes de votre entreprise ?

Options disponibles pour l'automatisation de la gestion de SQL Server



Points clés

La principale méthode d'automatisation des tâches de gestion, d'administration et des autres tâches courantes dans SQL Server 2012 consiste à utiliser SQL Server Agent.

Infrastructure pour SQL Server Agent

L'infrastructure de gestion fournie par SQL Server Agent repose sur deux objets principaux :

- travaux utilisés pour automatiser les tâches ;
- alertes utilisées pour répondre aux événements.

Les travaux peuvent être utilisés pour planifier plusieurs types de tâches, y compris les tâches nécessaires à l'implémentation d'autres fonctionnalités de SQL Server. Par exemple, la réPLICATION, la capture de données modifiées (CDC, Change Data Capture), la collecte de données et les fonctionnalités de gestion basées sur des stratégies (PBM, Policy Based Management) de SQL Server créent les travaux de SQL Server Agent. (La collecte de données est abordée dans le module 18).

Remarque La réPLICATION, la capture de données modifiées et la gestion basée sur des stratégies sont des rubriques avancées qui ne sont pas traitées dans ce cours.

Le système d'alerte fourni par SQL Server Agent peut répondre à plusieurs types d'alertes, y compris les messages d'erreur de SQL Server, les événements des compteurs de performances SQL Server et les alertes WMI (Windows Management Instrumentation).

En réponse à une alerte, une action peut être configurée, par exemple l'exécution d'un travail de SQL Server Agent ou l'envoi d'une notification à un administrateur. Dans SQL Server Agent, les administrateurs qui peuvent être notifiés sont appelés des opérateurs. Les opérateurs sont communément notifiés via la messagerie SMTP. (Les alertes sont traitées dans le Module 15).

Notez qu'il existe d'autres fonctionnalités SQL Server qui peuvent être utilisées pour automatiser des tâches de surveillance complexes. La fonctionnalité Événements étendus est un exemple, mais n'est pas traitée dans cette formation.

Question : réfléchissez à des événements susceptibles de se produire sur un système SQL Server et dont vous souhaiteriez être alerté.

Vue d'ensemble de SQL Server Agent

Agent SQL Server est le composant de SQL Server chargé de l'automatisation

- S'exécute en tant que service Windows
- Doit être en cours d'exécution pour
 - Exécuter des travaux
 - Déclencher des alertes
 - Contacter les opérateurs
- Le mode de démarrage doit être défini sur Automatique

Points clés

Comme indiqué précédemment dans ce module, SQL Server Agent est le composant de SQL Server qui gère l'automatisation des tâches d'administration de SQL Server. SQL Server Agent fonctionne comme un service Windows.

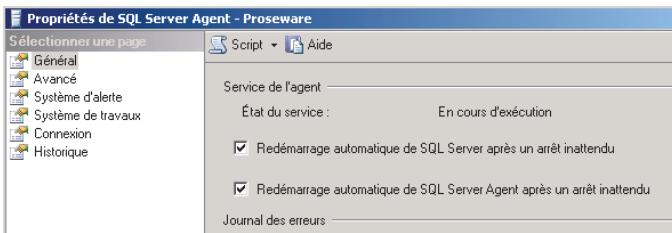
Démarrage de SQL Server Agent

Pour que SQL Server Agent puisse jouer son rôle principal d'exécution des travaux et de déclenchement des alertes, il doit toujours être en cours d'exécution. Pour cette raison, SQL Server Agent est généralement configuré pour démarrer automatiquement au lancement du système d'exploitation. Notez que l'option par défaut pendant l'installation de SQL Server est le démarrage manuel de SQL Server Agent. La conception du processus d'installation de SQL Server garantit que les services et les composants ne sont pas installés ou démarrés sauf s'ils sont requis. Cela signifie que l'option d'installation par défaut doit être modifiée si SQL Server Agent doit s'exécuter sur votre système.

Le mode de démarrage de SQL Server Agent est configuré dans les propriétés du service SQL Server Agent dans le Gestionnaire de configuration SQL Server, comme illustré sur la diapositive. Notez que trois modes de démarrage sont configurables :

Mode de démarrage	Description
Automatique	Le service s'exécute au démarrage du système d'exploitation.
Désactivé	Le service ne démarre pas, même si vous tentez de le démarrer manuellement.
Manuel	Le service doit être démarré manuellement.

En outre, vous pouvez configurer le service SQL Server Agent de manière à ce qu'il redémarre automatiquement s'il s'arrête de façon inattendue. L'option de redémarrage automatique est définie dans la page des propriétés de SQL Server Agent dans SSMS, comme illustré ci-dessous :



Pour redémarrer automatiquement, le compte de service SQL Server Agent doit être membre du groupe Administrateurs locaux sur l'ordinateur sur lequel SQL Server est installé, mais cela n'est pas considéré comme une méthode recommandée. Il est préférable d'utiliser un outil de surveillance externe tel que System Center Operations Manager pour surveiller et redémarrer le service SQL Server Agent si nécessaire.

Question : pourquoi le service SQL Server Agent doit-il toujours être configuré pour démarrer automatiquement ?

Démonstration 1A : Utilisation de SQL Server Agent

Dans cette démonstration, vous allez apprendre à

- Comment configurer Agent SQL Server
- Comment vérifier les travaux d'Agent SQL Server à l'aide de PowerShell

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_13_PRJ\22462A_13_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

Utilisation de SQL Server Agent

- Définition des travaux, des types d'étapes de travail et des catégories de travaux
- Création d'étapes de travail
- Programmation des travaux en vue de leur exécution
- Génération d'un script des travaux
- Démonstration 2A : Génération d'un script des travaux

Vous avez appris que SQL Server Agent est le principal outil pour automatiser les tâches dans SQL Server. Les administrateurs de base de données doivent être en mesure de créer et de configurer les travaux de SQL Server Agent. Des travaux peuvent être créés pour implémenter différents types de tâches et peuvent être classés pour faciliter la gestion.

La création d'un travail implique de créer une série d'étapes que le travail exécute, ainsi que le flux de travail qui détermine les étapes à exécuter, et leur ordre d'exécution.

Une fois que les étapes qu'un travail doit exécuter ont été définies, vous devez déterminer la date d'exécution du travail. La plupart des travaux de SQL Server Agent sont exécutés selon des planifications définies. SQL Server vous permet de créer un jeu flexible de planifications qui peuvent être partagées entre des travaux.

Il est important d'apprendre à générer un script pour les travaux créés. Le script de travaux permet de recréer rapidement les travaux en cas d'échec, mais aussi de les recréer dans d'autres environnements. Par exemple, les travaux ont peut-être été créés dans un environnement de test, mais doivent être exécutés dans un environnement de production.

Objectifs

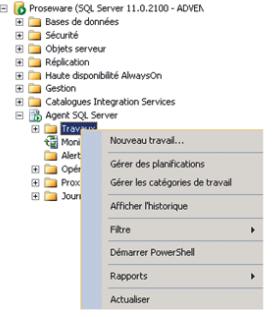
À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- définir les travaux, les types de travaux et les catégories de travaux ;
- créer des d'étapes de travail ;
- programmer des travaux en vue de leur exécution ;
- générer un script pour les travaux.

Définition des travaux, des types d'étapes de travail et des catégories de travaux

Les travaux d'Agent SQL Server correspondent à une série d'opérations spécifique exécutée de manière séquentielle par Agent SQL Server

- Ces travaux prennent en charge plusieurs types, notamment
 - Transact-SQL
 - Script de ligne de commande ou exécution d'une application
 - Exécution de script PowerShell
- Il est possible de configurer des travaux pour
 - S'exécuter une fois ou à plusieurs reprises
 - Démarrer à la mise en route d'Agent SQL Server ou manuellement
- Il est possible d'affecter des travaux à une catégorie



Nouveau travail...

Gérer des planifications

Gérer les catégories de travail

Afficher Historique

Filtre

Démarrer PowerShell

Rapports

Actualiser

Type :

Script Transact-SQL (T-SQL)

Agent de lecture de fil de l'attente de réplication

Commande SQL Server Analysis Services

Fusion de réplication

Instantané de réplication

Lecteur du journal des transactions de réplication

Package SQL Server Integration Services

PowerShell

Requête SQL Server Analysis Services

Script ActiveX

Script Transact-SQL (T-SQL)

Serveur de distribution de réplication

Système d'exploitation (CmdExec)

Points clés

Les travaux de SQL Server Agent se composent d'une série d'opérations qui doivent être exécutées dans l'ordre. Dans la plupart des travaux, les étapes sont effectuées de manière séquentielle, mais un administrateur peut choisir de contrôler leur ordre.

En configurant l'action pour qu'elle se produise en cas de réussite ou d'échec de chaque étape de travail, il est possible de créer un flux de travail qui détermine le flux logique global du travail.

Types d'étapes de travail

Notez que chaque étape du travail contient un type associé qui définit le type d'opération à exécuter. Les types les plus couramment utilisés sont :

- exécution d'un script de ligne de commande, d'un lot de commandes ou d'une application ;
- exécution d'une instruction T-SQL ;
- exécution d'un script Windows PowerShell® ;
- exécution d'un package SQL Server Integration Services ;
- exécution des commandes et requêtes Analysis Services.

 **Remarque** La possibilité d'exécuter les scripts ActiveX® a été conservée pour des raisons de rétrocompatibilité, mais cette option est déconseillée et ne devrait pas être utilisée pour un nouveau développement.

 **Remarque** D'autres types d'étapes de travail spécialisés sont utilisés par les fonctionnalités de SQL Server, telles que la réPLICATION. La réPLICATION est une rubrique avancée qui n'est pas traité dans ce cours.

Planifications de travaux

Une ou plusieurs planifications peuvent être définies pour chaque travail. Il est possible de définir les planifications comme étant périodiques, d'autres types de planification fournissent une exécution unique ou au premier démarrage de SQL Server Agent.

Vous devrez peut-être créer plusieurs planifications pour un travail lorsque la périodicité requise pour le travail est complexe et ne peut pas être prise en charge dans une seule planification de travail.

Chaque planification peut également être partagée entre plusieurs travaux, mais il est important de ne pas avoir trop de travaux qui commencent en même temps.

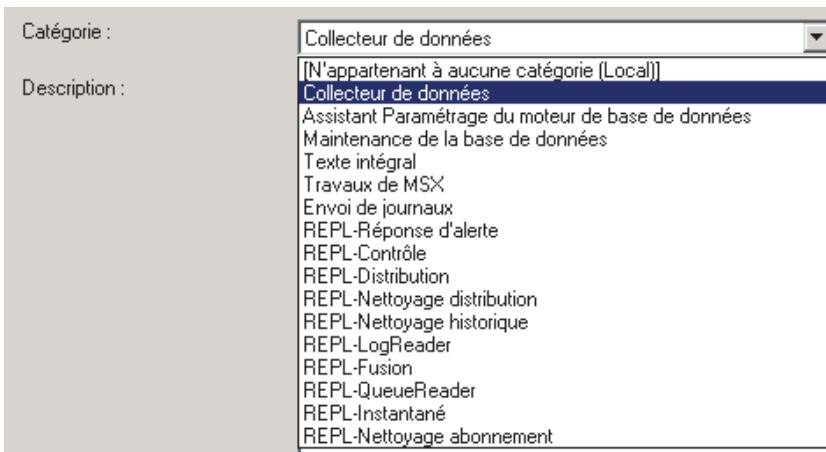
Création de travaux

Vous pouvez utiliser SQL Server Management Studio pour créer des travaux, ou vous pouvez exécuter la procédure stockée système sp_add_job. La création d'un travail requiert l'exécution de plusieurs procédures stockées système supplémentaires, pour ajouter des étapes et des planifications au travail.

La définition du travail est stockée dans la base de données msdb, ainsi que toutes les configurations de SQL Server Agent.

Catégories de travaux

Les travaux peuvent être placés dans des catégories. SQL Server dispose de plusieurs catégories prédéfinies, comme illustré ci-dessous, mais vous pouvez ajouter vos propres catégories.



Les catégories de travaux peuvent être utiles lorsque vous devez exécuter des actions associées aux travaux dans une catégorie spécifique. Par exemple, il est possible de créer une catégorie de travail appelée « Vérification des stratégies SQL Server 2005 » et d'écrire un script PowerShell pour exécuter tous les travaux de cette catégorie sur vos serveurs SQL Server 2005.



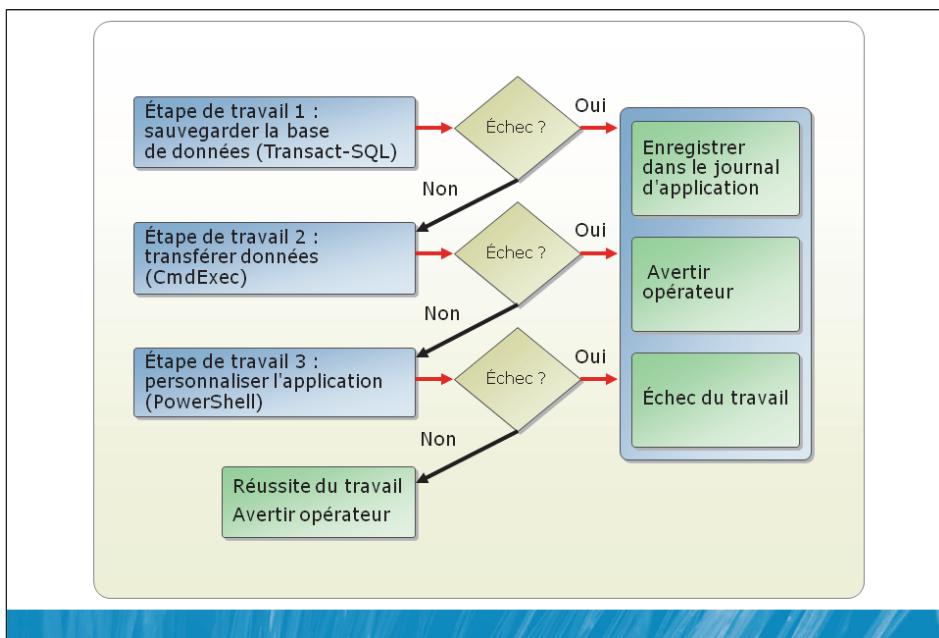
Remarque Une discussion détaillée sur l'utilisation de PowerShell est une rubrique avancée qui n'est pas traitée dans ce cours.



Remarque Il est également important de tenir compte du compte de sécurité dont chaque type d'étape de travail a besoin pour s'exécuter. Le module 14 aborde la sécurité de SQL Server Agent.

Question : pouvez-vous imaginer un exemple d'utilisation des catégories de travaux ?

Création d'étapes de travail



Points clés

Utilisez SQL Server Management Studio ou exécutez la procédure stockée système sp_add_jobstep pour définir chaque étape de travail requise pour automatiser une tâche. Un seul type d'exécution peut être défini pour chaque étape de travail, mais chacune peut avoir un autre type.

Flux de travail des étapes de travail

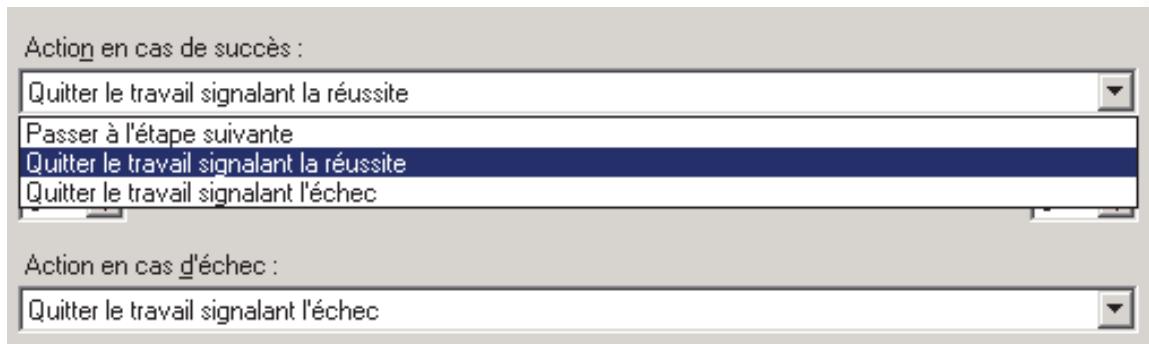
Chaque étape a des résultats qui définissent si l'étape a réussi ou échoué. Notez la liste des étapes de travail et les options « En cas de succès » et « En cas d'échec » pour chaque étape du travail suivant :

Liste des étapes du travail :

Et...	Nom	Type	En cas de succès	En cas d'échec
1	Collection_set_2_collection_collect	Système d'exploit...	Quitter le travail si...	Quitter le travail signalant l'é...
2	collection_set_2_collection_autostop	Script Transact-S...	Passer à l'étape s...	Quitter le travail signalant l'é...

Par défaut, SQL Server passe à l'étape de travail suivante en cas de réussite et s'arrête en cas d'échec d'une étape de travail. Mais il est possible de passer à n'importe quelle étape définie dans le travail en cas de réussite ou d'échec, pour définir un flux de travail particulier.

Dans les propriétés avancées de chaque étape de travail, une action peut être configurée en cas de réussite et d'échec, comme indiqué ci-dessous :



En configurant l'action pour qu'elle se produise en cas de réussite ou d'échec de chaque étape de travail, il est possible de créer un flux de travail qui détermine le flux logique global du travail. Notez qu'outre le fait que chaque étape du travail a des résultats définis, le travail global indique des résultats. Cela signifie que même si certaines étapes de travail réussissent, le travail global peut quand même signaler un échec.

Reprise d'étapes de travail

Vous pouvez spécifier combien de fois SQL Server doit tenter de reprendre l'exécution d'une étape de travail en cas d'échec de cette dernière. Vous pouvez également spécifier les intervalles entre chaque tentative (en minutes). Par exemple, si l'étape de travail requiert une connexion à un serveur distant, vous pouvez définir plusieurs tentatives de reprise au cas où la connexion ne peut pas être établie.

Question : quelles opérations ne doivent pas être regroupées dans un travail ?

Programmation des travaux en vue de leur exécution

The screenshot shows the 'Travail : Sauvegarde des journaux des transactions' (Job: Transaction Log Backup) properties. On the left, the 'Fréquence' (Frequency) section is visible, showing a weekly schedule from Monday to Sunday at 12:00:00. A callout box highlights two key points:

- Plusieurs planifications peuvent être définies par travail
- Les planifications peuvent être réutilisées dans d'autres travaux

On the right, the 'Planification : Lun-Dim Équipe 1' (Scheduling: Mon-Sun Team 1) and 'Planification : Lun-Dim Équipe 2' (Scheduling: Mon-Sun Team 2) sections are shown. Each section displays a grid representing time slots, with specific times highlighted for each team's schedule.

Points clés

Les planifications servent à démarrer des travaux à des moments demandés. Une ou plusieurs planifications peuvent être associées à un travail. Des noms sont affectés aux planifications qui peuvent être partagées par plusieurs travaux. Outre les planifications récurrentes standard, plusieurs types de périodicité particuliers sont également définis :

- exécution unique ;
- démarrage automatique au lancement de SQL Server Agent ;
- démarrage quand le processeur est inactif.

Même si un travail peut avoir plusieurs planifications, SQL Server limite le travail à une seule exécution simultanée. Lorsque vous exécutez un travail manuellement alors qu'il fonctionne comme prévu, SQL Server Agent refuse la demande. Si un travail est toujours en cours d'exécution lorsqu'il est planifié pour s'exécuter à nouveau, SQL Server Agent refuse de l'exécuter à nouveau.

Question : qu'est-ce qui peut être modifié si la base de données de l'exemple ci-dessus n'a pas besoin de sauvegarde horaire pendant le week-end ?

Génération d'un script des travaux

- Les travaux peuvent être écrits à des fins de documentation, d'archivage et de déploiement sur d'autres instances

```
USE AdminDB;
GO

ALTER PROCEDURE dbo.GatherLogSpace
AS
    INSERT INTO dbo.LogSpace (DatabaseName, LogSizeMB,
        EXEC ('DBCC SQLPERF(LOG_SPACE)');
    GO
```

-- Step 2: Attempt to execute the job manually.
Note that, as expected, the job did not
(In Object Explorer, right-click Gather
Log Statistics, and click Start Job at
the job fails, in the Start Jobs - Pros

Nouveau travail... Démarrer le travail à l'étape... Arrêter le travail Générer un script du travail en tant que CREATE To Nouvelle fenêtre d'éditeur de requête

Afficher Historique Activer Désactiver Démarrer PowerShell Rapports Renommer Supprimer Actualiser Propriétés

ALTER To DROP To DROP et CREATE To SELECT To INSERT To UPDATE To DELETE To EXECUTE To

File... Presse-papiers Travail de l'agent... Gather story)

Points clés

En général, les travaux sont d'abord créés dans SSMS, car plusieurs procédures stockées système sont nécessaires pour définir un travail unique. Toutefois, il est important que les travaux existants soient écrits dans T-SQL pour les raisons suivantes :

- Les scripts de travaux peuvent être utilisés dans la documentation et peuvent être archivés dans les systèmes de contrôle de code source.
- Les travaux peuvent facilement être recréés après un échec si nécessaire, si les scripts des travaux ont été créés.
- Il est également nécessaire de pouvoir créer un travail dans un environnement (tel qu'un environnement de test) et le déployer dans un autre environnement (tel qu'un environnement de production). La possibilité de générer un script des travaux facilite leur déploiement.
- Les scripts de travaux peuvent être utilisés lors de l'exécution de mises à niveau côté à côté de systèmes SQL Server.



Remarque Plusieurs travaux peuvent être sélectionnés dans les détails de l'Explorateur d'objets lors de la génération d'un script.

D'autres options plus avancées sont disponibles pour générer un script des travaux, par exemple l'utilisation de SQL Server Management Objects (SMO). SMO peut être utilisé conjointement avec la programmation .NET, dans les langages tels que Microsoft Visual Basic® ou C#, et peut être utilisé conjointement avec PowerShell.

Question : dans quels scénarios est-il utile de générer un script de plusieurs travaux à la fois ?

Démonstration 2A : Génération d'un script des travaux

Dans cette démonstration, vous allez apprendre à

- Créer un travail dans SSMS
- Générer un script d'un travail
- Déployer un travail sur une deuxième instance de SQL Server

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_13_PRJ\22462A_13_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

Gestion des travaux de SQL Server Agent

- Affichage de l'historique des travaux
- Interrogation des tables et des vues système spécifiques liées à SQL Server AgentRésolution des problèmes liés aux échecs des travaux
- Démonstration 3A : Affichage de l'historique des travaux et résolution des défaillances

L'automatisation des tâches administratives courantes et des autres tâches est importante, mais il est tout aussi important de s'assurer que ces tâches continuent à s'exécuter comme prévu. SQL Server fournit des informations concernant les exécutions précédentes et défaillantes en conservant un historique dans les tables contenues dans la base de données msdb. Cette leçon vous montre comment interroger les tables d'historique et vous fournit une approche pour résoudre les problèmes liés aux travaux qui échouent ou qui ne s'exécutent pas comme prévu.

Objectifs

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

- afficher l'historique des travaux ;
- interroger des tables et des vues système spécifiques liées à SQL Server Agent ;
- résoudre les problèmes liés aux échecs des travaux.

Affichage de l'historique des travaux

Date	ID de l'étape	Nom du travail	Nom du tâche
11/02/2013 13:27:51	1	MIA-SQL1MKTG	Gather Transaction Log Statistics
11/02/2013 13:27:44	2	MIA-SQL1MKTG	Gather Transaction Log Statistics
11/02/2013 13:27:38	3	MIA-SQL1MKTG	Gather Transaction Log Statistics

Détails de la ligne sélectionnée :

Date : 11/02/2013 13:27:51
ID de l'étape : 1
Nom du travail : MIA-SQL1MKTG
Nom du tâche : Gather Transaction Log Statistics
Durée : 00:00:00
ID de message SQL : 0
ID de message d'erreur : 0
Opérateur associé à cette exécution :
Opérateur ayant envoyé l'e-mail :
Opérateur ayant reçu l'e-mail :
Opérateurs en cours d'exécution : 0
Messages : Le travail a réussi. Le travail a été appellé par l'utilisateur ADVENTUREWORKS\Administrateur. La dernière étape exécutée est l'étape 1 (Gather Transaction Log Statistics).

- Agent SQL Server conserve les informations d'historique dans msdb
- L'historique des travaux peut être interrogé directement ou affiché via SSMS
- Le Moniteur d'activité des travaux affiche les informations en cours
- La rétention de l'historique peut être configurée selon le temps ou la taille

Points clés

SQL Server Agent conserve une trace des résultats des travaux dans les tables système de la base de données msdb. SQL Server Agent enregistre non seulement les résultats des travaux complets, mais aussi les résultats de chaque étape de travail.

Vous pouvez choisir d'écrire les résultats des travaux dans le journal des applications Windows et dans le journal SQL Server en définissant les propriétés de notification pour chaque travail. Les tables d'historique de la base de données msdb sont toujours écrites, quelle que soit la configuration de la sortie du journal.

Affichage de l'historique des travaux

Chaque travail et chaque étape de travail a des résultats. Si un travail échoue, les résultats manquants de l'étape doivent être examinés afin de connaître la raison de l'échec. L'historique de chaque travail peut être affiché dans SSMS, mais peut également être récupéré par programme en interrogeant directement les tables système. L'écriture de requêtes pour récupérer l'historique des travaux est illustrée dans la rubrique suivante de cette leçon.

Les 1000 entrées les plus récentes de l'historique des travaux sont conservées par défaut, mais la période de conservation des entrées de l'historique des travaux peut être configurée, selon l'âge ou la taille totale des données de l'historique, dans la fenêtre des propriétés du service SQL Server Agent.

L'Explorateur d'objets dans SSMS fournit également un Moniteur d'activité des travaux. Le Moniteur d'activité des travaux offre une vue des travaux en cours d'exécution et des données détaillant les résultats de la précédente exécution, ainsi que l'heure planifiée de la prochaine exécution du travail. Un exemple des données fournies par le Moniteur d'activité des travaux est illustré ci-dessous :

Activité du travail de l'Agent :						
Nom	Activé	État	Résulta...	Dernière exé...	Prochaine ...	Catégorie
collection_set_1_noncached_co...	oui	Inactif	Opérati...	18/02/201...	non planifié	[N'apparten...
collection_set_2_collection	oui	Exécuti...	Opérati...	18/02/201...	18/02/20...	[N'apparten...
collection_set_2_upload	oui	Inactif	Opérati...	18/02/201...	non planifié	[N'apparten...
collection_set_3_collection	oui	Exécuti...	Opérati...	18/02/201...	non planifié	[N'apparten...
collection_set_3_upload	oui	Inactif	Opérati...	18/02/201...	non planifié	[N'apparten...
mdw_purge_data_[MDW]	oui	Inactif	Opérati...	18/02/201...	non planifié	[N'apparten...
syspolicy_purge_history	oui	Inactif	Annulé	18/02/201...	19/02/20...	[N'apparten...
sysutility_get_cache_table_d...	oui	Inactif	Opérati...	18/02/201...	non planifié	[N'apparten...
sysutility_get_cache_tables_d...	oui	Inactif	Opérati...	18/02/201...	non planifié	[N'apparten...
sysutility_get_views_data_int...	oui	Inactif	Opérati...	18/02/201...	non planifié	[N'apparten...

Question : comment une base de données msdb endommagée affecte-t-elle SQL Server Agent ?

Interrogation des tables et des vues système spécifiques liées à SQL Server Agent

- Agent SQL Server conserve la configuration et l'historique dans msdb
 - Les tables système de l'agent se trouvent dans le schéma dbo
 - Les tables sont décrites dans la documentation en ligne
- Utilisez les tables d'historique pour automatiser la collection d'historiques des travaux sur plusieurs systèmes

```
SELECT j.name, jh.run_date, jh.run_time, jh.message
FROM msdb.dbo.sysjobhistory AS jh
INNER JOIN msdb.dbo.sysjobs AS j
ON jh.job_id = j.job_id
WHERE jh.step_id = 0;
GO
```

Points clés

Comme indiqué précédemment dans ce module, les informations relatives à la configuration de SQL Server Agent et de ses objets tels que les travaux, alertes, planifications et opérateurs sont écrites dans les tables système de la base de données msdb. Ces objets sont contenus dans le schéma dbo et peuvent être interrogés directement de là.

L'historique des travaux est écrit dans la table dbo.sysjobhistory et la liste des travaux est écrite dans la table dbo.sysjobs.

Exemple de requête

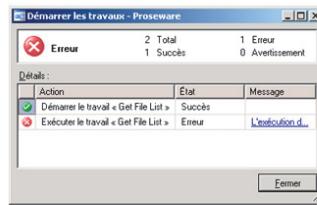
Dans l'exemple illustré sur la diapositive, la date et l'heure de la dernière exécution d'un travail et les résultats sont interrogés à partir de la table dbo.sysjobhistory de la base de données msdb. La requête établit une jointure avec la table dbo.sysjobs pour récupérer le nom du travail.

Notez que la clause WHERE spécifie un step_id de valeur zéro. Les étapes de travail commencent à un, pas zéro, mais une entrée est ajoutée à la table dbo.sysjobhistory, avec un step_id de valeur zéro pour stocker les résultats globaux du travail. Les résultats de chaque étape de travail peuvent être obtenus en interrogeant les valeurs step_id supérieures à zéro.

Question : pourquoi est-il important d'interroger les tables d'historique des travaux ?

Résolution des problèmes liés aux échecs des travaux

- Si Agent SQL Server ne s'exécute pas
 - Vérifiez les paramètres du service
 - Vérifiez la base de données msdb
- Examinez l'historique des travaux
 - Vérifiez les résultats du travail pour identifier la dernière étape exécutée
 - Vérifiez les résultats de l'étape du travail pour déterminer pourquoi elle a échoué
- Si le travail n'a pas commencé, vérifiez que
 - Le travail est activé
 - Le travail est planifié
 - La planification est activée
- Vérifiez que tous les objets dépendants tels que les bases de données, les fichiers et les procédures sont disponibles
 - Vérifiez que les paramètres de sécurité autorisent l'accès aux objets dépendants



Points clés

Les travaux ne s'exécutent pas toujours comme prévu et ils peuvent parfois ne pas s'exécuter du tout. Il est important de suivre un processus cohérent lorsque vous tentez de comprendre pourquoi un travail échoue.

Il existe quatre étapes de base qui doivent être suivies pour résoudre les problèmes liés aux travaux : vérification de l'état de SQL Server Agent, analyse de l'historique des travaux, vérification de l'exécution des travaux et vérification de l'accès aux dépendances.

Vérification de l'état de SQL Server Agent

Si SQL Server Agent n'est pas en cours d'exécution, aucun travail n'est exécuté. Vérifiez que le service est configuré pour démarrer automatiquement et tentez de le démarrer manuellement. Si le service ne démarre toujours pas, vérifiez les éléments suivants :

- Vérifiez que le compte du service est valide, que le mot de passe du compte n'a pas changé et que le compte n'est pas verrouillé. Si l'un de ces éléments est incorrect, le service ne démarre pas, mais les détails relatifs au problème sont écrits dans le journal des événements système de l'ordinateur.
- Vérifiez que la base de données msdb est en ligne. Si la base de données msdb est endommagée, suspecte ou en mode hors connexion, SQL Server Agent ne démarre pas.

Examen de l'historique des travaux

Examinez les résultats des travaux pour identifier la dernière exécution. Si le travail a échoué en raison de l'échec d'une étape de travail (ce qui est la situation la plus courante), l'erreur de l'étape de travail ne peut pas être affichée à ce niveau. Il est nécessaire d'examiner les résultats individuels de l'étape du travail qui a échoué.

Vérification de l'exécution des travaux

Si SQL Server Agent est en cours d'exécution mais un travail individuel ne s'exécute pas, vérifiez les éléments suivants :

- Vérifiez que le travail est activé. Les travaux désactivés ne s'exécutent pas.
- Vérifiez que le travail est planifié. Il se peut que la planification soit incorrecte ou que l'heure de la prochaine exécution planifiée ne soit pas encore arrivée.
- Vérifiez que la planification est activée. Les travaux et les planifications peuvent être désactivés. Un travail ne s'exécute pas sur une planification désactivée.

Vérification de l'accès aux dépendances

Vérifiez que tous les objets dépendants tels que les bases de données, les fichiers et les procédures sont disponibles. La démonstration 2A décrit une situation où le travail sur la deuxième instance de SQL Server n'a pas été exécuté, car les objets requis pour exécuter le travail n'étaient pas présents.

Les travaux s'exécutent souvent dans un contexte de sécurité différent pour l'utilisateur qui les crée. Des paramètres de sécurité incorrects sont l'un des problèmes courants qui provoquent l'échec de l'exécution des travaux. Le contexte de sécurité des étapes de travail est abordé en détail dans le module 14.

Question : lors de la migration d'un travail de l'environnement de test à l'environnement de production, quelle autre étape est requise, outre le déplacement du travail proprement dit ?

Démonstration 3A : Affichage de l'historique des travaux et résolution des défaillances

Dans cette démonstration, vous allez apprendre à

- Un travail qui n'a pas été exécuté correctement
- Comment examiner l'historique des travaux à l'aide de SSMS
- Comment examiner l'historique des travaux à l'aide de T-SQL

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_13_PRJ\22462A_13_PRJ.ssmsln**, puis cliquez sur **Ouvrir**.
 - Ouvrez et exécutez le fichier de script **00 – Setup.sql** dans l'Explorateur de solutions.
 - Dans le menu **Affichage**, cliquez sur **Explorateur de solutions**. Ouvrez le fichier de script **21 – Demonstration 2A.sql** et suivez les différentes étapes.
2. Ouvrez le fichier de script **31 – Demonstration 3A.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

Atelier pratique 13 : Automatisation de la gestion de SQL Server

- Exercice 1 : Créer un travail d'extraction de données
- Exercice 2 : Planifier le travail d'extraction de données
- Exercice difficile 3 : Résoudre les problèmes liés à un travail défaillant (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_13_PRJ\22462A_13_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

De nombreuses tâches courantes doivent être exécutées sur l'instance Proseware. Précédemment, ces tâches ont été exécutées manuellement et le manque de cohérence dans leur exécution a provoqué des problèmes pour l'organisation. Sur la nouvelle instance, vous devez automatiser ces tâches à l'aide de SQL Server Agent.

Il existe également un rapport sur un travail existant de SQL Server Agent qui ne s'exécute pas comme prévu. Si le temps le permet, vous devez résoudre les problèmes liés au travail.

Exercice 1 : Créer un travail d'extraction de données

Scénario

Dans le module 8, vous avez créé un package d'extraction de données SSIS. Le processus d'extraction identifie les prospects qui n'ont pas été contactés récemment. La sortie de ce processus d'extraction est utilisée pour planifier les activités marketing pendant la semaine. Vous devez créer un travail pour exécuter le package SSIS.

Les tâches principales de cet exercice sont les suivantes :

1. Créer le travail requis.
2. Vérifier que le travail s'exécute sans erreur.

► Tâche 1 : Créer le travail requis

- Créez le travail requis.appelez le travail « Extraire les prospects non contactés ». Le travail doit exécuter le package SSIS « Extraction hebdomadaire des prospects à contacter » qui se trouve sur l'instance du serveur Proseware.

► Tâche 2 : Vérifier que le travail s'exécute sans erreur

- À l'aide de l'Explorateur d'objets, démarrez le travail et vérifiez qu'il s'exécute correctement.

Résultats : À la fin de cet exercice, vous devez avoir créé un travail d'extraction de données.

Exercice 2 : Planifier le travail d'extraction de données

Scénario

Vous avez créé un travail pour extraire les prospects à contacter. Les informations fournies par ce travail sont utilisées deux fois pendant la semaine. Une réunion a lieu à 9h chaque lundi matin pour planifier les activités marketing de la semaine. Une deuxième réunion de planification a lieu à 19h chaque mardi soir. Vous devez vérifier qu'une liste mise à jour est disponible avant chaque réunion. Vous devez planifier le travail d'extraction pour qu'il s'exécute chaque lundi à 8 h 30 et chaque mardi à 18 h 30.

La tâche principale de cet exercice est la suivante :

1. Planifier le travail d'extraction de données.

► **Tâche 1 : Planifier le travail d'extraction de données**

- Créez une nouvelle planification du travail chaque lundi à 8h30.
- Créez une nouvelle planification du travail chaque mardi à 18 h 30.
- Affectez la planification au travail d'extraction de données.

Résultats : À la fin de cet exercice, vous devez avoir appliqué plusieurs planifications au travail d'extraction de données.

Exercice difficile 3 : Résoudre les problèmes liés à un travail défaillant (si le temps le permet)

Scénario

Sur le serveur Proseware, un nouveau travail de SQL Server Agent appelé Extraire les longues charges de page a été créé récemment. Le travail récupère les détails des pages Web qui ont mis du temps à se charger à partir de la table Marketing.WebLog pour les faire analyser par l'équipe de développement Web. Le travail est prévu pour s'exécuter chaque lundi à 6h. Le travail a été implémenté la semaine dernière, mais aucune donnée n'a été récupérée cette semaine. Vous devez examiner et corriger les problèmes liés à ce travail.

La tâche principale de cet exercice est la suivante :

1. Résoudre les problèmes liés au travail défaillant.

► Tâche 1 : Résoudre les problèmes liés au travail défaillant

- Examinez l'historique des travaux pour le travail qui a échoué et identifiez la cause de l'échec.
- Corrigez le problème qui empêche l'exécution du travail.
- Vérifiez que le travail s'exécute.
- Vérifiez que le travail est correctement planifié.

Résultats : À la fin de cet exercice, vous devez avoir résolu les problèmes de défaillance.

Contrôle des acquis et éléments à retenir

- Questions de contrôle des acquis
- Meilleures pratiques

Questions de contrôle des acquis

1. Quelles fonctions actuellement exécutées manuellement pourraient être intégrées dans un travail ?
2. Combien de temps l'historique des travaux est-il conservé dans la base de données msdb ?

Meilleures pratiques

1. Utilisez les travaux de SQL Server Agent pour planifier des travaux courants.
2. Créez des catégories personnalisées pour regrouper vos travaux.
3. Générez un script de vos travaux pour le déploiement distant.
4. Utilisez l'historique des travaux pour analyser les résultats des travaux et des étapes de travail.
5. Utilisez le Moniteur d'activité des travaux pour surveiller en temps réel les travaux.

Module 14

Configuration de la sécurité pour SQL Server Agent

Sommaire :

Leçon 1 : Comprendre la sécurité de SQL Server Agent	14-3
Leçon 2 : Configuration des informations d'identification	14-13
Leçon 3 : Configuration de comptes proxy	14-18
Atelier pratique 14 : Configuration de la sécurité pour SQL Server Agent	14-24

Vue d'ensemble du module

- Comprendre la sécurité de SQL Server Agent
- Configuration des informations d'identification
- Configuration de comptes proxy

Dans les modules précédents vous avez découvert l'utilité de réduire les autorisations accordées aux utilisateurs, de sorte que ceux-ci ne disposent que des autorisations dont ils ont besoin pour effectuer leurs tâches. La même logique s'applique à l'accord des autorisations à SQL Server Agent. Bien qu'il soit facile d'exécuter tous les travaux dans le contexte du compte de service SQL Server Agent et de configurer ce compte comme compte d'administrateur, il pourrait en résulter un environnement de sécurité médiocre. Il est important de comprendre comment créer un environnement de sécurité à priviléges minimaux pour les travaux qui s'exécutent dans SQL Server Agent.

Objectifs

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

- expliquer la sécurité de SQL Server Agent ;
- configurer les informations d'identification ;
- configurer des comptes proxy.

Leçon 1

Comprendre la sécurité de SQL Server Agent

- Vue d'ensemble de la sécurité de SQL Server Agent
- Rôles de SQL Server Agent
- Discussion : Dépendances du travail SQL Server Agent
- Attribution de contextes de sécurité à des étapes de travail de l'Agent
- Résolution des problèmes de sécurité de SQL Server Agent
- Démonstration 1A : Affecter un contexte de sécurité aux étapes de travail

Il est possible de faire appel à SQL Server Agent pour effectuer une grande variété de tâches. De nombreuses tâches effectuées par SQL Server Agent sont de nature administrative, mais bien d'autres sont effectuées pour le compte des utilisateurs. La nécessité de pouvoir effectuer une grande variété de tâches implique la nécessité d'une configuration de sécurité souple. Les travaux doivent pouvoir accéder à de nombreux types d'objets. Les travaux ont souvent besoin d'accéder à des ressources externes telles que les fichiers et les dossiers du système d'exploitation aussi bien qu'aux objets qui résident dans SQL Server. Ces dépendances au système d'exploitation (et autres) requièrent également un modèle de sécurité configurable et à plusieurs couches, afin d'éviter de devoir accorder trop d'autorisations au compte de service SQL Server. **Objectifs**

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

- mettre en œuvre la sécurité de SQL Server Agent ;
- décrire les rôles de SQL Server Agent ;
- affecter des contextes de sécurité aux étapes de travail SQL Server Agent ;
- résoudre les problèmes de sécurité de SQL Server Agent.

Vue d'ensemble de la sécurité de SQL Server Agent

Il est important de s'assurer que chaque étape de travail d'Agent SQL Server s'exécute dans un contexte de sécurité approprié

- Les autorisations réseau sont déterminées par le compte de service
 - Comptes intégrés tels que local et service réseau
 - Compte de domaine Windows
- Le compte utilisé pour exécuter les travaux doit se connecter à
 - Une instance de SQL Server pour les étapes de travail T-SQL
 - Des ressources Windows et réseau pour d'autres types de tâche
 - Des comptes proxy peuvent être utilisés

Points clés

Comme tous les services, le service SQL Server Agent a une identité dans le système d'exploitation Microsoft® Windows®. Le compte de démarrage du service définit le compte Windows sous lequel s'exécute SQL Server Agent. Le compte utilisé définit les autorisations dont dispose le service SQL Server Agent lors de l'accès aux ressources réseau.

Compte de service de l'agent

Pour le service SQL Server Agent, vous pouvez utiliser le compte de service local ou le compte de service réseau intégré. Il est préférable d'utiliser un autre compte de domaine Windows.



Remarque L'option de compte système local n'est fournie que pour des raisons de compatibilité descendante et le compte de service réseau n'est pas non plus recommandé, pour des raisons de sécurité. Le service réseau dispose de plus de fonctionnalités que n'en exige le compte de service. Il convient, à la place, de créer et d'utiliser un compte doté uniquement des autorisations nécessaires.

Il est préférable d'utiliser un compte de domaine Windows et de le configurer avec le moins de priviléges possibles qui permettent toujours le fonctionnement. Pendant l'installation de SQL Server, un groupe local est créé avec un nom dont le format est le suivant :

```
SQLServerSQLAgentUser$<ComputerName>$<InstanceName>
```

Ce groupe se voit accorder tous les privilèges d'accès requis par le compte SQL Server Agent. Notez que cela ne comprend que le strict minimum d'autorisations dont le compte a besoin pour que SQL Server Agent s'exécute. Lorsque SQL Server Agent doit accéder à d'autres ressources dans les étapes du travail, des autorisations supplémentaires sont nécessaires.

Lorsque le gestionnaire de configuration SQL Server est utilisé pour attribuer un compte au service (ce qui constitue la méthode prise en charge privilégiée pour modifier les comptes de service), il place le compte dans le groupe approprié. Il n'y a besoin d'accorder aucune autre autorisation particulière.

 **Remarque** Le compte SQL Server Agent ne peut pas utiliser l'authentification SQL Server pour sa connexion au moteur de base de données SQL Server.

Les travaux de SQL Server Agent sont exécutées dans le contexte du compte de service par défaut. L'alternative consiste à créer des comptes proxy qui serviront à l'exécution des travaux. Les comptes proxy seront décrits plus loin dans ce module.

Question : pour quelle raison un compte de service SQL Server Agent aurait-il besoin des privilèges sysadmin sur l'instance SQL Server ?

Rôles de SQL Server Agent

- Les membres du rôle fixe sysadmin peuvent administrer Agent SQL Server
- Les rôles de base de données fixes dans msdb contrôlent l'accès des autres utilisateurs

Rôle	Description
SQLAgentUserRole	Contrôle des autorisations des travaux et planifications dont ils sont propriétaires
SQLAgentReaderRole	Toutes les autorisations du rôle SQLAgentUserRole plus l'autorisation d'afficher la liste de tous les travaux et planifications de travail disponibles
SQLAgentOperatorRole	Autorisation de gérer les travaux locaux, les propriétés de vue des opérateurs et des proxies, et de dénombrer les proxies disponibles et les alertes

Points clés

Par défaut, seuls les membres du rôle serveur fixe sysadmin peuvent administrer SQL Server Agent. Comme le système des travaux de SQL Server Agent peut effectuer des tâches qui requièrent un accès non seulement à SQL Server, mais éventuellement à d'autres ressources Windows et réseau, il est important de contrôler qui est en mesure de l'administrer. Il n'est pas considéré comme recommandé d'établir une connexion à partir d'un membre du rôle serveur fixe sysadmin, si la seule raison de son appartenance à ce rôle consiste à administrer SQL Server Agent.

Rôles de base de données fixes de SQL Server Agent

Les rôles de base de données fixes dans la base de données msdb servent à contrôler l'accès des utilisateurs non sysadmin. SQL Server 2012 comporte des rôles de base de données fixes pour utiliser SQL Server Agent. Les rôles disponibles, répertoriés dans l'ordre de possibilités croissantes, sont les suivants :

- SQLAgentUserRole
- SQLAgentReaderRole
- SQLAgentOperatorRole

Lorsque des utilisateurs qui ne sont pas membres de l'un de ces rôles sont connectés à SQL Server® dans SSMS Server Management Studio, le nœud SQL Server Agent n'est pas visible dans l'Explorateur d'objets. Pour pouvoir utiliser SQL Server Agent, un utilisateur doit être membre d'au moins l'un ces rôles de base de données fixes ou du rôle de serveur fixe sysadmin.

SQLAgentUserRole

Le rôle SQLAgentUserRole est le plus restreint des rôles de base de données fixes de SQL Server Agent. Les membres du rôle SQLAgentUserRole disposent uniquement d'autorisations sur les travaux et les planifications de travaux locaux dont ils sont propriétaires. Ils ne peuvent pas exécuter de travaux multiserveurs (travaux du serveur maître et des serveurs cibles) et ne peuvent pas non plus modifier l'appartenance des travaux pour accéder aux travaux dont ils ne sont pas propriétaires. Les membres du rôle SQLAgentUserRole peuvent également consulter la liste des proxies disponibles dans la boîte de dialogue Propriétés de l'étape du travail de SSMS Server Management Studio.

SQLAgentReaderRole

Le rôle SQLAgentReaderRole inclut toutes les autorisations du rôle SQLAgentUserRole et il est de plus autorisé à consulter la liste des travaux multiserveurs disponibles, leurs propriétés et leur historique. Les membres de ce rôle peuvent également afficher la liste de tous les travaux et planifications de travaux disponibles ainsi que leurs propriétés, et pas seulement la liste des travaux et des planifications de travaux dont ils sont propriétaires. Les membres du rôle SQLAgentReaderRole ne peuvent pas modifier l'appartenance des travaux pour accéder aux travaux dont ils ne sont pas propriétaires.

SQLAgentOperatorRole

Le rôle SQLAgentOperatorRole est le moins restreint des rôles de base de données fixes de SQL Server Agent. Il inclut toutes les autorisations des rôles SQLAgentUserRole et SQLAgentReaderRole. Les membres de ce rôle peuvent également afficher les propriétés des opérateurs et des proxies, et énumérer les proxies et les alertes disponibles sur le serveur.

Les membres du rôle SQLAgentOperatorRole disposent d'autorisations supplémentaires sur les travaux et planifications locaux. Ils peuvent exécuter, arrêter ou démarrer tous les travaux locaux, et ils peuvent supprimer l'historique de n'importe quel travail local sur le serveur. Ils peuvent également activer ou désactiver tous les travaux et planifications locaux sur le serveur. Pour activer ou désactiver des travaux locaux ou des planifications locales, les membres de ce rôle doivent utiliser la procédure stockée sp_update_job ou sp_update_schedule, en spécifiant le paramètre nom du travail ou identificateur de la planification ainsi que le paramètre activé. S'ils spécifient d'autres paramètres, l'exécution de ces procédures stockées échoue. Les membres du rôle SQLAgentOperatorRole ne peuvent pas modifier l'appartenance des travaux pour accéder aux travaux dont ils ne sont pas propriétaires.

Question : pourquoi devez-vous faire attention lorsque vous attribuez l'accès aux membres n'ayant pas de rôle serveur fixe sysadmin ?

Discussion : Dépendances du travail SQL Server Agent

- De quelles ressources SQL Server les travaux d'Agent SQL Server peuvent-ils dépendre ?
- De quelles ressources en dehors de SQL Server les travaux d'Agent SQL Server peuvent-ils dépendre ?
- Quelles sont les identités requises pour accéder aux ressources externes ?



Rubriques de discussion

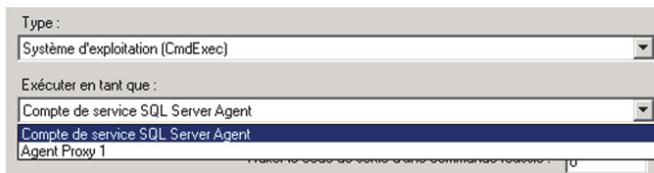
Question : de quelles ressources SQL Server les travaux de SQL Server Agent peuvent-ils dépendre ?

Question : de quelles ressources en dehors de SQL Server les travaux de SQL Server Agent peuvent-ils dépendre ?

Question : quelles sont les identités requises pour accéder aux ressources externes ?

Attribution de contextes de sécurité à des étapes de travail de l'Agent

- Étapes de travail T-SQL
 - Agent SQL Server emprunte l'identité du propriétaire du travail
 - Si le propriétaire est membre du rôle serveur fixe sysadmin, l'étape s'exécute sous le compte de service Agent SQL Server
 - Les membres du rôle serveur fixe sysadmin peuvent également spécifier un autre utilisateur
- Autres types d'étapes du travail
 - Les membres du rôle fixe sysadmin peuvent utiliser le compte Agent SQL Server (par défaut)
 - Les comptes proxy sont utilisés pour définir les informations d'identification à utiliser



Points clés

Chaque étape de travail peut être affectée d'un contexte de sécurité. Les étapes de travail qui exécutent du code T-SQL doivent être considérées indépendamment d'autres types d'étapes de travail.

Étapes de travail T-SQL

Les étapes de travail T-SQL n'utilisent pas les proxies de SQL Server Agent. Lorsqu'une étape de travail T-SQL est exécutée, SQL Server Agent emprunte l'identité du propriétaire du travail, sauf dans une situation où le propriétaire de l'étape de travail est membre du rôle serveur fixe sysadmin. Dans ce cas, l'étape du travail s'exécute dans le contexte de sécurité du service SQL Server Agent, sauf si le sysadmin choisit de faire emprunter à l'étape du travail l'identité d'un autre utilisateur. Les membres du rôle serveur fixe sysadmin peuvent spécifier que l'étape de travail doit emprunter l'identité d'un autre utilisateur.

Autres types d'étapes du travail

Pour les types d'étapes de travail qui ne sont pas fondés sur T-SQL, un modèle de sécurité différent est appliqué. Pour les membres du rôle serveur fixe sysadmin, par défaut, les autres types d'étapes de travail utilisent toujours le compte de service SQL Server Agent pour s'exécuter. Étant donné que différents types d'étapes de travail peuvent être exécutés dans SQL Server Agent, il est déconseillé de les exécuter sous ce compte. Pour fournir un contrôle plus strict, un système de proxy a été intégré.

Comptes proxy

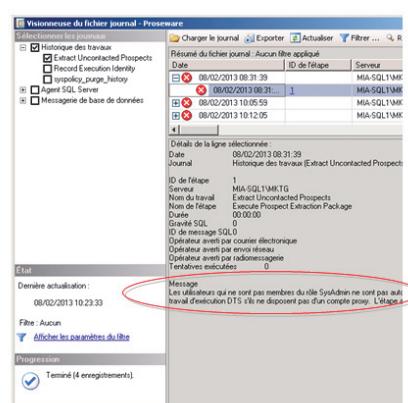
Un compte proxy sert à associer une étape de travail avec une identité Windows, via un objet appelé Informations d'identification. Les comptes proxy peuvent être créés pour tous les sous-systèmes disponibles, à l'exception des étapes de travail T-SQL. L'utilisation de comptes proxy implique que différentes identités Windows peuvent être utilisées pour réaliser les différentes tâches requises par les travaux, et permet d'assurer une sécurité plus étroite en évitant la nécessité qu'un compte unique dispose de toutes les autorisations nécessaires pour exécuter tous les travaux.

Les informations d'identification sont présentées dans la leçon 2 de ce module et les comptes proxy sont présentés dans la leçon 3.

Question : pourquoi un compte proxy doit-il être utilisé, même lorsque le propriétaire de l'étape est membre du rôle serveur fixe sysadmin ?

Résolution des problèmes de sécurité de SQL Server Agent

- Vérifiez les éléments suivants
 - Le travail s'exécute
 - Le compte de sécurité sous lequel le travail est exécuté
 - Le compte de service Agent SQL Server ou le proxy
 - L'utilisateur SQL pour les étapes de travail T-SQL
 - Les autorisations du compte
- Les tâches que le travail accomplit
- L'historique d'étapes de travail



Points clés

Quand les travaux de SQL Server Agent ne fonctionnent pas comme prévu, les problèmes sont souvent causés par des risques de sécurité. Pour résoudre les problèmes liés aux travaux de SQL Server Agent, vous devez suivre les étapes suivantes :

1. Vérifiez que le travail est bien en cours d'exécution. Examinez le journal d'activités des travaux et vérifiez si le travail a été exécuté. Pour chaque échec d'un travail indiqué par un X rouge (comme le montre l'exemple de la diapositive), développez le travail et recherchez les étapes qui ont échoué. Les étapes de travail qui ont échoué sont également marquées d'une icône X rouge.
2. Vérifiez le compte de sécurité. Lorsque vous cliquez sur l'étape de travail qui échoue, vous voyez dans la partie inférieure droite de la fenêtre une indication du contexte de sécurité dans lequel l'étape du travail a été exécutée. Vérifiez l'appartenance aux groupes pour le compte afin de vérifier que le compte dispose des autorisations requises.
3. Vérifiez les tâches que l'étape de travail doit effectuer. Cela inclut tous les objets T-SQL et tous les fichiers ou ressources Windows qui doivent être accessibles.
4. Vérifiez, pour chaque étape qui échoue, que le compte utilisé est capable d'accéder aux ressources nécessaires au déroulement de l'étape.



Remarque Un autre problème qui survient très fréquemment est que les étapes de travail indiquent des chemins d'accès locaux au lieu de chemins d'accès UNC. En général, il vaut mieux que les travaux utilisent des chemins d'accès UNC afin d'être portables. Autrement dit, un travail doit pouvoir être migré sur un autre serveur si nécessaire.

Question : qu'est-ce qui peut être à l'origine d'un travail qui s'exécute parfaitement bien sur un système de test mais échoue lorsqu'il est exécuté sur un système de production ?

Démonstration 1A : Affecter un contexte de sécurité aux étapes de travail

Dans cette démonstration, vous allez apprendre à

- Afficher l'identité sous laquelle a été exécutée l'étape de travail
- Modifier le contexte de sécurité pour les étapes de travail T-SQL
- Résoudre les problèmes d'une étape de travail

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_14_PRJ\22462A_14_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 pour exécuter chaque lot T-SQL contenu dans le fichier.

Leçon 2

Configuration des informations d'identification

- Vue d'ensemble des informations d'identification
- Configuration des informations d'identification
- Gestion des informations d'identification
- Démonstration 2A : Configuration des informations d'identification

Pour que les étapes de travail SQL Server puissent accéder aux ressources externes à SQL Server, les étapes de travail doivent être exécutées dans le contexte de sécurité d'une identité Windows qui a l'autorisation d'accéder aux ressources requises. Les identités Windows sont distinctes des identités SQL Server, même si SQL Server peut utiliser des connexions et des groupes Windows. Pour qu'une étape de travail puisse utiliser une identité Windows distincte, elle doit être en mesure de se connecter sous cette identité, et pour pouvoir ouvrir une session, le nom d'utilisateur Windows et le mot de passe doivent être enregistrés quelque part. Les informations d'identification sont des objets SQL Server utilisés pour stocker les noms et mots de passe d'utilisateur Windows.

Objectifs

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

- décrire les informations d'identification ;
- configurer les informations d'identification ;
- gérer les informations d'identification.

Vue d'ensemble des informations d'identification

Les informations d'identification sont des objets SQL Server qui stockent les informations d'authentification

- Les informations d'identification sont
 - Requises pour accéder aux ressources externes
 - À mot de passe chiffré au moyen de la clé principale du service
- Les informations d'identification servent à
 - Fournir une identité Windows pour les connexions SQL Server qui doivent accéder à des ressources externes
 - Fournir des identités aux comptes proxy utilisés dans les étapes de travail Agent SQL Server

Points clés

Une information d'identification est un objet SQL Server qui contient les informations d'authentification requises pour la connexion à une ressource extérieure à SQL Server. La plupart des informations d'identification contiennent un nom d'utilisateur et un mot de passe Windows.

Informations d'identification

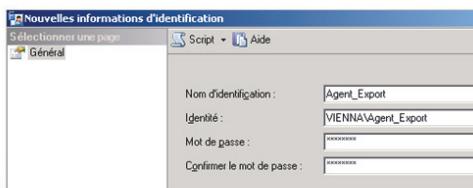
Un proxy d'agent SQL qui peut être utilisé pour l'exécution d'un travail est mappé à des informations d'identification dans SQL Server. Dans la leçon 3, vous allez découvrir comment mapper un compte proxy sur des informations d'identification.

SQL Server crée des informations d'identification système qui sont associées automatiquement à des points de terminaison spécifiques. Ces informations d'identification créées automatiquement sont appelées informations d'identification système et leur nom est préfixé par deux signes dièse (##).

Question : comment SQL Server accède-t-il aux ressources en dehors de SQL Server, lorsque l'utilisateur est connecté via une connexion Windows ?

Configuration des informations d'identification

- Crées avec T-SQL ou SSMS
- Stockées dans la base de données master, le secret chiffré à l'aide de la clé principale du service



```
USE master;
GO

CREATE CREDENTIAL Agent_Export
WITH IDENTITY = N'VIENNA\Agent_Export',
SECRET = N'Pa$$w0rd';
GO
```

Points clés

Les informations d'identification peuvent être créées à l'aide de l'instruction T-SQL CREATE CREDENTIAL ou au moyen de l'interface utilisateur graphique dans SSMS.

Configuration des informations d'identification

Le mot de passe d'une information d'identification est secret. Il est fortement chiffré et est enregistré dans la base de données master. Lorsque SQL Server a besoin pour la première fois de réaliser un chiffrement quelconque, le service SQL Server crée une clé de chiffrement principale. La clé principale du service est également utilisée pour protéger les clés principales de chaque base de données. (Toutes les bases de données ne possèdent pas de clés principales).

Bien souvent la stratégie des organisations implique de remplacer régulièrement les clés de chiffrement. Si la clé principale de service est régénérée, les secrets stockés relativement aux informations d'identification sont automatiquement déchiffrés et chiffrés à nouveau avec la nouvelle clé principale du service.

 **Remarque** Le chiffrement dans SQL Server est une rubrique avancée qui n'entre pas dans le cadre de ce cours. Notez que le chiffrement des secrets des informations d'identification par un fournisseur de gestion de clés extensible (EKM) est également pris en charge ; cela est également hors du cadre de ce cours.

Gestion des informations d'identification

- Les informations d'identification peuvent être répertoriées en consultant la vue système sys.credentials
- Les informations d'identification sont modifiées à l'aide de ALTER CREDENTIAL
 - L'identité et le secret sont toujours modifiés
- Les informations d'identification sont supprimées via DROP CREDENTIAL

```
SELECT * FROM sys.credentials;
GO
ALTER CREDENTIAL Agent_Export
WITH IDENTITY = N'VIENNA\Agent_Export',
SECRET = N'NewPa$$wOrd';
GO
DROP CREDENTIAL Agent_Export;
GO
```

Points clés

SQL Server offre la vue système sys.credentials pour fournir des informations de catalogue sur les informations d'identification existantes. Examinons la requête suivante :

```
SELECT * FROM sys.credentials;
```

Une fois exécutée, cette requête retourne des informations semblables aux résultats suivants :

	Résultats	Messages					
	credential_id	name	credential_identity	create_date	modify_date	target_type	target_id
1	65536	Agetn_Export	VIENNA\Agent_Export	2013-02-19 11:49:37.880	2013-02-19 11:49:37.880	NULL	NULL

Modification des informations d'identification

Le mot de passe d'un compte Windows peut changer au fil du temps. Vous pouvez mettre à jour des informations d'identification avec de nouvelles valeurs à l'aide de l'instruction ALTER CREDENTIAL. Dans l'exemple de la diapositive, remarquez comment les informations d'identification d'Agent_Export sont mises à jour. Le nom d'utilisateur et le mot de passe (autrement dit, le secret) sont fournis dans l'instruction ALTER CREDENTIAL. La commande ALTER CREDENTIAL met toujours à jour à la fois l'identité et le secret.

Les informations d'identification sont supprimées par l'instruction DROP CREDENTIAL.

Question : que se passe-t-il lorsque le mot de passe d'utilisateur Windows qui est mappé aux informations d'identification change ou expire ?

Démonstration 2A : Configuration des informations d'identification

Dans cette démonstration, vous allez apprendre à

- Créer un travail qui copie un fichier
- Créer une information d'identification 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_14_PRJ\22462A_14_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 **21 – Demonstration 2A.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

Leçon 3

Configuration de comptes proxy

- Vue d'ensemble des comptes proxy
- Utilisation de comptes proxy intégrés
- Gestion des comptes proxy
- Démonstration 3A : Configuration de comptes proxy

Vous venez de découvrir dans la dernière leçon que les informations d'identification sont utilisées dans SQL Server pour stocker les identités qui sont externes à SQL Server. Principalement, il s'agit d'identités Windows et les informations d'identification servent à stocker leurs noms et mots de passe d'utilisateur Windows. Pour permettre à une étape de travail d'une tâche SQL Server Agent d'utiliser des informations d'identification, l'étape de travail est mappée aux informations d'identification au moyen d'un compte proxy. Il existe un ensemble de comptes proxy intégrés et vous pouvez créer des comptes proxy manuellement.

Objectifs

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

- décrire les comptes proxy ;
- utiliser des comptes proxy intégrés ;
- gérer des comptes proxy.

Vue d'ensemble des comptes proxy

Un compte proxy fournit à Agent SQL Server l'accès aux informations d'identification de sécurité Microsoft Windows

- Créé à l'aide de SSMS ou de la procédure dbo.sp_add_proxy dans msdb
- Peut toujours être utilisé par les membres du rôle serveur fixe sysadmin
- Peut être utilisé, avec des autorisations, par
 - une connexion SQL
 - un rôle serveur ou msdb

Étape de travail → Proxy → Information d'identification → Ressource

Points clés

Un compte proxy SQL Server sert à définir le contexte de sécurité utilisé pour une étape de travail. Un compte proxy est généralement utilisé pour fournir à SQL Server Agent l'accès aux informations d'identification de sécurité d'un utilisateur Microsoft Windows. Chaque compte proxy peut être associé à un ou plusieurs sous-systèmes.

Une étape de travail qui utilise le compte proxy peut accéder aux sous-systèmes spécifiés en utilisant le contexte de sécurité de l'utilisateur Windows. Avant d'exécuter une étape de travail qui utilise un compte proxy, SQL Server Agent emprunte les informations d'identification définies dans le proxy, puis exécute l'étape de travail en utilisant ce contexte de sécurité.

 **Remarque** L'utilisateur Windows spécifié dans les informations d'identification doit avoir l'autorisation d'« Ouvrir une session en tant que tâche » sur l'ordinateur exécutant SQL Server.

La création d'un compte proxy ne modifie pas les autorisations existantes du compte Windows spécifié dans les informations d'identification. Par exemple, vous pouvez créer un compte proxy pour un utilisateur Windows qui n'est pas autorisé à se connecter à une instance SQL Server. Les étapes de travail qui utilisent ce compte proxy ne seraient alors pas en mesure de se connecter à SQL Server.

Notez qu'un utilisateur doit avoir l'autorisation d'utiliser le compte proxy avant de pouvoir spécifier celui-ci dans une étape de travail. Par défaut, seuls les membres du rôle serveur fixe sysadmin sont autorisés à accéder à tous les comptes proxy.

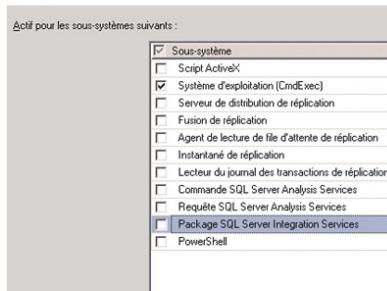
Les autorisations d'accès à un compte proxy peuvent être accordées à trois types de principaux de sécurité :

- connexions SQL Server ;
- rôles de serveur ;
- rôles dans la base de données msdb.

Question : quand les comptes proxy doivent-ils être utilisés ?

Utilisation de comptes proxy intégrés

- Les proxies sont définis pour des sous-systèmes spécifiques de SQL Server
- Les proxies peuvent être utilisés par un ou plusieurs sous-systèmes
 - Ils assurent une sécurité limitée pour des fonctions spécifiques
- Chaque type d'étape de travail est associé à un sous-système spécifique, sauf T-SQL



Points clés

Les comptes proxy de SQL Server sont utilisés par les sous-systèmes. Un sous-système est un objet prédéfini qui représente un jeu de fonctionnalités disponibles dans SQL Server. Chaque compte proxy peut être associé à plusieurs sous-systèmes.

Les sous-systèmes permettent d'assurer le contrôle de la sécurité car ils segmentent les fonctions disponibles à un compte proxy. Précédemment dans ce module, vous avez appris que chaque étape de travail s'exécute dans le contexte d'un compte proxy, à l'exception des étapes de travail T-SQL. Les étapes de travail T-SQL utilisent la commande EXECUTE AS pour définir le contexte de sécurité.

SQL Server Agent vérifie l'accès au sous-système pour un compte proxy à chaque exécution d'une étape de travail. Si l'environnement de sécurité a changé ou si le compte proxy n'a plus accès au sous-système, l'étape de travail échoue.

Question : pourquoi, en règle générale, les comptes proxy ne doivent pas être affectés à tous les sous-systèmes ?

Gestion des comptes proxy

- Les proxies sont définis dans msdb
- Les informations de configuration sont accessibles via les tables système de msdb
 - sysproxies, sysproxylogin, sysproxyloginsubsystem, syssubsystems

```
USE msdb;
GO

SELECT p.name AS ProxyName,
       c.name AS CredentialName,
       p.description AS ProxyDescription
  FROM dbo.sysproxies AS p
 INNER JOIN sys.credentials AS c
    ON p.credential_id = c.credential_id;
```

Points clés

La configuration de SQL Server Agent est stockée dans la base de données msdb. Les comptes proxy font partie de SQL Server Agent. Ainsi, la configuration des comptes proxy est également enregistrée dans la base de données msdb.

Les détails de la configuration actuelle du compte proxy peuvent être obtenus via un jeu de vues système répertoriées dans le tableau suivant :

Vue système	Description
dbo.sysproxies	Retourne une ligne par proxy défini dans SQL Server Agent.
dbo.sysproxylogin	Retourne quelles connexions SQL Server sont associées avec chaque compte proxy de SQL Server Agent. Notez qu'aucune entrée n'est enregistrée ou retournée pour les membres du rôle sysadmin.
dbo.sysproxyloginsubsystem	Retourne quels sous-systèmes de SQL Server Agent sont définis pour chaque compte proxy.
dbo.syssubsystems	Retourne des informations sur tous les sous-systèmes proxy disponibles de SQL Server Agent.

Vous avez appris précédemment que les informations d'identification peuvent être affichées depuis la vue système sys.credentials. Les informations d'identification sont stockées dans la base de données master, et non dans la base de données msdb.

Question : pourquoi les informations d'identification sont-elles stockées dans la base de données master au lieu de la base de données msdb ?

Démonstration 3A : Configuration de comptes proxy

Dans cette démonstration, vous allez apprendre à

- Comment définir un compte proxy
- Comment utiliser un compte proxy
- Comment afficher les comptes proxy et leurs propriétés avec T-SQL

Procédure de démonstration

1. Si les démonstrations 2A n'ont pas été effectuées :
 - 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_14_PRJ\22462A_14_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 – Démonstration 2A.sql** à partir de l'Explorateur de solutions et suivez les instructions contenues dans le fichier.
2. Ouvrez le fichier de script **31 – Démonstration 3A.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

Atelier pratique 14 : Configuration de la sécurité pour SQL Server Agent

- Exercice 1 : Résoudre la défaillance d'exécution du travail
- Exercice 2 : Résoudre le problème de sécurité
- Exercice difficile 3 : Dépannage plus avancé (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_14_PRJ\22462A_14_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 déployé un travail qui extrait les informations des clients potentiels qui n'ont pas été contactés récemment. Vous avez également programmé le travail pour qu'il s'exécute avant chacune des deux réunions marketing planifiées hebdomadaires. L'équipe marketing a déployé une nouvelle fonctionnalité dans l'application Promote pour améliorer les processus de planification. Plutôt que de planifier le travail, l'application Promote doit exécuter le travail à la demande.

L'application Promote se connecte sous une connexion SQL appelée PromoteApp. L'un des autres administrateurs de base de données, Terry Adams, a tenté de configurer SQL Server de façon à ce que la connexion PromoteApp puisse exécuter le travail. Mais il n'arrive pas à comprendre pourquoi le travail ne s'exécute toujours pas. Dans cet atelier pratique, vous devez comprendre et résoudre le problème.

Documentation

Actions qui ont déjà été effectuées par Terry Adams

1. Il a créé un utilisateur de base de données pour la connexion PromoteApp dans la base de données msdb.
2. Il a accordé à l'utilisateur de base de données PromoteApp l'autorisation d'exécuter la procédure stockée msdb.dbo.sp_start_job.
3. Il a ajouté l'utilisateur de base de données PromoteApp au rôle de base de données SQLAgentOperatorRole.
4. Il a modifié le travail Extraire les prospects non contactés pour définir la connexion PromoteApp comme propriétaire du travail.
5. Il a créé un utilisateur Windows nommé ExtractUser avec un mot de passe Pa\$\$w0rd.
6. Il a ajouté l'utilisateur Windows ExtractUser au rôle db_ssisoperator dans la base de données msdb.

Exercice 1 : Résoudre la défaillance d'exécution du travail

Scénario

Vous devez examiner les actions déjà entreprises par Terry Adams, puis examiner l'historique du travail qui échoue. Vous devez comprendre pourquoi ce travail échoue.

La tâche principale de cet exercice est la suivante :

1. Résoudre la défaillance d'exécution du travail.

► Tâche 1 : Résoudre la défaillance d'exécution du travail

- Examinez les actions effectuées précédemment par Terry Adams qui sont détaillées dans la documentation de l'exercice.
- Affichez le journal d'historique du travail Extraire les prospects non contactés.
- Déterminez à partir de l'historique la raison pour laquelle le travail est en échec.

Résultats : À la fin de cet exercice, vous devez avoir déterminé la raison pour laquelle le travail échoue.

Exercice 2 : Résoudre le problème de sécurité

Scénario

Vous avez déterminé qu'un compte proxy est nécessaire pour la bonne exécution de l'étape de travail qui échoue. Vous devez créer et affecter le compte proxy, puis tester que tous les problèmes ont été résolus.

Les tâches principales de cet exercice sont les suivantes :

1. Créer et affecter un compte proxy.
2. Effectuer des tests pour voir si tous les problèmes ont été résolus.

► Tâche 1 : Créer et affecter un compte proxy

- Au moyen de SQL Server Management Studio, créez des informations d'identification SQL Server appelées ExtractIdentity associées à l'utilisateur Windows 1077XA-MIA-SQL\ExtractUser et un mot de passe Pa\$\$w0rd.
- Créez un compte proxy SQL Server appelé ExtractionProxy qui est associé aux informations d'identification ExtractIdentity et qui est actif dans le sous-système du package SQL Server Integration Services. Veillez à accorder à la connexion PromoteApp l'autorisation d'utiliser ces informations d'identification.
- Attribuez le compte proxy au travail Extraire les prospects non contactés.

► Tâche 2 : Effectuer des tests pour voir si tous les problèmes ont été résolus

- Tentez d'exécuter le travail Extraire les prospects non contactés.
- Si le travail échoue, passez à l'exercice 3 si vous en avez le temps.

Résultats : À la fin de cet exercice, vous devez avoir corrigé un problème de sécurité lié à un travail.

Exercice difficile 3 : Dépannage plus avancé (si le temps le permet)

Scénario

Après avoir créé et affecté un compte proxy pour le travail, le problème initial était que SQL Server refusait d'exécuter les étapes de travail sans qu'un proxy ait été résolu. Cependant, le travail ne s'exécute toujours pas correctement. Vous devez essayer de résoudre le dernier problème.

La tâche principale de cet exercice est la suivante :

1. Dépannage plus avancé.

► Tâche 1 : Dépannage plus avancé

- Localisez et résolvez les autres problèmes qui empêchent le travail de s'exécuter correctement.
- Vérifiez que le travail s'exécute bien.

Résultats : Après cet exercice, vous devez avoir identifié et résolu les problèmes restants.

Contrôle des acquis du module et éléments à retenir

- Questions de contrôle des acquis
- Meilleures pratiques

Questions de contrôle des acquis

1. Quels types de comptes peuvent être utilisés pour démarrer le service SQL Server Agent ?
2. À quoi peuvent servir les informations d'identification ?

Meilleures pratiques liées à un domaine technologique particulier dans ce module

1. Utiliser un compte d'utilisateur de domaine Windows pour démarrer le compte de service SQL Server Agent.
2. Utiliser un compte avec le moins de privilèges possible.
3. Créer des comptes proxy avec le moins d'autorisations attribuées pour l'exécution d'un travail.

Module 15

Surveillance de SQL Server 2012 avec alertes et notifications

Sommaire :

Leçon 1 : Configuration de la messagerie de base de données	15-3
Leçon 2 : Surveillance des erreurs SQL Server	15-12
Leçon 3 : Configuration des opérateurs, des alertes et des notifications	15-19
Atelier pratique 15 : Surveillance des travaux de l'agent SQL avec alertes et notifications	15-31

Vue d'ensemble du module

- Configuration de la messagerie de base de données
- Surveillance des erreurs SQL Server
- Configuration des opérateurs, des alertes et des notifications

De nombreux administrateurs de base de données fonctionnent en mode réactif. Ils répondent aux demandes des utilisateurs qui signalent des erreurs ou lorsque des problèmes se produisent. Il est important d'essayer de passer d'un mode réactif à un mode plus proactif.

Un des aspects fondamentaux de la gestion de Microsoft® SQL Server® de manière proactive consiste à être sûr d'avoir connaissance des événements qui surviennent sur le serveur, en temps réel. Vous pouvez voir cela dans un premier temps comme une démarche réactive. Toutefois, il existe de nombreux types de problèmes qui peuvent survenir et qui ne sont pas directement apparents aux utilisateurs des applications de base de données. SQL Server consigne une grande quantité d'informations sur les problèmes. Vous pouvez le configurer pour vous en aviser automatiquement lorsqu'ils surviennent, via des alertes et des notifications.

La messagerie constitue la façon la plus courante pour les administrateurs de base de données SQL Server de recevoir des informations sur les événements pertinents. SQL Server peut être configuré pour envoyer un message via un serveur de messagerie SMTP existant.

Objectifs

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

- configurer la messagerie de base de données ;
- surveiller les erreurs de SQL Server ;
- configurer les opérateurs, les alertes et les notifications.

Leçon 1

Configuration de la messagerie de base de données

- Vue d'ensemble de la messagerie de base de données
- Profils de la messagerie de base de données
- Sécurité de la messagerie de base de données
- Journaux et rétention de la messagerie de base de données
- Démonstration 1A : Configuration de la messagerie de base de données

SQL Server doit être en mesure d'aviser les administrateurs de la survenue de problèmes qui requièrent leur attention. Le courrier électronique est le mécanisme le plus souvent utilisé pour les notifications de SQL Server. La fonctionnalité de messagerie de base de données SQL Server est utilisée pour se connecter à un serveur SMTP existant lorsque SQL Server a besoin d'envoyer un courrier électronique.

SQL Server peut être configuré avec plusieurs profils de messagerie et de manière à déterminer quels utilisateurs ont la possibilité d'utiliser les fonctionnalités de messagerie du produit. Il est important de pouvoir suivre et tracer les messages électroniques qui ont été envoyés. SQL Server vous permet de configurer une stratégie de rétention des messages électroniques.

Objectifs

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

- décrire la messagerie de base de données ;
- configurer les profils de messagerie de base de données ;
- configurer la sécurité de la messagerie de base de données ;
- configurer la rétention de la messagerie de base de données.

Vue d'ensemble de la messagerie de base de données

La messagerie de base de données est une implémentation du protocole SMTP standard qui permet aux composants de base de données d'envoyer des messages électroniques

- Configuration simple via l'Assistant Configuration de la messagerie de base de données
 - Différents profils disponibles
 - Remise en arrière-plan avec Service Broker
- Utilisée par la messagerie d'Agent SQL Server
 - Envoie des notifications de travaux et d'alertes
 - Requiert un profil de messagerie pour le compte d'Agent SQL Server



Points clés

La messagerie de base de données envoie le courrier électronique via un serveur de Protocole SMTP (Simple Mail Transport Protocol). Il doit y avoir un serveur SMTP disponible sur le réseau pour accepter le message.

Configuration de la messagerie de base de données

Pour activer et configurer les comptes et les profils de messagerie de base de données, utilisez l'assistant Configuration de la messagerie de base de données.

Bien que les détails de configuration de la messagerie de base de données soient stockés dans la base de données msdb avec toutes les autres informations de configuration de SQL Server Agent, SQL Server Agent met en cache les informations de profil dans la mémoire. Ainsi SQL Server Agent a la possibilité d'envoyer le courrier électronique même dans les cas où le moteur de base de données SQL Server n'est plus disponible.

La messagerie de base de données peut être utilisée pour envoyer un courrier électronique dans le cadre d'un travail de SQL Server Agent, en réponse à une alerte déclenchée, ou pour le compte d'un utilisateur par l'exécution de la procédure stockée système sp_send_dbmail.

SQL Mail

Notez que pour des raisons de compatibilité descendante, les versions antérieures de SQL Server comportaient une fonctionnalité appelée SQL Mail. SQL Mail était une fonctionnalité de messagerie électronique basée sur l'interface MAPI (Messaging Application Programming Interface) qui vous permettait de configurer SQL Server de manière à ce qu'il envoie et reçoive le courrier électronique via Microsoft Exchange Server ou d'autres serveurs de messagerie basés sur l'interface MAPI. SQL Mail n'est pas pris en charge dans SQL Server 2012.

Relais SMTP

La plupart des serveurs SMTP d'aujourd'hui sont configurés, par défaut, pour refuser tout relais de courrier électronique. Un serveur configuré pour autoriser le relais est prêt à accepter du courrier électronique à partir d'un autre serveur, même si la cible du courrier électronique n'est pas dans le domaine du serveur de messagerie. Le serveur de messagerie transfère ensuite le courrier électronique à sa destination finale.

Il est important de refuser de relayer le courrier électronique pour éviter les problèmes liés au courrier indésirable. Les serveurs de messagerie qui n'ont pas cette protection sont appelés serveurs « relais ouvert » et constituent une cible pour les usages abusifs. De nombreuses entreprises entretiennent des listes noires de serveurs de messagerie qui envoient régulièrement du courrier indésirable. Lorsque l'on pratique le relais par d'autres serveurs, il apparaît que ce sont ces autres serveurs qui envoient le courrier indésirable, et non le serveur qui l'a envoyé initialement. Étant donné que la configuration la plus courante pour les serveurs de messagerie est de refuser toute activité de relais, le serveur SMTP doit être configuré, si nécessaire, pour autoriser le relais des messages électroniques de SQL Server.

Question : pourquoi les administrateurs de messagerie doivent-ils être inclus dans les discussions, lorsque vous planifiez une configuration de la messagerie de base de données ?

Profils de la messagerie de base de données

- Un profil de messagerie de base de données définit un ou plusieurs comptes de messagerie
 - Définit la configuration utilisée pour envoyer des messages électroniques
 - Fournit plusieurs comptes de messagerie de base de données pour des raisons de fiabilité
- Le profil par défaut d'un compte de connexion est utilisé s'il n'est pas spécifié

Type de profil	Description
Privé	Accessible uniquement à des utilisateurs ou rôles spécifiques. Un profil privé par défaut est prioritaire sur le profil public par défaut
Public	Peut être utilisé par un utilisateur ou un rôle disposant des autorisations appropriées

Points clés

Un profil de messagerie de base de données est une collection de comptes de messagerie de base de données. Au moins un compte de messagerie de base de données est requis. Si plusieurs comptes de messagerie de base de données sont définis pour un profil, les comptes sont utilisés dans un ordre prédéfini lors de la tentative d'envoi des courriers. Le niveau de redondance fourni par l'emploi de plusieurs profils de messagerie peut améliorer la fiabilité de l'ensemble.

Les profils peuvent être privés ou publics. Les profils privés sont contrôlés de manière stricte et ne sont disponibles que pour certains utilisateurs ou rôles spécifiques. En comparaison, les profils publics peuvent être utilisés par n'importe quel utilisateur ou rôle bénéficiant de l'autorisation adéquate.

Profils multiples

Il est possible de créer plusieurs configurations par l'emploi de différents profils. Par exemple, un profil peut être créé pour envoyer le courrier à un serveur SMTP interne, à l'aide d'une adresse de messagerie interne, pour les courriers envoyés par SQL Server Agent. Un deuxième profil peut être créé pour être utilisé par une application de base de données qui doit envoyer aux clients des notifications par courrier électronique externe.

Chaque utilisateur de base de données peut avoir accès à plusieurs profils. Si aucun profil n'est spécifié lors de l'envoi d'un message, le profil par défaut est utilisé. S'il existe à la fois des profils privés et publics, la priorité est donnée à un profil privé par défaut sur un profil public par défaut. Si aucun profil par défaut n'est spécifié, ou s'il faut utiliser un profil non défini par défaut, le nom du profil doit être spécifié comme paramètre de la procédure stockée système sp_send_dbmail, comme l'indique le code suivant :

```
EXEC msdb.dbo.sp_send_dbmail
    @profile_name = 'Proseware Administrator',
    @recipients = 'admin@AdventureWorks.com',
    @body = 'Daily backup completed successfully.',
    @subject = 'Daily backup status';
```

Question : si un utilisateur a accès à plusieurs profils, lequel est utilisé si aucun profil n'est spécifié ?

Sécurité de la messagerie de base de données

- **Messagerie de base de données**
 - S'exécute sous le compte de service du moteur SQL Server dans un processus isolé
 - Utilise des procédures stockées qui sont désactivées par défaut
 - Envoie un message uniquement pour les membres de DatabaseMailUserRole dans msdb. (par défaut, les membres du rôle serveur sysadmin peuvent envoyer des messages)
 - Peut interdire l'utilisation d'extensions de fichier et de tailles de pièces jointes spécifiques
- **Les profils privés sont limités aux utilisateurs ou aux rôles spécifiques**

Points clés

Le choix du compte de service pour le service SQL Server est important lors de la configuration de la messagerie de base de données. Si SQL Server est configuré pour s'exécuter sous le compte de service local, il n'est pas autorisé à créer des connexions réseau sortantes. Dans ce cas, la messagerie de base de données ne peut pas contacter le serveur de messagerie situé sur un autre ordinateur.

Procédures stockées de messagerie de base de données

Pour réduire la surface de sécurité de SQL Server, les procédures stockées étendues système utilisées pour la messagerie de base de données sont désactivées par défaut. Lorsque vous exécutez l'assistant Configuration de la messagerie de base de données, les procédures sont activées automatiquement. Si vous voulez configurer manuellement la messagerie de base de données, vous pouvez activer les procédures stockées étendues système en définissant l'option « XP de messagerie de base de données » de sp_configure à la valeur 1.

Sécurité et limitations de pièce jointe

Tous les utilisateurs de SQL Server ne sont pas autorisés à envoyer des messages électroniques. La possibilité d'envoyer des messages électroniques est restreinte aux membres du rôle de base de données appelé DatabaseMailUserRole dans la base de données msdb. Les membres du rôle serveur fixe sysadmin peuvent également envoyer un message de messagerie de base de données.

Vous pouvez également limiter les types et la taille des pièces jointes qui peuvent être incluses dans les messages électroniques envoyés par la messagerie de base de données. Cette limitation peut être configurée à l'aide de l'assistant Configuration de la messagerie de base de données ou en appelant la procédure stockée système dbo.sysmail_configure_sp dans la base de données msdb.

Question : pourquoi est-ce-que la messagerie de base de données ne peut pas être utilisée avec un serveur SMTP distant lorsque le compte de service local est utilisé pour le moteur de base de données ?

Journaux et rétention de la messagerie de base de données

- La messagerie de base de données enregistre des informations à des fins de résolution
 - Audit des messages et des pièces jointes
 - Une stratégie de rétention doit être planifiée pour limiter la croissance de msdb

```
USE msdb;
GO

DECLARE @CutoffDate datetime ;
SET @CutoffDate = DATEADD(m, -1,SYSDATETIME());

EXECUTE dbo.sysmail_delete_mailitems_sp
@sent_before = @CutoffDate;

EXECUTE dbo.sysmail_delete_log_sp
@logged_before = @CutoffDate;
GO
```

Points clés

SQL Server enregistre les messages dans des tables internes de la base de données msdb. Les messages du journal peuvent être affichés en interrogeant la table dbo.sysmail_log. Le niveau de journalisation effectué par SQL Server peut être configuré comme l'un des trois niveaux suivants :

Niveau de journalisation	Description
Normal	Seules les erreurs sont enregistrées.
Étendu	Les messages d'erreur, d'avertissement et d'information sont enregistrés.
Détailé	Comme le niveau Étendu, avec en plus les messages de réussite et un certain nombre de messages internes.

Vous pouvez configurer le paramètre de niveau de journalisation à l'aide de la boîte de dialogue Configurer les paramètres du système de l'assistant Configuration de la messagerie de base de données, ou en appelant la procédure stockée dbo.sysmail_configure_sp dans la base de données msdb.

Le niveau Détailé ne doit être utilisé à des fins de dépannage, car il peut générer un grand volume d'entrées de journal.

Tables et vues de la messagerie de base de données

Les tables internes de la base de données msdb sont utilisées pour stocker les messages électroniques et les pièces jointes envoyés depuis la messagerie de base de données, ainsi que l'état actuel de chaque message. La messagerie de base de données met à jour ces tables à mesure que chaque message est traité.

Vous pouvez suivre l'état de remise d'un message individuel en consultant les informations dans les vues suivantes :

- dbo.sysmail_allitems ;
- dbo.sysmail_sentitems ;
- dbo.sysmail_unsentitems ;
- dbo.sysmail_faileditems.

Pour afficher les détails des pièces jointes d'un courrier, consultez la vue dbo.sysmail_mailattachments.

Stratégie de rétention

La messagerie de base de données conserve les messages sortants et leurs pièces jointes dans la base de données msdb. Cela implique qu'il est nécessaire d'organiser une stratégie de rétention des messages électroniques et des entrées du journal. Si le volume des messages de la messagerie de base de données et des pièces jointes associées est élevé, prévoyez une croissance substantielle de la base de données msdb.

Supprimez périodiquement les messages pour regagner l'espace et être conforme aux stratégies de rétention de documents de votre organisation. Par exemple, l'exemple de la diapositive montre comment supprimer les messages, les pièces jointes et les entrées de journal datant de plus d'un mois. Vous pouvez planifier l'exécution régulière de ces commandes en créant un travail SQL Server Agent.

Démonstration 1A : Configuration de la messagerie de base de données

Dans cette démonstration, vous allez apprendre à

- Comment configurer la messagerie de base de données ?
- Comment configurer Agent SQL Server pour utiliser le profil ?

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**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_15_PRJ\22462A_15_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

Surveillance des erreurs SQL Server

- Qu'est-ce qu'une erreur ?
- Gravité d'une erreur
- Configuration du journal des erreurs de SQL Server
- Démonstration 2A : Exécution en boucle du journal des erreurs

Il est important de comprendre les principaux aspects des erreurs applicables à SQL Server. Vous devez en particulier prendre en compte :

- la nature des erreurs ;
- l'endroit où peuvent survenir les erreurs lors de l'exécution de code T-SQL ;
- les données retournées par les erreurs ;
- les niveaux de gravité des erreurs.

Les erreurs graves de SQL Server sont stockées dans le journal des erreurs de SQL Server. Il est important de savoir configurer ce journal.

Objectifs

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

- décrire ce qu'est une erreur ;
- décrire les niveaux de gravité des erreurs ;
- configurer le journal des erreurs de SQL Server.

Qu'est-ce qu'une erreur ?

- Les erreurs déclenchées par le moteur de base de données présentent les attributs suivants

Attribut	Description
Numéro de l'erreur	Chaque message d'erreur possède un numéro d'erreur unique
Message d'erreur	Chaîne contenant des informations de diagnostic sur l'origine de l'erreur
Gravité	Indique la gravité de l'erreur
État	Valeur utilisée pour déterminer l'emplacement dans le code où une erreur s'est produite
Nom de la procédure	Nom de la procédure stockée ou du déclencheur dans lequel l'erreur s'est produite (le cas échéant)
Numéro de ligne	Indique la ligne d'un lot, d'une procédure stockée, d'un déclencheur ou d'une fonction où l'erreur a été déclenchée

Points clés

Une erreur est en elle-même un objet qui possède des propriétés, comme l'indique le tableau.

Attributs de l'erreur

Il peut ne pas être immédiatement évident qu'une erreur SQL Server (que l'on appelle parfois une exception) est en elle-même un objet. Les erreurs retournent plusieurs propriétés utiles (ou attributs).

Les numéros d'erreur sont utiles lorsque vous recherchez des informations sur l'erreur spécifique, notamment lors d'une recherche en ligne d'informations sur l'erreur.

Vous pouvez afficher la liste des messages d'erreur délivrés par le système en interrogeant la vue du catalogue sys.messages :

```
SELECT * FROM sys.messages
ORDER BY message_id, language_id;
```

Une fois exécutée, cette commande retourne ce qui suit :

	message_id	language_id	severity	is_event_logged	text
1	21	1028	20	0	警告: 繼重錯誤 %1! 發生於 %2!. 請記錄錯誤和時間，並連絡您的系統管理員。
2	21	1031	20	0	Warning: Schwerwiegender Fehler %1! um %2!. Notieren Sie den Fehler und die Warnung: Fatal error % occurred at %S_DATE. Note the error and time, and contact your system administrator.
3	21	1033	20	0	Avertissement : erreur irrécupérable %1! à %2!. Prenez note de l'erreur et de l'heure.
4	21	1036	20	0	Aviso: errore irreversibile %1! alle %2!. Prendere nota dell'errore e dell'ora in cui è avvenuta.
5	21	1040	20	0	Avviso: errore irreversibile %1! alle %2!. Prendere nota dell'errore e dell'ora in cui è avvenuta.
6	21	1041	20	0	警告: %2!で致命的なエラー-%1! が発生しました。エラーが発生した際に記録してシステム管理者に報告: %2!에 오류 %1이(가) 발생했습니다. 오류와 시간을 기록한 다음 시스템 관리자에게 보고하세요.
7	21	1042	20	0	경고: %2!에 오류 %1이(가) 발생했습니다. 오류와 시간을 기록한 다음 시스템 관리자에게 보고하세요.

Notez qu'il y a plusieurs messages portant le même identifiant de message, message_id. Les messages d'erreur sont traduisibles et peuvent être retournés dans plusieurs langues. Le « Language_id » 1033 correspond à la version en anglais du message. Vous pouvez voir un message en anglais à la troisième ligne de la sortie ci-dessus.

La gravité, comme son nom l'indique, renvoie au niveau de gravité de l'erreur. Il est décrit plus loin dans la rubrique suivante.

L'état est défini par l'auteur du code qui a déclenché l'erreur. Par exemple, si vous avez écrit une procédure stockée susceptible de déclencher une erreur dans le cas d'un client manquant et que cinq emplacements dans le code sont susceptibles de recevoir ce message, vous avez pu affecter un état différent aux emplacements où le message a été déclenché. Cela permet ultérieurement d'aider à corriger l'erreur.

Le nom de la procédure est le nom de la procédure stockée ou cette erreur s'est produite et le numéro de ligne est son emplacement dans cette procédure. En pratique, les numéros des lignes ne sont pas très utiles et pas toujours pertinents.

Question : dans quelle langue une erreur est-elle indiquée ?

Gravité d'une erreur

- La gravité d'une erreur indique le type de problème rencontré par SQL Server

Plage de numéros d'erreur	Description
0 à 9	Messages d'information
10	Messages d'information qui retournent des informations d'état ou signalent des erreurs qui ne sont pas graves
11 à 16	Erreurs pouvant être corrigées par l'utilisateur
17 à 19	Erreurs logicielles ne peuvent pas être corrigées par l'utilisateur
20 à 24	Erreurs système graves
25	Erreur de fin d'exécution du service SQL Server

Points clés

La gravité d'une erreur indique le type de problème rencontré par SQL Server. Les valeurs de gravité faibles correspondent à des messages d'information et n'indiquent pas les véritables erreurs. Les niveaux de gravité des erreurs sont organisés en plages.

Valeurs de 0 à 10

Les valeurs comprises entre 0 et 9 sont purement des messages d'information. Lorsque les requêtes qui les activent sont exécutées dans SQL Server Management Studio, les informations sont renvoyées mais sans fournir aucune information d'état d'erreur. Par exemple, considérez le code ci-dessous exécuté sur la base de données AdventureWorks :

```
SELECT COUNT(Color) FROM Production.Product;
```

Lors de son exécution il retourne un nombre, comme prévu. Toutefois, si vous cherchez dans l'onglet des messages dans SQL Server Management Studio, vous découvrez le contenu suivant :

```
Warning: Null value is eliminated by an aggregate or other SET operation.
```

```
(1 ligne(s) affectée(s))
```

Notez qu'aucune erreur ne s'est produite réellement mais SQL Server vous avertit qu'il a ignoré les valeurs NULL lors du comptage des lignes. Remarquez qu'aucune information d'état n'est retournée.

Le niveau 10 est le niveau supérieur des messages d'information.

Valeurs de 11 à 16

Les valeurs comprises entre 11 et 16 sont considérées comme correspondant à des erreurs que l'utilisateur peut corriger. En général, elles sont utilisées dans le cas d'erreurs où SQL Server considère que l'instruction en cours d'exécution est erronée.

Voici quelques exemples de ces erreurs :

Exemple de gravité de l'erreur
11 indique qu'un objet n'existe pas
13 indique un interblocage des transactions
14 indique des erreurs telles qu'une autorisation refusée
15 indique des erreurs de syntaxe

Valeurs de 17 à 19

Les valeurs comprises entre 17 et 19 sont considérées comme des erreurs logicielles graves que l'utilisateur ne peut pas corriger. Par exemple, le niveau de gravité 17 indique que SQL Server a échoué en raison d'un manque de ressources (mémoire, espace disque, verrous, etc.).

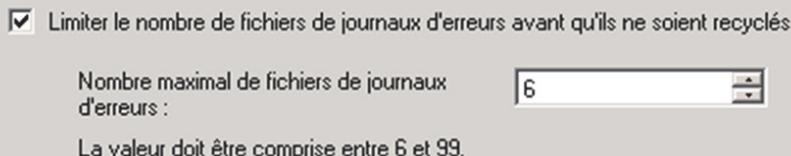
Valeurs supérieures à 19

Les valeurs supérieures à 19 sont susceptibles d'être des erreurs très graves qui impliquent normalement des erreurs inhérentes au matériel ou à SQL Server lui-même. Il est bon de veiller à ce que toutes les erreurs de gravité supérieure à 19 soient stockées et génèrent des alertes.

Question : parmi les plages de numéros d'erreur indiquées sur la diapositive, dans laquelle pensez-vous qu'une erreur de syntaxe peut se produire ?

Configuration du journal des erreurs de SQL Server

- Les erreurs graves sont consignées dans les journaux de l'application et SQL Server
 - Configurable avec sp_altermessages
- Un nouveau fichier journal d'erreurs SQL Server est créé au redémarrage de chaque instance
 - Six fichiers journaux sont conservés par défaut
 - Utilisez sp_cycle_errorlog pour passer à un nouveau fichier journal



Points clés

Les messages importants (en particulier ceux qui sont considérés comme des messages d'erreur grave) sont enregistrés dans le journal des événements des applications Windows et le journal des erreurs SQL Server. La vue sys.messages indique les messages d'erreur disponibles et indique lesquels seront enregistrés par défaut. Vous pouvez contrôler le comportement de journalisation des messages à l'aide de la procédure stockée système sp_altermessages.

Le journal des erreurs SQL Server est installé par défaut dans le dossier suivant :

```
Program Files\Microsoft SQL Server\MSSQL11.<Instance>\MSSQL\LOG\ERRORLOG
```

Les fichiers journaux sont nommés ERRORLOG.n où n correspond au numéro de fichier journal. Les fichiers journaux sont des fichiers texte qui peuvent être affichés à l'aide d'un éditeur de texte ou en utilisant la visionneuse de journal fournie dans SSMS.

Par défaut, SQL Server conserve des copies de sauvegarde des six derniers journaux et donne à la sauvegarde de fichier journal la plus récente l'extension .1, à la deuxième plus récente l'extension .2, et ainsi de suite. Le journal des erreurs en cours n'a aucune extension. Le nombre de fichiers journaux qui doivent être conservés peut être configuré dans SSMS en cliquant avec le bouton droit sur l'option Configurer du nœud Journaux SQL Server dans l'Explorateur d'objets.

Retraitements des fichiers journaux

Le fichier journal a un comportement cyclique à chaque redémarrage de l'instance SQL Server. Vous pouvez occasionnellement souhaiter supprimer les fichiers journaux trop volumineux. À l'aide de la procédure stockée système sp_cycle_errorlog, vous pouvez fermer le fichier journal existant et en ouvrir un nouveau. S'il est nécessaire de retraitre périodiquement le journal, vous pouvez créer un travail de SQL Server Agent pour exécuter la procédure stockée système sp_cycle_errorlog selon une planification. L'exécution en boucle du journal peut vous permettre de l'empêcher de devenir trop volumineux.

Démonstration 2A : Exécution en boucle du journal des erreurs

Dans cette démonstration, vous allez apprendre à

- Comment afficher le journal d'erreurs à l'aide d'un éditeur de texte et SSMS ?
- Comment recycler le fichier journal ?

Configuration de la 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**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_15_PRJ\22462A_15_PRJ.ssmsln** 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.

Leçon 3

Configuration des opérateurs, des alertes et des notifications

- Vue d'ensemble des opérateurs de SQL Server Agent
- Démonstration 3A : Configuration des opérateurs de SQL Server Agent
- Vue d'ensemble des alertes SQL Server
- Crédit d'une alerte
- Configuration des actions d'alerte
- Résolution des alertes et des notifications
- Démonstration 3B : Configuration des alertes et des notifications

Vous avez découvert précédemment dans ce module qu'il est important que SQL Server puisse envoyer des messages à un administrateur lors de la survenue d'événements qui requièrent son attention.

De nombreux systèmes SQL Server ont plusieurs administrateurs. SQL Server Agent vous permet de configurer des opérateurs associés à un ou plusieurs administrateurs et de déterminer quand chacun d'entre eux doit être contacté, ainsi que la méthode appliquée pour ce faire.

SQL Server peut également détecter de nombreux cas susceptibles d'intéresser les administrateurs. Vous pouvez configurer des alertes reposant sur les erreurs de SQL Server ou sur des événements système, tels qu'une faible disponibilité de l'espace disque. SQL Server peut être configuré pour vous informer de ces situations.

Objectifs

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

- décrire le rôle des opérateurs dans SQL Server Agent ;
- mettre en œuvre les alertes SQL Server ;
- créer des alertes ;
- configurer les actions qui doivent se produire en réponse à des alertes ;
- résoudre les alertes et les notifications.

Vue d'ensemble des opérateurs de SQL Server Agent

Un opérateur Agent SQL Server est une personne ou un groupe qui peut recevoir des notifications d'un travail ou d'une alerte

- Les opérateurs peuvent être informés comme suit
 - Courrier électronique
 - Radiomessagerie
 - Net send (éviter cette option)
- Un opérateur de prévention de défaillance peut être configuré

Points clés

Un opérateur de SQL Server Agent est un alias d'une personne ou d'un groupe de personnes pouvant recevoir des notifications électroniques à la fin de travaux ou en cas d'alerte.



Remarque Les opérateurs n'ont pas besoin d'être des connexions Windows, des connexions SQL Server ou des utilisateurs de la base de données. Par exemple, vous pouvez créer un opérateur qui est une référence à une adresse de radiomessagerie.

Les travaux de SQL Server Agent peuvent être configurés pour envoyer des messages lors de leur achèvement, de leur réussite ou de leur échec.

Vous pouvez définir de nouveaux opérateurs, soit à l'aide de SSMS, soit de la procédure stockée système dbo.sp_add_operator. Une fois qu'un opérateur est défini, sa définition est visible dans la table système dbo.sysoperators de la base de données msdb.

Contacter un opérateur

Vous pouvez configurer trois types de méthode de contact pour chaque opérateur :

- Adresse de messagerie : l'adresse de messagerie SMTP à laquelle les notifications doivent être envoyées. Il est souhaitable d'utiliser des adresses de groupe plutôt que des adresses individuelles, lorsque cela est possible. Il est possible de faire suivre plusieurs adresses de messagerie en les séparant par un point-virgule.

- Adresse de radiomessagerie : l'adresse de messagerie SMTP à laquelle est envoyé un message pendant des périodes et des jours spécifiés de la semaine.
- L'adresse Net Send : l'adresse Messenger à laquelle est envoyé un message.

 **Remarque** L'utilisation de Net Send pour l'envoi de notifications est déconseillée et ne devrait pas être mise en œuvre dans le cadre de nouveaux développements car cette fonction sera supprimée dans une prochaine version de SQL Server. L'option Net Send n'est pas utile car elle dépend du service Messenger de Microsoft Windows®. Ce service est généralement désactivé sur les systèmes.

Il est possible de définir un opérateur de prévention de défaillance pour répondre à une alerte si les notifications de radiomessagerie à d'autres opérateurs échouent en raison des limitations de temps qui ont été configurées. Par exemple, si aucun des opérateurs spécifiés n'est de service au moment du déclenchement d'une alerte, c'est l'opérateur de prévention de défaillance qui sera contacté.

Les utilisateurs et les groupes Active Directory® existants peuvent être utilisés comme groupes d'opérateurs si ce sont des groupes à extension messagerie.

Démonstration 3A : Configuration des opérateurs de SQL Server Agent

Dans cette démonstration, vous allez apprendre à

- Comment créer un opérateur ?
- Comment utiliser un opérateur dans un travail de SQL Server Agent ?

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**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_15_PRJ\22462A_15_PRJ.ssmsln** 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.

Vue d'ensemble des alertes SQL Server

Une alerte est une réponse prédefinie à un événement

- Les alertes peuvent être déclenchées par
 - Événements SQL Server journalisés
 - Conditions de performances SQL Server
 - Événements WMI
- Les alertes peuvent
 - Notifier un opérateur
 - Démarrer un travail

Points clés

Il existe de nombreux événements susceptibles de se produire dans un système SQL Server qui peuvent être intéressants pour les administrateurs. Une alerte est un objet SQL Server qui définit une condition qui requiert une attention, et une réponse qui doit être initiée lorsque l'événement se produit. Vous pouvez définir des alertes pour exécuter un travail ou pour avertir un opérateur lorsqu'un événement donné se produit, ou même quand un seuil de performances est franchi.

Alertes SQL Server

Les événements sont générés par SQL Server et entrés dans le journal des événements des applications Windows. Au démarrage, SQL Server Agent s'inscrit en tant que service de rappel avec le journal des événements des applications Windows. Cela signifie que SQL Server Agent sera averti directement par le journal des applications lors de la survenue d'événements pertinents. Ce mécanisme de rappel fonctionne efficacement car il implique que SQL Server Agent n'a pas besoin de lire (ou, plus formellement « d'interroger ») en continu le journal des applications pour rechercher des événements qui vous intéressent.

Lorsque SQL Server Agent est averti d'un événement journalisé, il le compare aux alertes qui ont été définies. Lorsque SQL Server Agent trouve une correspondance, il déclenche une alerte, laquelle est une réponse automatisée à un événement.



Remarque Le message d'erreur doit être consigné dans le journal des applications Windows à utiliser pour les alertes de SQL Server Agent.

Actions d'alertes (réponses)

Vous pouvez créer des alertes pour répondre à des numéros d'erreur individuels ou pour répondre à toutes les erreurs d'un certain niveau de gravité. Vous pouvez définir l'alerte pour toutes les bases de données, ou pour une base de données spécifique. Vous pouvez définir le délai entre les réponses.



Remarque On considère comme recommandé de configurer des notifications pour tous les messages d'erreur dont le niveau de gravité est supérieur ou égal à 19.

Événements système

En plus de surveiller les événements SQL Server, SQL Server Agent peut aussi surveiller les conditions qui peuvent être détectées au moyen des événements WMI (Windows Management Instrumentation). Les requêtes WQL écrites pour récupérer les données de performances sont exécutées plusieurs fois toutes les minutes. Par conséquent, le déclenchement de ces alertes peut prendre quelques secondes.

Les alertes de condition de performances peuvent également être configurées sur n'importe lequel des compteurs de performances qu'offre SQL Server.

Question : quels sont les événements que vous connaissez pour lesquels vous devriez configurer une alerte ?

Création d'une alerte

- Alerte créée avec SSMS ou `sp_add_alert`
- Définissez la détection de l'alerte et l'action à suivre

```
EXEC msdb.dbo.sp_add_alert
@name=N'Adventureworks Transaction Log Full',
@message_id=9002, @delay_between_responses=0,
@database_name=N'Adventureworks',
GO
```

Points clés

Les alertes sont créées à l'aide de l'interface utilisateur graphique de SSMS ou en appelant la procédure stockée système `dbo.sp_add_alert`. En définissant une alerte, vous pouvez également spécifier un travail de SQL Server Agent qui doit être démarré à son déclenchement. Dans l'exemple de la diapositive, l'ID d'un travail de SQL Server Agent qui a déjà été créé a été ajouté à la définition de l'alerte. Ce travail démarrera au déclenchement de l'alerte.

L'action que SQL Server Agent effectue en réponse à l'événement ou à la condition de performances peut inclure l'envoi d'une notification à un ou plusieurs opérateurs.

Événements enregistrés

Vous avez vu que l'alerte ne se déclenche en cas d'erreurs de SQL Server que si les messages d'erreur ont été écrits dans le journal des événements des applications Microsoft Windows. En général les niveaux de gravité d'erreur compris entre 19 et 25 sont automatiquement écrits dans le journal des applications, mais ce n'est pas toujours le cas. Pour vérifier quels messages sont automatiquement écrits dans le journal, consultez la colonne `is_event_logged` dans la table `sys.messages`.

La plupart des événements dont le niveau de gravité est inférieur à 19 ne déclencheront d'alertes que si vous avez appliqué l'une des options suivantes :

- le message d'erreur a été modifié à l'aide de la procédure stockée système `dbo.sp_altermessage` pour en faire un message journalisé ;
- l'erreur a été générée en code au moyen de l'option `RAISERROR WITH LOG` ;
- la procédure stockée système étendue `xp_logevent` a été utilisée pour forcer les entrées à s'écrire dans le journal.

Question : quel type d'alerte est nécessaire pour surveiller l'espace libre dans le système de fichiers ?

Configuration des actions d'alerte

- Les travaux peuvent être démarrés
- Les travaux peuvent utiliser les détails de l'étape de travail à l'aide de jetons
- Les opérateurs peuvent être notifiés



```
EXEC msdb.dbo.sp_add_notification  
@alert_name  
= N'AdventureWorks Transaction Log Full',  
@operator_name=N'SQL Admins',  
@notification_method = 1;  
GO
```

Points clés

Lorsqu'une alerte se déclenche, il existe deux mesures qui peuvent être configurées pour y répondre :

- exécuter un travail ;
- avertir les opérateurs.

Exécuter un travail

L'exécution d'un travail SQL Server Agent peut être configurée comme réponse à une alerte. Un seul travail peut être lancé. Toutefois, si vous devez démarrer plusieurs travaux lors d'une alerte, créez un nouveau travail qui exécute ces autres travaux puis configurez-le comme réponse à l'alerte.

Le travail à exécuter peut être configuré lors de la création de l'alerte au moyen de la procédure dbo.sp_add_alert, ou en appelant la procédure stockée système dbo.sp_update_alert après sa création.

Avertir les opérateurs

Contrairement à la configuration d'un travail pour qu'il s'exécute dans le cadre d'une alerte, la liste des opérateurs à avertir lors du déclenchement d'une alerte est définie à l'aide de la procédure stockée système dbo.sp_add_notification.

Lorsque vous envoyez aux opérateurs un message relatif à une alerte, il est important de pouvoir leur fournir suffisamment de contexte pour qu'ils puissent déterminer l'action à entreprendre. Il est possible d'inclure des jetons dans les messages pour y ajouter des informations. Les jetons spéciaux disponibles pour les alertes sont répertoriés dans le tableau suivant :

Jeton	Description
A-DBN	Nom de la base de données
A-SVR	Nom du serveur
A-ERR	Numéro de l'erreur
A-SEV	Gravité de l'erreur
A-MSG	Message d'erreur

Notez que pour des raisons de sécurité cette fonctionnalité est désactivée par défaut et peut être activée dans les propriétés de SQL Server Agent.

Question : si des notifications doivent être envoyées à une adresse de radiomessagerie, qu'est-ce-qui doit être configuré ?

Résolution des alertes et des notifications

1. Vérifier que SQL Server Agent est en cours de exécution
2. Vérifier que le message d'erreur est consigné dans le journal des applications
 - Vérifiez la configuration du journal des applications
3. Vérifiez que l'alerte est activée
4. Vérifiez que l'alerte a été déclenchée (dernier déclenchement)
 - Vérifiez que le délai entre les réponses n'est pas trop élevée
5. Si l'alerte est générée, mais qu'aucune action n'a été effectuée
 - Vérifiez le travail
 - Vérifiez la messagerie de base de données et la configuration du serveur SMTP
 - Testez le profil de messagerie de base de données utilisé dans SSMS

Points clés

Lors de la résolution des alertes et des notifications, appliquez la procédure suivante pour identifier les problèmes :

Étape	Description
Vérifier que SQL Server Agent est en cours de exécution.	Le journal des applications n'enverra les messages de SQL Server Agent que lorsque celui-ci sera en exécution. Le journal des applications ne maintient pas une file d'attente des notifications à effectuer ultérieurement.
Vérifier que le message d'erreur est consigné dans le journal des applications.	Pour les alertes d'événements SQL Server, vérifiez que le message d'erreur est écrit dans le journal des applications et vérifiez également que le journal des applications est configuré avec une taille suffisante pour contenir toutes les données détaillées du journal des événements.
Vérifier que l'alerte est activée.	Les alertes peuvent être activées ou désactivées et ne se déclenchent pas si elles sont désactivées.

(suite)

Étape	Description
Vérifier que l'alerte a été déclenchée.	Si l'alerte ne semble pas être générée, vérifiez que le paramètre du délai entre les réponses n'est pas fixé à une valeur trop élevée.
Si l'alerte est générée, mais qu'aucune action n'a été effectuée.	Vérifiez que le travail configuré pour répondre à l'alerte fonctionne comme prévu. Pour les notifications de l'opérateur, vérifiez que la messagerie de base de données fonctionne et que la configuration du serveur SMTP est correcte. Testez le profil de messagerie de base de données utilisé pour envoyer les notifications en envoyant manuellement un courrier à partir du profil utilisé par SQL Server Agent.

Question : pourquoi un message d'erreur ne pourrait-il pas être consigné dans le journal des applications ?

Démonstration 3B : Configuration des alertes et des notifications

Dans cette démonstration, vous allez apprendre à créer une alerte pour envoyer une notification lorsqu'un journal des transactions est saturé

Procédure de démonstration

1. Si la démonstration 1A 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**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_15_PRJ\22462A_15_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 le fichier de script **11 – Demonstration 1A.sql** et suivez les instructions de ce fichier.
 - Ouvrez le fichier de script **31 – Demonstration 3A.sql** et suivez les instructions de ce fichier.
2. Ouvrez le fichier de script **32 – Demonstration 3B.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

Atelier pratique 15 : Surveillance des travaux de l'agent SQL avec alertes et notifications

- Exercice 1 : Configurer la messagerie de base de données
- Exercice 2 : Implémenter des notifications
- Exercice difficile 3 : Implémentez des alertes
(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_15_PRJ\22462A_15_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

Vous avez configuré les tâches de gestion automatisées à l'aide de SQL Server Agent et configuré la sécurité pour ces tâches. Vous devez maintenant configurer des alertes et des notifications pour votre système Proseware. L'équipe d'assistance informatique chez AdventureWorks utilise une stratégie d'escalade définie pour les systèmes SQL Server. Comme Proseware fait partie de l'ensemble d'entreprises détenues par AdventureWorks, vous devez implémenter les éléments pertinents de cette stratégie.

L'équipe vous a communiqué les informations relatives à la stratégie qu'elle juge nécessaire pour le serveur Proseware. Pour certaines tâches automatiques, les notifications doivent être envoyées chaque fois que les tâches sont terminées, qu'elles aient fonctionné ou non comme prévu. Pour les autres tâches, les notifications doivent être envoyées seulement si la tâche échoue pour une raison quelconque.

Les notifications chez AdventureWorks s'effectuent avec la radiomessagerie. Vous devez configurer la messagerie de base de données dans SQL Server afin que SQL Server Agent puisse envoyer des messages de notification sur le système de radiomessagerie. L'équipe de support technique informatique de AdventureWorks a alloué à votre système deux administrateurs de base de données joignables. Vous devez configurer ces membres du personnel comme opérateurs en fonction de leur emploi du temps d'astreinte en cours et configurer également un opérateur de prévention de défaillance pour les périodes où aucun membre de l'équipe n'est en service.

Si vous avez suffisamment de temps, configurez également SQL Server pour vous alerter en cas d'erreurs graves sur le serveur.

Documentation

Paramètres de configuration de la messagerie de base de données

Nom du profil : Profil SQL Server Agent Proseware.

Compte SMTP	Élément	Valeur
Principal	Nom du compte	Administrateur Proseware
	Adresse de messagerie	prosewaresqladmin@adventureworks.com
	Nom d'affichage	Administrateur SQL Server de Proseware
	Répondre au courrier	prosewaresqladmin@adventureworks.com
Prévention de défaillance	Nom du serveur	mailserver.adventureworks.com
	Nom du compte	Administrateur AdventureWorks
	Adresse de messagerie	adventureworkssqladmin@adventureworks.com
	Nom d'affichage	Administrateur SQL Server AdventureWorks
	Répondre au courrier	adventureworkssqladmin@adventureworks.com
	Nom du serveur	mailserver.adventureworks.com

Profils publics : configurer le profil SQL Server Agent Proseware comme public et comme profil par défaut.

Profils privés : configurer le profil de SQL Server Agent comme profil par défaut pour le service SQL Server Agent (AdventureWorks\PWService).

Taille de fichier maximale de messagerie électronique : 4 Mo.

Spécifications relatives à l'opérateur de base de données sur appel

- L'administrateur de base de données senior Jeff Hay peut être contacté par radiomessagerie à l'adresse jeff.hay.page@adventureworks.com, 24 heures sur 24 et sept jours sur sept.
- L'administrateur de base de données Palle Petersen est joignable par radiomessagerie à l'adresse palle.petersen.page@adventureworks.com, 24 heures sur 24 et sept jours sur sept.
- Même s'il doit toujours y avoir administrateur joignable sur appel, il convient de configurer une adresse de radiomessagerie de prévention de défaillance, itsupport.page@adventureworks.com. L'opérateur d'assistance informatique doit également être disponible 24 heures sur 24 et sept jours sur sept.

Spécifications relatives à la notification des travaux

- Les travaux liés aux sauvegardes doivent envoyer des notifications à l'achèvement, et pas seulement en cas d'échec. Les notifications des travaux liés aux sauvegardes doivent être envoyées à Jeff Hay.
- Les travaux système n'ont pas besoin d'envoyer de notification. Les travaux système se reconnaissent au préfixe sys.
- Tous les autres travaux doivent envoyer une notification seulement en cas d'échec. Les notifications des autres travaux doivent être envoyées à Palle Petersen.

Spécifications relatives aux alertes d'erreur grave

- Les notifications d'erreurs de gravité 17 ou 18 doivent être envoyées à tous les opérateurs par radiomessagerie.
- L'erreur 9002 sur n'importe quelle base de données doit être envoyée à tous les opérateurs par radiomessagerie.

Exercice 1 : Configurer la messagerie de base de données

Scénario

Les notifications chez AdventureWorks s'effectuent avec la radiomessagerie. Vous devez configurer la messagerie de base de données dans SQL Server afin que SQL Server Agent puisse envoyer des messages de notification sur le système de radiomessagerie.

Les tâches principales de cet exercice sont les suivantes :

1. Configurer la messagerie de base de données.
2. Tester le fonctionnement de la messagerie de base de données.

► Tâche 1 : Configurer la messagerie de base de données

- Vérifiez les paramètres de configuration de la messagerie de base de données dans la documentation associée à l'exercice.
- Configurez la messagerie de base de données conformément aux paramètres indiqués.

► Tâche 2 : Tester le fonctionnement de la messagerie de base de données

- Envoyez un courrier électronique de test à l'aide de l'option de menu contextuel (bouton droit) sur le nœud de la messagerie de base de données dans l'Explorateur d'objets.
- Dans l'explorateur de solutions, ouvrez et exécutez le fichier de script 51 – Lab Exercice 1.sql pour afficher les éléments de courrier sortant.

Résultats : À la fin de cet exercice, vous devez avoir configuré et testé la messagerie de base de données.

Exercice 2 : Implémenter des notifications

Scénario

L'équipe d'assistance informatique chez AdventureWorks utilise une stratégie d'escalade définie pour les systèmes SQL Server. Comme Proseware fait partie de l'ensemble d'entreprises détenues par AdventureWorks, vous devez implémenter les éléments pertinents de cette stratégie.

L'équipe vous a communiqué les informations relatives à la stratégie qu'elle juge nécessaire pour le serveur Proseware. Pour certaines tâches automatiques, les notifications doivent être envoyées à chaque fois que les tâches sont terminées, qu'elles aient fonctionné ou non comme prévu. Pour les autres tâches, les notifications doivent être envoyées seulement si la tâche échoue pour une raison quelconque.

Les notifications chez AdventureWorks s'effectuent avec la radiomessagerie. L'équipe de support technique informatique de AdventureWorks a alloué à votre système deux administrateurs de base de données joignables. Vous devez configurer ces membres du personnel comme opérateurs en fonction de leur emploi du temps d'astreinte en cours et configurer également un opérateur de prévention de défaillance pour les périodes où aucun membre de l'équipe n'est en service.

Les tâches principales de cet exercice sont les suivantes :

1. Examiner la configuration requise.
2. Configurer les opérateurs requis.
3. Configurer la messagerie SQL Server Agent.
4. Configurer et tester les notifications dans les travaux de SQL Server Agent.

► Tâche 1 : Examiner la configuration requise

- Prenez connaissance des éléments requis fournis dans la documentation associée de l'exercice. En particulier, notez tous les opérateurs requis.

► Tâche 2 : Configurer les opérateurs requis

- Configurez les opérateurs dont vous avez déterminé qu'ils étaient requis dans la tâche 1. La documentation comporte des informations sur la façon dont les opérateurs doivent être configurés.

► Tâche 3 : Configurer la messagerie SQL Server Agent

- Configurez SQL Server Agent pour utiliser le profil de messagerie créé dans l'exercice 1.
- Configurez SQL Server Agent pour utiliser l'opérateur de prévention de défaillance de l'assistance informatique que vous avez configuré dans la tâche 2.

► Tâche 4 : Configurer et tester les notifications dans les travaux de SQL Server Agent

- Configurez les notifications des travaux conformément aux spécifications de la documentation.
- Testez les notifications en exécutant tous les travaux non système et en examinant l'élément de courrier envoyé.

Résultats : À la fin de cet exercice, vous devez avoir configuré les opérateurs de SQL Server Agent et les notifications des travaux.

Exercice difficile 3 : Implémenter des alertes (si le temps le permet)

Scénario

Si vous avez suffisamment de temps, configurez également SQL Server pour vous alerter en cas d'erreurs graves sur le serveur.

La tâche principale de cet exercice est la suivante :

1. Configurer et tester les alertes.

► Tâche 1 : Configurer et tester les alertes

- Examinez la documentation associée pour prendre connaissance des spécifications d'alertes.
- Configurez les alertes requises.
- Exécutez le script 71 – Lab Exercise 3.sql pour tester la fonctionnalité d'alerte.



Remarque Le script retourne l'erreur 9002.

Résultats : À la fin de cet exercice, vous devez avoir configuré et testé les fichiers les alertes SQL Server.

Contrôle des acquis du module et éléments à retenir

- Questions de contrôle des acquis
- Meilleures pratiques

Questions de contrôle des acquis

1. Qu'est-ce qu'un opérateur dans la terminologie de SQL Server Agent ?
2. Quelle est le niveau de gravité d'erreur le plus bas qui s'affiche comme message d'erreur dans SSMS ?

Meilleures pratiques

1. Utiliser la messagerie de base de données et non SQL Mail.
2. Configurer différents profils pour différents scénarios d'utilisation.
3. Fournir un accès limité à la fonction d'envoi des messages électroniques à partir du moteur de base de données.
4. Implémenter une stratégie de rétention pour le journal de la messagerie de base de données et les audits du courrier électronique.
5. Créer des opérateurs à qui envoyer des notifications sur les travaux et les alertes.
6. Définir des alertes pour les messages d'erreur grave.

Module 16

Exécution de la maintenance régulière des bases de données

Sommaire :

Leçon 1 : Vérification de l'intégrité de la base de données	16-3
Leçon 2 : Gestion des index	16-13
Leçon 3 : Automatisation de la maintenance régulière des bases de données	16-27
Atelier pratique 16 : Exécution de la maintenance régulière des bases de données	16-31

Vue d'ensemble du module

- Vérification de l'intégrité de la base de données
- Gestion des index
- Automatisation de la maintenance régulière des bases de données

Le moteur de base de données Microsoft® SQL Server® est tout à fait capable de fonctionner indéfiniment sans aucune maintenance courante. Cependant, l'obtention des meilleurs résultats implique la planification d'opérations de maintenance régulière.

L'altération de la base de données est un évènement relativement rare ; cependant, l'une des tâches les plus importantes de la maintenance consiste à vérifier qu'aucune altération ne s'y est produite. La récupération à partir d'une altération dépend de la détection rapide de celle-ci.

Les index SQL Server peuvent continuer à fonctionner sans maintenance, mais leur performance sera meilleure en éliminant régulièrement la fragmentation qui y apparaît.

SQL Server comprend un assistant Plan de maintenance pour aider à créer des travaux de SQL Server Agent qui effectuent cette action ainsi que d'autres tâches de maintenance courantes.

Objectifs

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

- vérifier l'intégrité de la base de données ;
- entretenir les index ;
- automatiser la maintenance régulière des bases de données.

Leçon 1

Vérification de l'intégrité de la base de données

- Discussion : Vérification de l'intégrité de la base de données
- Vue d'ensemble de DBCC CHECKDB
- Options DBCC CHECKDB
- Options de réparation DBCC CHECKDB
- Démonstration 1A : Utilisation de DBCC CHECKDB

Il est extrêmement rare que le moteur de base de données provoque l'altération directement. Toutefois, le moteur de base de données dépend de la plateforme matérielle sur laquelle il s'exécute, et cette plateforme peut être la cause de l'altération. En particulier, les problèmes de mémoire et des sous-systèmes d'E/S peuvent entraîner l'altération dans les bases de données.

Si vous ne détectez pas l'altération rapidement après sa survenue, d'autres problèmes sont susceptibles d'apparaître, significativement plus complexes ou fastidieux. Par exemple, il y a peu d'intérêt à récupérer une base de données endommagée à partir d'un jeu de sauvegardes dont chacune contient une copie altérée de la base de données.

La commande DBCC CHECKDB peut être utilisée pour détecter et, dans certaine circonstances, corriger une altération de bases de données. Il est important que vous soyez familiarisé avec l'utilisation de DBCC CHECKDB.

Objectifs

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

- utiliser DBCC CHECKDB ;
- expliquer les options de DBCC CHECKDB les plus courantes ;
- expliquer comment utiliser les options de réparation de DBCC CHECKDB.

Discussion : Vérification de l'intégrité de la base de données

- Qu'est-ce que l'intégrité d'une base de données ?
- Quelles sont les techniques que vous utilisez actuellement pour vérifier et gérer l'intégrité de la base de données ?



Rubriques de discussion

Question : qu'est-ce que l'intégrité d'une base de données ?

Question : quelles sont les techniques que vous utilisez actuellement pour activer et gérer l'intégrité de la base de données ?

Vue d'ensemble de DBCC CHECKDB

- Vérifie l'intégrité logique et physique dans la base de données
 - Allocation de toutes les pages dans la base de données
 - Cohérence des tables et des index
 - Cohérence du catalogue de la base de données
 - Cohérence au niveau des liens pour les objets FILESTREAM
 - Objets Service Broker
- :Offre des options de réparation
 - Certaines options autorisent la perte de données
- Exécution en ligne avec une capture instantanée de la base de données interne
- Doit être exécuté fréquemment
 - Synchronise les exécutions selon votre stratégie de sauvegarde, pour pouvoir récupérer les données dans les cas d'altération

Points clés

DBCC est un utilitaire qui est fourni avec SQL Server et qui offre un grand nombre de fonctionnalités de gestion. Dans la documentation antérieure, vous pouvez le trouver sous le nom de vérificateur de la cohérence de la base de données. Bien que son usage essentiel consiste à vérifier la cohérence des bases de données à l'aide de l'option CHECKDB, DBCC possède de nombreuses autres utilisations. Dans les versions actuelles du produit, il est connu sous le nom de l'utilitaire de commandes de console de base de données afin de mieux refléter la grande variété des tâches pour lesquelles il peut être utilisé.

DBCC CHECKDB

L'option CHECKDB de l'utilitaire DBCC effectue un contrôle particulièrement approfondi de la structure d'une base de données afin de détecter presque toutes les formes d'altération potentielle. Les fonctions contenues dans la commande DBCC CHECKDB sont également disponibles en tant qu'options pouvant être exécutées indépendamment si nécessaire.

Les plus importantes de ces options sont indiquées dans le tableau suivant :

Option	Description
DBCC CHECKALLOC	Vérifie la cohérence des structures d'allocation de l'espace disque pour une base de données spécifiée.
DBCC CHECKTABLE	Vérifie les pages associées à une table spécifiée et les pointeurs entre les pages qui sont associées à la table. DBCC CHECKDB exécute DBCC CHECKTABLE pour chaque table de la base de données.
DBCC CHECKCATALOG	Vérifie le catalogue de la base de données en effectuant des vérifications de cohérence logique dans les tables de métadonnées de la base de données. Ces tables de métadonnées servent à contenir les informations qui décrivent le système et les tables utilisateur ainsi que d'autres objets de base de données. DBCC CHECKCATALOG ne vérifie pas les tables utilisateur.

DBCC CHECKDB effectue également des contrôles sur d'autres types d'objets tels que les liens des objets FILESTREAM et des vérifications de cohérence sur les objets Service Broker.



Remarque FILESTREAM et Service Broker sont des rubriques avancées qui n'entrent pas dans le cadre de ce cours.

Options de réparation

Bien que DBCC CHECKDB propose des options de réparation, il n'est pas toujours possible de réparer une base de données sans perte de données. Habituellement, la meilleure solution pour récupérer la base de données consiste à la restaurer. Cela signifie que l'exécution de DBCC CHECKDB doit être synchronisée avec votre stratégie de rétention de sauvegarde, afin que vous puissiez toujours restaurer une base de données à partir d'une base de données non altérée et afin que toutes les sauvegardes de journaux nécessaires soient disponibles depuis ce moment.

Opérations simultanées en ligne

DBCC CHECKDB peut prendre beaucoup de temps pour s'exécuter et consomme une grande quantité de ressources d'E/S et de processeur. Pour cette raison, les administrateurs doivent souvent les exécuter alors que la base de données est en cours d'utilisation.

Dans les versions de SQL Server antérieures à SQL Server 2005, lorsque DBCC CHECKDB était exécuté pendant que la base de données était utilisée simultanément par d'autres utilisateurs, il était possible de recevoir des indications d'altération même s'il n'y en avait aucune. Depuis SQL Server 2005, DBCC CHECKDB s'exécute sur des captures instantanées de base de données internes pour garantir qu'il s'applique à une vue cohérente de la base de données. Si DBCC CHECKDB signale une altération, celle-ci doit être étudiée.

Si les besoins de performances des activités de base de données sont trop gourmands pour pouvoir exécuter DBCC CHECKDB simultanément, il est préférable d'exécuter DBCC CHECKDB sur une sauvegarde restaurée de la base de données que de ne pas l'exécuter du tout (même si cette solution est loin d'être idéale).

Espace disque

L'utilisation d'une capture instantanée interne a pour effet que DBCC CHECKDB a besoin d'espace disque supplémentaire. DBCC CHECKDB crée des fichiers masqués (à l'aide des flux de remplacement NTFS) sur les mêmes volumes qui contiennent les fichiers de la base de données. Il doit y avoir suffisamment d'espace libre sur les volumes pour que DBCC CHECKDB s'exécute correctement. La quantité d'espace disque nécessaire sur les volumes dépend de la quantité de données qui est modifiée durant l'exécution de DBCC CHECKDB.

DBCC CHECKDB utilise également de l'espace dans tempdb lors de son exécution. Pour fournir une estimation de la quantité d'espace nécessaire dans tempdb, DBCC CHECKDB propose l'option ESTIMATEONLY.

Sauvegardes et DBCC CHECKDB

Il est recommandé d'exécuter DBCC CHECKDB sur une base de données avant d'en réaliser la sauvegarde. Ce contrôle permet de garantir que la sauvegarde contient une version cohérente de la base de données.

Question : pourquoi est-il essentiel d'exécuter DBCC CHECKDB régulièrement ?

Options DBCC CHECKDB

Option	Description
PHYSICAL_ONLY	Vérifie uniquement l'intégrité physique avec moins de charge
NOINDEX	N'exécute pas les contrôles logiques sur des index non-cluster
EXTENDED_LOGICAL_CHECKS	Effectue des contrôles logiques supplémentaires des vues indexées, des index spatiaux et XML
TABLOCK	Utilise les verrous à la place des captures instantanées de base de données
ALL_ERRORMSG	Retourne tous les messages d'erreur contrairement à l'action par défaut qui retourne les 200 premiers messages
NO_INFOMSGS	Retourne uniquement les messages d'erreur et aucun message d'information
ESTIMATEONLY	Détermine le volume d'espace tempdb requis pour l'exécution

Points clés

DBCC CHECKDB offre plusieurs options qui modifient son comportement à l'exécution.

- L'option PHYSICAL_ONLY est souvent utilisée sur les systèmes de production car elle réduit considérablement le temps nécessaire pour exécuter DBCC CHECKDB sur des bases de données volumineuses. Si vous utilisez régulièrement l'option PHYSICAL_ONLY, vous devez quand même exécuter régulièrement la version complète de l'utilitaire. La fréquence à laquelle vous exécutez la version complète dépend des besoins spécifiques de l'activité.
- L'option NOINDEX indique qu'il ne faut pas exécuter de contrôle intensif des index non cluster pour les tables utilisateur. Elle réduit également la durée totale d'exécution, mais n'affecte pas les tables système car les contrôles d'intégrité sont toujours effectués sur les index des tables système. Quand vous utilisez l'option NOINDEX, vous faites l'hypothèse que vous pouvez reconstruire les index non cluster s'ils sont endommagés.
- L'option EXTENDED_LOGICAL_CHECKS ne peut être exécutée que lorsque la base de données est dans le niveau de compatibilité de la base de données 100 (SQL Server 2008) ou supérieur. Elle consiste à effectuer des contrôles détaillés de la structure interne des objets tels que les types de données CLR définis par l'utilisateur et les types de données spatiales.
- L'option TABLOCK est utilisée pour faire en sorte que DBCC CHECKDB mette un verrou de table sur chacune lors des vérifications de cohérence, au lieu d'utiliser les captures instantanées internes de la base de données. Cela réduit l'espace disque nécessaire, avec comme contrepartie le fait d'empêcher d'autres utilisateurs de mettre les tables à jour.

- Les options ALL_ERRORMSG et NO_INFOMSGS n'affectent que la sortie de la commande, pas les opérations effectuées par la commande.
- L'option ESTIMATEONLY permet d'estimer l'espace nécessaire dans tempdb, comme décrit dans la rubrique précédente.

Question : quelle option DBCC CHECKDB peut être utilisée sur des systèmes en production de grande taille ?

Options de réparation DBCC CHECKDB

- La base de données doit être en mode SINGLE_USER
- Le résultat de l'option DBCC CHECKDB indique l'option minimum pour la récupération
 - REPAIR_REBUILD pour les réparations qui peuvent être effectuées sans perte de données
 - REPAIR_ALLOW_DATA_LOSS implique la perte de données
- Pensez à restaurer une base de données au lieu d'autoriser la perte de données

```
DBCC CHECKDB('ReportServer$MKTGTempDB') WITH ALL_ERRORMSG;
Résultats DBCC pour 'sys FileStream_tombstone_2073058421'.
Il y a 0 lignes dans 0 pages pour l'objet "sys FileStream_tombstone_2073058421".
Résultats DBCC pour 'sys.syscommittab'.
Il y a 0 lignes dans 0 pages pour l'objet "sys.syscommittab".
Résultats DBCC pour 'sys.sysavailable_updates_2105058535'.
Il y a 0 lignes dans 0 pages pour l'objet "sys.sysavailable_updates_2105058535".
CHECKDB a trouvé 0 erreurs d'allocation et 0 erreurs de cohérence dans la base de données 'ReportServer$MKTGTempDB'.
Exécution de DBCC terminée. Si DBCC vous a adressé des messages d'erreur, contactez l'administrateur système.
```

Requête terminée avec des erreurs.

Points clés

Tout en donnant des informations détaillées sur les erreurs trouvées, la sortie de DBCC CHECKDB indique l'option de réparation qui serait nécessaire pour corriger le problème. Dans l'exemple de la diapositive, quatre erreurs de cohérence ont été trouvées et l'option REPAIR_ALLOW_DATA_LOSS est requise pour réparer la base de données.

Options de réparation

DBCC CHECKDB propose deux options de réparation. Pour les deux options, la base de données doit être en mode mono-utilisateur. Ces options sont les suivantes :

- REPAIR_REBUILD reconstruit les index. Les pages de données endommagées sont supprimées. Cette option ne fonctionne que pour certaines formes légères d'altération et n'implique pas de perte de données.
- REPAIR_ALLOW_DATA_LOSS entraîne presque toujours une perte de données. Elle libère les pages endommagées et modifie les autres pages qui y font référence. Une fois cette opération terminée, la base de données est cohérente, mais uniquement du point de vue de l'intégrité de la base de données physique. Il peut s'entreproduire une perte de données significative. Les opérations de réparation ne prennent en compte aucune des contraintes susceptibles d'exister sur des tables ou entre elles. Si la table spécifiée est impliquée dans une ou plusieurs contraintes, il est recommandé d'exécuter DBCC CHECKCONSTRAINTS après l'opération de réparation.

L'utilisation de DBCC CHECKDB est présentée dans la démonstration 1A.

Il est fortement recommandé de sauvegarder une base de données avant d'y appliquer toute option de réparation. La réparation d'une base de données doit être une option de dernier recours. Lorsqu'une base de données est endommagée, il est généralement préférable de restaurer la base de données à partir d'une sauvegarde après avoir résolu la cause de l'altération. L'altération peut très bien se produire à nouveau à moins que vous n'en trouviez et résolviez la cause. L'altération dans les bases de données SQL Server est principalement causée par des échecs dans les sous-systèmes d'E/S ou de mémoire.

Altération du journal des transactions

Si le journal des transactions est endommagé, il est possible d'utiliser une option spéciale appelée réparation en mode urgence, mais dans cette situation il est fortement recommandé de restaurer la base de données. Une réparation en mode urgence ne doit être tentée que si aucune sauvegarde n'est disponible.

Question : pourquoi est-il préférable de restaurer une base de données au lieu d'utiliser l'option REPAIR_ALLOW_DATA_LOSS ?

Démonstration 1A : Utilisation de DBCC CHECKDB

Dans cette démonstration, vous allez apprendre à utiliser les diverses options de la commande DBCC CHECKDB

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**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_16_PRJ\22462A_16_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

Gestion des index

- Comment les index affectent les performances
- Types d'index SQL Server
- Fragmentation des index
- FILLFACTOR et PAD_INDEX
- Maintenance continue des index
- Opérations d'index en ligne
- Mise à jour des statistiques
- Démonstration 2A : Gestion des index

Un autre point important de SQL Server qui requiert une maintenance courante pour assurer des performances optimales est la gestion des index. Les index sont utilisés pour accélérer les opérations lorsque SQL Server doit accéder à des données dans une table. Au fil du temps, les index peuvent être fragmentés et les performances des applications de base de données qui les utilisent peuvent être amoindries. La défragmentation ou la reconstruction des index a pour effet de rétablir les performances de la base de données.

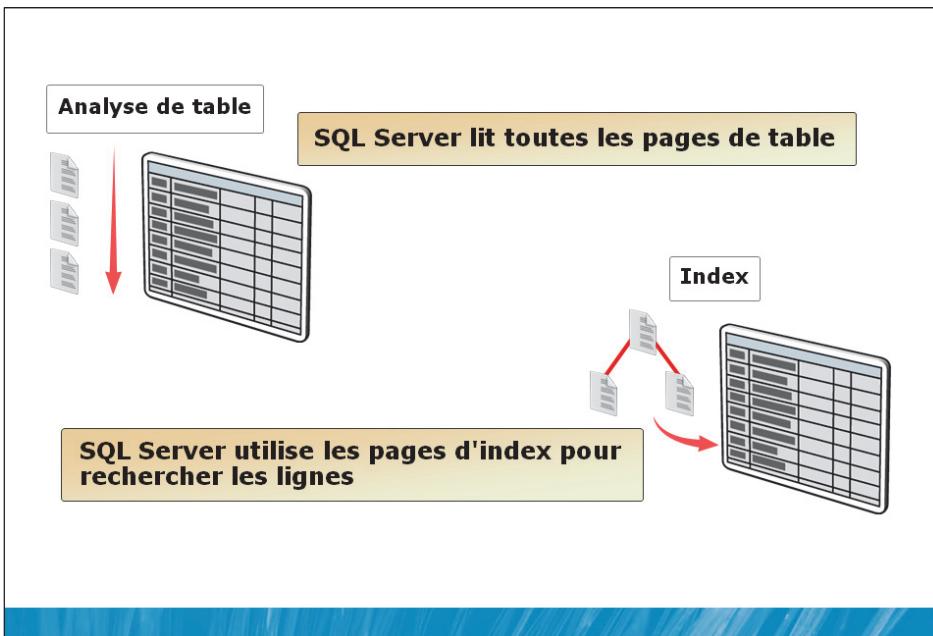
Les options de gestion des index sont souvent incluses dans les plans réguliers de maintenance des bases de données. Avant d'apprendre à configurer les plans de maintenance, il est important de mieux comprendre la façon dont les index sont exécutés ainsi que la façon dont ils sont conservés.

Objectifs

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

- décrire comment les index affectent les performances ;
- décrire les différents types d'index de SQL Server ;
- décrire la manière dont les index sont progressivement fragmentés ;
- utiliser les options FILLFACTOR et PAD_INDEX ;
- expliquer les raisons de la maintenance continue des index ;
- mettre en œuvre les opérations d'index en ligne ;
- décrire comment les statistiques sont créées et utilisées par SQL Server.

Comment les index affectent les performances



Points clés

SQL Server peut accéder aux données contenues dans une table en lisant toutes les pages de celle-ci (opération également appelée analyse de table) ou en utilisant les pages d'index pour rechercher les lignes requises.

Index

Lorsque SQL Server doit accéder à des données contenues dans une table, il décide soit de lire toutes les pages de la table, soit d'utiliser un ou plusieurs index qui existent pour cette table, si cela peut réduire l'effort de recherche des lignes demandées.

Il est toujours possible de répondre aux requêtes en lisant les données sous-jacentes de la table. L'utilisation d'un index n'est pas obligatoire, mais il est nettement plus lent d'accéder aux données en lisant un grand nombre de pages qu'en utilisant les index appropriés.

Les index peuvent aider à améliorer la recherche, le tri et la performance des opérations de jointure, mais ils peuvent influer sur la performance des opérations de modification des données, ils impliquent une maintenance continue et ils nécessitent de l'espace disque supplémentaire.

Occasionnellement, SQL Server crée ses propres index temporaires pour améliorer la performance des requêtes. Cependant, comme cette action relève de l'optimiseur et non du programmeur ou de l'administrateur de base de données, ces index temporaires ne sont pas abordés dans ce module. Les index temporaires ne sont utilisés que pour améliorer un plan de requête, si aucun index approprié n'existe déjà.

Dans ce module, vous étudierez les index standard créés sur des tables. SQL Server comprend d'autres types d'index :

- la recherche en texte intégral intégrée (iFTS) utilise un type spécial d'index qui assure une recherche de texte flexible ;
- les index spatiaux sont utilisés avec les types de données géométrie et géographie ;
- les index XML primaires et secondaires contribuent à l'interrogation de données XML ;
- les index columnstore sont généralement utilisés dans les grands entrepôts de données. Les tables sont essentiellement en lecture seule lorsqu'il existe des index columnstore.

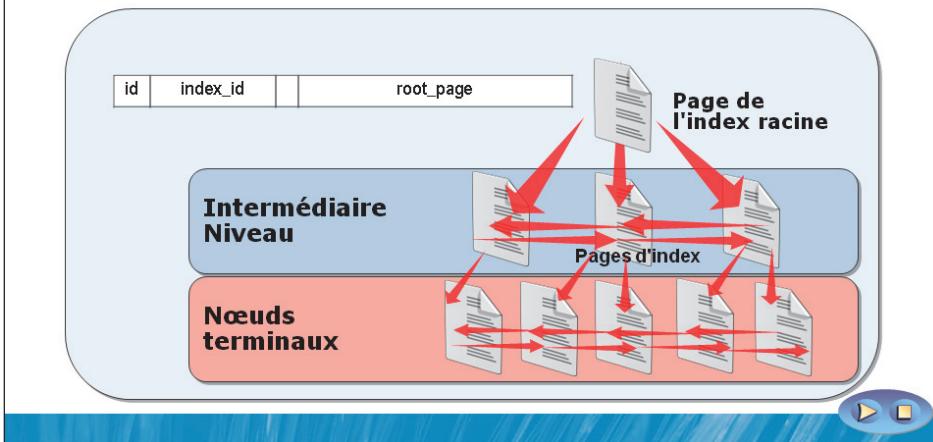


Remarque Les index iFTS, XML et columnstore sortent du cadre de ce cours, mais les index iFTS, spatiaux et XML sont décrits dans le cours 10776A : Developing Microsoft SQL Server 2012 Databases (*en anglais*) ; les index cluster et non cluster y sont également présentés de manière plus approfondie.

Question : quand une analyse de table est-elle plus efficace que l'utilisation d'un index ?

Types d'index SQL Server

- Les index sont organisées comme des arbres B (B-tree)
- L'index cluster possède des pages de données au niveau feuille
- L'index non cluster possède un pointeur vers les lignes de données au niveau feuille



Points clés

Plutôt que d'inscrire les lignes d'une donnée en tant que segment, il est possible de concevoir les tables avec un ordre logique interne. Il s'agit d'un index cluster.

Index cluster

Une table dotée d'un index cluster possède un ordre prédéfini des lignes dans une page et des pages au sein de la table. Cet ordre est fondé sur une clé composée d'une ou plusieurs colonnes. On appelle communément cette clé une clé de clustering.

Comme les lignes d'une table ne peuvent être que dans un seul ordre, il ne peut exister qu'un seul index cluster sur une table. Une entrée IAM (Index Allocation Map) sert à pointer sur un index cluster. Les index cluster ont toujours un ID d'index = 1.

Il existe une idée fausse très répandue qui consiste à penser que les index cluster « sont stockés physiquement dans l'ordre ». Bien que cela soit possible dans de rares situations, ce n'est généralement pas le cas. Si c'était vrai, il n'y aurait pas de fragmentation des index cluster. SQL Server essaie d'aligner l'ordre physique et l'ordre logique lors de la création d'un index, mais il peut se créer du désordre à mesure que les données sont modifiées.

L'index et les pages de données sont liés selon une hiérarchie logique et également doublement liés sur toutes les pages au même niveau de la hiérarchie afin de faciliter l'analyse d'un index. Par exemple, imaginez une table avec dix étendues et des numéros de page alloués de 201 à 279, toutes liées dans l'ordre (chaque étendue contient huit pages). Si une page devait être placée au milieu de l'ordre logique, SQL Server trouverait une étendue comportant une page libre, ou allouerait une nouvelle étendue pour l'index. Cette page serait liée logiquement dans la bonne position, mais elle pourrait se trouver n'importe où parmi les pages de la base de données.

Index non-cluster

Un index non-cluster est un type d'index qui n'affecte pas la disposition des données dans la table à la manière dont le fait un index cluster.

Si la table sous-jacente est un segment (autrement dit, elle ne comporte pas d'index cluster), le niveau feuille d'un index non cluster contient des pointeurs vers l'endroit où sont stockées les lignes de données.

Les pointeurs incluent un numéro de fichier, un numéro de page, et un numéro d'emplacement dans la page.

Si la table sous-jacente a un index cluster (autrement dit, les pages et les données sont logiquement liées dans l'ordre d'une clé de clustering), le niveau feuille d'un index non-cluster contient la clé de clustering qui permet ensuite de faire une recherche dans les pages de l'index cluster pour localiser les lignes souhaitées.

Fragmentation des index

- Comment se déroule la fragmentation ?
 - SQL Server réorganise les pages d'index lorsque les données sont modifiées et qu'elles entraînent un fractionnement des pages d'index
- Types de fragmentation
 - Interne : les pages ne sont pas complètes
 - Externe : les pages ne relèvent pas d'une séquence logique
- Détection de la fragmentation
 - SQL Server Management Studio – Propriétés des index
 - Fonction système - sys.dm_db_index_physical_stats

Points clés

La fragmentation des index correspond à l'utilisation inefficace des pages au sein d'un index.

La fragmentation se produit au fur et à mesure de la modification des données.

Fragmentation des index

Pour les opérations qui lisent les données, les index offrent les meilleures performances lorsque chaque page de l'index est aussi complète que possible. Même si les index sont complets au démarrage (ou relativement complets), les modifications apportées aux données peuvent provoquer le fractionnement des pages d'index. L'ajout d'un nouvelle d'index à la fin d'un index est simple ; cependant, ce processus est plus compliqué si l'entrée doit être effectuée en milieu de page d'un index complet existant.

Fragmentation interne et fragmentation externe

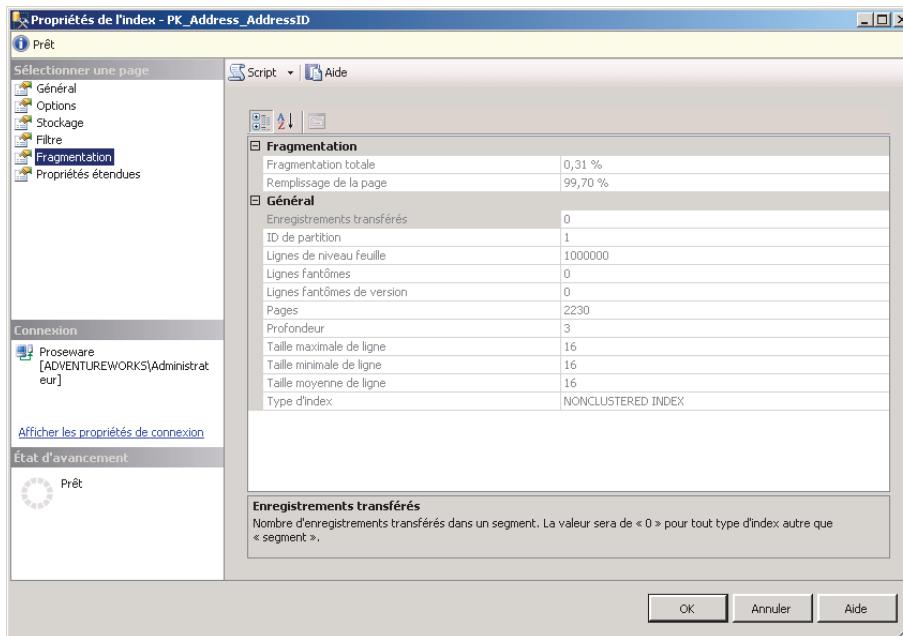
La fragmentation interne se produit lorsque les pages ne contiennent pas autant de données qu'elles le peuvent. Cela se produit souvent lorsque la page est fractionnée pendant une opération d'insertion, et cela peut également se produire lorsqu'une opération de mise à jour entraîne le déplacement d'une ligne vers une autre page. Dans l'un ou l'autre cas, il y a de l'espace laissé vide dans les pages.

La fragmentation externe se produit lorsque des pages qui sont logiquement organisées ne sont pas conservées dans des numéros de page organisés. Si une nouvelle page d'index doit être allouée, elle est logiquement insérée dans l'emplacement approprié dans la liste des pages, mais elle peut aussi bien être placée à la fin de l'index. Cela implique qu'un processus qui doit lire les pages d'index dans l'ordre doit suivre les pointeurs pour localiser les pages ; le processus implique alors d'accéder à des pages non séquentielles dans la base de données.

Détection de la fragmentation

SQL Server fournit une mesure utile dans la colonne avg_fragmentation_in_percent de la vue de gestion dynamique sys.dm_db_index_physical_stats.

SQL Server Management Studio fournit également des informations de fragmentation de l'index dans la page des propriétés pour chaque index, comme le montre la capture d'écran suivante de la base de données AdventureWorks :



Question : pourquoi la fragmentation nuit-elle aux performances ?

FILLFACTOR et PAD_INDEX

- Des espaces libres peuvent être laissés dans les index, y compris les index cluster
 - FILLFACTOR (niveau feuille uniquement)
 - PAD_INDEX (niveaux intermédiaires et racine également)

```
ALTER TABLE Person.Contact
ADD CONSTRAINT PK_Contact_ContactID
PRIMARY KEY CLUSTERED
(
    ContactID ASC
) WITH (PAD_INDEX = OFF, FILLFACTOR = 70);
GO
```

Points clés

Les options FILLFACTOR et PAD_INDEX permettent de libérer de l'espace dans les pages d'index. Cela peut améliorer les performances des commandes INSERT et UPDATE dans certains cas, mais souvent au détriment souvent des opérations SELECT.

FILLFACTOR et PAD_INDEX

L'espace libre disponible dans une page d'index peut influer grandement sur les performances des opérations de mise à jour des index. Si un enregistrement d'index doit être inséré et qu'il n'y a pas d'espace libre, il est nécessaire de créer une nouvelle page d'index et de répartir le contenu de l'ancienne page sur les deux pages. Les performances peuvent s'en trouver pénalisées si cette action se produit trop souvent.

L'impact sur la performance des fractionnements de page peut être atténué en laissant de l'espace vide sur chaque page lors de la création d'un index, notamment un index cluster. Cela s'effectue en spécifiant une valeur de FILLFACTOR. La valeur par défaut de FILLFACTOR est 0, ce qui signifie « remplit à 100 % ». Toute autre valeur (y compris 100) correspond au pourcentage de remplissage souhaité pour chaque page. Si l'on reprend l'exemple de la diapositive, cela signifie complet à 70 % et 30 % d'espace libre sur chaque page.



Remarque La différence entre les valeurs 0 et 100 peut sembler déroutante. Si les deux valeurs aboutissent au même résultat, 100 indique qu'une valeur particulière de FILLFACTOR a été spécifiée. La valeur 0 indique qu'aucune valeur n'a été spécifiée pour le paramètre FILLFACTOR.

FILLCODE s'applique uniquement aux pages de niveau feuille dans un index. PAD_INDEX est une option qui, lorsqu'elle est activée, entraîne l'allocation du même espace libre dans les niveaux non-feuille de l'index.

Question : il est possible d'éviter le fractionnement de plusieurs pages en définissant un FILLCODE de 50, mais quel est l'inconvénient de cette opération ?

Question : quand un FILLCODE de 100 est-il utile ?

Question : pourquoi est-il important d'appliquer un facteur FILLCODE à un index cluster et non à un index non cluster ?

Maintenance continue des index

- Reconstruire
 - Reconstruit l'index tout entier
 - Nécessite de l'espace libre dans la base de données
 - Exécution comme transaction unique avec besoins potentiels d'un très grand espace pour le journal des transactions
- Réorganiser
 - Trie les pages et est toujours en ligne
 - Utilisation moindre du journal des transactions
 - Peut être interrompu, mais conserve toujours le travail effectué jusqu'ici

```
ALTER INDEX CL_LogTime ON dbo.LogTime  
REBUILD;
```

```
ALTER INDEX ALL ON dbo.LogTime  
REORGANIZE;
```

Points clés

Comme les index sont mis à jour lors des modifications de données, ils peuvent être fragmentés au fil du temps. SQL Server fournit deux options pour éliminer la fragmentation des index cluster et non-cluster :

- reconstruire ;
- réorganiser.

Reconstruire

La reconstruction d'un index supprime l'index et le recrée. Cela supprime la fragmentation, libère de l'espace disque en compactant les pages d'après le paramètre spécifié ou existant du facteur de remplissage, et réorganise les lignes d'index en pages contigües. Si l'option ALL est spécifiée, tous les index de la table sont supprimés et reconstruits en une seule opération. Si une partie de l'opération échoue, l'opération entière est annulée.

Étant donné que les reconstructions sont exécutées en une seule opération et sont consignées, une seule opération de reconstruction peut utiliser une grande quantité d'espace dans le journal des transactions. Il est possible d'effectuer l'opération de reconstruction en l'enregistrant de manière minimale lorsque la base de données est dans le mode de récupération simple ou BULK_LOGGED. Une opération de reconstruction enregistrée de façon minimale consomme moins d'espace dans le journal des transactions et s'effectue plus rapidement.

Il doit y avoir de l'espace libre lors de la reconstruction des index.

Réorganiser

La réorganisation d'un index utilise au minimum les ressources système. Elle défragmente le niveau feuille des index cluster et non-cluster des tables : les pages de niveau feuille sont réorganisées physiquement afin de correspondre à l'ordre logique de gauche à droite des nœuds feuille. La réorganisation entraîne aussi une compression des pages d'index. La compression est basée sur la valeur du facteur de remplissage existante. Il est possible d'interrompre une réorganisation sans perdre le travail effectué jusqu'alors. Par exemple, cela implique qu'il est possible de réaliser une réorganisation partielle chaque jour sur un index volumineux.

Pour les index très fragmentés > (30%) la reconstruction est généralement l'option la plus appropriée.

Les plans de maintenance de SQL Server comportent des options de reconstruction ou de réorganisation des index. Si les plans de maintenance ne sont pas utilisés, il est important de créer une tâche qui réalise la défragmentation des index de vos bases de données.

Question : quelle est généralement la meilleure option pour défragmenter des tables et des index volumineux (index cluster) ?

Opérations d'index en ligne

- Les index peuvent être créés, reconstruits et supprimés en ligne
 - Autorise l'accès simultané d'utilisateurs à la table sous-jacente et aux index
 - Nécessite uniquement des verrous partagés à court terme au début et à la fin de l'opération et des verrous de schéma pendant l'opération
- En général, plus lent qu'une opération équivalente en mode déconnecté, mais autorise l'accès utilisateur

```
ALTER INDEX IX_Contact_EmailAddress
ON Person.Contact REBUILD
WITH ( PAD_INDEX = OFF,
       FILLFACTOR = 80,
       ONLINE = ON,
       MAXDOP = 4 );
```

Points clés

Pour la plupart des organisations, le motif d'achat de SQL Server Enterprise Edition est que ces éditions peuvent effectuer des opérations d'index en ligne, alors même que les utilisateurs accèdent à la base de données. Cela est très important parce que de nombreuses organisations n'ont aucune fenêtre de maintenance disponible pendant laquelle effectuer des opérations de maintenance de base de données, telles que la reconstruction d'index.

Quand il réalise une opération en ligne de reconstruction d'index, SQL Server crée un index de mappage temporaire qui suit les modifications de données qui surviennent pendant l'opération de reconstruction. Pour des raisons de cohérence, SQL Server met un verrou partagé de très courte durée sur l'objet au début de l'opération et, une fois encore, à la fin. Pendant l'opération de reconstruction en ligne, des verrous de schéma sont mis en place pour empêcher la modification des métadonnées. Cela signifie que les utilisateurs ne peuvent pas modifier la structure de la table au moyen de commandes telles que ALTER TABLE pendant l'opération en ligne de reconstruction d'index.

En raison du travail supplémentaire impliqué, les opérations en ligne de reconstruction d'index sont généralement plus lentes que leurs équivalents hors connexion.

 **Remarque** Certains index ne peuvent pas être reconstruits en ligne, notamment les index cluster avec des données d'objet volumineuses ou les index non cluster qui contiennent des données d'objet volumineuses.

Question : quand les opérations d'index en ligne sont-elles les plus importantes ?

Mise à jour des statistiques

- Lorsque les données changent, les statistiques de distribution deviennent obsolètes
- Les statistiques peuvent être mises à jour automatiquement ou à la demande
 - La mise à jour automatique est définie à partir d'une option de base de données et doit être activée

Option	Description
AUTO_UPDATE_STATISTICS	Option de base de données qui permet à SQL Server de mettre à jour les statistiques automatiquement
UPDATE STATISTICS	Instruction permettant de mettre à jour les statistiques d'une table ou des statistiques spécifiées à la demande
sp_updatestats	Met à jour toutes les statistiques de la base de données

Points clés

L'une des principales tâches effectuées par SQL Server lorsqu'il optimise les requêtes qu'il doit exécuter consiste à décider quels index utiliser. SQL Server prend les décisions sur les index à utiliser en fonction des statistiques qu'il conserve sur la distribution des données dans l'index.

Les statistiques sont principalement mises à jour automatiquement par SQL Server, et AUTO_UPDATE_STATISTICS est activée par défaut dans toutes les bases de données. Il est recommandé de ne pas désactiver cette option.

Alternatives à la mise à jour automatique des statistiques

Pour les tables volumineuses, l'option AUTO_UPDATE_STATISTICS_ASYNC demande à SQL Server de mettre à jour les statistiques de manière asynchrone au lieu de retarder l'exécution de la requête, car sinon SQL Server mettrait à jour des statistiques obsolètes requises pour la compilation de la requête.

Les statistiques peuvent également être mises à jour à la demande. L'exécution de la commande UPDATE STATISTICS sur une table entraîne la mise à jour de toutes les statistiques de la table.

La procédure système stockée sp_updatestats peut servir à mettre à jour toutes les statistiques de la base de données.

Question : pourquoi pourriez-vous choisir de mettre à jour les statistiques en dehors des heures de bureau et non automatiquement ?

Démonstration 2A : Gestion des index

Dans cette démonstration, vous allez apprendre à

- Comment afficher la fragmentation des index ?
- Comment réorganiser des index ?

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**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_16_PRJ\22462A_16_PRJ.ssmsln** 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 **11 – Demonstration 1A.sql**.
2. Ouvrez le fichier de script **21 – Demonstration 2A.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

Leçon 3

Automatisation de la maintenance régulière des bases de données

- Vue d'ensemble des plans de maintenance de base de données SQL Server
- Surveillance des plans de maintenance des bases de données
- Démonstration 3A : Configuration d'un plan de maintenance des bases de données

Vous avez maintenant découvert comment effectuer manuellement certaines tâches de maintenance de base de données qui doivent être accomplies régulièrement. SQL Server fournit un assistant Plan de maintenance de base de données qui peut être utilisé pour créer des travaux de SQL Server Agent qui réalisent les tâches les plus courantes de maintenance de base de données.

Bien que l'assistant Plan de maintenance de base de données facilite ce processus, il est important de se rendre compte que vous pouvez utiliser la sortie de l'assistant comme point de départ de la création de vos propres plans de maintenance, ou vous pouvez créer ces plans de toutes pièces.

Objectifs

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

- configurer les plans de maintenance de base de données SQL Server ;
- surveiller les plans de maintenance de bases de données.

Vue d'ensemble des plans de maintenance de base de données SQL Server

- Assiste les administrateurs de base de données à planifier les tâches de maintenance principales
- Crée un ou plusieurs travaux d'Agent SQL Server
- Utilise SSIS pour exécuter les tâches
- Peut être créé manuellement au lieu de passer par l'Assistant

[+] Gouverneur de ressources
[+] Événements étendus
[+] Plans de maintenance
[+] AdventureWorksLT
[+] Journaux SQL Server
[+] Messagerie de base de données
[+] Coordinateur de transmission
[+] Catalogues Integration Server
[+] Agent SQL Server

Exécuter Rapports Renommer Supprimer Actualiser

The screenshot shows the 'Assistant Plan de maintenance' window. In the center, there's a list titled 'Sélectionnez une ou plusieurs tâches de maintenance:' (Select one or more maintenance tasks). Several checkboxes are checked, including 'Vérifier l'intégrité de la base de données', 'Réorganiser l'index', 'Mettre à jour les statistiques', and 'Tâche de nettoyage de maintenance'. Below the list is a note: 'La tâche de nettoyage de maintenance supprime les fichiers résiduels résultant de l'exécution d'un plan de maintenance.' At the bottom of the window are buttons for 'Aide' (Help), '< Précédent' (Previous), 'Suivant >', 'Terminer' (Finish), and 'Annuler' (Cancel).

Points clés

L'assistant Plan de maintenance de SQL Server crée des travaux de SQL Server Agent qui effectuent les tâches courantes de maintenance de base de données, et planifie ces travaux de sorte à garantir que la base de données est sauvegardée régulièrement, qu'elle fonctionne correctement et que sa cohérence est contrôlée. L'assistant crée des packages SQL Server Integration Services exécutés par les tâches de SQL Server Agent.

Vous pouvez planifier de nombreuses tâches de maintenance de façon à s'exécuter automatiquement, y compris les suivantes :

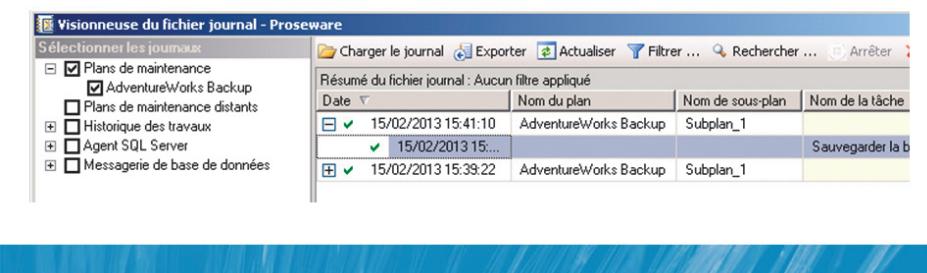
- Sauvegarde de la base de données et des fichiers journaux des transactions. Les sauvegardes de la base de données et des journaux peuvent être conservées pendant une période donnée, puis supprimées de manière automatique.
- Exécution de travaux de SQL Server Agent qui effectuent diverses actions.
- Compactage des fichiers de données en supprimant les pages vides de la base de données.
- Exécution de vérifications de la cohérence interne des données et des pages de données de la base de données pour s'assurer qu'un problème système ou logiciel n'a pas endommagé des données.
- Réorganisation des informations sur les pages de données et d'index en reconstruisant les index.
- Mise à jour des statistiques d'index pour s'assurer que l'optimiseur de requête dispose des toutes dernières informations de répartition des valeurs de données dans les tables.

Remarque Les plans de maintenance peuvent être créés avec une planification de toutes les tâches ou avec des planifications individuelles pour chaque tâche sélectionnée.

Question : quels types de tâches de maintenance devraient être automatisés ?

Surveillance des plans de maintenance des bases de données

- Surveillance en temps réel avec le Moniteur d'activité des travaux
- Les résultats d'exécution sont stockés dans msdb et peuvent également être
 - Enregistrés dans un fichier texte
 - Envoyés à un opérateur
- Les tâches de nettoyage sont utilisées pour implémenter la rétention



Points clés

Les plans de maintenance de base de données SQL Server sont mis en œuvre au travers de travaux de SQL Server Agent qui exécutent des packages SQL Server Integration Services (SSIS). Étant donné qu'ils utilisent les travaux de SQL Server Agent, les plans de maintenance peuvent être surveillés au moyen du Moniteur d'activité des travaux standard de SSMS. Comme avec les autres travaux de SQL Server Agent, l'historique des travaux est écrit mais les plans de maintenance enregistrent des informations supplémentaires.

Résultats des plans de maintenance

Les résultats générés par les tâches de maintenance sont enregistrés dans les tables du plan de maintenance dbo.sysmaintplan_log et dbo.sysmaintplan_log_detail de la base de données msdb. Les entrées de ces tables peuvent être affichées en interrogeant directement ces tables au moyen de TSQL ou à l'aide de la visionneuse du fichier journal.

En outre, les rapports texte peuvent être écrits dans le système de fichiers et peuvent également être envoyés automatiquement aux opérateurs définis dans SQL Server Agent.

Notez que les tâches de nettoyage qui font partie des plans de maintenance servent à mettre en œuvre une stratégie de rétention pour les fichiers de sauvegarde, l'historique des travaux, les fichiers de rapport du plan de maintenance et les entrées de table de la base de données msdb.

Question : les enregistrements de l'historique du plan de maintenance sont-ils nettoyés automatiquement ?

Démonstration 3A : Configuration d'un plan de maintenance des bases de données

Dans cette démonstration, vous allez apprendre à

- Comment créer et exécuter un plan de maintenance ?
- Comment examiner l'historique d'un plan de maintenance ?

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**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_16_PRJ\22462A_16_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 et exécutez le fichier de script **11 – Demonstration 1A.sql**.
2. Ouvrez le fichier de script **31 – Demonstration 3A.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

Atelier pratique 16 : Exécution de la maintenance régulière des bases de données

- Exercice 1 : Contrôler l'intégrité de la base de données avec DBCC CHECKDB
- Exercice 2 : Corriger la fragmentation des index
- Exercice 3 : Créer un plan de maintenance de la base de données
- Exercice difficile 4 : Étudiez les performances du verrou de table (uniquement 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_16_PRJ\22462A_16_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

Une défaillance du disque s'est produite dans le sous-système d'E/S. Le disque a été remplacé, mais vous souhaitez vérifier la cohérence de vos bases de données existantes. Vous devez exécuter DBCC CHECKDB pour contrôler l'intégrité logique et physique de toutes les bases de données sur l'instance Proseware.

Vous avez identifié un problème de fragmentation dans un certain nombre de tables de la base de données MarketDev et vous êtes certain que les performances diminuent au fur et à mesure que le niveau de fragmentation augmente. Vous allez reconstruire les index des tables de base de données principales qui sont extrêmement fragmentés.

Vous avez également identifié une dégradation des performances de l'application si aucune maintenance d'index appropriée n'a été effectuée. Vous souhaitez vérifier qu'il existe une détection anticipée des problèmes de cohérence dans la base de données MarketDev et que la maintenance d'index est automatiquement exécutée de manière planifiée. Pour vous assurer que la maintenance régulière est effectuée, vous allez créer un plan de maintenance de base de données pour planifier ces opérations une fois par semaine.

Alors que l'exécution de DBCC CHECKDB se déroule assez rapidement, vous vous intéressez à la différence en termes de performances qui peut être obtenue en utilisant des verrous de table à la place de captures instantanées de base de données pendant les opérations DBCC CHECKDB. Si vous avez le temps, étudiez les différences de performances.

Documentation

Spécifications du plan de maintenance de base de données

Élément	Configuration
Nom du plan	Maintenance hebdomadaire de Proseware.
Planification	Une fois par semaine pour toutes les tâches, le dimanche soir à 18 h.
Tâches requises	Vérifier l'intégrité de toutes les bases de données sur l'instance de serveur Proseware.
	Reconstruire les index de la base de données MarketDev.
Remarques	Les contrôles d'intégrité de la base de données doivent inclure les index.
	Lors de la reconstruction des index de la base de données MarketDev, les pages dans les index doivent être complètes à 90 %.
	Comme Proseware utilise une licence Enterprise Edition, les reconstructions d'index en ligne sont prises en charge et doivent être utilisées.
	Les rapports doivent être enregistrés dans le dossier L:\MKTG.

Exercice 1 : Contrôler l'intégrité de la base de données avec DBCC CHECKDB

Scénario

Une défaillance du disque s'est produite dans le sous-système d'E/S. Le disque a été remplacé, mais vous souhaitez vérifier la cohérence de vos bases de données existantes. Vous devez exécuter DBCC CHECKDB pour contrôler l'intégrité logique et physique de toutes les bases de données sur l'instance Proseware.

Les tâches principales de cet exercice sont les suivantes :

1. Vérifier la cohérence des bases de données sur l'instance Proseware.
2. Corriger les problèmes détectés.

► Tâche 1 : Vérifier la cohérence des bases de données sur l'instance Proseware

- Exécutez DBCC CHECKDB sur toutes les bases de données sur l'instance de serveur Proseware. Notez toutes les bases de données qui ont des erreurs.

► Tâche 2 : Corriger les problèmes détectés

- Pour les bases de données qui comportent des erreurs, utilisez l'option DBCC pour la réparation tout en permettant la perte de données. (Notez qu'il s'agit d'une action extrême qui doit être entreprise uniquement dans les cas d'urgence, lorsqu'il n'existe aucune sauvegarde disponible pour être restaurée).

Résultats : À la fin de cet exercice, vous devez avoir utilisé la commande DBCC CHECKDB pour vérifier la cohérence de toutes les bases de données sur l'instance de serveur Proseware et avoir corrigé les problèmes qui ont été trouvés.

Exercice 2 : Corriger la fragmentation des index

Scénario

Vous avez identifié un problème de fragmentation dans un certain nombre de tables de la base de données MarketDev et vous êtes certain que les performances diminuent au fur et à mesure que le niveau de fragmentation augmente. Vous allez reconstruire les index des tables de base de données principales qui sont extrêmement fragmentés.

Les tâches principales de cet exercice sont les suivantes :

1. Examiner la fragmentation des index de la base de données MarketDev pour déterminer les index à défragmenter et les index à reconstruire.
2. Défragmenter les index comme déterminé.
3. Reconstruire les index comme déterminé.

► **Tâche 1 : Examiner la fragmentation des index de la base de données MarketDev pour déterminer les index à défragmenter et les index à reconstruire**

- Écrivez une requête à l'aide de la fonction sys.dm_db_index_physical_stats pour rechercher les index qui sont fragmentés à plus de 30 %.

► **Tâche 2 : Défragmenter les index comme déterminé**

- Écrivez une requête pour défragmenter les index dont vous avez déterminé que leur fragmentation était supérieure à 30 % et inférieure à 70 %.

► **Tâche 3 : Reconstruire les index comme déterminé**

- Écrivez une requête pour reconstruire les index dont vous avez déterminé que la fragmentation était supérieure à 70 %.

Résultats : À la fin de cet exercice, vous devez avoir reconstruit ou défragmenté tous les index dont la fragmentation était importante.

Exercice 3 : Créer un plan de maintenance de la base de données

Scénario

Vous avez également identifié une dégradation des performances de l'application si aucune maintenance d'index appropriée n'a été effectuée. Vous souhaitez vérifier qu'il existe une détection anticipée des problèmes de cohérence dans la base de données MarketDev et que la maintenance d'index est automatiquement exécutée de manière planifiée. Pour vous assurer que la maintenance régulière est effectuée, vous allez créer un plan de maintenance de base de données pour planifier ces opérations une fois par semaine.

La tâche principale de cet exercice est la suivante :

1. Créer un plan de maintenance de base de données requis.

► **Tâche 1 : Créer un plan de maintenance de base de données requis**

- Prenez connaissance des éléments requis de l'exercice fournis dans la documentation associée.
- Créez un plan de maintenance qui répond aux besoins.

Résultats : À la fin de cet exercice, vous devez avoir créé le plan de maintenance de base de données requis.

Exercice difficile 4 : Étudier les performances du verrou de table (uniquement si le temps le permet)

Scénario

Alors que l'exécution de DBCC CHECKDB se déroule assez rapidement, vous vous intéressez à la différence en termes de performances qui peut être obtenue en utilisant des verrous de table à la place de captures instantanées de base de données pendant les opérations DBCC CHECKDB. Si vous avez le temps, étudiez les différences de performances.

Les tâches principales de cet exercice sont les suivantes :

1. Exécuter DBCC CHECKDB avec des captures instantanées de base de données.
2. Exécuter DBCC CHECKDB avec des verrous de table.

► Tâche 1 : Exécuter DBCC CHECKDB avec des captures instantanées de base de données

- Exécutez DBCC CHECKDB sur toutes les bases de données de l'instance de serveur Proseware en utilisant des captures instantanées de base de données (option par défaut).
- Enregistrez la durée totale de l'exécution.

► Tâche 2 : Exécuter DBCC CHECKDB avec les verrous de table

- Exécutez DBCC CHECKDB sur toutes les bases de données de l'instance de serveur Proseware en utilisant des verrous de table (option TABLOCK).
- Enregistrez la durée totale de l'exécution.
- Comparez la durée de l'exécution à celle enregistrée dans la tâche 1.

Résultats : À la fin de cet exercice, vous devez avoir comparé les performances de DBCC CHECKDB selon qu'elle est utilisée avec des captures instantanées de base de données ou des verrous de table.

Contrôle des acquis du module et éléments à retenir

- Questions de contrôle des acquis
- Meilleures pratiques

Questions de contrôle des acquis

1. Quelles tâches régulières doivent être implémentées pour les bases de données en lecture seule ?
2. Quelle option devez-vous envisager d'utiliser lors de l'exécution de DBCC CHECKDB sur des bases de données en production volumineuses ?

Meilleures pratiques

1. Exécuter DBCC CHECKDB régulièrement.
2. Synchroniser DBCC CHECKDB avec votre stratégie de sauvegarde.
3. Examiner la possibilité d'utiliser RESTORE avant la réparation en cas d'altération.
4. Défragmenter vos index si nécessaire.
5. Mettre à jour les statistiques selon un calendrier, si vous ne souhaitez pas le faire pendant les heures d'opérations normales.
6. Utiliser des plans de maintenance pour implémenter les tâches régulières.

Module 17

Trace de l'accès à SQL Server 2012

Sommaire :

Leçon 1 : Capture de l'activité avec SQL Server Profiler et le générateur de profils des événements étendus	17-3
Leçon 2 : Amélioration des performances avec l'Assistant Paramétrage du moteur de base de données	17-17
Leçon 3 : Utilisation des options de suivi	17-26
Atelier pratique 17 : Trace de l'accès à SQL Server 2012	17-38

Vue d'ensemble du module

- Capture de l'activité avec SQL Server Profiler et le générateur de profils des événements étendus
- Amélioration des performances avec l'Assistant Paramétrage du moteur de base de données
- Utilisation des options de suivi

Microsoft® SQL Server® fonctionne correctement avec de nombreux types de charges de travail SQL, mais les performances de la plupart des systèmes SQL Server peuvent être améliorées par le paramétrage des requêtes et des structures des bases de données. Pour la plupart des organisations, le réglage des performances n'est pas une tâche qui s'effectue en une seule fois. Le réglage des performances implique généralement un processus d'améliorations incrémentielles continues.

Il est important de concentrer vos efforts sur le réglage des performances qui apporteront le plus d'améliorations. Les développeurs devinent souvent comment leurs applications seront utilisées et ils essaient d'optimiser les applications en fonction de leur intuition. La capacité de tracer l'activité d'un SQL Server supprime la part d'intuition de ce processus et vous permet de concentrer vos efforts sur les domaines des applications réellement utilisées par les utilisateurs.

SQL Server Profiler et le générateur de profils des événements étendus sont utilisés pour capturer les traces de l'activité d'un SQL Server. L'Assistant Paramétrage du moteur de base de données peut être utilisé pour analyser les traces capturées par SQL Server Profiler et pour suggérer des améliorations à effectuer, en particulier par rapport aux index et aux statistiques.

Objectifs

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

- capturer l'activité avec SQL Server Profiler et le générateur de profils des événements étendus ;
- améliorer les performances avec l'Assistant Paramétrage du moteur de base de données ;
- utiliser les options de traçage.

Leçon 1

Capture de l'activité avec SQL Server Profiler et le générateur de profils des événements étendus

- Vue d'ensemble de SQL Server Profiler
- Options de sortie de trace disponibles
- Événements de trace couramment utilisés
- Colonnes de trace couramment utilisées
- Filtrage des traces
- Utilisation des modèles de trace
- Démonstration 1A : Capture de l'activité à l'aide de SQL Server Profiler
- Utilisation du générateur de profils d'événements étendus
- Démonstration 1B : Capture de l'activité avec le Générateur de profils des événements étendus

SQL Server Profiler et le générateur de profils d'événements étendus vous permettent de tracer l'activité qui se produit dans le moteur de base de données dans SQL Server. SQL Server Profiler peut également effectuer le traçage de l'activité sur Analysis Services. (Le traçage de l'activité d'Analysis Services n'entre pas dans le cadre de ce cours).

Les traces peuvent être utilisées pour le réglage des performances, à des fins de dépannage et de diagnostic et pour relire des charges de travail. La capacité de relire une charge de travail vous permet de tester l'impact des modifications de performances sur des systèmes de test ou de tester les charges de travail d'une application sur les versions plus récentes de SQL Server.

Il est important de savoir configurer ces outils, afin d'éviter de créer des impacts excessifs du processus de traçage lui-même.

Objectifs

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

- utiliser SQL Server Profiler ;
- décrire les options de sortie de trace disponibles ;
- détailler les événements de trace couramment utilisés ;
- détailler les colonnes de trace couramment utilisées ;
- filtrer les traces ;
- utiliser des modèles de trace ;
- utiliser le générateur de profils d'événements étendus.

Vue d'ensemble de SQL Server Profiler

SQL Server Profiler est un outil utilisé pour tracer l'activité de SQL Server

- Inspiré de l'interface de programmation Trace SQL
- Utilisé dans de nombreux scénarios tels que le débogage, l'analyse des performances, l'analyse des blocages et bien d'autres encore
- Fonction de relecture des tests de contrainte

The screenshot shows a table of trace results. The columns include Event Class, Event Subclass, Text Data, Status, Duration, and SPID. The table contains several rows of event data, such as 'SP:StmtCompleted' and 'RPC:Completed' events, along with some XML data at the bottom.

Points clés

SQL Server Profiler est un outil important pour le réglage des performances des requêtes SQL Server. Il capture l'activité des applications clientes vers SQL Server et les stocke dans une trace. Ces traces peuvent ainsi être analysées.

SQL Server Profiler

SQL Server profiler capture des données lorsque des événements se produisent. Seuls les événements sélectionnés sont capturés. Diverses informations (affichées sous la forme d'un ensemble de colonnes) sont disponibles lorsque chaque événement se produit. La trace créée contient uniquement les colonnes sélectionnées pour les événements sélectionnés.

Au lieu de devoir sélectionner les événements et les colonnes chaque fois que vous exécutez SQL Server profiler, un ensemble de modèles existants sont disponibles. Vous pouvez également enregistrer vos propres sélections en tant que nouveau modèle.

Les traces capturées sont utiles pour paramétriser les performances d'une application et diagnostiquer les problèmes spécifiques qui se produisent. Lorsque vous utilisez des traces pour diagnostiquer les problèmes, les données du journal de l'Analyseur de performances Windows peuvent être chargées. Cela permet d'établir des relations entre les impacts sur les ressources système et l'exécution des requêtes dans SQL Server à effectuer.

Les traces peuvent également être relues. La capacité de relire des traces est utile pour les systèmes de test de charge ou pour s'assurer que les versions mises à jour de SQL Server peuvent être utilisées avec les applications existantes. SQL Server Profiler vous permet également de parcourir les requêtes lors du diagnostic des problèmes.

Trace SQL

SQL Server Profiler est un outil graphique et il est important de savoir qu'il peut impacter de façon significative les performances sur le serveur en cours de traçage, en fonction des éléments sélectionnés.

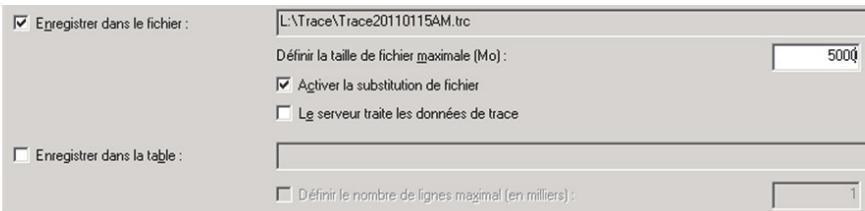
Trace SQL est une bibliothèque des procédures stockées système qui peuvent être utilisées pour le traçage lorsqu'il est nécessaire de réduire l'impact sur les performances du traçage. En interne, SQL Server Profiler utilise l'interface de programmation fournie par Trace SQL.

-  **Remarque** Le système d'événements étendus qui a été introduit dans SQL Server 2008 propose également des fonctionnalités pour tracer l'activité et les ressources de SQL Server. L'utilisation des événements étendus pour tracer l'activité est abordée ultérieurement dans cette leçon. L'utilisation des événements étendus à d'autres fins d'analyse n'entre pas dans le cadre de ce cours.

Question : dans quel cas la relecture d'une trace peut s'avérer être utile ?

Options de sortie de trace disponibles

- Les traces du générateur de profils peuvent être enregistrées dans les tables et les fichiers
- Pour la sortie des fichiers, veillez à configurer les éléments suivants
 - Taille de fichier maximale (la valeur par défaut de 5 Mo n'est jamais suffisante)
 - Activer la substitution de fichier (le cas échéant)



Points clés

Lorsqu'une trace SQL Server Profiler est active, les événements capturés sont chargés dans une grille graphique dans l'interface utilisateur de SQL Server Profiler.

De plus, SQL Server Profiler peut envoyer les détails des événements capturés dans des fichiers du système d'exploitation ou dans des tables de base de données.

Capture dans des fichiers

Capturer dans un fichier du système d'exploitation est l'option la plus efficace pour les traces SQL Server Profiler. Lorsque vous configurez la sortie du fichier, vous devez fournir un nom de fichier pour la trace. Le type de fichier par défaut pour un fichier de trace est « .trc ». La taille de fichier par défaut de SQL Server Profiler est de 5 Mo, ce qui est beaucoup trop petit pour la plupart des traces. Une valeur plus réaliste dans la plupart des grands systèmes serait 500 Mo ou 5000 Mo, selon le volume d'activité qui doit être enregistré.

Lorsque la taille de fichier allouée est épuisée, SQL Server Profiler ouvre un nouveau fichier avec une variation du nom de fichier actuel et commence à écrire dedans si l'option « Activer la substitution de fichier » a été activée. Il est recommandé d'utiliser une taille de fichier maximale et d'éviter de recourir à des fichiers de substitution, à moins qu'il soit nécessaire de déplacer les traces capturées sur des supports comme des DVD ou sur des sites de téléchargement qui ne peuvent pas utiliser des fichiers trop volumineux.

Selectionner l'option « Le serveur traite les données de trace » oblige le service SQL Server à écrire le fichier de sortie au lieu du système client qui exécute SQL Server Profiler. Cette option peut améliorer les performances mais requiert l'accès au système de fichiers du serveur.

Capturer dans des tables

SQL Server Profiler peut également capturer des données de trace dans des tables de base de données. L'interface de programmation Trace SQL sous-jacente ne prend pas directement en charge la sortie vers des tables. Cela signifie que le programme SQL Server Profiler doit extraire les données d'événement dans sa grille graphique et, à mesures que les lots et les lignes sont reçus, SQL Server Profiler écrit ces lignes dans la table de base de données sélectionnée.

 **Remarque** Il est très important d'éviter d'écrire des données de trace directement dans le système SQL Server surveillé, sauf si est un niveau d'activité très faible est prévu.

SQL Server Profiler fournit également une option pour enregistrer les données d'événement capturées existantes qui sont affichées dans la grille graphique dans une table de base de données.

Question : quel peut être l'avantage d'enregistrer des événements dans une table SQL Server ?

Événements de trace couramment utilisés

Exemple d'événement	Description
SQL:BatchCompleted	Un lot SQL Server terminé
SQL:StmtCompleted	Une instruction T-SQL terminée
RPC:Completed	Un appel de procédure distante (procédure stockée) terminé
Audit Login/Audit Logout	Une connexion ou une déconnexion s'est produite
Deadlock Graph	Une description XML d'un blocage a été capturée

Points clés

Les informations enregistrées dans une trace sont réparties en catégories. Les catégories contiennent des événements, dont chacun possède des attributs, eux-mêmes définis par des colonnes.

Catégories de trace

Dans le Générateur de profils SQL Server, une catégorie correspond à un groupe de classes d'événements associées. Les classes d'événements se composent de types d'événements pouvant être tracés. La classe d'événements contient toutes les colonnes de données qu'un événement peut signaler.

Événements

Un événement est défini comme l'occurrence d'une action au sein d'une instance du moteur de base de données SQL Server. Les événements sont définis par leurs attributs, qui sont répertoriés dans des colonnes de données.

Les événements les plus souvent tracés sont :

Événement	Description
SQL:BatchCompleted	Lorsqu'un lot d'instructions T-SQL est terminé, l'événement SQL:BatchCompleted est déclenché. Notez qu'il existe également un événement généré lorsque le lot est lancé pour la première fois mais que l'événement terminé contient des informations utiles comme des détails sur les ressources utilisées pendant l'exécution du lot.
SQL:StmtCompleted	Si le traçage au niveau du lot SQL est trop brut, il est possible d'extraire les détails de chaque instruction contenue dans le lot.

(suite)

Événement	Description
RPC: Completed	L'événement RPC:Completed se déclenche lorsque l'exécution d'une procédure stockée se termine. Il existe un événement traçable lorsque la procédure stockée démarre, mais comme pour l'événement SQL:BatchCompleted, l'événement RPC:Completed est utile car il contient des détails sur les ressources utilisées pendant l'exécution de la procédure stockée. Vous pouvez afficher une répartition instruction par instruction des ressources utilisées dans la procédure stockée via l'événement SP:StmtCompleted.
Audit Login/ Audit Logout	Vous pouvez inclure dans vos traces les détails de chaque événement de connexion et de déconnexion qui se produisent pendant l'activité tracée.
Deadlock Graph	Les blocages non pris en charge entraînent souvent la transmission d'erreurs des applications aux utilisateurs finaux. Si votre système subit des blocages, l'événement Deadlock Graph se déclenche lorsque des blocages se produisent et il capture les détails de ce qui a provoqué le blocage. Les détails sont capturés dans un document XML qui peut être affiché graphiquement dans SQL Server Profiler.

Question : pourquoi les événements déclenchés à l'issue de l'exécution des lots ou des instructions sont-ils souvent plus intéressants que ceux générés au début de l'exécution des lots ou des instructions ?

Colonnes de trace couramment utilisées

Les colonnes de trace représentent les données (souvent des attributs) qui peuvent être capturées lorsqu'un événement se produit

- Évitez de capturer trop de colonnes afin de réduire la taille de la trace
- Certains événements ne fournissent pas de données dans chaque colonne
- Les colonnes peuvent être utilisées pour regrouper les données d'événement

TextData	ApplicationName	NTUserName	LoginName	CPU	Reads	Writes	Duration
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>				
	<input checked="" type="checkbox"/>						
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>				<input type="checkbox"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Points clés

Les colonnes de données contiennent les attributs des événements. SQL Server Profiler utilise les colonnes de données dans le résultat de la trace pour décrire les événements capturés lors de son exécution.

SQL Server Profiler propose un grand nombre de colonnes, mais les événements n'écrivent pas tous dans toutes les colonnes possibles. Par exemple, dans l'événement SQL:BatchStarting, les colonnes Reads, Writes, Duration et CPU ne sont pas disponibles car les valeurs ne sont pas disponibles au moment de l'événement. Ces colonnes sont disponibles dans l'événement SQL:BatchCompleted.



Remarque La sortie dans la grille graphique de SQL Server Profiler peut être groupée en fonction des valeurs des colonnes.

Vous devez réduire le nombre de colonnes que vous capturez lorsque des événements se produisent, afin de réduire la taille globale de la trace capturée.

Vous pouvez également organiser les colonnes en groupes associés en utilisant la fonction Organiser les colonnes.

Colonnes utiles sont souvent omises

L'une des colonnes les plus intéressantes est la colonne TextData. De nombreux événements ne l'incluent pas par défaut, même si les valeurs qu'elle contient sont très utiles. Par exemple, pour l'événement RPC:Completed, la colonne TextData contient l'instruction T-SQL utilisée pour exécuter la procédure stockée.

Deux autres colonnes utiles sont l'ID et le nom de la base de données. Vous devrez souvent créer un filtre en fonction de la base de données pour laquelle vous souhaitez tracer l'activité. Le traçage par l'ID de la base de données est plus efficace que celui par le nom de la base de données. Toutefois, les modèles de trace qui filtrent par l'ID de la base de données sont moins portables que celles qui filtrent par le nom de la base de données car une base de données restaurée sur un autre serveur aura généralement une ID de base de données différente. Vous pouvez également utiliser des valeurs génériques dans la clause LIKE d'un filtre sur le nom de la base de données.

Question : quelles sont les informations renvoyées par la colonne TextData ?

Filtrage des traces

- Évitez de capturer trop d'événements pour minimiser
 - La taille de la trace
 - L'impact sur les performances
 - La complexité de l'analyse
- Le filtre est appliqué uniquement lorsque la colonne filtrée est prise en charge par l'événement sélectionné



Points clés

Des filtres peuvent être définis pour chacune des colonnes capturées dans une trace. Il est important de vérifier que vous ne capturez que les événements qui vous intéressent en limitant les événements via des filtres. L'utilisation efficace des filtres permet de réduire la taille globale de la trace capturée, permet d'éviter de surcharger le serveur avec l'activité de traçage et diminue le nombre d'événements contenus dans la trace, ce qui réduit la complexité de l'analyse. Les plus petites traces sont généralement plus rapides à analyser.

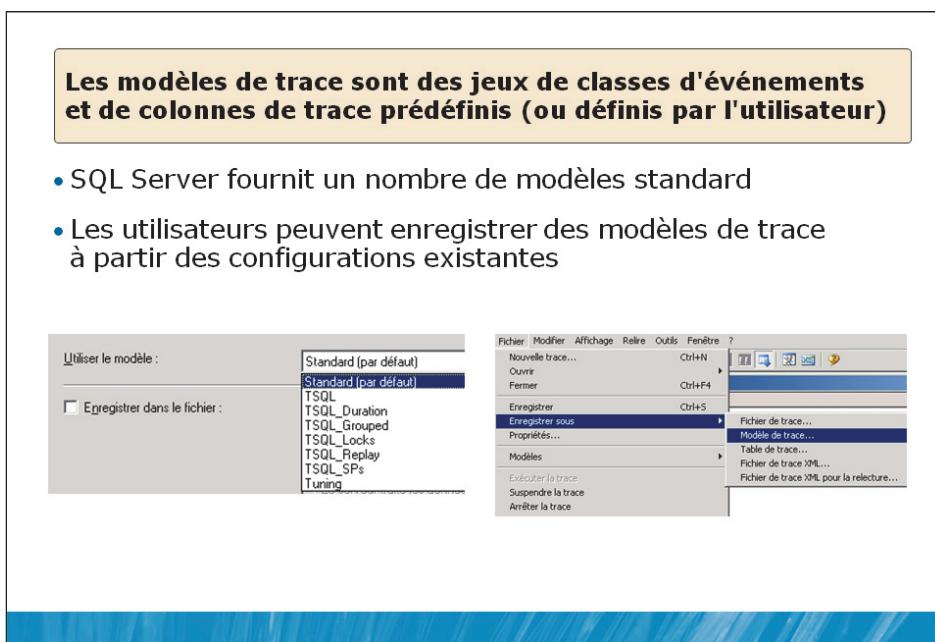
Les filtres sont appliqués uniquement si un événement écrit dans une colonne particulière. Par exemple, si vous définissez un filtre pour DatabaseName = AdventureWorks et définissez la capture de l'événement Deadlock Graph, tous les événements d'interblocage s'affichent, car la colonne DatabaseName n'est pas présente dans l'événement Deadlock Graph.

Question : quel filtre allez-vous utiliser pour localiser les instructions ou les traitements longs ?

Utilisation des modèles de trace

Les modèles de trace sont des jeux de classes d'événements et de colonnes de trace prédéfinis (ou définis par l'utilisateur)

- SQL Server fournit un nombre de modèles standard
- Les utilisateurs peuvent enregistrer des modèles de trace à partir des configurations existantes



Points clés

Vous pouvez utiliser SQL Server Profiler pour créer des modèles qui définissent les classes d'événements et les colonnes de données à inclure dans des traces. Après avoir défini et enregistré le modèle, vous pouvez exécuter une trace qui enregistre les données de chaque classe d'événements sélectionnée.

Les modèles peuvent être générés dans SQL Server Profiler en créant une trace à l'aide de l'interface graphique, en démarrant et en arrêtant la trace au moins une fois, puis en utilisant Fichier > Enregistrer sous > Modèle de trace.

SQL Server Profiler fournit des modèles de traces prédéfinis qui vous permettent de configurer aisément les classes d'événements dont vous avez besoin pour des types de traces spécifiques. Le modèle Standard, par exemple, permet de créer une trace générique pour enregistrer les connexions, les déconnexions, les lots terminés et les informations de connexion. Vous pouvez utiliser ce modèle pour exécuter des traces sans modification ou comme point de départ pour d'autres modèles avec des configurations d'événements différents.

Question : comment les modèles SQL Server Profiler vous aident-ils à tracer l'activité ?

Démonstration 1A : Capture de l'activité à l'aide de SQL Server Profiler

Dans cette démonstration, vous allez apprendre à

- Créer et exécuter une trace à l'aide de SQL Server Profiler
- Utiliser la trace pour capturer l'activité de SQL Server
- Enregistrer la trace dans un fichier

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**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_17_PRJ\22462A_17_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.
- Question :** pourquoi existe-t-il un seul événement Batch Starting et un seul événement Batch Completed pour la charge de travail que nous avons exécutée ?

Utilisation du générateur de profils d'événements étendus

- Les événements étendus constituent une infrastructure d'événements légère dans SQL Server
 - Même technologie sous-jacente que celle utilisée par SQL Server Audit
- Générateur de profils pour les événements étendus
 - Peut capturer l'activité du moteur de base de données
 - Non encore pris en charge pour Analysis Services
 - Concepts familiers pour les utilisateurs de SQL Server Profiler
 - Incorporé dans SQL Server Management Studio



Points clés

SQL Server 2012 présente un nouvel outil pour gérer les sessions qui capturent des événements lorsqu'ils se produisent, basés sur l'infrastructure des événements étendus qui a été introduite dans SQL Server 2008. Il s'agit de la même technologie qui sous-tend SQL Server Audit.

Le générateur de profils d'événements étendus est directement intégré dans SQL Server Management Studio et peut être utilisé pour capturer l'activité du moteur de base de données SQL Server à peu près de la même façon que SQL Server Profiler.

Au fil du temps, le Générateur de profils d'événements étendus remplacera SQL Server Profiler comme outil de premier choix pour tracer l'activité dans SQL Server. Dans cette version, les deux outils possèdent des ensembles de fonctionnalités courantes et les deux sont susceptibles d'être utilisés.

Démonstration 1B : Capture de l'activité avec le Générateur de profils des événements étendus

Dans cette démonstration, vous allez apprendre à capturer l'activité avec le Générateur de profils des événements étendus

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**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_17_PRJ\22462A_17_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 **12 – Demonstration 1B.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

Leçon 2

Amélioration des performances avec l'Assistant Paramétrage du moteur de base de données

- Vue d'ensemble du réglage des performances
- Options disponibles pour le réglage des performances
- Présentation de l'Assistant Paramétrage du moteur de base de données
- Options de l'Assistant Paramétrage du moteur de base de données
- Démonstration 2A : Utilisation de l'Assistant Paramétrage du moteur de base de données

Le paramétrage les performances d'un système SQL Server est l'une des tâches les plus importantes effectuée par les administrateurs de base de données. Comme indiqué précédemment dans ce module, le réglage des performances a tendance à être un processus continu, et non une tâche qui s'effectue en une seule fois.

Il est important de suivre une méthodologie standard pour le réglage des performances. SQL Server inclut de nombreux outils qui peuvent faciliter le réglage des performances, notamment l'Assistant Paramétrage du moteur de base de données (DETA). Il est important de savoir comment configurer et utiliser DETA.

Objectifs

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

- expliquer le réglage des performances ;
- décrire les options disponibles pour le réglage des performances ;
- utiliser l'Assistant Paramétrage du moteur de base de données ;
- décrire brièvement les options disponibles de l'Assistant Paramétrage du moteur de base de données.

Vue d'ensemble du réglage des performances



Points clés

Le réglage des performances est un processus itératif d'amélioration incrémentielle effectuée de façon continue.

Tout d'abord, il est important de définir un objectif initial en fonction des problèmes qui peuvent survenir. « Cela doit être plus rapide » n'est généralement pas un objectif assez bien défini. En fonction de l'objectif choisi, vous devez sélectionner les outils appropriés pour surveiller le système et fournir les mesures qui vous aideront à atteindre l'objectif. SQL Server fournit plusieurs outils pour vous aider dans cette étape. (Une vue d'ensemble des outils disponibles sera proposée dans la rubrique suivante).

Dans les outils sélectionnés, vous devez identifier les mesures que vous utiliserez dans la liste des mesures disponibles. La base doit être créée grâce à une analyse et une prise de décision à propos des modifications à apporter.

En fonction des résultats, une stratégie est ensuite développée pour résoudre les problèmes détectés. La stratégie est ensuite mise en œuvre, les résultats sont analysés et le processus recommence.

Travail sur les causes plutôt que sur les symptômes

Un concept clé pour le réglage des performances consiste à essayer de déterminer les causes sous-jacentes des problèmes, plutôt que de batailler avec les symptômes. Par exemple, un client peut vous demander de résoudre un problème de blocage. Il arrive souvent que le blocage excessif se produit à cause de requêtes dont l'exécution est longue. En résolvant les requêtes et en les indexant afin qu'elles soient exécutées plus rapidement, les problèmes de blocage disparaissent souvent.

Un autre exemple courant concerne les systèmes qui rencontrent des problèmes de performances d'E/S, même si dans de nombreux cas, cela peut être dû au manque de mémoire disponible dans le système.

Question : quand pouvez-vous savoir que le réglage des performances est terminé ?

Options disponibles pour le réglage des performances

- Assistant Paramétrage du moteur de base de données
- SSMS
 - Moniteur d'activité
 - Rapports standard
 - Rapports personnalisés
- Objets de gestion dynamique
- Collecte de données SQL Server
- SQL Server Profiler
- Trace SQL
- Événements étendus de SQL Server
- Moniteur de fiabilité et de performances

Points clés

SQL Server fournit plusieurs outils que vous pouvez utiliser pour effectuer l'analyse et le réglage des performances. Chacun d'eux est utile dans certains scénarios et vous devrez souvent combiner plusieurs outils pour obtenir les résultats souhaités.

Outil	Description et emplacement
Assistant Paramétrage du moteur de base de données	Un outil fourni avec SQL Server pour paramétrier les index et les statistiques. Cet assistant est traité dans cette leçon.
SSMS	Le principal outil de gestion fourni avec SQL Server. Les aspects utiles pour le réglage des performances (moniteur d'activité, rapports standard et personnalisés) sont traités dans le module 18.
Objets de gestion dynamique	Des objets de base de données qui donnent un aperçu des opérations internes de SQL Server. Un grand nombre de ces objets utiles sont traités dans le module 18.
Collecte de données SQL Server	Un système automatisé pour la collecte et le stockage des données de performances avec un ensemble de rapports standard. La collecte de données est traitée dans le module 18.
SQL Server Profiler	Un outil de traçage et de profilage traité dans la première leçon de ce module.

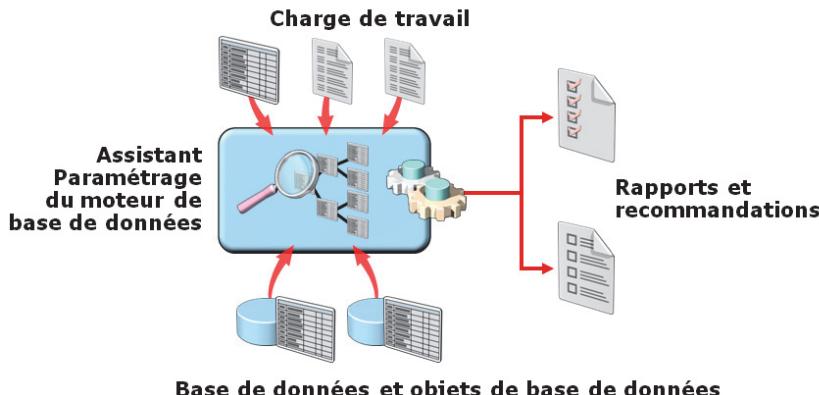
(suite)

Outil	Description et emplacement
Trace SQL	Une interface de programmation pour suivre l'accès à SQL Server traitée dans la dernière leçon de ce module.
Événements étendus de SQL Server	Une architecture d'événements légère. Vous venez de découvrir comment utiliser le Générateur de profils d'événements étendus pour capturer l'activité d'un serveur. L'utilisation directe des événements étendus (sans passer par une interface utilisateur graphique) n'entre pas dans le cadre de ce cours.
Distributed Replay	Un outil avancé pour relire des charges de travail sur un ensemble de serveurs potentiellement non distribué. Le Distributed Replay n'entre pas dans le cadre de cours.
Moniteur de fiabilité et de performances	L'outil de performance Windows standard traité dans le module 18.

Question : quels outils peuvent être utiles pour une surveillance en temps réel ?

Présentation de l'Assistant Paramétrage du moteur de base de données

- Utilisé pour suggérer des modifications d'index et de statistiques afin d'améliorer les performances
- Traite les charges de travail capturées par SQL Server Profiler comme traces



Points clés

L'utilitaire Assistant Paramétrage du moteur de base de données analyse l'impact des charges de travail sur les performances d'une ou plusieurs bases de données. En général, ces charges de travail sont obtenues à partir de traces capturées par SQL Server Profiler ou la fonctionnalité Trace SQL. (Trace SQL est traitée ultérieurement dans la leçon suivante). Après avoir analysé l'impact d'une charge de travail sur vos bases de données, l'Assistant Paramétrage du moteur de base de données fournit ses recommandations pour améliorer les performances de votre système.

Assistant Paramétrage du moteur de base de données

Dans SQL Server 2000 et les versions antérieures, une ancienne version de cet outil était fournie. Elle s'appelait « Assistant Paramétrage d'index ». Dans SQL Server 2005, le nom a été modifié avec l'évolution de l'outil pour pouvoir fournir un plus grand nombre de recommandations.

L'Assistant Paramétrage du moteur de base de données a encore été amélioré dans SQL Server 2008 au niveau de l'analyse des charges de travail, du paramétrage intégré et de la possibilité de paramétrier plusieurs bases de données simultanément.

Charges de travail

Une charge de travail est un ensemble d'instructions Transact-SQL exécutées sur des bases de données à paramétriser. La source de la charge de travail peut être un fichier contenant des instructions Transact-SQL, un fichier trace généré par le Générateur de profils SQL ou une table d'informations de trace également générée par le Générateur de profils SQL. SQL Server Management Studio propose également la possibilité de lancer l'Assistant Paramétrage du moteur de base de données pour analyser une instruction individuelle.

Recommandations

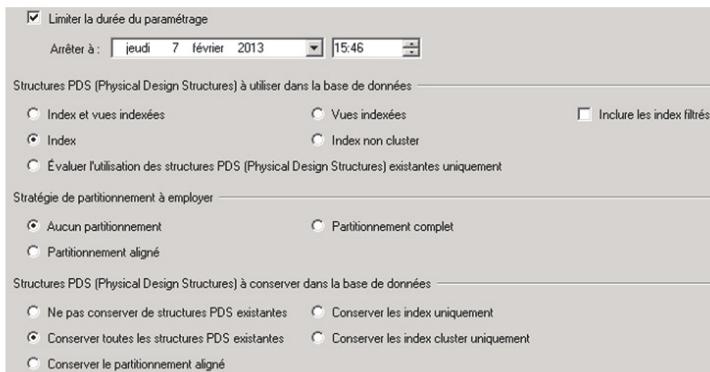
Les recommandations qui peuvent être produites incluent les modifications à apporter à la base de données, comme des index à créer ou à supprimer, et selon les options de paramétrage définies, des recommandations sur le partitionnement. Les recommandations produites sont fournies comme un ensemble d'instructions Transact-SQL qui implémenteraient les modifications suggérées. Vous pouvez consulter le code Transact-SQL et l'enregistrer en vue d'une révision ou application ultérieure, ou choisir d'implémenter immédiatement les modifications recommandées.

N'appliquez pas des modifications à une base de données sans examen préalable détaillé, notamment dans des environnements de production. De plus, vérifiez que toute analyse que vous exécutez se base sur des charges de travail correctement dimensionnées afin que les recommandations ne soient pas émises en fonction d'informations partielles.

Question : pourquoi est-il important de paramétrier une charge de travail entière plutôt que des requêtes individuelles ?

Options de l'Assistant Paramétrage du moteur de base de données

- Limiter la durée d'analyse
 - L'analyse peut prendre beaucoup de temps pour les charges de travail et les schémas volumineux
- Déterminer les types de recommandations à retourner



Points clés

L'Assistant Paramétrage du moteur de base de données propose un vaste ensemble d'options de configuration qui vous permettent de configurer l'analyse à exécuter et la façon dont les recommandations d'optimisation doivent être émises.

L'exécution de l'Assistant Paramétrage du moteur de base de données sur des charges de travail volumineuses peut prendre beaucoup de temps, en particulier sur les systèmes qui contiennent également de nombreux objets de base de données. Vous pouvez configurer l'Assistant Paramétrage du moteur de base de données pour limiter la durée de l'analyse et pour retourner les résultats obtenus à cette limite de durée.

Vous pouvez également configurer les types de recommandations à émettre, et si vous souhaitez consulter ou non les recommandations qui impliquent la suppression d'objets existants.

Analyse exploratoire

Les administrateurs de base de données peuvent aussi utiliser l'Assistant Paramétrage du moteur de base de données pour effectuer une analyse exploratoire. L'analyse exploratoire associe le paramétrage manuel et le paramétrage assisté par des outils. Pour effectuer une analyse exploratoire avec l'Assistant Paramétrage du moteur de base de données, utilisez la fonction de configuration spécifiée par l'utilisateur. La fonction de configuration spécifiée par l'utilisateur vous permet de spécifier des fonctionnalités de paramétrage pour les structures de conception physique existantes et hypothétiques, telles que les index, les vues indexées et le partitionnement. L'avantage de spécifier des structures hypothétiques est que vous pouvez évaluer leurs effets sur vos bases de données sans avoir besoin de les implémenter dans un premier temps.

Vous pouvez créer un fichier de configuration XML pour spécifier une configuration hypothétique. La configuration peut être utilisée pour l'analyse. L'analyse peut être exécutée en isolement ou en relation avec la configuration en cours. Ce type d'analyse peut également être effectuée à l'aide d'une interface de ligne de commande.

Question : quel est l'inconvénient de limiter la durée d'analyse ?

Démonstration 2A : Utilisation de l'Assistant Paramétrage du moteur de base de données

Dans cette démonstration, vous allez apprendre à utiliser l'Assistant Paramétrage du moteur de base de données pour générer des recommandations sur l'indexation et le partitionnement d'une requête

Configuration de la 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**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_17_PRJ\22462A_17_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 – Démonstration 2A.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

Question : devez-vous immédiatement appliquer les recommandations émises par DETA sur votre serveur ?

Leçon 3

Utilisation des options de suivi

- Vue d'ensemble de Trace SQL
- Comparaison Trace SQL/SQL Server Profiler
- Démonstration 3A : Configuration de Trace SQL
- Récupération de la sortie d'une trace
- Relecture des traces
- Trace par défaut
- Combinaison des traces avec les journaux de l'Analyseur de performances
- Démonstration 3B : Combinaison des traces avec les journaux de l'Analyseur de performances

Dans de nombreuses grandes organisations, vous ne pourrez pas utiliser l'outil graphique de SQL Server Profiler dans les environnements de production car il place trop haut une charge sur les systèmes profilés.

Il est encore possible, toutefois, d'utiliser l'interface de programmation Trace SQL basée sur les procédures stockées système, pour créer des traces plus légères. Il est important de savoir comment utiliser ces traces au lieu de SQL Server Profiler et comment utiliser SQL Server profiler pour vous aider à configurer Trace SQL.

Objectifs

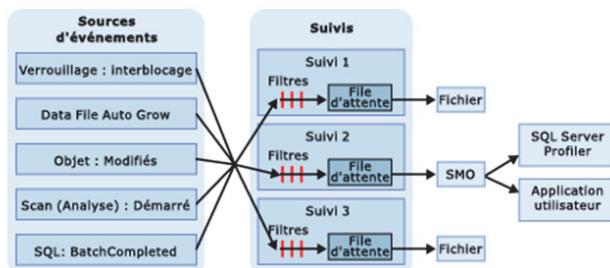
À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- décrire Trace SQL ;
- décrire les différences entre Trace SQL et SQL Server Profiler ;
- récupérer la sortie d'une trace ;
- relire les traces ;
- utiliser la trace par défaut ;
- combiner des traces avec les journaux de l'analyseur de performances et de fiabilité.

Vue d'ensemble de Trace SQL

Trace SQL est une interface de programmation basé sur les procédures stockées système qui est utilisée pour capturer l'activité de SQL Server

- SQL Server Profiler peut être utilisé pour créer des scripts pour les traces SQL
- Les événements peuvent être envoyés vers des fichiers ou des objets SMO
- SQL Server Profiler utilise SMO pour créer et utiliser des traces



Points clés

Trace SQL est une fonction exécutée dans le moteur de base de données pour créer et exécuter des traces. Les traces sont gérées à l'aide de procédures stockées système. En interne, SQL Server Profiler effectue des appels à la fonctionnalité de Trace SQL dans SQL Server lorsque SQL Server Profiler doit créer ou gérer des traces.

Les traces s'exécutent dans le processus du moteur de base de données SQL Server et peuvent écrire des événements dans un fichier ou une application à l'aide des objets SMO (SQL Server Management Objects).

Les informations que vous avez obtenues sur la façon dont les événements, les colonnes et filtrage fonctionnent dans SQL Server Profiler s'appliquent également pour leur fonctionnement dans Trace SQL.

L'implémentation des traces peut à priori paraître difficile car vous devez effectuer de nombreux appels aux procédures stockées pour définir et exécuter une trace. Toutefois, l'interface graphique de SQL Server Profiler peut être utilisée pour créer une trace, puis pour écrire le script de la trace à utiliser avec Trace SQL. Très peu de modifications doivent généralement être apportées aux fichiers de script Trace SQL créés par SQL Server Profiler, telles que le chemin d'accès des fichiers de sortie.

Comparaison Trace SQL/SQL Server Profiler

Trace SQL	SQL Server Profiler
<ul style="list-style-type: none">Défini à l'aide de procéduresS'exécute directement dans le moteur de base de donnéesEnregistre les événements dans des fichiers ou SMOUtilisé pour<ul style="list-style-type: none">Analyse à long termeTraces axées sur les performancesTraces de taille importante	<ul style="list-style-type: none">Utilisation avec un outil graphiqueUtilise Trace SQLEnregistre les données dans des fichiers ou des tables de base de donnéesUtilisé pour<ul style="list-style-type: none">Débogage sur des systèmes de testAnalyse à court termeTraces de petite taille

Points clés

Il est important de comprendre les différences entre Trace SQL et SQL Server Profiler et de comprendre quand chaque outil doit être utilisé.

- Trace SQL doit être défini à l'aide d'une série d'appels de procédure stockée système, alors que SQL Server Profiler fournit une interface graphique pour configurer et contrôler le traçage de l'activité.
- Trace SQL s'exécute directement dans le moteur de base de données alors que SQL Server Profiler s'exécute sur un système client (ou sur le serveur) et communique avec le moteur de base de données en employant les procédures de Trace SQL.
- Trace SQL peut écrire des événements dans des fichiers ou des applications basées sur SMO, alors que SQL Server Profiler peut écrire des événements dans des fichiers ou des tables de base de données.
- Trace SQL est utile pour les traces longues et ayant un impact critique sur les performances, ou pour les traces très volumineuses qui impactent considérablement les performances du système cible, alors que SQL Server Profiler est plus généralement utilisé pour déboguer sur des systèmes de test, pour effectuer des analyses de court terme ou pour capturer des petites traces.



Remarque L'option « Le serveur traite les données de trace » dans SQL Server Profiler n'est pas la même option qui consiste à écrire le script d'une trace et à la lancer via des procédures stockées. L'option crée deux traces : une trace qui écrit directement dans un fichier et une deuxième trace pour envoyer les événements via SMO dans SQL Server Profiler.

Les traces ne redémarrent pas automatiquement après le redémarrage de l'instance de serveur. Si une trace doit être exécutée en permanence, il faut écrire le script de la trace, la lancer à partir d'une procédure stockée et la procédure stockée doit être signalée comme procédure de démarrage.

Question : quelle option aura moins d'impact sur un système tracé ?

- a) Trace SQL avec sortie vers un fichier.
- b) Le Générateur de profils SQL avec sortie dans une table.

Démonstration 3A : Configuration de Trace SQL

Dans cette démonstration, vous allez apprendre à

- Créer un script pour une trace avec SQL Server Profiler
- Démarrer Trace SQL
- Afficher une trace SQL avec SQL Server Profiler

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**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_17_PRJ\22462A_17_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.

Récupération de la sortie d'une trace

- Les traces SQL Server sont écrites dans les fichiers
- Les traces peuvent être analysées en procédant comme suit
 - Ouvrez la trace dans SQL Server Profiler
 - Importez la trace dans une table SQL Server

```
CREATE TABLE dbo.tracetable
(
    TextData nvarchar(max) NULL,
    BinaryData varbinary(max) NULL,
    ...
);
INSERT INTO dbo.tracetable
SELECT * FROM
fn_trace_gettable('L:\Traces\adworks.trc', default);
```

Points clés

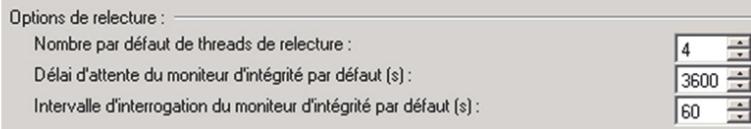
Les traces qui sont écrites dans des fichiers de trace ne sont pas faciles à lire directement et sont quelque peu difficiles à analyser à partir d'une application.

Deux options sont proposées par le Générateur de profils SQL pour faciliter l'utilisation du contenu des fichiers de trace :

- SQL Server Profiler peut être utilisé pour ouvrir des fichiers de trace. Avec SQL Server Profiler, la sortie de trace peut être filtrée et groupée pour être analysée. SQL Server Profiler est particulièrement utile lorsque vous travaillez avec des petits fichiers de trace.
- Les fichiers de trace peuvent être importés dans SQL Server à l'aide de la fonction système fn_trace_gettable. La lecture des fichiers dans une table est particulièrement utile lorsque vous devez analyser de grandes quantités de données de trace, car la table peut être indexée pour améliorer la vitesse des requêtes courantes sur les données capturées. Les requêtes T-SQL peuvent être utilisées pour analyser et filtrer les données.

Relecture des traces

- Des événements et colonnes spécifiques doivent figurer dans le fichier de trace
 - Utiliser le modèle de trace TSQL_Replay
- SQL Server Profiler fournit un moteur de lecture pour reproduire l'activité
 - Valider les modifications de configuration
 - Tester les modifications des performances



- SQL Server 2012 propose un utilitaire de relecture distribué
 - Simulation des ordinateurs clients simultanés
 - Simulation des charges de travail critiques

Points clés

SQL Server Profiler permet de relire les fichiers de trace. La capacité de relire les fichiers de trace est utile pour valider les modifications que vous envisagez d'effectuer sur un système ou pour tester une charge de travail sur du nouveau matériel, des index ou des modifications physiques de mise en page.

La relecture ne doit pas nécessairement être exécutée sur le même système sur lequel les événements de trace ont été capturés, mais le système doit être configuré de manière très similaire. Cela s'applique en particulier aux objets tels que les bases de données et les connexions.

Distributed Replay

SQL Server 2012 a introduit une fonctionnalité Distributed Replay pour les traces capturées par SQL Server Profiler ou Trace SQL. Distributed Replay ne se limite pas à la relecture d'une charge de travail à partir d'un seul ordinateur et propose une solution plus évolutive que l'option de relecture fournie par SQL Server Profiler. Cette fonctionnalité peut améliorer la simulation de charges de travail critiques.



Remarque En plus des fonctionnalités de relecture de SQL Server Profiler et de l'utilitaire Distributed Replay, un ensemble d'utilitaires non pris en charge proposés par le groupe de support produit SQL Server sont disponibles. Ces utilitaires non pris en charge sont appelés les Utilitaires RML et peuvent relire des charges de travail SQL Server avec des fonctionnalités avancées comme la relecture synchronisée via plusieurs clients. Les utilitaires RML peuvent également être utilisés pour analyser les fichiers de trace et sont téléchargeables sur le site support.microsoft.com.

Avant de pouvoir relire des traces capturées, vous devez vérifier que toutes les colonnes nécessaires à la relecture ont bien été capturées. Pour rendre cette opération plus simple, SQL Server Profiler propose un modèle appelé TSQL Replay. Vous pouvez utiliser ce modèle comme point de départ, mais assurez-vous de ne supprimer aucune des colonnes incluses dans le modèle.

Question : quand est-il judicieux de tester des systèmes à l'aide de la fonctionnalité de relecture ?

Trace par défaut

- Assure principalement le suivi des changements de configuration à des fins de dépannage
 - Enregistrée dans le répertoire du journal de SQL Server
 - Utilise des fichiers de substitution
- Activée par défaut
 - Peut être configurée avec sp_configure

```
EXEC sp_configure 'show advanced options' , 1;
RECONFIGURE;
```

```
EXEC sp_configure 'default trace enabled' , 1;
RECONFIGURE;
```

```
EXEC sp_configure 'show advanced options' , 0;
RECONFIGURE;
```

Points clés

Depuis SQL Server 2005, une trace côté serveur par défaut est démarrée automatiquement chaque fois que SQL Server démarre. Cette trace est appelée la trace par défaut et a l'ID de trace 1. La trace par défaut est une trace légère qui conserve uniquement 5 Mo de données à n'importe quel moment. Elle capture les événements suivants :

Objet	Événements
Base de données	Data file auto grow, Data file auto shrink, Database mirroring status change, Log file auto grow, Log file auto shrink
Erreurs et avertissements	Errorlog, Hash warning, Missing Column Statistics, Missing Join Predicate, Sort Warning
Texte intégral	FT Crawl Aborted, FT Crawl Started, FT Crawl Stopped
Objets	Object Altered, Object Created, Object Deleted

(suite)

Objet	Événements
Audit de sécurité	Audit de l'événement Add DB user Audit de l'événement Add login to server role Audit de l'événement Add member to DB role Audit de l'événement Add role Audit de l'événement Add login Audit de l'événement Backup/Restore Audit Change Database owner Audit de l'événement DBCC Audit de l'événement Database Scope GDR Audit de l'événement Login Change Property Audit Login Failed Audit de l'événement Login GDR Audit de l'événement Schema Object GDR Audit Schema Object Take ownership Audit Server Starts and Stops
Serveur	Server Memory Change

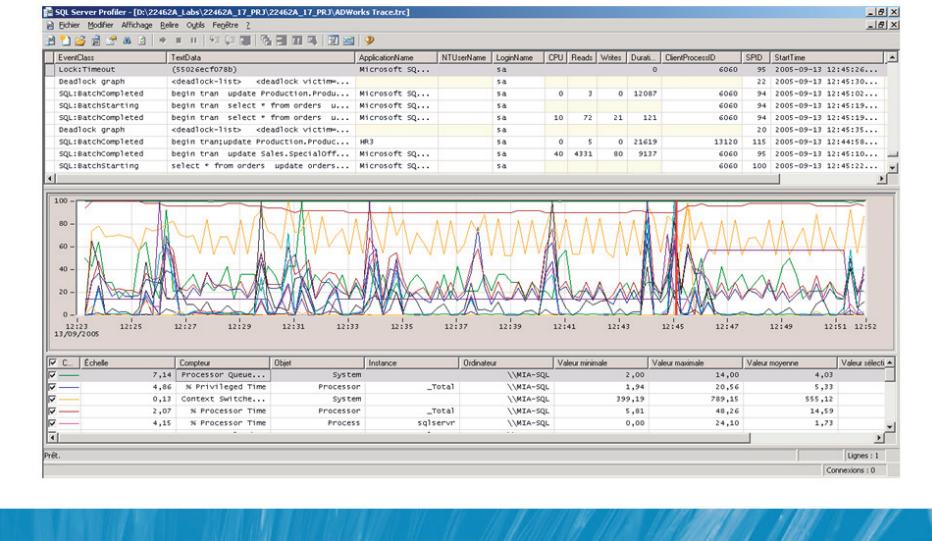
La trace par défaut peut être activée ou désactivée à l'aide de sp_configure. Dans l'exemple de la diapositive, vous pouvez voir comment réactiver la trace par défaut si nécessaire.

La trace par défaut ne peut pas être modifiée, ainsi que l'emplacement du fichier.

Question : la trace par défaut doit-elle être toujours activée ?

Combinaison des traces avec les journaux de l'Analyseur de performances

- Mettre en corrélation l'Analyseur de performances avec les événements SQL Trace dans SQL Server Profiler



Points clés

Avec SQL Server Profiler, vous pouvez ouvrir un journal de performances Microsoft Windows®, choisir les compteurs à mettre en corrélation avec une trace et afficher les compteurs de performances sélectionnés en même temps que la trace dans l'interface utilisateur graphique SQL Server Profiler. Lorsque vous sélectionnez un événement dans la fenêtre de la trace, une barre verticale rouge dans le volet Moniteur système de SQL Server Profiler indique les données du journal de performances en corrélation avec l'événement de trace sélectionné.

Pour mettre en corrélation une trace avec des compteurs de performances, ouvrez un fichier ou une table de trace contenant les colonnes de données StartTime et EndTime, puis cliquez sur Importer les données de performances dans le menu Fichier de SQL Server Profiler. Vous pouvez alors ouvrir un journal de performances, puis sélectionner les objets et compteurs du Moniteur système que vous voulez corrélérer avec la trace.

Question : quel est le principal avantage de l'utilisation de cette fonctionnalité ?

Démonstration 3B : Combinaison des traces avec les journaux de l'Analyseur de performances

Dans cette démonstration, vous allez apprendre à combiner une trace SQL Server avec un journal de l'Analyseur de performances

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**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_17_PRJ\22462A_17_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 **32 – Démonstration 3B.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

Atelier pratique 17 : Trace de l'accès à SQL Server

- Exercice 1 : Capturer une trace avec SQL Server Profiler
- Exercice 2 : Analyser une trace avec l'Assistant Paramétrage du moteur de base de données
- Exercice difficile 3 : Configurer Trace SQL
(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_17_PRJ\22462A_17_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

Les développeurs de la nouvelle application marketing sont préoccupés par les performances de leurs requêtes. Lorsque les développeurs testaient l'application, ils travaillaient avec une petite quantité de données et les performances étaient acceptables. Ils ne sont pas certains d'avoir créé les index appropriés pour prendre en charge l'application.

Vous allez utiliser SQL Server Profiler pour capturer les traces de l'exécution de l'application. Vous analyserez ensuite les traces à l'aide de l'Assistant Paramétrage du moteur de base de données.

Si vous avez le temps, vous configurerez des traces à l'aide de procédures stockées système de Trace SQL.

Exercice 1 : Capturer une trace avec SQL Server Profiler

Scénario

Vous avez isolé la charge de travail à l'origine des performances médiocres et vous devez tester les requêtes dans un environnement de développement. Dans cet exercice, vous allez générer une exemple de charge de travail et capturer l'activité de la base de données avec SQL Server Profiler.

Les tâches principales de cet exercice sont les suivantes :

1. Créer et démarrer une trace SQL Server Profiler appropriée.
2. Exécuter la charge de travail.
3. Arrêter la trace.

► Tâche 1 : Créer et démarrer une trace SQL Server Profiler appropriée

- Créez une trace SQL Server Profiler en fonction des paramètres suivants :
 - Modèle de paramétrage
 - Filtré dans la base de données MarketDev uniquement
 - Fichiers de substitution désactivés
 - La taille maximum du fichier de trace est de 500 Mo
- Démarrez la trace.

► Tâche 2 : Exécuter la charge de travail

- Ouvrez et exécutez le fichier 51 – Lab Exercise 1.sql qui est un fichier de charge de travail à analyser.



Remarque Assurez-vous que vous configurez la fenêtre de la charge de travail d'après les instructions en haut du script.

► Tâche 3 : Arrêter la trace

- Lorsque la requête se termine, arrêtez la trace.
- Fermez SQL Server Profiler.

Résultats : À la fin de cet exercice, vous devez avoir capturé une charge de travail à l'aide de SQL Server Profiler.

Exercice 2 : Analyser une trace avec l'Assistant Paramétrage du moteur de base de données

Scénario

Vous allez consulter les résultats capturés à partir de Trace SQL et identifier les modifications à apporter pour améliorer les performances. Dans cet exercice, vous analyserez l'activité capturée à l'aide de l'Assistant Paramétrage du moteur de base de données.

Les tâches principales de cet exercice sont les suivantes :

1. Analyser la trace capturée dans l'Assistant Paramétrage du moteur de base de données.
2. Examiner les modifications suggérées.

► **Tâche 1 : Analyser la trace capturée dans l'Assistant Paramétrage du moteur de base de données**

- Analysez le fichier de trace capturé à l'aide de l'Assistant Paramétrage du moteur de base de données.

► **Tâche 2 : Examiner les modifications suggérées**

- Examiner les modifications suggérées.
- Fermez l'Assistant Paramétrage du moteur de base de données.

Résultats : À la fin de cet exercice, vous devez avoir analysé la trace à l'aide de l'Assistant Paramétrage du moteur de base de données et avoir examiné les modifications suggérées.

Exercice difficile 3 : Configurer Trace SQL (si le temps le permet)

Scénario

Vous avez remarqué que lorsque le Générateur de profils SQL est en cours d'exécution sur le serveur de production, ses performances sont réduites. Vous souhaitez capturer les mesures de performances et réduire l'impact sur le serveur. Dans cet exercice, vous allez capturer l'activité en utilisant les procédures stockées de Trace SQL pour illustrer le mode d'utilisation à la place du Générateur de profils SQL, pour réduire l'impact sur le serveur.

Les tâches principales de cet exercice sont les suivantes :

1. Créer un script qui utilise les procédures de Trace SQL pour implémenter le même type de capture que vous avez effectué dans l'exercice 1 mais avec un nom de trace différent.
2. Tester que le script s'exécute comme prévu en utilisant la même charge de travail.
3. Analyser la nouvelle sortie capturée et noter si les modifications suggérées sont identiques à celles suggérées dans l'exercice 2.

► **Tâche 1 : Créez un script qui utilise les procédures de Trace SQL pour implémenter le même type de capture que vous avez effectué dans l'exercice 1 mais avec un nom de trace différent**

- Créez, démarrez et arrêtez la trace même que celle utilisés dans l'exercice 1, mais appelez la trace ProsewareTrace2.
- Exportez la définition de trace dans un fichier.

► **Tâche 2 : Tester que le script s'exécute comme prévu en utilisant la même charge de travail**

- Configurez et démarrez le fichier de définition de trace enregistré. Notez l'ID de la trace.
- Exécutez à nouveau le fichier de charge de travail 51 – Lab Exercise 1.sql.

► **Tâche 3 : Analyser la nouvelle sortie capturée et noter si les modifications suggérées sont identiques à celles suggérées dans l'exercice 2**

- Arrêter la trace.
- Ouvrez et examinez le fichier de trace capturés.

Résultats : À la fin de cet exercice, vous devez avoir capturé une trace à l'aide de Trace SQL.

Contrôle des acquis du module et éléments à retenir

- Questions de contrôle des acquis
- Meilleures pratiques

Questions de contrôle des acquis

1. Quel est l'objectif principal de l'Assistant Paramétrage du moteur de base de données ?
2. Que pouvez-vous utiliser pour tester la charge de travail après les modifications de configuration ?

Meilleures pratiques

1. Utilisez SQL Server Profiler pour effectuer des traces de petite taille pour le débogage et d'autres opérations.
2. Utilisez Trace SQL pour des traces importantes et dont l'exécution est longue.
3. Utilisez SQL Server Profiler pour définir les traces et les mettre dans un script pour Trace SQL.
4. Importez les données de trace dans une table de base de données pour une analyse avancée.
5. Utilisez l'Assistant Paramétrage du moteur de base de données pour analyser la base de données en fonction d'une charge de travail, plutôt qu'en vous concentrant sur des requêtes individuelles.

Module 18

Surveillance de SQL Server 2012

Sommaire :

Leçon 1 : Surveillance de l'activité	18-3
Leçon 2 : Capture et gestion des données de performances	18-15
Leçon 3 : Analyse des données de performances collectées	18-25
Atelier pratique 18 : Surveillance de SQL Server 2012	18-34

Vue d'ensemble du module

- Surveillance de l'activité
- Capture et gestion des données de performances
- Analyse des données de performances collectées

Le moteur de base de données Microsoft® SQL Server® peut s'exécuter sur de très longues périodes sans nécessiter d'administration. Toutefois, un processus de surveillance continue de l'activité sur le serveur de base de données vous permet de gérer de manière proactive les problèmes potentiels avant qu'ils ne se produisent.

SQL Server fournit plusieurs outils qui peuvent être utilisés pour surveiller l'activité en cours et enregistrer les détails de l'activité précédente. Vous devez vous familiariser avec la fonction de chacun des outils, et leur utilisation.

Lorsque vous enregistrez l'activité à l'aide des outils de surveillance, il est facile d'être submergé par le volume de sortie qu'un outil de surveillance peut fournir. Vous devez également apprendre les techniques d'analyse de la sortie fournie par les outils de surveillance.

Objectifs

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

- surveiller l'activité en cours ;
- capturer et gérer les données de performances ;
- analyser les données de performances collectées.

Leçon 1

Surveillance de l'activité

- Vue d'ensemble des vues et fonctions de gestion dynamique
- Affichage de l'activité à l'aide des vues de gestion dynamique
- Démonstration 1A : Affichage de l'activité à l'aide des vues de gestion dynamique
- Utilisation du Moniteur d'activité dans SQL Server Management Studio
- Démonstration 1B : Utilisation du Moniteur d'activité dans SQL Server Management Studio
- Utilisation de l'Analyseur de performances
- Utilisation des compteurs SQL Server
- Démonstration 1C : Utilisation de l'Analyseur de performances

Les vues de gestion dynamique (DMV, Dynamic Management Views) et les fonctions de gestion dynamique (DMF, Dynamic Management Functions) donnent un aperçu des opérations internes du moteur de base de données SQL Server et sont utiles pour la surveillance. Il est important que les administrateurs de base de données SQL Server se familiarisent avec quelques-unes des vues et fonctions de gestion dynamique les plus utiles dans le cadre d'un processus de surveillance continue du serveur.

SQL Server Management Studio (SSMS) fournit le Moniteur d'activité qui peut être utilisé pour examiner les problèmes en cours tels que « Un processus est-il bloqué par un autre processus ? » et les problèmes d'historique récents tels que « Quelle requête a utilisé le plus de ressources depuis le dernier redémarrage du serveur ? » Vous devez vous familiariser avec les fonctionnalités du Moniteur d'activité.

Les processus SQL Server exposent également un jeu d'objets et de compteurs de performance à l'Analyseur de performances Windows. Ces objets et compteurs vous permettent de surveiller SQL Server dans le cadre de la surveillance de l'intégralité du serveur.

Objectifs

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

- expliquer les vues et fonctions de gestion dynamique ;
- afficher l'activité à l'aide des vues de gestion dynamique ;
- utiliser le Moniteur d'activité dans SSMS ;
- utiliser l'Analyseur de performances ;
- utiliser les compteurs SQL Server.

Vue d'ensemble des vues et fonctions de gestion dynamique

Les objets de gestion dynamique représentent des vues et des fonctions virtuelles qui fournissent des données sur l'état des systèmes SQL Server

- Organisation par catégorie

Catégorie	Description
sys.dm_exec_%	Exécution et connexion
sys.dm_os_%	Informations liées au système d'exploitation de SQL
sys.dm_tran_%	Gestion des transactions
sys.dm_io_%	Informations liées aux entrées/sorties
sys.dm_db_%	Informations dont l'étendue est définie au niveau de la base de données

- Nombreuses autres catégories disponibles

Points clés

Dans les versions antérieures de SQL Server, les administrateurs de base de données utilisaient souvent des outils tiers pour surveiller l'état interne de SQL Server. La plupart des outils tiers effectuaient cette surveillance à l'aide de procédures stockées étendues. L'utilisation des procédures stockées étendues n'est pas souhaitable, car elles s'exécutent dans l'espace mémoire du processus SQL Server.

Les programmes mal écrits qui s'exécutent dans ces zones de mémoire peuvent provoquer l'instabilité ou le blocage de SQL Server.

SQL Server 2005 et les versions ultérieures fournissent des objets de gestion dynamique pour donner un aperçu du fonctionnement interne du moteur de base de données sans qu'il soit nécessaire d'utiliser des procédures stockées étendues. Certains des objets ont été créés en tant que vues et sont appelés des vues de gestion dynamique (DMV). D'autres objets ont été créés en tant que fonctions et sont appelés des fonctions de gestion dynamique (DMF).



Remarque Les informations exposées par les vues et fonctions de gestion dynamique ne sont pas généralement conservées dans la base de données, comme c'est le cas avec les vues de catalogue. Les vues et fonctions sont des objets virtuels qui retournent des informations relatives à l'état. L'état est désactivé lorsque l'instance du serveur est redémarrée.

Vues et fonctions de gestion dynamique

Les vues et fonctions de gestion dynamique retournent des informations sur l'état du serveur qu'il est possible d'utiliser pour surveiller l'état d'une instance du serveur, diagnostiquer des problèmes et améliorer les performances. Il existe deux types de vues et fonctions de gestion dynamique :

- les vues et fonctions de gestion dynamique dont l'étendue est définie au niveau du serveur ;
- les vues et fonctions de gestion dynamique dont l'étendue est définie au niveau de la base de données.

Toutes les vues et fonctions de gestion dynamique existent dans le schéma sys et respectent la convention d'affectation des noms dm_%. Elles sont définies dans la base de données de ressources masquée et sont mappées aux autres bases de données. Les vues et fonctions de gestion dynamique sont organisées en un ensemble de catégories :

Catégorie	Description
sys.dm_exec_%	Ces objets fournissent des informations sur les connexions, les sessions, les demandes et l'exécution des requêtes. Par exemple, sys.dm_exec_sessions fournit une ligne pour chaque session actuellement connectée au serveur.
sys.dm_os_%	Ces objets fournissent un accès aux informations relatives au système d'exploitation SQL. Par exemple, sys.dm_os_performance_counters fournit un accès aux compteurs de performance SQL Server sans avoir à y accéder à l'aide des outils du système d'exploitation.
sys.dm_tran_%	Ces objets fournissent un accès à la gestion des transactions. Par exemple, sys.dm_os_tran_active_transactions fournit des informations sur les transactions actuellement actives.
sys.dm_io_%	Ces objets fournissent des informations sur les processus d'E/S. Par exemple, sys.dm_io_virtual_file_stats fournit des informations sur les performances et les statistiques d'E/S pour chaque fichier de base de données.
sys.dm_db_%	Ces objets fournissent des informations relatives à l'étendue de la base de données. Par exemple, sys.dm_db_index_usage_stats fournit des informations sur l'utilisation de chaque index de la base de données.

Autorisations requises

Pour interroger une vue ou fonction de gestion dynamique, une autorisation SELECT sur l'objet est requise, ainsi qu'une autorisation VIEW SERVER STATE ou VIEW DATABASE STATE, selon que l'objet a une étendue définie au niveau du serveur ou de la base de données. Cela permet de limiter sélectivement l'accès d'un utilisateur ou d'une connexion aux fonctions et vues de gestion dynamique. Pour contrôler l'accès d'un utilisateur, créez d'abord l'utilisateur dans master (à l'aide d'un nom d'utilisateur), puis refusez à cet utilisateur l'autorisation SELECT sur les vues et fonctions de gestion dynamique auxquelles vous souhaitez interdire l'accès. Une fois cette opération effectuée, l'utilisateur ne peut pas sélectionner ces vues et fonctions de gestion dynamique, quel que soit le contexte de la base de données de l'utilisateur car la commande DENY dans ce contexte de base de données est traitée en premier.

Question : quelles sont les raisons qui justifient l'utilisation des vues de gestion dynamique pour afficher l'activité en cours ?

Affichage de l'activité à l'aide des vues de gestion dynamique

- Doivent être référencées dans le schéma sys
- Deux types de base
 - Informations sur l'état actuel
 - Informations historiques accumulées

```
SELECT s.original_login_name, s.program_name,
       t.wait_type, t.wait_duration_ms
  FROM sys.dm_os_waiting_tasks AS t
 INNER JOIN sys.dm_exec_sessions AS s
    ON t.session_id = s.session_id
 WHERE s.is_user_process = 1
   AND t.wait_duration_ms > 3000;

SELECT * FROM sys.dm_os_wait_stats
 ORDER BY wait_time_ms DESC;
```

Points clés

Vous pouvez afficher la liste des vues de gestion dynamique disponibles dans l'Explorateur d'objets dans le nœud Vues système d'une base de données. Les fonctions de gestion dynamique s'affichent dans le nœud Fonctions table sous le nœud Fonctions système de la base de données master.



Remarque Les vues et fonctions de gestion dynamique doivent être référencées dans les instructions T-SQL en utilisant le schéma sys comme préfixe. Elles ne peuvent pas être référencées par des noms en une seule partie.

Il existe deux types d'objets de gestion dynamique de base :

- les objets qui retournent des informations d'état en temps réel à partir du système ;
- les objets qui retournent des informations d'historique récentes.

Objets qui retournent des informations d'état en temps réel à partir du système

La plupart des vues et fonctions de gestion dynamique sont conçues pour fournir des informations sur l'état actuel du système. Dans l'exemple illustré sur la diapositive, deux vues de gestion dynamique sont jointes. La vue sys.dm_exec_sessions retourne une ligne pour chaque session d'utilisateur en cours. La vue sys.dm_os_waiting_tasks retourne une ligne pour chaque tâche en attente sur une ressource. En joignant les deux vues et en ajoutant un filtre, vous pouvez rechercher une liste de tâches d'utilisateur en attente depuis plus de 3 000 millisecondes (3 secondes).

-  **Remarque** Lorsqu'une tâche doit attendre une ressource, elle est envoyée vers une liste d'attente. Elle reste sur cette liste jusqu'à ce qu'elle reçoive un signal lui indiquant que la ressource demandée est maintenant disponible. Elle est ensuite renvoyée à la liste d'exécution, où elle attend d'être planifiée pour une nouvelle exécution. Cette analyse du type d'attente est très utile lors du réglage des performances système, car elle vous permet d'identifier les goulets d'étranglement du système.

Dans de nombreux cas, lorsqu'une tâche est en attente, la cause de l'attente est une forme de verrou. Les verrous sont étudiés en détail dans le module 20.

Objets qui retournent des informations relatives à l'historique

Le second type d'objet de gestion dynamique retourne des informations relatives à l'historique. Par exemple, vous avez appris que la vue sys.dm_os_waiting_tasks a retourné les détails des tâches en attente sur les ressources. Par contre, la vue sys.dm_os_wait_stats retourne des informations sur la fréquence et le délai d'attente d'une tâche pour un wait_type spécifique depuis le démarrage de l'instance SQL Server.

Un autre exemple utile d'une fonction d'historique est la fonction sys.dm_io_virtual_file_stats() qui retourne des informations sur les performances des fichiers de base de données.

Question : quels types de ressources SQL Server devra-t-il attendre ?

Démonstration 1A : Affichage de l'activité à l'aide des vues de gestion dynamique

Dans cette démonstration, vous allez apprendre à utiliser les vues de gestion dynamique pour afficher les informations sur les performances

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**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_18_PRJ\22462A_18_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.

Utilisation du Moniteur d'activité dans SQL Server Management Studio

The screenshot displays the SQL Server Management Studio Activity Monitor. It includes four main panes:

- Vue d'ensemble:** Shows four graphs: % du temps processeur (CPU), Tâches en attente (Tasks waiting), ES de la base de données (I/O), and Nombre de requêtes de lots (Batch processing).
- Processus:** A table listing active processes with columns for ID, Process ID, Command, Application, CPU time, Type, and Status.
- Attentes de ressources:** Displays resource wait statistics.
- E/S du fichier de données:** Shows data file I/O statistics.

 The process table data is as follows:

ID	Process ID	Connexions	Base de données	Connexions	Application	CPU	Type	Statut	Procédures stockées	Nombre de requêtes	Génération
51	1 ADVENTURER	maine	AdventureworksLT	0	Microsoft SQL Server	0	0	0	0	0	16 Mo SQL1
52	1 ADVENTURER	maine	AdventureworksLT	0	Microsoft SQL Server	0	0	0	0	0	16 Mo SQL1
53	1 ADVENTURER	maine	AdventureworksLT	0	Microsoft SQL Server	0	0	0	0	0	16 Mo SQL1
54	1 ADVENTURER	maine	AdventureworksLT	0	Microsoft SQL Server	0	0	0	0	0	16 Mo SQL1
55	1 ADVENTURER	requête	RUNNING SELECT	0	Microsoft SQL Server	0	0	0	0	0	16 Mo SQL1
56	1 ADVENTURER	requête	Report Server	0	Report Server	0	0	0	0	0	16 Mo SQL1
57	1 ADVENTURER	requête	SQLAgent - Empl	0	Report Server	0	0	0	0	0	16 Mo SQL1
58	1 ADVENTURER	requête	Report Server	0	Report Server	0	0	0	0	0	16 Mo SQL1
61	1 ADVENTURER	MakaleDev	SUSPEN...	23 WAITFOR	Microsoft SQL Server	0	23 WAITFOR	0	0	0	16 Mo SQL1

- Affiche des informations sur les processus SQL Server, les attentes, les entrées/sorties et les requêtes coûteuses
- Accède à SQL Server via les objets de gestion dynamique
- Nécessite l'autorisation VIEW SERVER STATE
- Peut être utilisé pour arrêter les processus qui bloquent les autres

Points clés

Le Moniteur d'activité est un outil dans SQL Server Management Studio qui affiche des informations sur les processus, les attentes, les performances des ressources d'E/S et les requêtes coûteuses récentes. Pour démarrer le Moniteur d'activité, dans SQL Server Management Studio, cliquez avec le bouton droit sur le nom du serveur, puis cliquez sur Moniteur d'activité.

Le Moniteur d'activité comporte cinq sections :

- La section Vue d'ensemble contient des informations graphiques sur l'utilisation du processeur, les tâches en attente, les E/S de la base de données et les requêtes de traitement par lots par seconde.
- La section Processus inclut les informations détaillées sur les processus et leurs ID, les connexions, les bases de données et les commandes. Cette section affiche également des informations sur les processus qui bloquent d'autres processus.
- La section Attentes de ressources affiche les catégories de processus en attente de ressources, ainsi que des informations sur les temps d'attente.
- La section E/S du fichier de données présente des informations sur les fichiers de base de données physiques en cours d'utilisation et leurs performances récentes.
- La section Requêtes coûteuses récentes affiche des informations détaillées sur les requêtes récentes les plus coûteuses et les ressources consommées par ces requêtes. Vous pouvez cliquer avec le bouton droit sur les requêtes de cette section pour afficher la requête ou un plan d'exécution de la requête.

Vous pouvez filtrer les données affichées en cliquant sur les en-têtes de colonne et choisissant le paramètre sur lequel afficher des informations.

Question : pourquoi est-il important de surveiller les attentes ?

Démonstration 1B : Utilisation du Moniteur d'activité dans SQL Server Management Studio

Dans cette démonstration, vous allez apprendre à

- Comment utiliser le Moniteur d'activité pour afficher des informations sur les processus ?
- Comment terminer un processus à l'aide du Moniteur d'activité ?

Procédure de démonstration

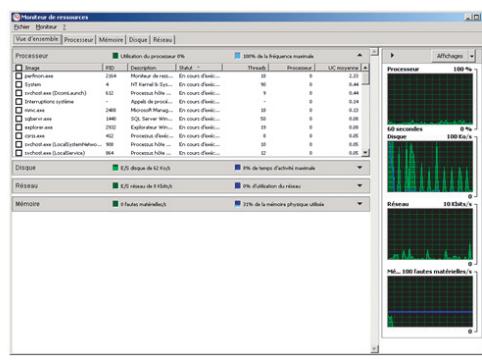
1. Si la démonstration 1A n'a pas été effectuée :
 - Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A_Labs\Revert.txt**.
 - Sur l'ordinateur virtuel, cliquez sur **Démarrer, Tous les programmes, Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**. Dans la fenêtre Se connecter au serveur, tapez **Proseware**, puis cliquez sur **Se conn.** Dans le menu **Fichier** cliquez sur **Ouvrir**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_18_PRJ\22462A_18_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 **12 – Demonstration 1B.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

Utilisation de l'Analyseur de performances

- Utilisé pour collecter et afficher les mesures système
- Fournit la surveillance en temps réel de l'intégrité et des compteurs de performance
- Crée des jeux d'éléments de collecte de données qui capturent les données de différents collecteurs de données, à la demande ou planifiés

Domaines clés à analyser

- UC
- Mémoire
- Système de disques
- Réseau
- Compteurs SQL Server



Points clés

L'Analyseur de performances Windows est un outil Microsoft Windows Server 2008 (et versions ultérieures) qui regroupe plusieurs outils d'analyse et de performances Microsoft Windows Server 2003 qui étaient auparavant disparates. En regroupant plusieurs sources d'informations à un même emplacement, l'Analyseur de performances Windows permet aux administrateurs d'obtenir les informations nécessaires pour le diagnostic des performances du serveur et des problèmes d'instabilité.

Les ensembles de collecteurs de données constituent une nouvelle fonctionnalité majeure de l'Analyseur de performances Windows. Ils regroupent des collecteurs de données en éléments réutilisables, destinés à être utilisés dans des scénarios d'analyse de performances spécifiques. Une fois qu'un groupe de collecteurs de données est stocké en tant qu'ensemble de collecteurs de données, il est possible d'appliquer des paramètres de configuration, telles qu'un paramètre de planification, à tout l'ensemble en ne modifiant qu'une seule propriété.

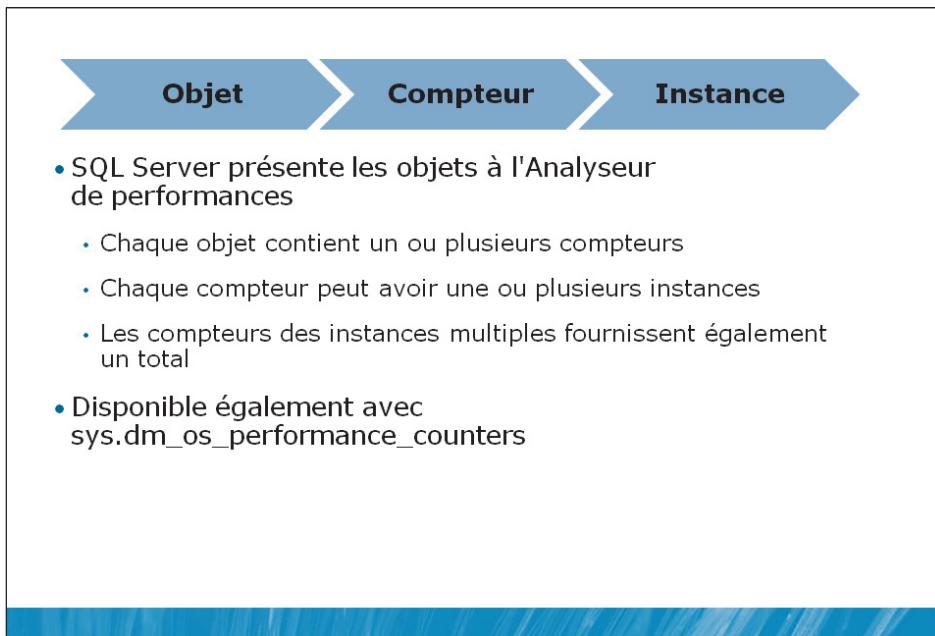
L'Analyseur de performances Windows contient également des modèles d'ensembles de collecteurs de données par défaut qui permettent aux administrateurs système de commencer immédiatement à rassembler des données de performances propres au rôle serveur ou à surveiller des scénarios.

L'Analyseur de performances Windows est le principal outil d'analyse des systèmes Microsoft Windows®. Étant donné que SQL Server s'exécute sur le système d'exploitation Windows, il est important d'effectuer une analyse au niveau du serveur et au niveau du moteur de base de données, car les problèmes rencontrés dans le moteur de base de données sont peut-être causés par des problèmes externes au moteur de base de données.

L'Analyseur de performances Windows se concentre principalement sur l'analyse du processeur, de la mémoire, du système de disques et du réseau. Après avoir installé SQL Server, plusieurs objets et compteurs SQL Server sont disponibles dans l'Analyseur de performances Windows.

Question : pourquoi l'Analyseur de performances est-il utile pour déterminer les problèmes de performances ?

Utilisation des compteurs SQL Server



Points clés

Il est important de comprendre la terminologie de base utilisée dans l'Analyseur de performances Windows :

- Le terme objet est utilisé pour représenter une ressource qui peut être analysée.
- Chaque objet expose un ou plusieurs compteurs.
- Un compteur peut avoir plusieurs instances si plusieurs ressources de ce type existent.

Par exemple, un objet appelé Processeur comprend plusieurs compteurs. L'objet Processeur fournit les mesures associées aux processeurs disponibles sur le serveur. L'un des compteurs couramment utilisés pour l'objet Processeur est le compteur « % du temps processeur ». Ce compteur offre un ensemble d'instances qui représentent les cœurs de processeur individuels présents sur le système. De plus, une valeur appelée _Total représente toutes les instances combinées.

Objets et compteurs d'application

De nombreuses applications exposent des données statistiques liées aux applications à l'Analyseur de performances. SQL Server expose un grand nombre d'objets et de compteurs.

Les objets SQL Server ont la convention d'affectation de noms suivante :

Format du nom d'objet	Utilisation
SQLServer:<nom de l'objet>	Utilisé pour les instances par défaut
MSSQL\$<instance>:<objet >	Utilisé pour les instances nommées
SQLAgent\$<instance >:<objet >	Utilisé pour SQL Server Agent

SQL Server fournit également les mêmes valeurs de compteur dans la vue de gestion dynamique sys.dm_os_performance_counters.

Question : pourquoi peut-il être utile d'interroger les compteurs de performance sys.dm_os_performance_counters pour accéder aux compteurs SQL Server au lieu de l'Analyseur de performances ?

Démonstration 1C : Utilisation de l'Analyseur de performances

Dans cette démonstration, vous allez apprendre à utiliser l'Analyseur de performances Windows

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**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_18_PRJ\22462A_18_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 **13 – Demonstration 1C.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

Leçon 2

Capture et gestion des données de performances

- Vue d'ensemble du collecteur de données
- Conception d'une topologie pour le collecteur de données
- Configuration du collecteur de données
- Sécurité du collecteur de données
- Surveillance du collecteur de données
- Démonstration 2A : Configuration du collecteur de données

Vous avez appris que les vues et fonctions de gestion dynamique fournissent des informations utiles sur l'état du système. Les valeurs fournies par les vues et fonctions de gestion dynamique ne sont pas généralement conservées et résident uniquement dans la mémoire pendant l'exécution du serveur. Lorsque l'instance du serveur est redémarrée, ces valeurs sont réinitialisées.

Lorsque les vues et fonctions de gestion dynamique ont été introduites dans SQL Server 2005, il était courant que les utilisateurs souhaitent conserver les valeurs fournies par les vues et fonctions de gestion dynamique. À cette fin, de nombreux utilisateurs créent une base de données pour stocker les valeurs, puis créent un travail qui collecte et enregistre régulièrement les valeurs.

Le système Collecteur de données qui a été introduit avec SQL Server 2008 formalise ce concept en créant un entrepôt central pour contenir les données de performances, des travaux pour collecter et transférer les données dans l'entrepôt et un ensemble de rapports de haute qualité qui peuvent être utilisés pour analyser les données. Cette leçon explique comment installer et configurer le collecteur de données. La leçon 3 décrit les rapports disponibles à partir des données collectées par le collecteur de données.

Objectifs

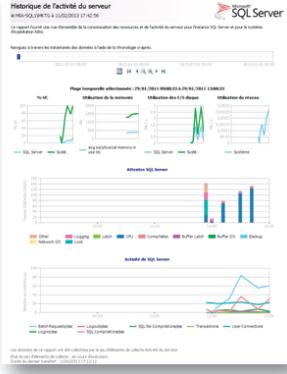
À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- expliquer le rôle du collecteur de données ;
- concevoir une topologie à utiliser avec le collecteur de données ;
- configurer le collecteur de données ;
- configurer la sécurité pour le collecteur de données ;
- surveiller le collecteur de données.

Vue d'ensemble du collecteur de données

Le collecteur de données est un composant SQL Server qui collecte des données sur la planification des capacités et les performances au fil du temps

- Collecte de données avec faible charge
- Persistance des données de diagnostics
- Rétention des données
- Création de rapports détaillés
- Facilité d'extensibilité
- Référentiel central pour plusieurs instances SQL Server



Points clés

Précédemment dans ce module, vous avez appris comment des informations utiles peuvent être obtenues par l'utilisation des vues et fonctions de gestion dynamique. Un des inconvénients de l'utilisation des objets de gestion dynamique est que la plupart des vues et fonctions ne retournent que des données en temps réel et que les vues et fonctions qui retournent des données d'historique sont des agrégations des occurrences dans le temps. Toutefois, pour effectuer un réglage et une analyse efficaces des performances, une vue d'ensemble dans le temps est nécessaire, ainsi que la possibilité d'aller à des niveaux plus détaillés pour examiner les problèmes. Le collecteur de données SQL Server vous permet d'y parvenir.

Jeux d'éléments de collecte de données

Lorsque vous configurez le collecteur de données SQL Server, plusieurs jeux d'éléments de collecte de données système sont créés. Ces jeux définissent les données devant être collectées, la fréquence de téléchargement des données vers un référentiel central et la durée de conservation des données dans ce référentiel.

Le collecteur de données peut collecter des informations à partir de plusieurs emplacements :

- Il peut interroger les vues et fonctions de gestion dynamique pour récupérer des informations détaillées sur le fonctionnement du système.
- Il peut récupérer les compteurs de performance qui fournissent des mesures sur les performances de SQL Server et de l'intégralité du serveur.
- Il peut également capturer les événements Trace SQL qui se sont produits.

Outre les jeux d'éléments de collecte de données système, le collecteur de données SQL Server peut être étendu en créant des jeux d'éléments de collecte de données définis par l'utilisateur. La possibilité d'ajouter des jeux d'éléments de collecte de données définis par l'utilisateur permet aux utilisateurs de spécifier les données qu'ils souhaitent collecter, et d'utiliser l'infrastructure fournie par le collecteur de données SQL Server pour collecter et centraliser les données.

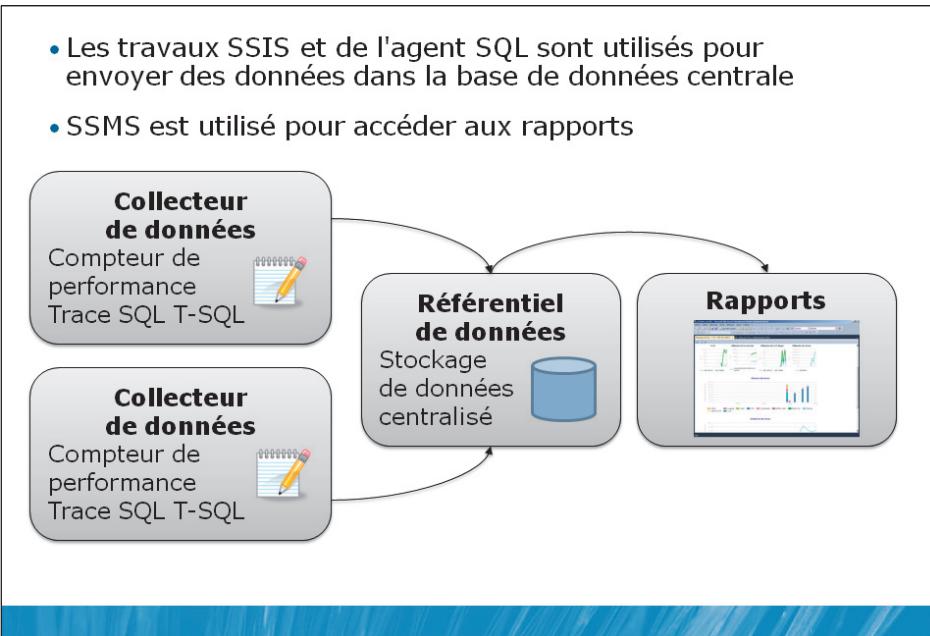
Entrepôt de données de gestion et rapports

Les données collectées par le collecteur de données SQL Server sont stockées dans un référentiel central appelé un entrepôt de données de gestion. L'entrepôt de données de gestion est une base de données SQL Server créée par un Assistant qui est également utilisé pour configurer la collecte de données.

Trois rapports standard et un ensemble complet de sous-rapports ont été fournis avec le collecteur de données, mais il est possible d'écrire vos propres rapports en fonction des données collectées par les jeux d'éléments de collecte de données système ou des données collectées par les jeux d'éléments de collecte de données définis par l'utilisateur.

Question : le collecteur de données peut-il être utilisé pour la surveillance en temps réel ?

Conception d'une topologie pour le collecteur de données



Points clés

Le système de collecte de données de SQL Server est constitué de trois composants :

- Le collecteur de données est composé d'un ensemble de travaux qui s'exécutent sur le serveur local et collectent des données à partir des objets de gestion dynamique, des compteurs de performance et des événements Trace SQL et les téléchargent vers un référentiel central.
- Le référentiel central peut consolider des données de plusieurs instances de serveur.
- Les rapports riches sont disponibles pour analyser les données de l'entrepôt de données de gestion. Les rapports sont accessibles à l'aide de SSMS.

Une grande entreprise doit envisager d'utiliser un système autonome pour l'entrepôt de données de gestion. La création d'un entrepôt de données de gestion centralisé répond à deux objectifs :

- Vous pouvez accéder aux rapports qui combinent des informations pour toutes les instances de serveur de votre entreprise.
- Il n'est pas nécessaire de stocker les données collectées et de créer des rapports à partir des serveurs de production.

Le collecteur de données fournit deux méthodes pour télécharger les données de performances capturées dans l'entrepôt central. Les informations à faible volume sont envoyées immédiatement à l'entrepôt. Les informations à volume élevée sont d'abord mises en cache localement puis téléchargées vers l'entrepôt à l'aide de SSIS.

Question : est-il possible de configurer toutes les fonctionnalités du collecteur de données sur une instance unique ?

Configuration du collecteur de données

- Installation basée sur l'assistant
- Processus en deux étapes
 1. Création de l'entrepôt de données de gestion
 2. Configuration du collecteur de données sur toutes les instances à surveiller
- Prévoyez suffisamment d'espace dans l'entrepôt de données de gestion
- La configuration crée des jeux d'éléments de collecte de données système
 - Données à collecter
 - Fréquence de collecte
 - Période de rétention
- Vous pouvez ajouter des jeux d'éléments de collecte de données personnalisés



Points clés

L'installation du système de collecte de données dans SQL Server s'effectue en deux étapes :

- configuration d'un entrepôt de données de gestion central pour stocker les données collectées ;
- configuration de chaque instance de serveur pour collecter et télécharger les données requises.

SQL Server fournit un Assistant qui est utilisé pour les deux tâches. La première étape lors de l'exécution de l'Assistant consiste à créer un entrepôt de données de gestion. L'entrepôt de données de gestion peut stocker les données collectées à partir de plusieurs instances de SQL Server.

Planification de l'espace requis

Un espace disque suffisant doit être disponible pour répondre aux besoins de l'entrepôt de données de gestion. Dans une configuration classique, vous devez prévoir au moins 300 Mo par jour pour chaque instance gérée.

Impact sur les systèmes locaux

Les seuls processus qui sont exécutés sur les instances locales sont les travaux utilisés pour collecter et télécharger les données vers l'entrepôt de données de gestion. Certaines données sont collectées très régulièrement, mises en cache localement et téléchargées ultérieurement à l'aide de SSIS. D'autres données sont rarement capturées et téléchargées immédiatement.

Les jeux d'éléments de collecte de données système sont créés automatiquement pendant l'installation. Ils peuvent être activés et désactivés si nécessaire, et la fréquence de collecte et les périodes de rétention des données collectées peuvent être personnalisées pour les jeux d'éléments de collecte de données système et les jeux d'éléments de collecte de données définis par l'utilisateur.

Il est également important de tenir compte des exigences de sécurité du collecteur de données. Elles sont décrites dans la rubrique suivante.

Question : qu'est-ce qui peut être intéressant d'inclure sous forme de jeu d'éléments de collecte personnalisé ?

Sécurité du collecteur de données

Rôles de l'entrepôt de données de gestion

Rôle	Description
mdw_admin	Accès complet à l'entrepôt de données de gestion
mdw_writer	Accès en lecture et en écriture. Nécessaire pour les collecteurs de données
mdw_reader	Accès en lecture. Requis par les utilisateurs qui accèdent aux rapports

Rôles de configuration du collecteur de données

Rôle	Description
dc_admin	Accès administrateur complet à la configuration
dc_operator	Accès en lecture et mise à jour à la configuration
dc_proxy	Accès en lecture à la configuration

Points clés

Le système Collecteur de données SQL Server comporte deux aspects de sécurité qui doivent être pris en compte :

- Qui peut accéder à l'entrepôt de données de gestion ?
- Qui peut configurer la collecte de données ?

Les rôles de base de données existent à ces deux fins, comme indiqué dans les tableaux de la diapositive.

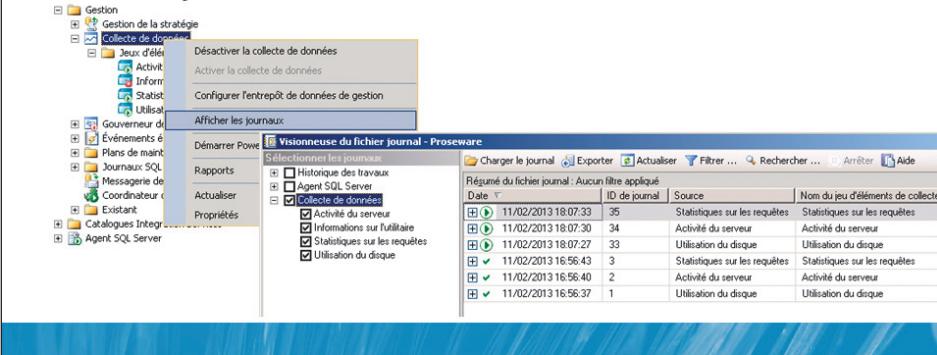
 **Remarque** Un utilisateur doit être membre du rôle mdw_reader pour pouvoir accéder aux rapports et aux données collectées.

Outre la nécessité d'être membre du rôle mdw_writer pour télécharger les données, les travaux qui collectent les données ont besoin des autorisations requises pour accéder aux données qu'ils collectent.

Question : si le compte de service SQL Server Agent n'a pas de priviléges suffisants pour accéder à l'entrepôt de données de gestion pour télécharger les données, quelles modifications de configuration pouvez-vous effectuer pour permettre ceci ?

Surveillance du collecteur de données

- Configuration et journal enregistrés dans msdb
 - Implémenté via les procédures stockées et les fonctionnalités de journalisation SSIS
- Trois niveaux de journalisation sont disponibles
- La rétention est basée sur la rétention du jeu d'éléments de collecte
- Les journaux peuvent être affichés avec T-SQL ou la Visionneuse du fichier journal



Points clés

Comme avec les autres travaux de SQL Server Agent, l'historique des travaux utilisés par le collecteur de données SQL Server pour capturer et télécharger les données de performances est conservé dans les tables d'historique des travaux de SQL Server Agent et peut être affiché à l'aide de la visionneuse standard de l'historique des travaux de SQL Server Agent.

Outre l'historique des travaux standard, le collecteur de données enregistre également les informations de configuration et les autres informations de consignation dans les tables de la base de données msdb. Le collecteur de données appelle les procédures stockées pour ajouter les informations de consignation et utilise également les fonctionnalités de journalisation de SSIS pour les packages SSIS qu'il exécute.

Les données enregistrées dans la base de données msdb sont conservées avec les mêmes paramètres de période de rétention que les jeux d'éléments de collecte de données auxquels elles sont associées. Les informations conservées peuvent être affichées à l'aide de la visionneuse du fichier journal ou en interrogeant les vues suivantes :

- fn_syscollector_get_execution_details()
- fn_syscollector_get_execution_stats()
- syscollector_execution_log
- syscollector_execution_log_full
- syscollector_execution_stats

Trois niveaux de journalisation sont disponibles et peuvent être définis en appelant la procédure stockée système `sp_syscollector_update_collection_set`. Le niveau de journalisation le plus faible enregistre les démarriages et arrêts de l'activité du collecteur. Le niveau de journalisation suivant ajoute les statistiques d'exécution et les rapports de progression. Le niveau de journalisation le plus élevé ajoute les détails de la journalisation des packages SSIS.

Question : pourquoi il est préférable d'utiliser les journaux du collecteur de données que l'historique du collecteur de données pour la résolution des problèmes du collecteur de données ?

Démonstration 2A : Configuration du collecteur de données

Dans cette démonstration, vous allez apprendre à configurer le collecteur de données

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**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_18_PRJ\22462A_18_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 **21 – Demonstration 2A.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

Leçon 3

Analyse des données de performances collectées

- Vue d'ensemble des rapports du collecteur de données
- Rapport d'utilisation du disque
- Démonstration 3A : Affichage du rapport d'utilisation du disque
- Rapport d'activité du serveur
- Démonstration 3B : Affichage du rapport d'activité du serveur
- Rapport des statistiques sur les requêtes
- Démonstration 3C : Affichage du rapport des statistiques sur les requêtes

Une fois que les données de performances ont été collectées à partir de plusieurs instances de serveur et consolidées dans un entrepôt de données de gestion, elles peuvent être analysées. Vous pouvez écrire vos propres rapports personnalisés à l'aide de SQL Server Reporting Services, via les rapports personnalisés de SSMS ou via les requêtes T-SQL. La plupart des utilisateurs trouvent que les rapports standard fournis avec SQL Server sont suffisants et qu'il n'est pas nécessaire d'écrire des rapports supplémentaires. Il est important de vous familiariser avec les informations contenues dans les rapports standard, et la navigation dans les rapports.

Objectifs

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

- expliquer les rapports du collecteur de données disponibles ;
- utiliser le rapport d'utilisation du disque ;
- utiliser le rapport d'activité du serveur ;
- utiliser le rapport des statistiques sur les requêtes.

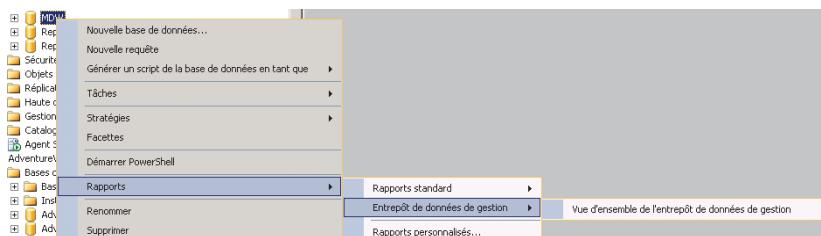
Vue d'ensemble des rapports du collecteur de données

Le collecteur de données fournit une série de rapports en référence croisée sur des données historiques. Les rapports sont visibles dans SQL Server Management Studio

- Résumé sur l'utilisation du disque
 - Tendances et détails sur l'utilisation des disques et des fichiers
- Historique des statistiques sur les requêtes
 - Requêtes les plus coûteuses classées par : UC, durée, lectures, écritures
- Historique de l'activité du serveur
 - UC, mémoire, disque et E/S réseau
 - Attentes et activité de SQL Server

Points clés

Le collecteur de données SQL Server fournit une série de rapports standard qui sont basés sur les données collectées à partir des jeux d'éléments de collecte de données système et consolidées dans un entrepôt de données de gestion centralisé. Les rapports sont accessibles à partir de SQL Server Management Studio sous la forme d'une option de menu contextuel dans la base de données de l'entrepôt de données de gestion, comme indiqué dans la capture d'écran ci-dessous :



La page de présentation des rapports de l'entrepôt de données de gestion fournit des liens vers les zones principales du rapport, comme indiqué dans la capture d'écran suivante :

À partir de cette page de présentation, trois zones principales sont associées :

- Résumé sur l'utilisation du disque.
- Historique des statistiques sur les requêtes.
- Historique de l'activité du serveur.

Remarque Chaque rapport peut être imprimé et exporté dans des fichiers PDF ou Microsoft Excel® pour une analyse plus approfondie.

Question : quelle est la période couverte par les rapports du collecteur de données ?

Rapport d'utilisation du disque

- Basé sur le jeu d'éléments de collecte de données système sur l'utilisation des disques
 - Type de collecteur T-SQL
 - Non mis en cache, exécuté toutes les 6 heures par défaut

Nom de la base de données	Base de données			Journal				
	Taille de début (Mo)	Tendance	Taille actuelle (Mo)	Croissance moyenne (Mo/jour)	Taille de début (Mo)	Tendance	Taille actuelle (Mo)	Croissance moyenne (Mo/jour)
AdminDB	1.25		2.25	1	0.49		0.49	0
MarketDev	229.25		229.25	0	1,102.13		1,102.13	0
master	4.00		4.00	0	1.00		1.00	0
MDW	100.00		100.00	0	10.00		10.00	0
model	1.25		1.25	0	0.50		0.50	0
msdb	14.75		16.25	1.5	2.25		2.25	0
ReportServer\$MKTG	4.25		4.25	0	6.75		6.75	0
ReportServer\$MKTGTempDB	2.25		2.25	0	0.75		0.75	0
tempdb	8.00		8.00	0	1.00		1.00	0

Points clés

Les rapports d'utilisation du disque sont basés sur le jeu d'éléments de collecte de données système sur l'utilisation du disque. Par défaut, le jeu d'éléments de collecte recueille les données d'utilisation du disque toutes les six heures et les conserve pendant deux années.

Notez que par défaut, le jeu d'éléments de collecte système sur l'utilisation du disque conserve les données sur une période plus longue que les autres jeux d'éléments de collecte de données système. Toutefois, comme la quantité de données collectées par ce jeu d'éléments de collecte est assez faible, cela ne devrait pas poser problème et la possibilité de suivre l'utilisation individuelle de l'espace fichier dans le temps justifie l'utilisation d'une période de rétention plus longue.

Ce rapport contient également plusieurs liens hypertexte qui renvoient à un ensemble de sous-rapports associés pour analyser les données d'utilisation.

Question : comment doit se présenter un bon rapport personnalisé pouvant être créé sur des données d'utilisation du disque ?

Démonstration 3A : Affichage du rapport d'utilisation du disque

Dans cette démonstration, vous allez apprendre à afficher les informations disponibles dans un rapport d'utilisation du disque

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 cliquez 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_18_PRJ\22462A_18_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 le fichier de script **21 – Démonstration 2A.sql** et suivez les instructions qu'il contient.
2. Ouvrez le fichier de script **31 – Démonstration 3A.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

Rapport d'activité du serveur

- Basé sur le jeu d'éléments de collecte de données système de l'activité du serveur
 - Types de collecteurs de capture instantanée d'objet de gestion dynamique et de compteur de performance
 - S'exécute toutes les 60 secondes et est téléchargé toutes les 15 minutes par défaut
 - Conservé pendant 14 jours par défaut
- Fournit de nombreux sous-rapports avec des informations détaillées sur les processus SQL Server et Windows
- Référence croisée avec les rapports de statistiques sur les requêtes



Points clés

Le rapport d'activité du serveur est basé sur le jeu d'éléments de collecte de données système sur l'activité du serveur. Le jeu d'éléments de collecte recueille un grand nombre de statistiques SQL Server telles que les attentes, les verrous, les statistiques de mémoire qui sont accessibles à l'aide des vues de gestion dynamique. En outre, le jeu d'éléments de collecte recueille les compteurs de performance Windows et SQL Server pour récupérer des informations telles que l'utilisation du processeur et de la mémoire à partir du système et des processus qui s'exécutent sur le système. La collecte s'exécute toutes les soixante secondes et est téléchargée par défaut toutes les quinze minutes. L'historique est conservé par défaut pendant quatorze jours.

Ce rapport contient un grand nombre de sous-rapports associés qui fournissent des informations plus approfondies que celles fournies dans le résumé initial. Le rapport initial est un tableau de bord qui fournit une vue d'ensemble. Si vous examinez ce rapport, vous remarquerez que presque tous les éléments affichés sont des points d'exploration possibles. Par exemple, vous pouvez cliquer sur la ligne de tendance d'un graphique pour connaître les valeurs qui composent la tendance.

Question : pourquoi est-ce important que la collecte de données suive également l'utilisation de la mémoire des autres processus exécutés sur le système ?

Démonstration 3B : Affichage du rapport d'activité du serveur

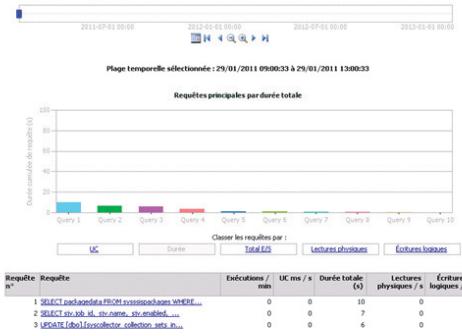
Dans cette démonstration, vous allez examiner les informations disponibles dans un rapport d'activité de serveur

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**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_18_PRJ\22462A_18_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 **32 – Demonstration 3B.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

Rapport des statistiques sur les requêtes

- Basé sur le jeu d'éléments de collecte de données système des statistiques sur les requêtes
 - Activité des requêtes, type de collecteur
 - S'exécute toutes les 10 secondes et téléchargé toutes les 15 minutes par défaut
 - Conservé pendant 14 jours par défaut
- Les requêtes les plus coûteuses sont regroupées, y compris le texte de la requête et le plan de la requête



Points clés

Le rapport des statistiques sur les requêtes est basé sur le jeu de collecte de données système sur les statistiques des requêtes qui extrait les détails des requêtes coûteuses. Le jeu d'éléments de collecte est le système de collecte par défaut le plus intensif et, par défaut, s'exécute toutes les dix secondes. Pour éviter la surcharge liée au téléchargement constant, les données collectées par ce jeu d'éléments de collecte sont mises en cache dans le système de fichiers local et téléchargées à l'aide de SSIS toutes les quinze minutes. Les données collectées sont conservées par défaut pendant quatorze jours, mais cette valeur peut être étendue.

SQL Server détermine les requêtes coûteuses en fonction de ce qui suit :

- temps écoulé ;
- temps de travail ;
- lectures logiques ;
- écritures logiques ;
- lectures physiques ;
- nombre d'exécutions.

Ce rapport contient également un grand nombre de sous-rapports associés qui peuvent être utilisés pour accéder à des niveaux de détail plus élevés. Par exemple, il est possible de récupérer des plans de requête à partir des requêtes coûteuses qui étaient en mémoire au moment où la capture a été effectuée.

Question : que signifie le terme « requêtes coûteuses » ?

Démonstration 3C : Affichage du rapport des statistiques sur les requêtes

Dans cette démonstration, vous verrez des informations disponibles dans un rapport des statistiques sur les requêtes

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**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_18_PRJ\22462A_18_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 **33 – Demonstration 3C.sql**.
3. Suivez les instructions contenues dans les commentaires du fichier de script.

Atelier pratique 18 : Surveillance de SQL Server 2012

- Exercice 1 : Étudier les vues de gestion dynamique
- Exercice 2 : Configurer l'entrepôt de données de gestion
- Exercice 3 : Configurer les instances de la collecte de données
- Exercice difficile 4 : Utiliser les rapports du collecteur de données (uniquement 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_18_PRJ\22462A_18_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

Les versions antérieures de SQL Server que vous avez utilisées n'incluaient pas les fonctions et vues de gestion dynamique. Vous avez récemment découvert ces objets et souhaitez savoir comment ils peuvent être utilisés pour collecter des informations sur les performances.

Au lieu de collecter des informations séparément pour chaque instance SQL Server, vous avez décidé de collecter des informations de performances sur un serveur central. Cela permet de traiter les problèmes qui n'ont pas été signalés au support technique au moment où ils se sont produits.

Exercice 1 : Étudier les vues de gestion dynamique

Scénario

On vous a fourni une série d'exemples de requêtes qui peuvent être utilisées pour fournir des informations sur les vues de gestion dynamique et une analyse de l'activité de SQL Server. Dans cet exercice, vous allez exécuter ces requêtes pour mieux cerner les utilisations possibles des vues de gestion dynamique.

La tâche principale de cet exercice est la suivante :

1. Étudier l'utilisation des vues et des fonctions de gestion dynamique.

► **Tâche 1 : Étudier l'utilisation des vues et des fonctions de gestion dynamique**

- Ouvrez le fichier de script 51 - Lab Exercise 1.sql et suivez les instructions qu'il contient.

Résultats : À la fin de cet exercice, vous devez avoir étudié l'utilisation des vues et des fonctions de gestion dynamique.

Exercice 2 : Configurer l'entrepôt de données de gestion

Scénario

Vous devez pouvoir collecter de manière centralisée des informations sur les performances afin de pouvoir analyser toutes les données de performances Adventure Works à un stade ultérieur. Dans cet exercice, vous allez configurer un serveur pour contenir l'entrepôt de données de gestion pour la collecte des données de performances.

La tâche principale de cet exercice est la suivante :

1. Créer un entrepôt de données de gestion pour la collection centrale des données de performances.

► Tâche 1 : Créer un entrepôt de données de gestion pour la collection centrale des données de performances

- Créer un entrepôt de données de gestion pour la collection centrale des données de performances.
Créez un entrepôt de données de gestion portant le nom de base de données MDW sur l'instance Proseware.

Résultats : À la fin de cet exercice, vous devez avoir créé un entrepôt de données de gestion.

Exercice 3 : Configurer les instances de la collecte de données

Scénario

Vous devez configurer chaque instance Adventure Works SQL Server qui enverra des informations de performances sur un serveur central. Dans cet exercice, vous allez configurer chaque instance de SQL Server pour envoyer des données de performances dans l'entrepôt de données récemment configurée.

Les tâches principales de cet exercice sont les suivantes :

1. Configurer la collecte de données sur chaque instance.
2. Configurer la sécurité de la collecte de données pour l'instance AdventureWorks.

► Tâche 1 : Configurer la collecte de données sur chaque instance

- Configurez la collecte de données sur l'instance Proseware.
- Définissez la cible des données de performances sur l'entrepôt de données de gestion créé dans l'exercice 2.
- Configurez la collecte de données sur l'instance AdventureWorks.

Résultats : À la fin de cet exercice, vous devez avoir configuré les deux instances pour la collecte de données.

Exercice difficile 4 : Utiliser les rapports du collecteur de données (uniquement si le temps le permet)

Scénario

Vous devez créer un rapport sur les performances qui résumé les informations de performances de toutes les instances d'Adventure Works SQL Server. Dans cet exercice, vous allez examiner les rapports fournis par le système de collecte de données.

-  **Remarque** Les données de performances mettent du temps à être collectées. Pour faciliter cet exercice, la sauvegarde d'un autre entrepôt de données de gestion vous a été fournie pour tester les rapports.

Les tâches principales de cet exercice sont les suivantes :

1. Désactiver les collecteurs de données sur les deux instances.
2. Restaurer une sauvegarde de la base de données MDW.
3. Examiner les rapports disponibles.

► **Tâche 1 : Désactiver les collecteurs de données sur les deux instances**

- Désactivez la collecte de données sur l'instance Proseware.
- Désactivez la collecte de données sur l'instance AdventureWorks.

► **Tâche 2 : Restaurer une sauvegarde de la base de données MDW**

- Restaurez la base de données MDW à partir du fichier de sauvegarde D:\22462A_Labs\22462A_18_PRJ\22462A_18_PRJ\MDW.bak.

► **Tâche 3 : Examiner les rapports disponibles**

- Créez un rapport d'activité du serveur.
- Créez un rapport d'utilisation de l'espace disque.
- Créez un rapport des statistiques sur les requêtes.

Résultats : À la fin de cet exercice, vous devez avoir utilisé les rapports de l'entrepôt de données de gestion.

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 préférable d'avoir un entrepôt de données de gestion centralisée pour la collecte de données à la place d'installations locales ?
2. Que sont les objets de gestion dynamique ?

Meilleures pratiques

1. Utiliser les objets de gestion dynamique pour effectuer une surveillance en temps réel et une résolution des problèmes.
2. Utiliser le Moniteur d'activité pour accéder facilement à la majeure partie des informations importantes.
3. Utiliser l'Analyseur de performances pour rassembler des mesures pour Windows et SQL Server.
4. Créer un entrepôt de données de gestion centralisée pour stocker les informations de performances d'historique.
5. Configurer la collecte de données pour regrouper les données sur les performances de vos instances SQL Server.

Module 19

Gestion de plusieurs serveurs

Sommaire :

Leçon 1 : Utilisation de plusieurs serveurs	19-3
Leçon 2 : Virtualisation de SQL Server	19-9
Leçon 3 : Déploiement et mise à niveau des applications de la couche Données	19-16
Atelier pratique 19 : Gestion de plusieurs serveurs	19-23

Vue d'ensemble du module

- Utilisation de plusieurs serveurs
- Virtualisation de SQL Server
- Déploiement et mise à niveau des applications de la couche Données

Les administrateurs de base de données qui utilisent Microsoft® SQL Server® doivent de plus en plus gérer un grand nombre de serveurs. La nécessité de gérer plusieurs serveurs crée des difficultés supplémentaires. Microsoft a fait des investissements considérables dans l'outillage SQL Server pour faciliter la gestion de plusieurs serveurs.

Bien que la virtualisation de systèmes d'exploitation et d'applications ne soit pas un phénomène nouveau, les systèmes SQL Server virtualisés sont moins courants que les autres types de systèmes. Toutefois, les systèmes SQL Server virtualisés connaissent désormais un déploiement rapide. Il est important que vous compreniez les concepts de base impliqués dans la virtualisation de systèmes SQL Server, les outils couramment utilisés pour implémenter la virtualisation et le mode de gestion de ces systèmes.

Dans SQL Server 2012, Microsoft a introduit un nouveau type de projet de développement de base de données appelé applications de la couche Données. Même si les administrateurs de base de données n'ont pas forcément besoin de comprendre comment créer des applications de la couche Données, ils doivent savoir comment déployer et mettre à niveau des applications de la couche Données.

Objectifs

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

- utiliser plusieurs serveurs ;
- décrire les options de virtualisation de SQL Server ;
- déployer et mettre à niveau des applications de la couche Données.

Leçon 1

Utilisation de plusieurs serveurs

- Vue d'ensemble des serveurs de gestion centralisée
- Exécution de requêtes multiserveurs
- Démonstration 1A : Exécution de requêtes multiserveurs

L'un des premiers outils avec lequel vous devez vous familiariser lorsque vous devez gérer plusieurs systèmes SQL Server est le serveur de gestion centralisée (CMS, Central Management Server). Le serveur de gestion centralisée a été introduit pour la première fois dans SQL Server 2008. Il vous permet de définir des groupes de serveurs, d'exécuter des requêtes et d'appliquer des stratégies sur les groupes de serveurs définis.

Il est important que vous sachiez comment configurer un serveur de gestion centralisée, définir des groupes de serveurs et exécuter des requêtes sur les groupes de serveurs. Il est également important que vous connaissiez les restrictions relatives aux requêtes multiserveurs.

Objectifs

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

- configurer les serveurs de gestion centralisée ;
- exécuter des requêtes multiserveurs.

Vue d'ensemble des serveurs de gestion centralisée

Les serveurs de gestion centralisée sont utilisés pour gérer des coordonnées des serveurs dans une organisation et organiser les serveurs dans les groupes

- Configuration requise
 - Définissez le serveur de CMS (généralement un seul serveur)
 - Créer des groupes de serveurs
 - Enregistrer un serveur IAS
- Actions de groupe autorisées
 - Exécutez l'instruction T-SQL
 - Évaluez et importer les stratégies de la gestion basée sur des stratégies
 - Parcourez l'Explorateur d'objets
- L'utilisation de l'authentification Windows et l'appartenance au rôle ServerGroupReaderRole

Points clés

Pour de nombreuses versions du produit, SQL Server Management Studio a permis aux utilisateurs de définir des groupes de serveurs dans la fenêtre Outil de serveurs inscrits. Cette fonctionnalité était généralement utilisée pour regrouper les serveurs dans différents environnements. Par exemple, vous pouviez créer un groupe de serveurs de développement, un groupe de serveurs de test, un groupe de serveurs intermédiaires et un groupe de serveurs de production.

Aucune action n'était autorisée sur les groupes entiers de serveurs et la principale difficulté liée à cette fonctionnalité était que la définition des groupes de serveurs était enregistrée dans un fichier XML sur l'ordinateur client local qui exécutait SSMS.

Si un utilisateur changeait d'ordinateur, les groupes de serveurs n'existaient plus et devaient être recréés. En cas de changement de la liste des serveurs d'un groupe, les groupes devaient être modifiés sur chaque ordinateur qui exécutait SSMS, si l'accès aux groupes était nécessaire.

Serveur de gestion centralisée

Un serveur de gestion centralisée est un serveur configuré pour gérer une liste centralisée de groupes de serveurs qui doit être définie une seule fois et est accessible par tous les utilisateurs autorisés. La liste centralisée est stockée dans les tables de la base de données msdb du serveur de gestion centralisée.

Les utilisateurs qui ont besoin d'accéder à la liste de serveurs doivent être membres du rôle ServerGroupReaderRole de la base de données msdb. Toutes les éditions et versions de SQL Server peuvent être enregistrées grâce au serveur de gestion centralisée, mais tous les enregistrements nécessitent l'utilisation de l'authentification Windows. Le serveur qui fournit le serveur de gestion centralisée doit être SQL Server 2008 ou une version ultérieure.

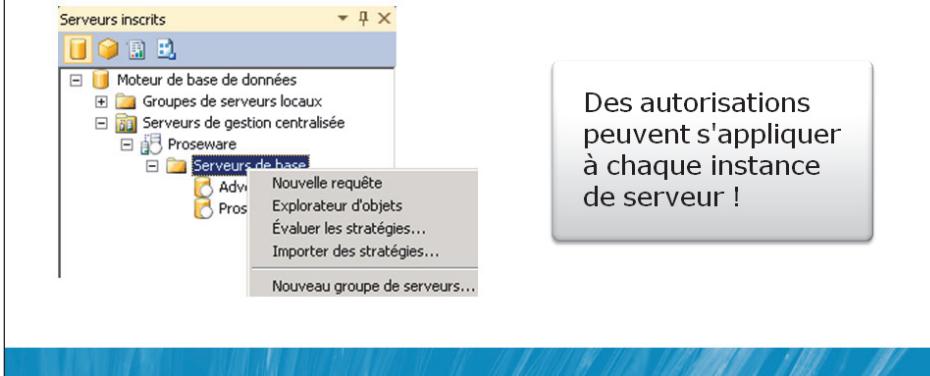
Bien qu'il soit utile de pouvoir gérer des listes centralisées, SQL Server 2008 et les versions ultérieures permettent d'exécuter des actions sur des groupes entiers d'ordinateurs. En cliquant avec le bouton droit sur un groupe dans la fenêtre Outil de serveurs inscrits, les utilisateurs peuvent exécuter des instructions T-SQL sur tous les serveurs du groupe en même temps. Ils peuvent également utiliser l'Explorateur d'objets pour parcourir plusieurs serveurs simultanément.

 **Remarque** Les utilisateurs peuvent également importer et évaluer les stratégies qui font partie de la gestion basée sur des stratégies (PBM, Policy-Based Management) de SQL Server, mais la gestion basée sur les stratégies est une rubrique avancée qui n'est pas traitée dans ce cours.

Question : quel est le principal avantage d'utiliser un serveur de gestion centralisée plutôt que des serveurs inscrits standards dans SSMS ?

Exécution de requêtes multiserveurs

- Les requêtes T-SQL peuvent être effectuées sur plusieurs serveurs simultanément
- Les résultats peuvent être fusionnés
- Les colonnes supplémentaires sont fournies pour le serveur source et les informations de connexion



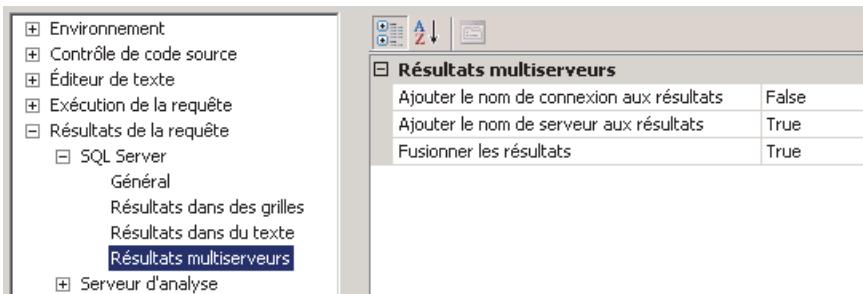
Points clés

Lorsque vous ouvrez une fenêtre de requête multiserveur, SSMS modifie la couleur de la barre d'état pour vous indiquer que la requête n'est pas une requête standard. En outre, au lieu d'afficher un nom de serveur en bas de la fenêtre, SSMS affiche le nom du groupe de serveurs. La capture d'écran suivante affiche le résultat de l'exécution de la commande suivante sur un groupe contenant les serveurs AdventureWorks et Proseware :

```
SELECT SERVERPROPERTY('IsFullTextInstalled') AS IsFullTextInstalled;
```

	Nom de serveur	IsFullTextInstalled
1	AdventureWorks	1
2	Proseware	1

Notez la colonne supplémentaire appelée Nom du serveur qui est incluse dans les résultats, mais ne l'était pas dans l'instruction SELECT. Pour l'exécution de requêtes multiserveurs, SSMS fournit des options qui déterminent comment les résultats des requêtes doivent être présentés. Dans le menu Outils, cliquez sur Options, sur Résultats de la requête, sur SQL Server, puis sur Résultats multiserveurs, comme indiqué dans la capture d'écran suivante :



Par défaut, SSMS fusionne les résultats de chaque serveur en un seul jeu de résultats et insère une première colonne contenant le nom du serveur. Vous pouvez également supprimer ce nom du serveur, ajouter le nom de connexion ou demander que les résultats ne soient pas fusionnés.

Question : que se passe-t-il si l'utilisateur n'est pas autorisé à exécuter une instruction sur l'un des serveurs d'une requête multiserveur ?

Démonstration 1A : Exécution de requêtes multiserveurs

Dans cette démonstration, vous allez apprendre à

- Créer un serveur de gestion centralisée
- Exécutez la procédure les requêtes à plusieurs

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**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_19_PRJ\22462A_19_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

Virtualisation de SQL Server

- Discussion : Avantages et inconvénients de la virtualisation de SQL Server
- Vue d'ensemble de la virtualisation de SQL Server
- Scénarios de virtualisation courants
- Éléments à prendre en compte pour la virtualisation de SQL Server
- Vue d'ensemble de System Center Virtual Machine Manager

La virtualisation de serveurs est une pratique courante depuis de nombreuses années, notamment en ce qui concerne les efforts de consolidation de serveurs. La virtualisation de SQL Server est l'exception. Il est beaucoup moins fréquent de voir des systèmes SQL Server virtualisés. Cela est principalement causé par les problèmes liés aux exigences d'E/S complexes qui sont associées aux systèmes SQL Server. Cette situation évolue rapidement et il est très important que les administrateurs de base de données qui utilisent SQL Server connaissent les techniques de virtualisation de SQL Server ainsi que les avantages et inconvénients associés à la virtualisation de SQL Server.

Objectifs

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

- discuter des avantages et inconvénients de la virtualisation de SQL Server ;
- décrire la virtualisation ;
- décrire les options de virtualisation disponibles ;
- expliquer les éléments à prendre en compte pour la virtualisation de SQL Server ;
- décrire System Center Virtual Machine Manager.

Discussion : Avantages et inconvénients de la virtualisation de SQL Server

- Utilisez-vous la virtualisation de SQL Server dans votre environnement ?
- Si vous utilisez la virtualisation, pour quelle raison le faites-vous ?
- Quels sont les principaux avantages fournis par la virtualisation ?
- Quels sont les principaux problèmes de la virtualisation de SQL Server ?



Points de discussion

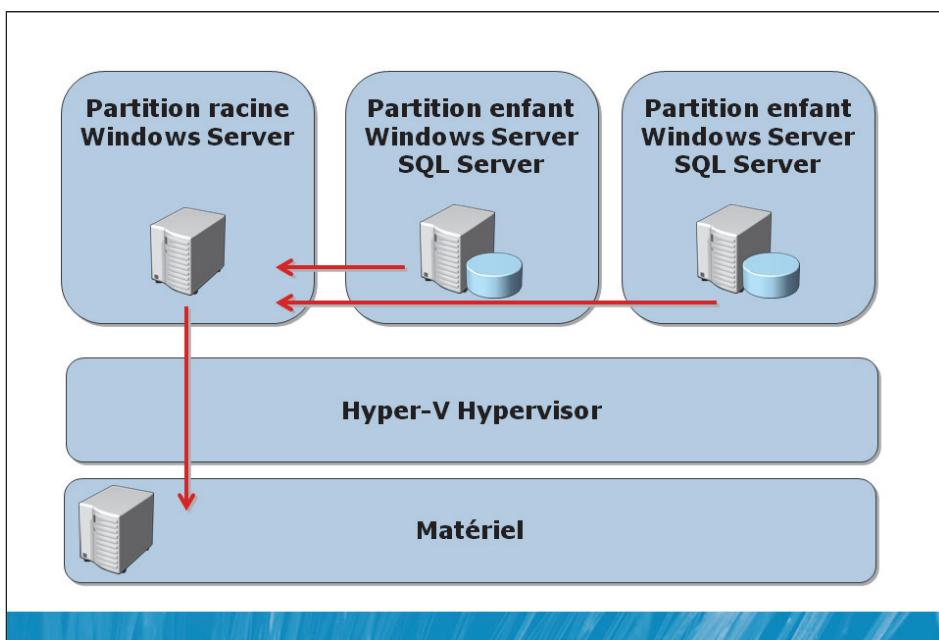
Question : utilisez-vous la virtualisation de SQL Server dans votre environnement ?

Question : si vous utilisez la virtualisation, pour quelle raison le faites-vous ?

Question : quels sont les principaux avantages fournis par la virtualisation ?

Question : quels sont les principaux problèmes de la virtualisation de SQL Server ?

Vue d'ensemble de la virtualisation de SQL Server



Points clés

La virtualisation est une technologie qui vous permet de créer et de gérer des charges de travail qui s'exécutent sur des ordinateurs virtuels distincts plutôt que des ordinateurs physiques distincts. Microsoft Hyper-V® est une fonctionnalité de Windows Server® 2008 (R2) x64 et est installé en tant que rôle du système d'exploitation.

Hyper-V prend en charge l'isolation via un concept de partitions. Une partition Hyper-V est une unité logique qui exécute un système d'exploitation. L'accès au matériel s'effectue par la partition racine qui est initialement créée dans le cadre du système d'exploitation. Aucune des partitions enfants n'a un accès direct au matériel. Les partitions enfants accèdent aux ressources via l'hyperviseur qui contrôle l'ensemble du système.

Chaque ordinateur doit avoir une partition racine sur laquelle s'exécute Windows Server. La partition racine est utilisée pour gérer le système, ainsi que les partitions enfants.

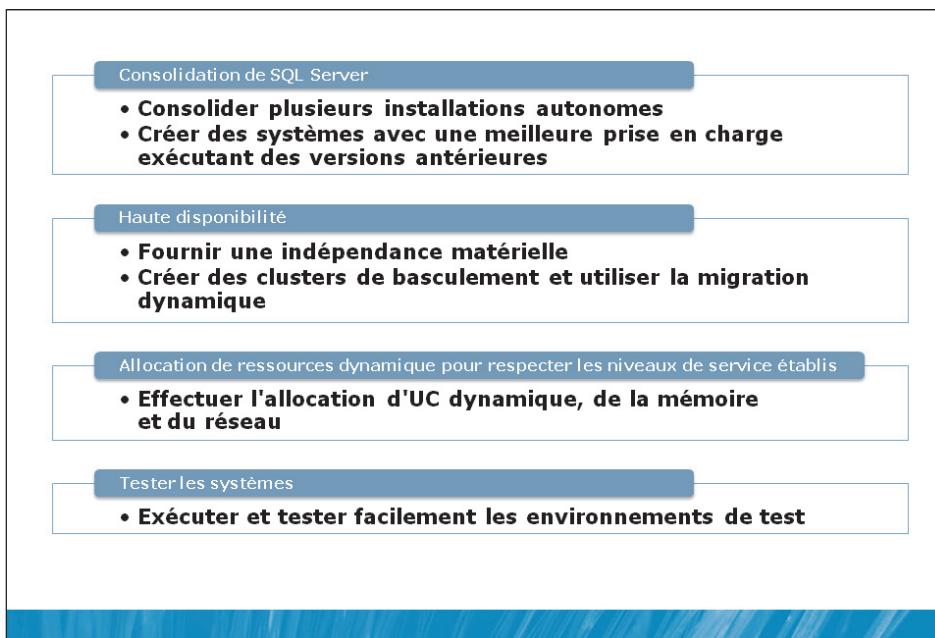
La virtualisation fournie à l'aide de la technologie Hyper-V ne fournit pas seulement une meilleure utilisation des ressources matérielles, mais aussi une gestion simplifiée et une haute disponibilité grâce à la migration dynamique, aux captures instantanées et à de nombreuses autres fonctionnalités.

La migration dynamique permet de déplacer une partition enfant d'un ordinateur vers la partition enfant d'un autre ordinateur, tout en conservant les connexions utilisateur tout au long du processus.

Les captures instantanées permettent de créer des points nommés dans le temps vers lesquels des ordinateurs virtuels entiers peuvent être restaurés à la demande.

Question : quel est l'avantage de la migration dynamique par rapport à d'autres options telles que le clustering de basculement SQL Server ?

Scénarios de virtualisation courants



Points clés

Il existe de nombreuses raisons de virtualiser une installation SQL Server :

- L'une des principales raisons de la virtualisation de SQL Server est la consolidation de plusieurs serveurs autonomes sur un seul serveur physique plus puissant. La consolidation peut offrir plusieurs avantages, notamment en termes d'espace physique et de consommation d'énergie.
- Une autre raison de la virtualisation de SQL Server est le déplacement de versions antérieures de SQL Server et d'anciens systèmes d'exploitation serveur qui doivent être conservés pour des raisons de compatibilité des applications, vers de nouvelles plateformes matérielles qui offrent une meilleure prise en charge. De nombreux anciens serveurs ne peuvent pas être réparés en cas de panne. La virtualisation des serveurs atténue considérablement ce problème. Les outils PTV (outils physiques à virtuels) sont disponibles pour faciliter cette migration.
- Les serveurs virtuels sont indépendants du matériel sous-jacent et sont beaucoup plus faciles à déplacer vers le matériel auxiliaire.

- La virtualisation basée sur Hyper-V prend en charge la création de clusters Hyper-V sur plusieurs serveurs physiques. Les clusters de basculement SQL Server peuvent être construits au-dessus des clusters Hyper-V pour fournir des options de haute disponibilité. La migration dynamique peut être utilisée pour le basculement manuel des ordinateurs virtuels.
- Il est souvent difficile d'équilibrer les besoins en ressources de différentes applications. En séparant les applications dans des serveurs virtuels distincts, l'équilibrage dynamique des ressources entre les ordinateurs virtuels est possible.
- Les environnements de test sont faciles à créer et à supprimer lorsqu'ils sont basés sur la technologie de virtualisation. Les captures instantanées permettent de restaurer l'effet des installations d'application ou des exécutions de test.

Question : pourquoi la virtualisation serait-elle très utile lorsque vous testez des applications ?

Éléments à prendre en compte pour la virtualisation de SQL Server

- Utilisez le POC pour vérifier que la collecte de virtualisation vos besoins
- Fournissez les ressources pertinentes en E/S
 - Utiliser des disques de transfert direct
 - Appliquez la recommandation sous forme pour des systèmes non virtualisés
 - Utilisez SQLIO pour le test avant de déploiement
- Fournissez les ressources pertinentes UC
 - Fournissez l'UC supplémentaire pour les systèmes puissants réseau
- Évitez d'utiliser des périphériques émulés
 - Installez les composants d'intégration pour Hyper-V qui fournissent des périphériques synthétiques

Points clés

Avant de virtualiser une charge de travail SQL Server, vous devez effectuer une mise à l'épreuve appropriée (POC, Proof Of Concept), en veillant à utiliser des tailles de base de données et des charges de travail normales.

Une erreur très courante qui est souvent commise en virtualisant des systèmes SQL Server est de négliger de fournir des ressources d'E/S suffisantes pour les serveurs virtuels. Il existe une idée fausse répandue selon laquelle un ensemble de serveurs virtualisés nécessite moins de ressources que les serveurs individuels. Les disques directs acheminent les appels d'E/S vers le système hôte pour exécution et ajoutent une charge minimale aux appels. Les disques directs doivent être utilisés dans tous les cas. Les disques émulés ajoutent une charge considérable et doivent toujours être évités.

L'utilitaire SQLIO doit être utilisé pour exécuter un test de contrainte sur le sous-système d'E/S avant de déployer SQL Server, comme cela doit être effectué avec un système SQL Server non virtualisé. Le test doit être exécuté à partir du système virtuel utilisé pour exécuter SQL Server afin que toutes les couches logicielles soient testées. Les mêmes suggestions concernant la séparation des lecteurs de données et de journaux sur des lecteurs physiques distincts qui s'appliquent aux serveurs non virtualisés le sont également pour les serveurs virtualisés.

Les serveurs virtualisés ont tendance à utiliser davantage les ressources processeur que les serveurs non virtualisés. Les systèmes qui sollicitent beaucoup le réseau nécessitent des ressources processeur encore plus importantes.

Hyper-V fournit un ensemble de composants d'intégration qui augmentent les performances du système virtuel en remplaçant les périphériques émulés sur des logiciels par leurs versions synthétiques.

Question : quelles sont les différences entre les besoins d'un système SQL Server virtualisé et d'un système SQL Server non virtualisé en termes de besoins d'E/S ?

Vue d'ensemble de System Center Virtual Machine Manager

System Center ordinateur virtuel manager (SCVMM) est un outil qui vous permet de gérer les allocations des ressources entre plusieurs ordinateurs virtuels

- Suite centrale de partie du système
- Souvent utilisé conjointement avec operations manager (SCOM) de System Center
- Autorise les ressources de serveur sur plusieurs ordinateurs virtuels
 - Il n'est plus nécessaire d'équilibrer les ressources sur chaque ordinateur

Points clés

SQL Server 2008 a introduit le gouverneur de ressources SQL Server pour aider à équilibrer les besoins en ressources de plusieurs applications sur un serveur unique. Bien que l'utilisation du gouverneur de ressources aide à équilibrer les besoins en processeur et mémoire associés au moteur de base de données et à certains types de requêtes, les utilisateurs souhaitent souvent équilibrer les ressources à un niveau de granularité supérieur et sur d'autres composants SQL Server.

Une option courante consiste à installer plusieurs ordinateurs virtuels qui exécutent chacun SQL Server et à séparer les applications sur les ordinateurs virtuels distincts. System Center Virtual Machine Manager (SCVMM) a ensuite pour rôle d'équilibrer les ressources allouées à chacun des ordinateurs virtuels.

SCVMM est un composant de la suite System Center, avec d'autres composants tels que System Center Operations Manager (SCOM) qui fournit une infrastructure de gestion globale pour une organisation. SCVMM est souvent utilisé conjointement avec SCOM.

Leçon 3

Déploiement et mise à niveau des applications de la couche Données

- Vue d'ensemble des applications de la couche Données
- Déploiement des applications de la couche Données
- Mise à niveau des applications de la couche Données
- Extraction des applications de la couche Données
- Démonstration 3A : Utilisation des applications de la couche Données

La création d'applications .NET dans Microsoft Visual Studio® est un processus bien affiné. La création de projets pour prendre en charge les aspects liés à la base de données des projets a été moins bien définie. Depuis Visual Studio 2005, il est possible de créer des projets de base de données dans Visual Studio. Les projets de base de données peuvent faire partie de solutions comprenant d'autres types de projets Visual Studio. Les projets de base de données continuent à être utilisés pour les applications métier ou les grandes applications stratégiques.

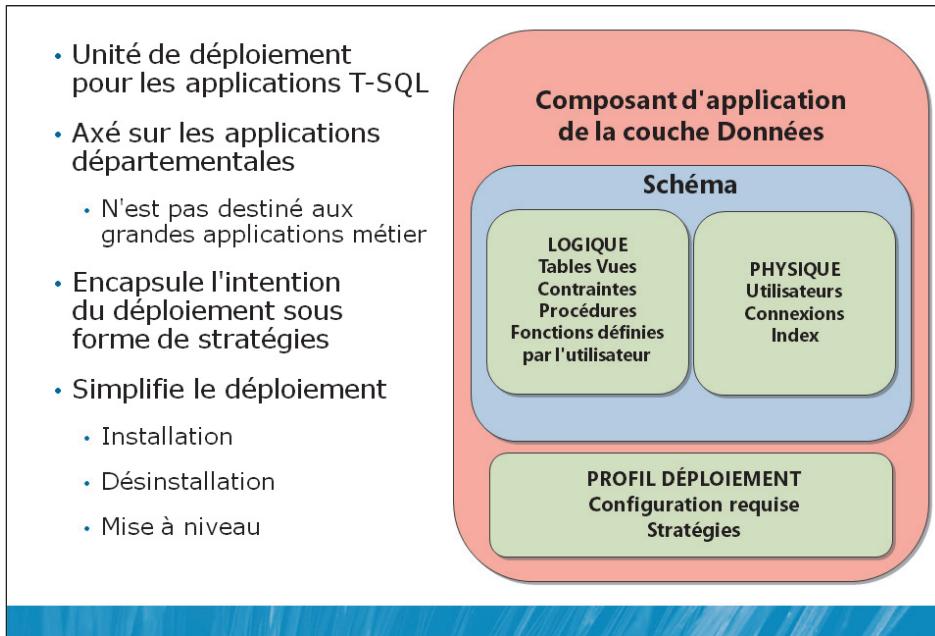
Les applications de la couche Données sont un autre type de projet de base de données qui offrent la plupart des avantages des projets de base de données, mais sont beaucoup plus faciles à créer, à déployer et à mettre à niveau. Même si les administrateurs de base de données n'ont pas forcément besoin de savoir comment créer des applications de la couche Données, il est important qu'ils sachent comment déployer et mettre à niveau les applications de la couche Données qui leur ont été fournies par les développeurs.

Objectifs

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

- décrire une application de la couche Données ;
- déployer des applications de la couche Données ;
- mettre à niveau des applications de la couche Données ;
- extraire des applications de la couche Données.

Vue d'ensemble des applications de la couche Données



Points clés

Les applications de la couche Données fournissent un mécanisme pour déployer et mettre à niveau des applications de service simples.

Problèmes liés aux approches précédentes

Examinez comment la plupart des applications de base de données sont installées. L'application proprement dite est normalement installée par un programme d'installation Windows. Le programme d'installation demande les informations de configuration et installe tous les fichiers nécessaires, enregistre les composants, ajoute les entrées de Registre requises, etc. Les mises à niveau fonctionnent généralement de façon très similaire.

Examinons maintenant comment les bases de données requises par les applications sont installées et mises à niveau. La majorité des projets de base de données actuels sont installés grâce à des scripts T-SQL. Certaines applications fournissent des programmes d'installation qui exécutent automatiquement les commandes T-SQL. L'installation basée sur des scripts présente de nombreuses inconvénients.

Les développeurs ont eu très peu d'aide dans leurs environnements de développement pour faciliter la création des scripts d'installation requis. Les scripts doivent souvent être modifiés manuellement. Ces processus manuels entraînent invariablement des erreurs. De plus, il est difficile pour un développeur d'afficher le schéma global de la base de données sur laquelle il travaille, sans créer une base de données basée sur le schéma. Dans de nombreux cas, cela implique de récupérer et d'exécuter un script « de départ » qui crée une base de données, et d'appliquer un grand nombre de scripts de mise à niveau pour définir la base de données sur le niveau requis.

L'administrateur de base de données doit souvent exécuter manuellement les scripts d'installation et effectuer d'autres tâches telles que la création de connexions et d'autres objets. Si ces tâches ne sont pas exécutées parfaitement, les scripts d'installation et de mise à niveau échouent souvent et peuvent laisser la base de données dans un état incohérent (voire inconnu).

Applications de la couche Données

Les applications de la couche Données sont conçues pour fournir une expérience d'installation et de mise à niveau des applications de base de données similaire à celle des applications Windows. Le développeur crée une application de la couche Données à l'aide de Visual Studio, avec tous les objets requis, et définit les stratégies qui limitent l'installation de l'application. Par exemple, une stratégie de déploiement peut indiquer qu'une application peut uniquement être installée sur SQL Server 10.5 et les versions ultérieures.

Lorsque le projet d'application de la couche Données est créé, la sortie est un fichier .dACPAC qui peut être remis à l'administrateur de base de données. Un seul fichier .dACPAC peut être utilisé pour l'installation et la mise à niveau d'une application et est portable dans différents environnements tels que l'environnement de développement, de test, intermédiaire et de production.

L'installation et la mise à niveau des applications de la couche Données sont automatisées. Les applications de la couche Données ne sont pas conçues pour les grandes applications métier. L'objectif est de faciliter l'installation et la mise à niveau d'un grand nombre d'applications plus simples.

Question : quel est l'inconvénient de déployer des mises à niveau grâce à des scripts T-SQL ?

Déploiement des applications de la couche Données

- Déploiement effectué à l'aide d'un Assistant
- Nécessite le nom de la base de données cible
- Vérifie la stratégie de sélection des serveurs
- Crée une base de données et des objets
 - Les paramètres de base de données par défaut sont utilisés
 - Mode de récupération de la base de données modèle
- Vérification et test des fichiers .dacpac avant son déploiement en production



Points clés

Le déploiement d'une application de la couche Données a été conçu pour être très simple et automatisé. SSMS inclut un Assistant d'installation des applications de la couche Données. L'Assistant commence par demander un nom de base de données et l'emplacement (chemins d'accès) des fichiers de base de données.

Avant d'apporter des modifications, l'Assistant vérifie toutes les stratégies incluses. Dans SQL Server 2012, la seule stratégie disponible est une stratégie de sélection du serveur qui détermine les versions de SQL Server sur lesquelles l'application peut être installée. Les versions ultérieures des applications de la couche Données peuvent inclure d'autres types de stratégie, par exemple une stratégie qui détaille les besoins en matière de basculement ou de haute disponibilité.

Avant de déployer un fichier .dacpac en tant qu'application de la couche Données, assurez-vous d'approuver la source du fichier .dacpac. Sinon, vous devez vérifier les fichiers .dacpac avant de les déployer. Il est recommandé, comme avec tout type d'application, d'installer les applications de la couche Données dans un environnement de test avant de les installer dans un environnement de production.

Vous pouvez afficher le contenu d'un fichier .dacpac en l'important dans un projet d'application de la couche Données dans Visual Studio. Si Visual Studio n'est pas disponible, une option de menu contextuel « Décompresser » est disponible pour les fichiers .dacpac sur les systèmes où SQL Server 2012 a été installé.

Question : pourquoi est-il important de vérifier un fichier de dacpac avant son déploiement ?

Mise à niveau des applications de la couche Données

- Processus de mise à niveau effectué à l'aide de l'Assistant
 - Utilisé pour modifier le schéma et les propriétés des applications déployées
 - Non basé sur le script T-SQL
 - Le nom de l'application doit correspondre au fichier actuellement déployé
- La base de données est mise à niveau
 - Différente de la version antérieure dans laquelle les données ont été migrées
 - L'utilisation de l'espace doit être prise en compte
- Le processus peut être géré via PowerShell

Points clés

Les mises à niveau des applications de la couche Données sont également automatisées. Une option de menu contextuel sur une application existante de la couche Données déployée dans SSMS fournit une option pour mettre à niveau l'application. Les mises à niveau sont également basées sur des Assistants et conçues pour être faciles à appliquer.

Dans les versions antérieures de SQL Server, lorsque vous mettiez à niveau une application de la couche Données, une base de données était créée et toutes les données de la base de données mise à niveau étaient migrées vers la nouvelle base de données. Dans SQL Server 2012, les bases de données sont mises à niveau sur place au lieu d'être migrées.

L'espace disque requis pour la base de données mise à niveau doit également être pris en compte.

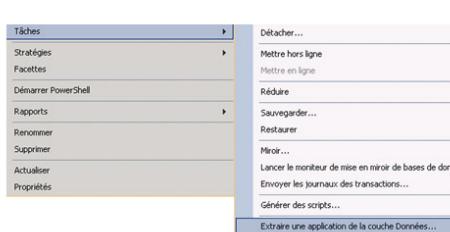
Tous les aspects des applications de la couche Données qu'il est possible de gérer dans SSMS peuvent également l'être à l'aide du script Windows PowerShell®.

Le principal avantage des mises à niveau des applications de la couche Données est que lorsqu'une erreur se produit, la mise à niveau est annulée. Contrairement aux mises à niveau basées sur des scripts où les erreurs produisent souvent des bases de données dans un état incohérent ou inconnu, les mises à niveau des applications de la couche Données fonctionnent ou ne fonctionnent pas.

Question : quel est le principal avantage des mises à niveau basées sur .dacpac par rapport à celles basées sur script ?

Extraction des applications de la couche Données

- Les applications de la couche Données peuvent être extraites de plusieurs bases de données existantes
- Toutes les bases de données ne peuvent pas être extraites
- Processus d'ingénierie inverse
 - Extraire via SSMS
 - Créer un projet Visual Studio
 - Développer une nouvelle version d'application
 - Compiler et créer un build
 - Mettre à niveau l'application avec une nouvelle version



Points clés

Certaines applications sont démarrées entièrement. Il est souvent nécessaire de développer davantage les applications existantes.

SSMS fournit deux options permettant d'utiliser les applications existantes :

- Vous pouvez enregistrer une base de données existante en tant qu'application de la couche Données.
- Vous pouvez extraire une application de la couche Données (via un fichier .dacpac) à partir d'une base de données existante.

Notez que les applications de la couche Données ne prennent pas en charge tous les objets SQL Server pour le moment. Par exemple, les collections de schémas XML et les objets basés sur CLR SQL ne sont pas pris en charge. Pour cette raison, certaines bases de données peuvent être enregistrées en tant qu'applications de la couche Données ou extraites dans des fichiers .dacpac. Lorsque SQL Server ne peut pas effectuer un enregistrement ou une extraction, les objets non pris en charge sont affichés dans l'Assistant.

Question : quel est l'objectif principal d'extraire des applications de la couche Données ?

Démonstration 3A : Utilisation des applications de la couche Données

Dans cette démonstration, vous allez apprendre à

- Déployer une application de la couche Données
- Mettre à niveau une application de la couche Données

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**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_19_PRJ\22462A_19_PRJ.ssmsln**, 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.

Atelier pratique 19 : Gestion de plusieurs serveurs

- Exercice 1 : Configurer CMS et exécuter des requêtes multiserveurs
- Exercice 2 : Déployer une application de la couche Données
- Exercice 3 : Enregistrer et extraire une application de la couche Données
- Exercice difficile 4 : Mettre à niveau une application de la couche Données (uniquement 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_19_PRJ\22462A_19_PRJ\22462A_19_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

Vous devez configurer une solution qui vous permet de gérer facilement plusieurs instances de SQL Server. Vous avez remarqué que sur chaque ordinateur auquel vous connectez votre réseau SQL Server, un ensemble différent de serveurs a été configuré. Vous décidez de configurer un serveur de gestion centralisée pour fournir une liste cohérente de groupes de serveurs. Les développeurs ont commencé à utiliser les applications de la couche Données dans le cadre partiel de leur développement. Vous devez déployer l'une de ces applications sur le nouveau serveur, enregistrer une base de données existante comme application de la couche Données et, si le temps le permet, mettre à niveau une application de la couche Données.

Exercice 1 : Configurer CMS et exécuter des requêtes multiserveurs

Scénario

Vous allez créer une liste d'instances SQL Server afin que chaque administrateur de base de données ait une liste d'instances SQL Server cohérente dans SQL Server Management Studio. Vous allez gérer cette liste de serveurs de façon centralisée. Dans cet exercice, vous allez configurer un serveur de gestion centralisée et exécuter des requêtes sur plusieurs serveurs.

Les tâches principales de cet exercice sont les suivantes :

1. Créer un serveur de gestion centralisée.
2. Créer un groupe de serveurs sur le serveur de gestion centralisée.
3. Exécuter une commande pour rechercher toutes les bases de données sur n'importe quel serveur de base en mode de restauration complète.

► **Tâche 1 : Créer un serveur de gestion centralisée**

- Créez un serveur de gestion centralisée sur l'instance Proseware.

► **Tâche 2 : Créer un groupe de serveurs sur le serveur de gestion centralisée**

- Créez un groupe de serveurs appelé Serveurs de base.
- Enregistrez l'instance Proseware dans le groupe CoreServers.
- Enregistrez l'instance AdventureWorks dans le groupe CoreServers.

► **Tâche 3 : Exécuter une commande pour rechercher toutes les bases de données sur n'importe quel serveur de base en mode de restauration complète**

- Ouvrez une fenêtre de requête multiserveur.
- Exécutez une requête multiserveur pour rechercher toutes les bases de données en mode de restauration complète.
- Conseil : recherchez la valeur FULL dans la colonne recovery_model_desc de la vue sys.databases.

Résultats : À la fin de cet exercice, vous devez avoir créé un groupe de serveurs de façon centralisée et avoir exécuté une requête multiserveur.

Exercice 2 : Déployer une application de la couche Données

Scénario

Vous allez déployer une application de la couche Données fournie par le groupe de développement d'applications. Cette application de la couche Données contient tous les objets de base de données et d'instance utilisés par une application. Dans cet exercice, vous allez utiliser SQL Server Management Studio pour déployer une application de la couche Données.

La tâche principale de cet exercice est la suivante :

1. Déployer l'application de la couche Données à l'aide de SSMS.

► Tâche 1 : Déployer l'application de la couche Données à l'aide de SSMS

- Déployez l'application de la couche Données D:\22462A_Labs\22462A_19_PRJ\22462A_19_PRJ\CityCode.dacpac en tant que nom de la base de données CityCode sur l'instance Proseware.
- Testez l'application de la couche Données en exécutant le fichier de script 61 – Lab Exercise 2.sql.

Résultats : À la fin de cet exercice, vous devez avoir déployé une application de la couche Données.

Exercice 3 : Enregistrer et extraire une application de la couche Données

Scénario

Vous allez encapsuler tous les objets de base de données et d'instance utilisés par une application existante dans un package unique que vous pourrez envoyer par courrier électronique. Dans cet exercice, vous allez enregistrer une base de données existante en tant qu'application de la couche Données. Vous allez également extraire une application de la couche Données que vous enverrez à l'équipe de développement.

Les tâches principales de cet exercice sont les suivantes :

1. Enregistrer la base de données existante en tant qu'application de la couche Données.
2. Extraire un dacpac de la base de données à envoyer à l'équipe de développement.

► **Tâche 1 : Enregistrer la base de données existante en tant qu'application de la couche Données**

- Enregistrez la base de données Recherche en tant qu'application de la couche Données.

► **Tâche 2 : Extraire un dacpac de la base de données à envoyer à l'équipe de développement**

 - Extrayez un dacpac de la base de données Recherche dans le fichier D:\MKTG\Research.dacpac.

Résultats : À la fin de cet exercice, vous devez avoir enregistré la base de données Recherche en tant qu'application de la couche Données et avoir extrait un fichier de package DAC.

Exercice difficile 4 : Mettre à niveau une application de la couche Données (uniquement si le temps le permet)

Scénario

Le groupe de développement d'applications vous a fourni une nouvelle version du schéma de base de données sous la forme d'une application de la couche Données. Vous souhaitez déployer cette nouvelle version du schéma. Dans cet exercice, vous allez mettre à niveau l'application de la couche Données existante.

Remarque : deux versions de l'application de la couche Données mise à jour ont été fournies. La première (CityCode_v2) provoque une erreur lorsque vous tentez de la déployer. Le but est que vous puissiez voir que le système n'est pas laissé dans un état inconnu. La deuxième version (CityCode_v3) devrait être déployée avec succès.

Les tâches principales de cet exercice sont les suivantes :

1. Tenter de déployer la mise à niveau v2.
2. Déployer la mise à niveau v3.

► Tâche 1 : Tenter de déployer la mise à niveau v2

- Tentez de déployer la mise à niveau v2 à partir du fichier D:\22462A_Labs\22462A_19_PRJ\22462A_19_PRJ\CityCode_v2.dacpac.

► Tâche 2 : Déployer la mise à niveau v3

- Déployez la mise à niveau v3 à partir du fichier D:\22462A_Labs\22462A_19_PRJ\22462A_19_PRJ\CityCode_v3.dacpac.

Résultats : À la fin de cet exercice, vous devez avoir mis à niveau l'application de la couche Données.

Contrôle des acquis du module et éléments à retenir

- Questions de contrôle des acquis
- Meilleures pratiques

Questions de contrôle des acquis

1. Que devez-vous prendre en compte pour exécuter des requêtes sur plusieurs serveurs ?
2. À quoi sert la migration dynamique dans le cadre de Hyper-V ?

Meilleures pratiques

1. Utiliser le serveur de gestion centralisée pour gérer une configuration centrale de serveurs et groupes enregistrés.
2. Utiliser les requêtes sur plusieurs serveurs pour exécuter des lots sur plusieurs serveurs simultanément.
3. Prendre en compte la virtualisation de la haute disponibilité, la consolidation, l'utilisation améliorée du matériel, ainsi que les environnements de test.
4. Tenir compte de l'utilisation de l'application de la couche Données pour les applications de base de données départementales.

Module 20

Résolution des problèmes courants liés à l'administration de SQL Server 2012

Sommaire :

Leçon 1 : Méthodologie de résolution des problèmes SQL Server	20-3
Leçon 2 : Résolution des problèmes liés aux services	20-7
Leçon 3 : Résolution des problèmes de connexion et de connectivité	20-13
Leçon 4 : Résolution des problèmes liés aux accès concurrentiels	20-17
Atelier pratique 20 : Résolution des problèmes courants	20-25

Vue d'ensemble du module

- Méthodologie de résolution des problèmes SQL Server
- Résolution des problèmes liés aux services
- Résolution des problèmes de connexion et de connectivité
- Résolution des problèmes liés aux accès concurrentiels

Un rôle important que les administrateurs de base de données qui utilisent Microsoft® SQL Server® doivent endosser est celui d'expert en résolution de problèmes lorsque des problèmes surviennent, notamment ceux qui empêchent les utilisateurs de travailler.

Il est important d'utiliser une bonne méthodologie pour résoudre les problèmes en général, et de se familiariser avec les problèmes les plus courants qui peuvent survenir lors de l'utilisation des systèmes SQL Server.

Objectifs

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

- expliquer la méthodologie de résolution des problèmes SQL Server ;
- résoudre les problèmes liés aux services ;
- résoudre les problèmes de connexion et de connectivité ;
- résoudre les problèmes liés aux accès concurrentiels.

Leçon 1

Méthodologie de résolution des problèmes SQL Server

- Discussion : Caractéristiques des experts en résolution de problèmes
- Application d'une méthodologie de résolution des problèmes

Avant de commencer à essayer de résoudre un problème, il est important d'utiliser une méthodologie logique de résolution des problèmes et de l'appliquer de manière cohérente. La résolution de problèmes est souvent considérée un peu comme un art autant que comme une science. Il existe de nombreuses caractéristiques communes à tous les experts en résolution de problèmes. Vous devez veiller à développer ou à émuler ces caractéristiques pour garantir une résolution efficace des problèmes.

Objectifs

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

- discuter des caractéristiques des experts en résolution de problèmes ;
- appliquer une méthodologie de résolution des problèmes.

Discussion : Caractéristiques des experts en résolution de problèmes

- Quelles caractéristiques avez-vous constatées chez les utilisateurs que vous considérez comme doués pour la résolution des problèmes ?
- Quelles caractéristiques avez-vous remarquées chez les utilisateurs peu doués pour résoudre les problèmes ?



Points de discussion

Question : quelles caractéristiques avez-vous constatées chez les utilisateurs que vous considérez comme doués pour la résolution des problèmes ?

Question : quelles caractéristiques avez-vous remarquées chez les utilisateurs peu doués pour résoudre les problèmes ?

Application d'une méthodologie de résolution des problèmes

- Rechercher
 - Définissez clairement le problème tel qu'il est perçu par l'utilisateur. Comment savez-vous s'il est résolu ?
 - Qu'est-ce qui fonctionne ? Qu'est-ce qui ne fonctionne pas ?
 - Est-ce que cela a déjà fonctionné ? Quelle est la dernière fois où cela a fonctionné ? Quoi d'autre a changé ?
- Analyser
 - Réfléchissez à toutes les causes possibles. Quelles causes sont probables ? Comment peuvent-elles être testées et éliminées ?
- Implémenter
 - Éliminez les causes possibles dans l'ordre décroissant de leur probabilité
- Valider
 - Vérifiez que le problème est réellement résolu
- Documenter pour servir à l'avenir

Points clés

Vous avez appris dans la discussion de la dernière rubrique qu'une des caractéristiques principales des experts en résolution de problèmes est qu'ils suivent une méthodologie claire de façon logique. Il existe de nombreuses méthodologies différentes qui sont souvent appliquées pour résoudre des problèmes, mais la liste suivante décrit quatre phases communes à la plupart des méthodologies.

Phase d'examen

Cette phase est importante. Trop de personnes sautent cette étape et se concentrent directement sur la recherche de solutions. Avant de pouvoir résoudre un problème, vous devez avoir une compréhension très claire du problème que vous résolvez.

Un concept très important dans cette phase est que le problème doit être défini du point de vue de l'utilisateur concerné, pas du point de vue présumé d'un informaticien. Par exemple, il ne sert à rien de dire à un utilisateur qu'un système fonctionne s'il ne peut pas l'utiliser pour une raison quelconque, quelle que soit la façon dont votre point de vue informatique peut vous indiquer le contraire.

Vous devez comprendre ce qui fonctionne et ce qui ne fonctionne pas. Une erreur courante dans cette phase est de supposer quand un utilisateur se plaint que quelque chose ne fonctionne pas que cela n'a jamais fonctionné. Vérifiez que le problème n'existe pas à un moment donné et déterminez la date exacte. Déterminez également tout ce qui a changé depuis cette date, même si cela peut sembler sans rapport à ce stade.

Enfin, vous devez savoir comment l'utilisateur décide que le problème est résolu. Une erreur de résolution courante est de trouver un problème, de supposer qu'il est la cause de l'incident, de résoudre ce problème et de supposer que l'incident d'origine est désormais résolu.

Phase d'analyse

Dans la phase d'analyse, vous devez déterminer toutes les causes possibles du problème que vous essayez de résoudre. À ce stade, il est important de ne pas exclure des causes possibles, même si vous les considérez comme peu probables.

Une séance de réflexion avec une autre personne est souvent utile dans cette phase, surtout si cette personne peut constamment apporter d'autres points de vue pendant la discussion. (Dans de nombreux pays, cette personne serait décrite comme étant parfaite dans le rôle de l'avocat du diable). La phase d'analyse tire souvent parti de deux types de personnes, dont l'une a d'excellentes connaissances techniques du produit, et l'autre oblige constamment la première personne à justifier ses opinions et à réfléchir de manière logique et transversale.

Phase d'implémentation

Dans la phase d'implémentation, vous devez éliminer chaque cause possible. Ce processus d'élimination retourne généralement des résultats optimum lorsque les causes possibles sont éliminées dans l'ordre, de la cause la plus probable à la cause la moins probable.

L'aspect essentiel de la phase d'implémentation est de s'assurer que vos raisons d'éliminer des causes possibles sont valides d'un point de vue logique.

Si vous arrivez à la fin de la liste des causes possibles sans avoir trouvé de solution au problème, vous devez retourner à la phase d'analyse et revoir les causes possibles. Si votre analyse est correcte, vous devrez peut-être revoir vos hypothèses initiales dans la phase d'examen.

Phase de validation

Trop de personnes, notamment celles qui débutent dans la résolution de problèmes, supposent que les problèmes sont résolus alors qu'ils ne le sont pas. Ne partez pas du principe que, comme vous avez trouvé et résolu un problème, vous avez résolu le problème d'origine.

Dans la phase d'examen, vous devez avoir déterminé comment l'utilisateur décide que le problème est résolu. Dans la phase de validation, vous devez appliquer ce test pour voir si le problème est vraiment résolu.

Documentation

Une fois que le problème est résolu, il est vraiment important de veiller à ce que tout le personnel de l'organisation tire les leçons de ce qui s'est passé.

Question : comment savoir qu'un problème est résolu ?

Leçon 2

Résolution des problèmes liés aux services

- Résolution des problèmes liés aux services
- Journal des erreurs SQL Server
- Journaux d'événements Windows
- Démonstration 2A : Résolution des problèmes liés aux services

Dans le reste de ce module, vous apprendrez à résoudre des types de problèmes courants qui peuvent survenir lorsque vous utilisez les systèmes SQL Server. Cette leçon couvre le premier de ces types de problèmes qui concerne les problèmes liés aux services SQL Server.

SQL Server comprend une série de services Windows. La résolution des problèmes liés à ces services s'apparente beaucoup à la résolution des problèmes liés aux autres services Windows, mais certaines considérations sont propres à SQL Server.

Objectifs

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

- résoudre les problèmes liés aux services ;
- utiliser le journal des erreurs SQL Server ;
- utiliser les journaux d'événements Windows.

Résolution des problèmes liés aux services

Problème standard : un ou plusieurs services SQL Server ne démarrent pas ou ne sont pas accessibles

- Consultez les journaux d'erreurs Windows et SQL
- Si SQL Server peut démarrer, mais n'est pas accessible
 - Vérifiez les problèmes liés au réseau
 - Essayez d'accéder à SQL Server via la connexion DAC
- Si SQL Server ne démarre pas
 - Consultez le journal système Windows
 - Vérifiez si les bases de données master et model sont endommagées
 - Vérifiez que les chemins d'accès aux fichiers tempdb sont accessibles
 - Essayez de démarrer le service à partir de l'invite de commandes



Points clés

Le problème le plus courant lié aux services est que l'un des services SQL Server ne démarre pas ou n'est pas accessible. Comme avec tous les services Windows, les services SQL Server ne sont pas autorisés à interagir directement avec la console système, ce qui limite les options dont disposent le service pour vous informer des problèmes. En outre, aucun utilisateur ne peut être connecté à la console pour recevoir des notifications même s'il a été autorisé, car les services s'exécutent en arrière-plan et sont principalement démarrés automatiquement, même avant qu'un utilisateur se connecte.

Journaux Windows et SQL Server

La cause la plus courante d'un service qui ne démarre pas est que la connexion du compte de service échoue pour une raison quelconque. Cela peut être causé par un mot de passe incorrect ou expiré, ou le compte peut être verrouillé. Les échecs de connexion des services apparaissent généralement dans le journal des événements système Windows. Les problèmes plus complexes tels que des fichiers manquants provoquent l'écriture des entrées dans les journaux SQL Server.

Vérifiez les journaux Windows et SQL Server comme première étape de la résolution des problèmes de démarrage de service.

Autres problèmes liés aux services

Si SQL Server démarre mais n'est pas accessible, le problème est souvent lié à des problèmes de réseau qui sont décrits plus loin dans ce module. Dans de rares cas, le planificateur SQL Server peut se bloquer et ne plus accepter de nouvelles connexions. Dans ce cas, et dans des cas similaires, vous devez également essayer de vous connecter à SQL Server à l'aide de la connexion administrateur dédiée (DAC, Dedicated Administration Connection). Vous devrez peut-être utiliser la connexion administrateur dédiée pour accéder à l'instance et analyser le problème. Vous devrez peut-être parfois supprimer les sessions individuelles à l'origine du problème.

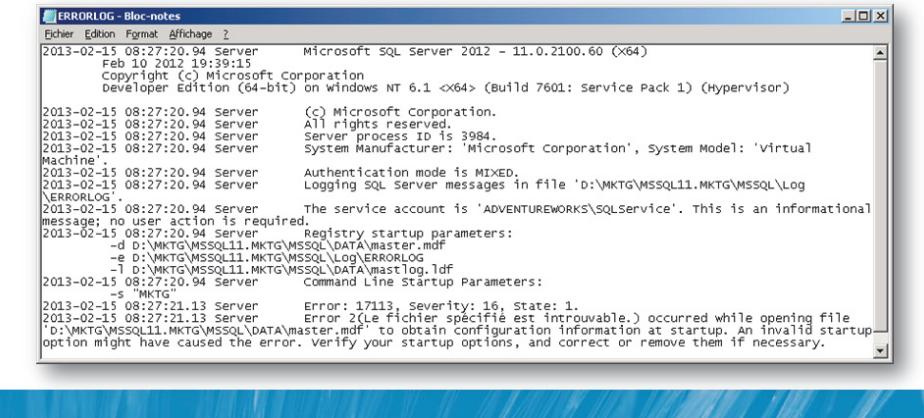
Si SQL Server ne démarre pas, mais le problème n'est pas lié à la connexion, vérifiez les éléments suivants :

- Vérifiez si les fichiers journaux SQL Server indiquent que la base de données master ou model est endommagée. Si l'une d'entre elles est endommagée, suivez les procédures de récupération des bases de données qui sont décrites dans le module 7.
- Vérifiez si les chemins d'accès aux fichiers de la base de données tempdb sont accessibles. La base de données tempdb est recréée à chaque démarrage de l'instance du serveur, mais le chemin d'accès aux fichiers de la base de données (tels qu'ils sont configurés dans la base de données master) doit exister et être accessible.
- Essayez de démarrer l'instance à l'aide de l'invite de commandes. S'il est possible de démarrer SQL Server à partir d'une invite de commandes, vérifiez la configuration du service et assurez-vous que les conditions d'autorisation sont remplies.

Question : qu'est-ce qui doit d'abord être examiné lorsqu'un problème relatif à un service se produit avec SQL Server ?

Journal des erreurs SQL Server

- La Visionneuse du fichier journal ne peut pas être utilisée si l'instance n'a pas démarré
 - Utilisez l'éditeur pour ouvrir le fichier journal actuel et l'examiner
 - Examinez également les journaux précédents parce que le problème a pu se produire auparavant



The screenshot shows a Windows Notepad window with the title 'ERRORLOG - Bloc-notes'. The content of the log file is as follows:

```
Fichier Edition Format Affichage 2
2013-02-15 08:27:20.94 server Microsoft SQL Server 2012 - 11.0.2100.60 (x64)
Feb 10 2013 10:39:15
Copyright (c) Microsoft Corporation
Developer Edition (64-bit) on windows NT 6.1 <x64> (Build 7601: service Pack 1) (Hypervisor)

2013-02-15 08:27:20.94 Server (C) Microsoft Corporation.
2013-02-15 08:27:20.94 Server All rights reserved.
2013-02-15 08:27:20.94 Server Server process ID is 3984.
2013-02-15 08:27:20.94 Server System Manufacturer: 'Microsoft Corporation', System Model: 'virtual
Machine'
2013-02-15 08:27:20.94 Server Authentication mode is MIXED.
2013-02-15 08:27:20.94 Server Logging SQL Server messages in file 'D:\MKTG\MSSQL11.MKTG\MSSQL\Log
\ERRORLOG'.
2013-02-15 08:27:20.94 Server The service account is 'ADVENTUREWORKS\SQLService'. This is an informational
message; no user action is required.
2013-02-15 08:27:20.94 Server Registry startup parameters:
-td D:\MKTG\MSSQL11.MKTG\MSSQL\DATA\master.mdf
-ld D:\MKTG\MSSQL11.MKTG\MSSQL\LOG\ERRORLOG
-9 D:\MKTG\MSSQL11.MKTG\MSSQL\DATA\mastlog.ldf
2013-02-15 08:27:20.94 Server Command Line Startup Parameters:
-s "MKTG"

2013-02-15 08:27:21.13 Server Error: 17113, Severity: 16, State: 1.
2013-02-15 08:27:21.13 Server Error 2[Le fichier spécifié est introuvable.] occurred while opening file
'D:\MKTG\MSSQL11.MKTG\MSSQL\DATA\master.mdf' to obtain configuration information at startup. An invalid startup
option might have caused the error. Verify your startup options, and correct or remove them if necessary.
```

Points clés

Le journal SQL Server fournit souvent des informations détaillées sur les problèmes de démarrage. Il peut être affiché à l'aide de la visionneuse du journal dans le cadre de SSMS.

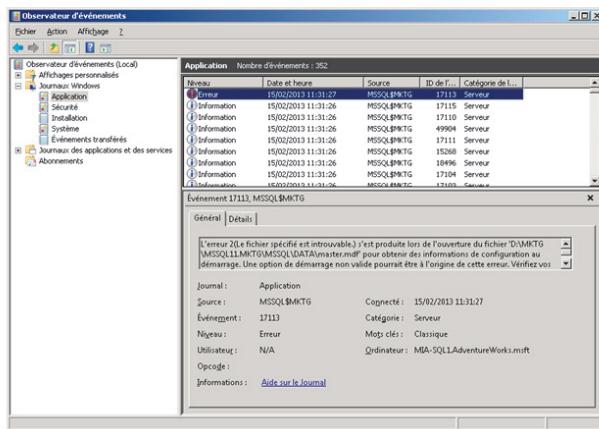
Toutefois, si SQL Server ne peut pas démarrer, vous ne pouvez pas utiliser la visionneuse du journal dans SSMS pour afficher le contenu des journaux SQL Server. Mais vous pouvez ouvrir ces journaux dans un éditeur tel que le Bloc-notes, comme illustré dans l'exemple sur la diapositive.

Le journal des erreurs SQL Server a été traité plus en détail dans le module 15. Bien que le dernier journal d'erreurs fournit généralement les informations les plus utiles, SQL Server conserve un jeu de fichiers journaux d'archive que vous devez également examiner, car les problèmes se sont peut-être produits depuis un certain temps.

Question : quel est le problème lié au démarrage de l'instance dans l'exemple affiché dans la diapositive ?

Journaux d'événements Windows

- Journal système pour examiner les informations Windows
- Journal des applications pour rechercher les messages liés à l'application



Points clés

Les services Windows ne sont pas autorisés à interagir directement avec la console. Le principal emplacement qu'ils peuvent utiliser pour vous informer des problèmes est le journal des événements système Windows.

En outre, comme avec d'autres applications, SQL Server écrit une grande quantité d'informations dans le journal des applications Windows au fil du temps.

Ces deux journaux doivent être vérifiés régulièrement, car certaines erreurs qui y sont consignées produisent des symptômes qui sont apparents aux utilisateurs.

Question : dans quel journal Windows les échecs de connexion s'affichent-ils pour le service SQL Server ?

Démonstration 2A : Résolution des problèmes liés aux services

Dans cette démonstration, vous allez apprendre à résoudre un problème relatif à un service

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**, sur **Projet/Solution**, naviguez jusqu'à **D:\22462A_Labs\22462A_20_PRJ\22462A_20_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 – Demonstration 2A.sql**.
5. Suivez les instructions contenues dans les commentaires du fichier de script.

Leçon 3

Résolution des problèmes de connexion et de connectivité

- Résolution des problèmes de connectivité
- Résolution des échecs de connexion

Le dernier type de problèmes que les administrateurs de base de données qui utilisent SQL Server sont généralement invités à résoudre sont les problèmes de connectivité et de connexion. Les utilisateurs peuvent rencontrer des problèmes de connexions à SQL Server, en rapport avec les noms de réseau, les protocoles et les ports, et les utilisateurs peuvent rencontrer des problèmes avec les mots de passe qui doivent être résolus. En général, vous devez essayer de créer des connexions basées sur l'appartenance au groupe Windows. Cela vous évite en tant qu'administrateur de base de données de résoudre la plupart des problèmes liés aux mots de passe et aux comptes Windows.

Objectifs

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

- résoudre les problèmes de connectivité ;
- résoudre les échecs de connexion.

Résolution des problèmes de connectivité

Problème standard : impossible de se connecter à SQL Server

- Essayez d'accéder à la mémoire partagée sur le serveur
 - Si aucun accès n'est possible via la mémoire partagée, corrigez le problème de connexion et de service
- Testez la connectivité réseau
 - Le nom du serveur peut-il être résolu ?
 - Le réseau et le serveur sont-ils accessibles ?
 - Le service Browser s'exécute-t-il pour les instances nommées qui n'utilisent pas de ports fixes ?
 - Le client est-il configuré pour utiliser le protocole et les paramètres appropriés ?
 - Un pare-feu bloque-t-il la connectivité ?

Points clés

Le problème le plus courant de ce type que vous devrez peut-être résoudre est que les utilisateurs ne peuvent pas se connecter à SQL Server.

Supprimer le réseau

La première étape pour essayer de résoudre ces problèmes consiste à déterminer si le problème est lié à la connectivité réseau. Vous pouvez souvent ne plus tenir compte du réseau en essayant de vous connecter via une connexion de mémoire partagée lorsqu'elle est locale au serveur. Si vous ne pouvez pas accéder à SQL Server à l'aide d'une connexion de mémoire partagée, vous devez réparer la connexion ou le service. La résolution des problèmes de connexion est traitée dans la rubrique suivante. La résolution des problèmes liés aux services a été abordée dans la leçon 2.

Problèmes liés au réseau

Si la connexion à l'aide de la mémoire partagée réussit, le problème est presque toujours lié au réseau. Dans de rares cas, vous pouvez rencontrer un problème de bibliothèques réseau incompatibles sur le client et le serveur, mais la plupart des problèmes sont beaucoup moins subtiles.

Problème	Actions
Le nom du serveur peut-il être résolu ?	Le système client doit pouvoir rechercher l'adresse réseau du serveur. Dans les réseaux TCP/IP, cela implique généralement une recherche DNS. Vérifiez que vous pouvez exécuter une commande Ping sur le serveur sur lequel SQL Server est installé et qu'une adresse IP est renvoyée par la commande Ping.
Le réseau et le serveur sont-ils accessibles ?	Lorsqu'un nom de serveur peut être correctement résolu en une adresse réseau, aucune connectivité n'est possible entre les systèmes client et serveur. Sur les réseaux TCP/IP, vérifiez que vous pouvez exécuter une commande Ping sur le serveur sur lequel SQL Server est installé, en supposant qu'il ne peut pas envoyer en écho les commandes Ping désactivées. Dans de nombreux cas, les problèmes dans ce domaine sont liés à des adresses de sous-réseau ou de passerelle/routeur par défaut incorrectes.
Le service Browser s'exécute-t-il pour les instances nommées qui n'utilisent pas de ports fixes ?	Si vous utilisez des instances nommées de SQL Server, les systèmes clients doivent être en mesure de résoudre les noms des instances sur un numéro de port. Par défaut, il suffit de se connecter au service SQL Browser sur le port UDP 1434. Si le service SQL Browser est désactivé, les instances nommées doivent être configurées pour utiliser des ports fixes et les systèmes clients doivent être configurés pour accéder à ces ports, éventuellement par la création d'alias. Vérifiez que les alias sont créés pour la bibliothèque réseau appropriée telle que 32 bits et 64 bits.
Le client est-il configuré pour utiliser le protocole et les paramètres appropriés ?	Le protocole que le client tente d'utiliser pour se connecter au serveur est-il réellement activé sur le serveur ? Les paramètres du protocole sont-ils appropriés ?
Un pare-feu bloque-t-il la connectivité ?	Vérifiez qu'un pare-feu ne bloque pas les ports sur lesquels vous essayez de vous connecter. Si un pare-feu bloque l'accès, une exception ou une règle devra probablement être configurée dans le pare-feu.

Question : pourquoi une connexion locale doit-elle être testée en premier ?

Résolution des échecs de connexion

Problème standard : impossible d'ouvrir une session SQL Server

- Connexions Windows
 - Le contrôleur de domaine est-il disponible ?
 - SQL Server peut-il communiquer avec le contrôleur de domaine ?
- Connexions SQL Server
 - SQL Server est-il configuré pour le mode mixte ?
 - Le mot de passe est-il correct ?
 - La connexion est-elle verrouillée ou existe-t-il des règles pour changer le mot de passe ?
- Considérations d'ordre général
 - La connexion est-elle activée et a-t-elle l'autorisation CONNECT ?
 - La base de données par défaut/demandée est-elle disponible et dispose-t-elle des autorisations d'accès ?

Points clés

Le problème standard avec ce type de problème est qu'un utilisateur peut établir une connexion réseau à SQL Server mais ne peut pas se connecter. Les actions de résolution des problèmes que vous devez exécuter varient selon le type de connexion utilisé :

- Pour les connexions Windows, vérifiez que SQL Server peut se connecter à un contrôleur de domaine pour traiter les connexions. Pour examiner les problèmes potentiels, consultez les journaux Windows.
- Pour les connexions SQL Server, vérifiez que SQL Server est configuré pour l'authentification SQL Server. Des connexions SQL Server peuvent être créées et activées, même si SQL Server est configuré pour l'authentification Windows uniquement. L'erreur la plus courante rencontrée dans ce cas est qu'une connexion approuvée n'est pas disponible.

De même, lorsque vous utilisez les connexions SQL Server, vérifiez que le mot de passe fourni est correct et que la connexion n'a pas été verrouillée par la stratégie du compte. Un autre problème courant pour les connexions SQL Server est que la connexion peut expirer mais l'application qui tente de se connecter à SQL Server ne comprend pas l'expiration du mot de passe. Cette situation est courante avec les applications conçues pour les versions antérieures de SQL Server qui n'ont pas implémenté la stratégie du compte pour les connexions SQL Server.

Pour les deux types de connexion, vérifiez que la connexion est autorisée à se connecter à SQL Server et qu'elle a obtenu l'autorisation d'accéder à la base de données à laquelle elle tente de se connecter. Cette vérification doit inclure la base de données par défaut pour la connexion.

Si un problème de connexion se produit avec plusieurs utilisateurs différents, vérifiez si un déclencheur de connexion n'a pas échoué. Lorsqu'un déclencheur de connexion empêche les utilisateurs de se connecter, le message d'erreur retourné aux utilisateurs interactifs indique qu'un déclencheur de connexion a bloqué la connexion.

Question : une connexion SQL Server peut-elle être créée et activée lorsque SQL Server est configuré pour l'authentification Windows uniquement ?

Leçon 4

Résolution des problèmes liés aux accès concurrentiels

- Principaux concepts liés aux accès concurrentiels
- Résolution des problèmes de blocage
- Démonstration 4A : Résolution des problèmes de blocage
- Résolution des problèmes d'interblocage
- Démonstration 4B : Résolution des problèmes d'interblocage

SQL Server essaie de donner à chaque l'utilisateur l'illusion qu'il est le seul utilisateur sur le système. Il n'est pas possible pour SQL Server de donner cette illusion dans toutes les circonstances, et les actions d'un utilisateur ont un impact sur celles des autres utilisateurs.

Il est important de comprendre les principaux concepts liés aux accès concurrentiels et de savoir comment résoudre les problèmes de blocage et d'interblocage les plus courants.

Objectifs

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

- décrire les principaux concepts liés aux accès concurrentiels ;
- résoudre les problèmes de blocage ;
- résoudre les problèmes d'interblocage.

Principaux concepts liés aux accès concurrentiels

- SQL Server doit vérifier les propriétés ACID des transactions
- Le verrouillage est le mécanisme utilisé pour synchroniser l'accès simultané de plusieurs utilisateurs à la même donnée
- Les niveaux d'isolement définissent le type et la longueur des verrous utilisés
- Deux principaux types de verrous
 - **Verrous partagés** : autorisent l'accès en lecture, mais pas en écriture
 - **Verrous exclusifs** : interdisent l'accès en lecture ou en écriture
- Les verrous empêchent des conflits de mise à jour
 - Le verrouillage garantit la sérialisation des transactions
 - Le verrouillage est automatique
 - Les verrous permettent un accès concurrentiel sécurisé aux données

Points clés

Examinez vos attentes lorsque vous transférez de l'argent d'un compte bancaire à un autre. Vous vous attendez à ce que la même somme d'argent soit retirée d'un compte et ajoutée à l'autre compte. Vous vous attendez également à ce que l'opération entière se produise ou ne se produise pas, même si un type d'échec survient pendant le transfert. En revanche, vous ne vous attendez pas à ce que les fonds soient retirés d'un compte sans être ajoutés à l'autre compte.

Examinez également ce que vous vous attendriez à voir si vous consultiez l'état des deux comptes pendant que le transfert était en cours. Vous vous attendriez à toujours voir les comptes dans un état cohérent, selon le solde des comptes avant le transfert ou selon le solde des comptes à la fin du transfert. Vous ne vous attendriez pas à pouvoir voir les comptes dans un état intermédiaire.

Le dernier problème à prendre en compte est ce que vous vous attendriez à voir si vous consultiez l'état des comptes le lendemain de la transaction réussie. Vous vous attendriez à ce que la transaction soit toujours affichée comme étant terminée.

Propriétés ACID

Vos attentes sont étroitement liées à un ensemble de propriétés que le secteur de l'informatique désigne comme étant les propriétés ACID attendues des transactions :

- Atomicité : une transaction a un début et une fin définis. Elle est exécutée ou restaurée entièrement.
- Cohérence : la base de données n'est pas laissée dans un état intermédiaire à la fin de la transaction. Toutes les contraintes logiques sont appliquées.
- Isolation : les transactions sont empêchées d'interférer entre elles.
- Durabilité : une fois qu'une transaction est validée, les détails de la transaction sont conservés dans le système et survivent à un redémarrage.

Pour répondre aux exigences d'isolation, SQL Server utilise un schéma de verrouillage. Le verrouillage est un mécanisme utilisé par le moteur de base de données pour synchroniser l'accès simultané de plusieurs utilisateurs à la même donnée. L'objectif est d'optimiser les accès concurrentiels tout en conservant les propriétés ACID.

 **Remarque** Les accès concurrentiels sont traités en détail dans le cours 10776A : Developing Microsoft SQL Server 2012 Databases (*en anglais*).

Comportement de verrouillage

Avant qu'une transaction acquière une dépendance sur l'état actuel d'un élément de données, par exemple en lisant ou en modifiant les données, elle doit se protéger des effets d'une autre transaction qui tente de modifier les mêmes données. Pour ce faire, la transaction demande un verrou sur l'élément de données. Les verrous ont différents modes, par exemple partagé ou exclusif. Le mode de verrouillage définit le niveau de dépendance de la transaction sur les données.

Aucune transaction ne peut obtenir un verrou qui entre en conflit avec le mode d'un verrou qui a déjà été accordé à une autre transaction sur les mêmes données. Si une transaction demande un mode de verrou qui entre en conflit avec un verrou qui a déjà été accordé sur les mêmes données, le moteur de base de données bloque la transaction qui demande un verrou tant que le premier verrou n'est pas libéré.

Pour les opérations UPDATE, SQL Server maintient toujours des verrous à la fin de la transaction. Pour les opérations SELECT, il maintient le verrou qui protège la ligne pendant une période qui dépend du niveau d'isolation de la transaction. Tous les verrous qui sont encore maintenus à la fin d'une transaction sont libérés, si la transaction est validée ou restaurée.

Le verrouillage est primordial pour le traitement des transactions et est un comportement normal pour le système. Des problèmes se produisent uniquement lorsque les verrous sont maintenus trop longtemps et que les autres transactions sont bloquées trop longtemps en raison du maintien des verrous.

Question : pourquoi les instructions SELECT acquièrent-elles les verrous dans la plupart des cas ?

Résolution des problèmes de blocage

Problème standard : les applications ne semblent pas être en cours de traitement, mais également ne retournent pas d'erreurs

- Les processus sont bloqués lorsqu'ils attendent des ressources verrouillées
 - Comportement normal, essentiel pour le système
 - La ressource est affichée en attente d'acquisition d'un verrou
 - Problème uniquement lorsque la durée d'attente est longue
- Surveillance des scénarios de blocage interminables
 - Moniteur d'activité SQL Server
 - Vues de gestion dynamique
 - Collecteur de données SQL Server

Points clés

Le problème le plus courant qui se produit avec le blocage est que les applications apparaissent comme étant bloquées. Les symptômes associés sont que l'application n'est pas occupée et a des tâches à exécuter, mais aucun travail n'est effectué. Cette situation indique généralement que l'application est bloquée alors qu'elle attend une ressource d'un certain type.

Vous avez appris dans la dernière rubrique que le verrouillage consiste à acquérir et à maintenir les verrous qui sont utilisés pour implémenter le contrôle des accès concurrentiels, et que le blocage désigne ce qui arrive à un processus pendant qu'il doit attendre une ressource verrouillée par un autre processus. Un blocage est une occurrence normale pour les systèmes et ne constitue pas un problème. Un blocage excessif peut être un problème.

Un blocage peut être surveillé en temps réel à l'aide du Moniteur d'activité SQL Server et des vues de gestion dynamique. Vous apprendrez à surveiller un blocage dans la démonstration 3A.

Le collecteur de données SQL Server qui a été traité dans le module 18 peut être utilisé pour collecter des données sur les scénarios de blocage, notamment des informations sur tous les processus impliqués dans le blocage. Les rapports fournis avec le collecteur de données via l'entrepôt de données de gestion peuvent s'avérer utiles lors de l'examen des problèmes de blocage qui se sont produits précédemment.

Question : à quoi selon vous un blocage « excessif » peut-il faire référence ?

Démonstration 4A : Résolution des problèmes de blocage

Dans cette démonstration, vous allez apprendre à résoudre un problème de blocage

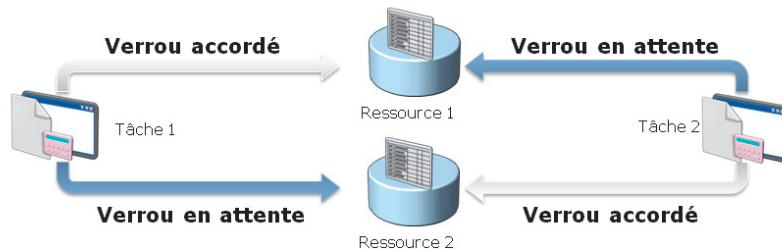
Procédure de démonstration

1. Si la démonstration 2A n'a pas été effectuée :
 - Rétablissez les ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A_Labs\Revert.txt**.
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_20_PRJ\22462A_20_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 **41 – Démonstration 4A.sql** à partir de l'**Explorateur de solutions**.
5. Suivez les instructions contenues dans les commentaires du fichier de script.

Résolution des problèmes d'interblocage

Problème standard : des erreurs de blocage (1205) sont retournées

- Les interblocages sont les scénarios de blocage circulaire
- SQL Server détecte automatiquement les blocages
 - Restaure les transactions avec l'erreur 1205 sur le système victime du blocage
- Surveiller l'utilisation des occurrences de blocage avec le Générateur de profils et Trace SQL



Points clés

Les erreurs d'interblocage sont un type particulier d'erreur de blocage nécessitant l'intervention de SQL Server, faute de quoi les verrous ne sont pas libérés. Le type d'interblocage le plus courant se produit lorsque deux transactions ont des verrous sur des objets séparés et que chaque transaction demande un verrou sur l'objet de l'autre transaction. Par exemple :

- La tâche 1 maintient un verrou partagé sur la ligne 1.
- La tâche 2 maintient un verrou partagé sur la ligne 2.
- La tâche 1 demande un verrou exclusif sur la ligne 2, mais il ne sera pas accordé tant que la tâche 2 ne libère pas le verrou partagé.
- La tâche 2 demande un verrou exclusif sur la ligne 1, mais il ne sera pas accordé tant que la tâche 1 ne libère pas le verrou partagé.
- Chaque tâche doit attendre que l'autre libère le verrou, ce qui ne se produit jamais.

Un interblocage peut se produire lorsque plusieurs transactions longues s'exécutent simultanément dans la même base de données. Un interblocage peut également se produire en raison de l'ordre dans lequel l'optimiseur traite une requête complexe, telle qu'une jointure.

Comment SQL Server met fin à un interblocage

SQL Server met fin à un interblocage en arrêtant automatiquement l'une des transactions. SQL Server effectue les actions suivantes :

- Il choisit une victime de l'interblocage. SQL Server donne la priorité au processus qui a la priorité d'interblocage la plus élevée. Si les deux processus ont la même priorité d'interblocage, SQL Server restaure la transaction qui est la moins coûteuse à restaurer.
- Il restaure la transaction de la victime de l'interblocage.
- Il notifie l'application de la victime de l'interblocage (avec le numéro de message 1205).
- Il autorise l'autre transaction à continuer.



Important Dans un environnement multi-utilisateurs, chaque client doit rechercher le numéro de message 1205, qui indique que la transaction a été restaurée. Si le message 1205 est détecté, l'application doit se reconnecter et tenter de réexécuter la transaction.

Utilisation des interblocages

Les interblocages ne sont normalement pas enregistrés. La seule indication qu'un interblocage s'est produit est qu'un message d'erreur est retourné au client qui a été sélectionné comme victime.

Pour surveiller les interblocages, utilisez SQL Server Profiler et/ou Trace SQL. Plusieurs événements d'interblocage sont disponibles. L'événement qui contient le plus d'informations et qui est le plus facile à lire est l'événement Deadlock Graph. Un exemple de graphique d'interblocage est disponible dans la démonstration 3B.

Notez qu'il existe d'autres outils qui peuvent être utiles pour les interblocages qui ne sont pas traités dans ce cours :

- Les indicateurs de trace peuvent être définis pour écrire les informations relatives aux interblocages dans le journal des erreurs SQL Server.
- Les notifications d'événements peuvent être générées.
- Les événements étendus peuvent être suivis.

Question : avez-vous rencontré des problèmes d'interblocage dans votre environnement actuel ? Si oui, comment avez-vous déterminé que les interblocages constituaient un problème, et comment le problème a-t-il été résolu ?

Démonstration 4B : Résolution des problèmes d'interblocage

Dans cette démonstration, vous allez apprendre à

- Tracer un interblocage avec SQL Server Profiler
- Extraire des événements d'interblocage de SQL Server Profiler
- Ouvrir des fichiers d'interblocage avec SSMS

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**.
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_20_PRJ\22462A_20_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 **42 – Demonstration 4B.sql**.
5. Suivez les instructions contenues dans les commentaires du fichier de script.

Atelier pratique 20 : Résolution des problèmes courants

- Exercices 1 à 5 : Dépanner et résoudre les problèmes liés à l'administration SQL Server

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 :

- Rétablissement des ordinateurs virtuels conformément aux instructions fournies dans **D:\22462A_Labs\Revert.txt**.
- Sur l'ordinateur virtuel, cliquez sur **Démarrer**, sur **Tous les programmes**, sur **Microsoft SQL Server 2012**, puis sur **SQL Server Management Studio**.
- Dans la fenêtre Se connecter au serveur, tapez **Proseware** dans la zone de texte **Nom du serveur**.
- Dans la zone de liste déroulante **Authentification**, sélectionnez **Authentification Windows**, puis cliquez sur **Se conn.**
- Dans le menu **Fichier**, cliquez sur **Ouvrir**, puis sur **Projet/Solution**.
- Dans la fenêtre Ouvrir un projet, sélectionnez le projet **D:\22462A_Labs\22462A_20_PRJ\22462A_20_PRJ.ssmssqln**.
- 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 être en mesure de résoudre les problèmes courants liés aux processus et services SQL Server lorsqu'ils se produisent. Il y a cinq exercices qui créent des problèmes. Vous devez essayer de dépanner et résoudre le plus de problèmes possibles.

Documentation

Exercice 1

Les utilisateurs de l'application Promote se plaignent que l'application ne peut plus se connecter au serveur. L'application se connecte à l'aide de la connexion SQL PromoteApp.

Exercice 2

Un administrateur de base de données junior a créé une sauvegarde de la base de données AdminDB de production à envoyer à l'équipe de développement à des fins de test. Les utilisateurs de cette base de données se plaignent qu'ils ne peuvent plus s'y connecter.

Exercice 3

Les utilisateurs se plaignent que le travail permettant d'obtenir la liste des fichiers dans SQL Server Agent ne s'exécute pas.

Exercice 4

Les performances de la base de données AdminDB ont enregistré une baisse constante depuis le déploiement de la base de données. Les requêtes qui s'exécutaient en quelques secondes s'exécutent maintenant en plusieurs minutes.

Exercice 5

Une situation étrange se produit avec la base de données CityDetails. La plupart des bases de données fonctionnent plus lentement lorsque d'autres utilisateurs sont ajoutés. Mais la base de données CityDetails est moins performante lorsqu'un seul utilisateur y est connecté. Le délai de connexion pour cette seule base de données peut être très long et expire parfois. Les autres bases de données fonctionnent normalement.

Exercices 1 à 5 : Dépanner et résoudre les problèmes liés à l'administration SQL Server

Scénario

Plusieurs problèmes ont été signalés par les utilisateurs du serveur Proseware. Vous devez dépanner et résoudre les problèmes.

Les tâches principales de chaque exercice sont les suivantes :

1. Lire la documentation associée à l'exercice.
2. Dépanner et résoudre le problème.

► **Tâche 1 : Lire la documentation associée à l'exercice**

- Lire la documentation associée à l'exercice.

► **Tâche 2 : Dépanner et résoudre le problème**

- Dépannez le problème.
- Résolvez le problème.
- Si vous avez des difficultés, consultez la solution dans le corrigé.

Résultats : À la fin de ces exercices, vous devez avoir examiné et résolu plusieurs problèmes.

Contrôle des acquis du module et éléments à retenir

- Questions de contrôle des acquis
- Meilleures pratiques

Questions de contrôle des acquis

1. Que peut-on utiliser pour surveiller les problèmes de blocage ?
2. Que se passe-t-il lorsqu'un blocage se produit ?

Meilleures pratiques

1. Surveiller votre système et stocker les données historiques à l'aide du collecteur de données par exemple pour faciliter le dépannage des problèmes.
2. Identifier clairement les problèmes sous-jacents plutôt que de s'attaquer aux symptômes.
3. Appliquer l'analyse de l'interblocage avec Trace SQL.

Contrôle des acquis et évaluation du cours



Votre évaluation de ce cours aidera Microsoft à comprendre la qualité de votre expérience de formation.

Consultez votre formateur pour accéder au formulaire d'évaluation du cours.

Votre évaluation est strictement confidentielle et vos réponses à cette enquête seront utilisées dans le seul but d'améliorer la qualité des prochains cours Microsoft. Vos commentaires ouverts et honnêtes sont très précieux.

Annexe A

Concepts principaux de la haute disponibilité et de la réPLICATION dans SQL Server

Sommaire :

Leçon 1 : Concepts principaux de la haute disponibilité	A-3
Leçon 2 : Concepts principaux de la réPLICATION	A-12

Vue d'ensemble de l'annexe

- Concepts principaux de la haute disponibilité
- Concepts principaux de la réPLICATION

À mesure que vous acquérez de l'expérience en matière de bases de données, vous risquez de devoir implémenter des technologies avancées disponibles dans la plateforme Microsoft SQL Server. Parmi les domaines essentiels on peut citer la haute disponibilité et la réPLICATION.

Dans cette annexe, vous allez vous familiariser avec les concepts principaux de chacune de ces technologies. En particulier, vous allez apprendre quand utiliser chacune d'elles.

Objectifs

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

- expliquer les différentes options de disponibilité offertes pour Microsoft SQL Server et savoir quand les utiliser ;
- expliquer les différentes options de réPLICATION offertes pour Microsoft SQL Server et savoir quand les utiliser.

Leçon 1

Concepts principaux de la haute disponibilité

- Discussion : Expérience passée en matière de haute disponibilité
- Mise en miroir de bases de données
- Clustering avec basculement
- Groupes de disponibilité AlwaysOn
- Envoi de journaux
- Choix parmi les options de haute disponibilité

SQL Server offre un large éventail d'options pour garantir la plus haute disponibilité possible des serveurs et des bases de données. Il est important que vous compreniez les concepts principaux impliqués et que vous sachiez faire un choix parmi les diverses options disponibles.

Objectifs

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

- décrire la mise en miroir de bases de données ;
- décrire le clustering avec basculement ;
- expliquer le rôle des technologies AlwaysOn ;
- décrire l'envoi de journaux ;
- choisir les cas d'usage appropriés pour chaque option de la haute disponibilité.

Discussion : Expérience passée en matière de haute disponibilité

- Que signifie pour vous le terme « haute disponibilité » ?
- Quelles techniques avez-vous déjà utilisées pour mettre en œuvre une configuration à haute disponibilité ?
- Ces tentatives ont-elles abouti ?



Points clés

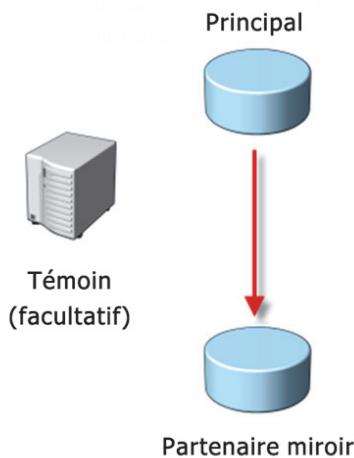
Dans cette discussion, les membres de la classe partageront leurs expériences en matière de stratégies de haute disponibilité. En particulier, considérez les questions suivantes (mais ne vous y limitez pas) :

- Que signifie pour vous le terme « haute disponibilité » ?
- Quelles techniques avez-vous déjà utilisées pour mettre en œuvre une configuration à haute disponibilité ?
- Ces tentatives ont-elles abouti ?

La haute disponibilité est un concept qui va au-delà d'une technologie. Elle implique également des processus définis, de la documentation, du savoir-faire individuel et bien plus encore. Si le personnel concerné ne sait pas que faire lorsque les problèmes se produisent, la technologie la plus avancée n'est d'aucune aide. Une bonne stratégie de sauvegarde fait également partie d'une solution de haute disponibilité.

Mise en miroir de bases de données

- Solution logicielle
 - Améliore la disponibilité d'une base de données
 - Améliore la protection des données
- Modes de fonctionnement
 - Haute sécurité
 - Performances élevées
- Basculement de rôle
 - Manuel
 - Automatisée (nécessite le témoin)



Points clés

La mise en miroir de bases de données est une solution logicielle qui vise à atteindre la haute disponibilité des bases de données et qui a été introduite dans Microsoft SQL Server 2005. Le journal des transactions d'une base de données sur un serveur est continuellement appliqué à une copie de la base de données résidant sur un autre serveur. Les bases de données à mettre en miroir doivent être en mode de restauration complète.

Dans la mesure où un serveur entièrement distinct est utilisé pour le partenaire mis en miroir, le risque d'un point de défaillance unique est fortement réduit. Bien que cela ne soit généralement pas souhaitable, les serveurs utilisés dans la mise en miroir de bases de données peuvent également être implémentés selon des architectures matérielles très différentes. Toutefois, la mise en miroir de bases de données est plus facile à gérer lorsque les serveurs concernés sont similaires, et lorsque la disposition du disque de chaque serveur est la même.

Il existe deux modes d'actions disponibles. En mode haute sécurité, le serveur principal envoie les informations de mise à jour à son propre journal des transactions et au journal des transactions du serveur partenaire en miroir, puis attend l'exécution des écritures sur celui-ci et ensuite attend l'exécution des écritures sur son propre journal. En mode haute performance, le serveur principal envoie les informations de mise à jour aux deux journaux mais attend seulement l'écriture de son propre journal. Le mode haute performance sacrifie la cohérence des données au profit de la performance.

Si une défaillance se produit, le partenaire miroir peut assurer le rôle principal. Cela peut se produire de deux manières. L'administrateur de base de données peut initier un basculement manuel, ou il peut se produire un basculement automatique. Les basculements automatisés requièrent les éléments suivants :

- le mode haute sécurité ;
- le partenaire de mise en miroir doit avoir été synchronisé avec le serveur primaire au moment du basculement. (Cela peut ne pas être le cas si le partenaire de mise en miroir était en cours de récupération après une défaillance ou un problème réseau) ;
- un serveur témoin a été configuré.

Un serveur témoin peut être n'importe quelle édition de SQL Server, y compris Express Édition, et doit résider sur un autre serveur physique, distinct à la fois du serveur principal et du partenaire de mise en miroir. Le serveur témoin sert à fournir un quorum au serveur partenaire de mise en miroir, où ils peuvent convenir que le serveur principal est en échec et qu'il faut opérer un basculement. Lors d'un échec, s'il est impossible de réaliser un basculement automatique, il est possible de « forcer le service » en donnant le rôle principal au partenaire de mise en miroir, bien qu'une perte de données soit une éventualité.

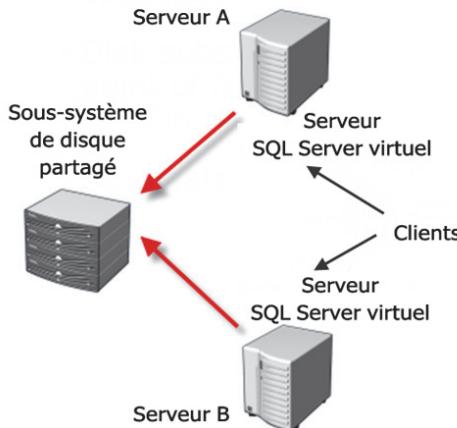
Notez que les bases de données système telles que master, msdb, tempdb, et la base de données model ne peuvent pas être mises en miroir. SQL Server 2008 a amélioré les performances de la mise en miroir de bases de données via l'introduction de la compression de flux du journal. Bien qu'il ne soit possible de fournir qu'un seul miroir pour chaque base de données, on peut mettre en miroir de nombreuses bases de données.

La mise en miroir peut améliorer la disponibilité des bases de données, y compris lors des périodes maintenance régulière telles que l'application nécessaire des services packs. Les données conservées sur le partenaire miroir ne sont pas accessibles, à moins de créer une capture instantanée de base de données sur le miroir, qui permet un accès en lecture seule des données.

La plus importante limite de la mise en miroir des bases de données provient du fait que les bases de données ont souvent des dépendances externes, telles que des connexions à la base de données master. Il convient de mettre en place des mécanismes ou des processus afin de garantir que ces objets dépendants sont présents sur les deux serveurs.

Clustering avec basculement

- Instance ou instances SQL Server® installées sur les différents nœuds de cluster de basculement Windows Server
- Protège contre les défaillances des serveurs en fournissant un serveur virtuel auquel les applications clientes se connectent
- Le sous-système de disque est un point de défaillance unique et nécessite des options de redondance intégrées
- Basculement automatisé



Points clés

Le clustering avec basculement de SQL Server est construit sur le clustering avec basculement de Windows Server. Il exige que chaque serveur soit un nœud de cluster de basculement Windows. Le clustering avec basculement est conçu pour se prémunir de la défaillance d'un serveur. Les clients se connectent à un SQL Server virtuel plutôt qu'à l'un ou l'autre des serveurs physiques.

Contrairement à la mise en miroir de bases de données, lorsqu'un basculement se produit, l'instance entière bascule. Cela supprime la complexité de la conservation des dépendances externes de base de données, car les bases de données système (telles que la base de données master) sont également basculées. Les bases de données n'ont pas besoin d'être en mode de restauration complète.

Les serveurs impliqués dans le clustering avec basculement doivent utiliser un matériel identique. Un outil de validation de cluster (CVT) est fourni pour vérifier la configuration d'un cluster de basculement avant que celui-ci ne soit créé. Passer les tests fournis par le CVT est une condition préalable pour obtenir le support technique Microsoft sur la configuration du cluster.

Le clustering de basculement utilise un sous-système de disque partagé. Il est important de noter qu'une seule copie des données est conservée. Cela signifie que le sous-système de disque doit intégrer ses propres mécanismes de redondance.

Un cluster SQL Server présente un serveur virtuel aux systèmes client. Quel que soit le serveur en cours de fonctionnement, les clients continuent à se connecter au même nom du serveur virtuel ou à la même adresse. Chaque serveur surveille les autres serveurs du cluster, de sorte que l'action de basculement puisse être initiée si nécessaire. Cette communication est normalement effectuée sur une connexion réseau privée (souvent appelée un réseau de pulsation). Le réseau privé est maintenu séparé des connexions du réseau public de chaque serveur.

À tout moment, chaque instance de SQL Server s'exécute uniquement sur un seul serveur. Lorsqu'un basculement de serveur se produit, le service SQL Server pour chaque instance du serveur secondaire doit être redémarré et les bases de données doivent être récupérées. Ce processus peut prendre du temps, selon le temps pris par la récupération de chaque base de données.

Groupes de disponibilité AlwaysOn

- Autre solution haute disponibilité d'entreprise
- Spécifiez les options permettant d'améliorer la disponibilité des bases de données et l'utilisation des ressources
 - Prend en charge une réplica principale et jusqu'à quatre réplicas secondaires
 - Les réplicas peuvent prendre en charge d'autres modes de disponibilité (asynchrone et synchrone)
 - Prend en charge d'autres options de basculement, comme le basculement manuel planifié
 - Tentative de basculement d'un groupe de disponibilité plutôt qu'une seule base de données
- Les réplicas secondaires peuvent fournir une prise en charge supplémentaire
 - Accès en lecture seule
 - Opérations de sauvegarde

Points clés

SQL Server 2012 a introduit les technologies AlwaysOn. Une de ces technologies, les groupes de disponibilité AlwaysOn, fournit une alternative à la mise en miroir de bases de données au niveau de l'entreprise. Les groupes de disponibilité AlwaysOn doivent être installés sur un nœud de cluster Windows et améliorent la mise en miroir de bases de données de la manière suivante :

Aspect	Mise en miroir de bases de données	Groupes de disponibilité AlwaysOn
S'applique à	Base de données	Groupe de bases de données
Réplicas	Un	Quatre
Utilisation du réplica	Il est possible de le rendre accessible en lecture seule au moyen de captures instantanées de base de données, mais ce n'est pas très utile en pratique.	Chaque réplica peut être mis en lecture seule et peut être utilisé pour les sauvegardes.
Mode de fonctionnement du réplica	Synchrone ou asynchrone	Les réplicas peuvent être synchrones ou asynchrones. Jusqu'à deux réplicas synchrones sont pris en charge. Les combinaisons des réplicas synchrones et asynchrones sont prises en charge.
Connexion client	Options disponibles de la redirection automatisée.	Options plus simples de la redirection automatisée disponibles.

Envoi de journaux

- Technologie au point et fiable
- Largement déployé
- Pas d'automatisation du basculement
- Processus principal
 - Sauvegarder le journal des transactions
 - Copier les journaux sur un autre serveur
 - Restaurer le journal des transactions sur un second serveur

Points clés

L'envoi de journaux est une technologie déjà rodée et fiable qui a été introduite dans SQL Server 2000. De nombreuses organisations et utilisateurs tiers ont également mis en œuvre leurs propres variantes de l'envoi de journaux. Fondamentalement, l'envoi de journaux est un processus simple et répétitif qui implique :

- la sauvegarde du journal des transactions sur le serveur principal ;
- la copie du journal des transactions sur le serveur secondaire ;
- la restauration du journal des transactions sur le serveur secondaire.

Chaque fois que la restauration est effectuée sur le serveur secondaire, les restaurations de base de données sont effectuées sans l'option de récupération, de sorte que les journaux de transactions supplémentaires peuvent être restaurés ultérieurement.

Lorsqu'une défaillance se produit, la mise en production du serveur secondaire est une tâche manuelle. Le produit ne fournit aucun moyen d'automatiser cette tâche.

L'envoi de journaux permet l'emploi de plusieurs serveurs secondaires.

Choix parmi les options de haute disponibilité

Aspect	Mise en miroir de bases de données	Clustering avec basculement	Groupes de disponibilité	Envoi de journaux
Unité de basculement	Base de données	Serveur	Groupe de bases de données	Base de données
Basculement automatisé	Oui, avec témoin	Oui	Oui	Non
Réplicas de données	1	0	4	Illimité
Complexité	Moyenne	Moyenne	Moyenne	Faible
Disponibilité des réplicas	Lecture seule avec capture instantanée	N/A	Lecture seule, sauvegarde	Lecture seule entre les restaurations

Points clés

Le tableau ci-dessus fournit une comparaison d'un certain nombre d'aspects de chaque option de haute disponibilité, tels que décrits dans les quatre rubriques qui précèdent. Notez qu'une solution de haute disponibilité peut comporter une combinaison de ces différentes technologies.

Avec le clustering de basculement, bien qu'une seule copie des données soit conservée, le sous-système de lecteur partagé doit toujours être configuré avec des options de redondance.

Avec l'envoi de journaux, l'accès en lecture seule des données du réplica n'est disponible que dans le cadre du mode veille.

Leçon 2

Concepts principaux de la réPLICATION

- Architecture de réPLICATION SQL Server
- RéPLICATION de capture instantanée
- RéPLICATION transactionnelle
- RéPLICATION transactionnelle d'égal à égal
- RéPLICATION de fusion
- Choix parmi les options de réPLICATION

SQL Server fournit un large éventail d'options de réPLICATION de données entre serveurs. Il est important que vous compreniez les concepts principaux impliqués et que vous sachiez faire un choix parmi les diverses options disponibles.

Objectifs

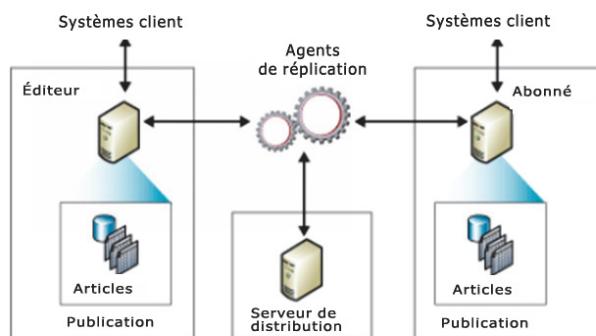
À la fin de cette leçon, vous serez à même d'effectuer les tâches suivantes :

- expliquer la métaphore serveur de publication/abonné sur laquelle est fondée la réPLICATION ;
- décrire la réPLICATION de capture instantanée ;
- décrire la réPLICATION transactionnelle ;
- décrire la réPLICATION transactionnelle d'égal à égal ;
- décrire la réPLICATION de fusion ;
- choisir entre les différentes options de réPLICATION.

Architecture de réPLICATION SQL Server

- Peut jouer un rôle dans le domaine de la haute disponibilité, mais destiné principalement à la distribution de données
- Pour reprendre une métaphore d'un éditeur de magazines

- Serveur de publication
- Publication
- Article
- Serveur de distribution
- Abonné



Points clés

Bien que la réPLICATION SQL Server puisse jouer un rôle dans la haute disponibilité (notamment par rapport à la réPLICATION transactionnelle d'égal à égal), son rôle majeur est lié à la distribution des données.

Il est important de comprendre la terminologie utilisée par la réPLICATION.

- Un serveur de publication est une instance qui publie des données que vont utiliser les abonnés. Une base de données doit être configurée pour autoriser la publication des données contenues dans celle-ci. (Dans la métaphore, cela correspond à la maison d'édition).
- Un article est l'objet en cours de publication. Un article peut comporter une table ou une vue. La table ou la vue entière peut être publiée, ou un article peut être basé sur le filtrage vertical (colonnes), le filtrage horizontal (lignes) ou les deux. (Dans la métaphore, cela a trait au concept d'article dans un magazine).
- Une publication est un groupe d'articles en cours de publication. C'est l'unité de comptage de l'abonnement. (Dans la métaphore, cela correspond au magazine).
- Un serveur de distribution est utilisé pour déplacer les publications depuis les serveurs de publication vers les abonnés. Une base de données de distribution est utilisée pour conserver les publications qui doivent être fournies à un abonné. Un serveur de distribution qui se trouve sur le même réseau local que le serveur de publication est appelé un serveur de distribution local. Les autres serveurs de distribution (peut-être même situés sur un continent différent) sont considérés comme des serveurs de distribution distants.

- Un abonné est une instance qui reçoit les publications.
- Un agent est un service qui exécute les opérations requises par l'un des rôles de réPLICATION.

Un serveur de distribution distant peut être utilisé pour réduire le trafic réseau entre le serveur de distribution et un certain nombre d'abonnés, en rapprochant le serveur de distribution de ses abonnés. Dans un abonnement par envoi de données, le serveur envoie les données à l'abonné. Pour un abonnement par extraction, l'abonné extrait les données sur le serveur de distribution.

RéPLICATION DE CAPTURE INSTANTANÉE

- Forme de réPLICATION la plus simple
- Toutes les données publiées copiées chaque fois
- Cadre d'utilisation
 - Les données ne changent pas souvent
 - Les données obsolètes chez l'abonné sont acceptables
 - Le volume de données est relativement faible
 - D'importantes modifications se produisent sur de courtes périodes

POINTS CLÉS

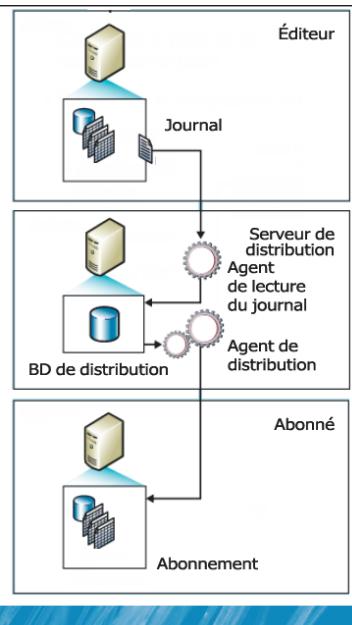
La capture instantanée est la forme la plus simple de réPLICATION. Régulièrement, l'agent de capture instantanée crée une copie des données publiées. La copie complète des données est alors utilisée pour remplacer les données chez l'abonné.

La réPLICATION de capture instantanée est utile dans plusieurs situations :

- quand il est acceptable d'autoriser des données obsolètes chez l'abonné. Tous les systèmes ne permettent pas cela ;
- si les données ne changent pas souvent, il est plus acceptable qu'il y en ait de périmées chez l'abonné ;
- si le volume de données est très petit, il est plus simple de copier toutes les données que de déterminer quelles données il faut copier ;
- si un très grand nombre de modifications ont été apportées aux données, il peut être moins onéreux de recopier toutes les données que de copier les modifications, car le volume de celles-ci peut être plus grand que le volume des données sources ;
- le serveur de publication n'est pas une base de données SQL Server. (La publication à partir d'une base de données Oracle est prise en charge).

RéPLICATION transactionnelle

- Les modifications sont appliquées aux abonnés rapidement
- Activités impliquées
 - La capture instantanée initiale des données est effectuée
 - Les modifications apportées aux données (et au schéma) sont remises à l'abonné lorsqu'elles se produisent
- Les modifications sont appliquées à l'abonné dans l'ordre d'apparition
 - Également dans les mêmes limites des transactions où elles se produisent sur le serveur de publication



Points clés

La réPLICATION transactionnelle est utilisée pour envoyer des modifications aux abonnés rapidement après que les modifications sont intervenues sur les serveurs de publication. La réPLICATION transactionnelle est initiée d'abord en créant une capture instantanée des données, puis en appliquant cette capture à l'abonné. Il y a plusieurs options disponibles pour créer cette copie des données, notamment par la réalisation d'une sauvegarde de base de données et de sa restauration. Les modifications sont appliquées dans le même ordre et dans les mêmes limites transactionnelles qu'elles sont intervenues sur le serveur de publication.

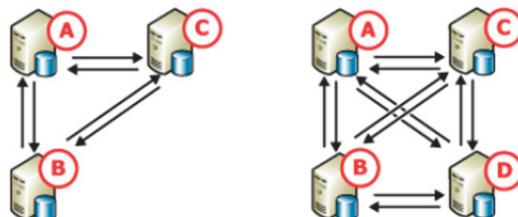
La réPLICATION transactionnelle doit, de manière générale, être considérée comme une réPLICATION unidirectionnelle. En général, les modifications effectuées au niveau de l'abonné ne sont pas renvoyées au serveur de publication ; néanmoins certaines options de configuration autorisent les mises à jour chez l'abonné.

La réPLICATION transactionnelle est utile dans les situations suivantes :

- les modifications incrémentielles des données doivent être appliquées à l'abonné rapidement après leur survenue ;
- l'application requiert que le délai entre une modification et son application soit court. Notez que ce délai est configurable et peut aller de quasi-instantané à plusieurs heures ;
- l'abonné doit voir toute modification apportée aux données. (Cela permet à l'abonné d'entreprendre des actions sur chaque modification, telles que des déclencheurs) ;
- il se déroule au niveau du serveur de publication une activité de modification des données en grand volume ;
- Le serveur de publication n'est pas une base de données SQL Server. (La publication à partir d'une base de données Oracle est prise en charge).

RéPLICATION TRANSACTIONNELLE D'ÉGAL À ÉGAL

- Option configurable dans la réPLICATION TRANSACTIONNELLE
- Généralement utilisée avec des serveurs autonomes
 - Les données partitionnées de manière appropriée sont essentielles
 - Des conflits peuvent être détectés, mais doivent être évités



Points clés

« Autoriser les abonnés d'égal à égal » est une option qui peut être configurée sur les publications de réPLICATION TRANSACTIONNELLE. Cela peut permettre de faire monter en charge une topologie de réPLICATION qui permette les opérations de lecture sur n'importe quel nœud.

Les lectures et les écritures peuvent se produire sur tous les nœuds, mais pour prévenir les conflits, il convient d'appliquer un schéma de partitionnement des données approprié. Les opérations d'écriture sur toutes les lignes individuelles se font toujours sur un nœud unique. Quand il est possible de détecter des conflits, cela cause l'échec de la réPLICATION ou la conservation des modifications. Par conséquent, l'aspect le plus important de la réPLICATION TRANSACTIONNELLE D'ÉGAL À ÉGAL réside dans la conception du partitionnement des données gérées par chaque serveur afin d'éviter tout risque de conflit.

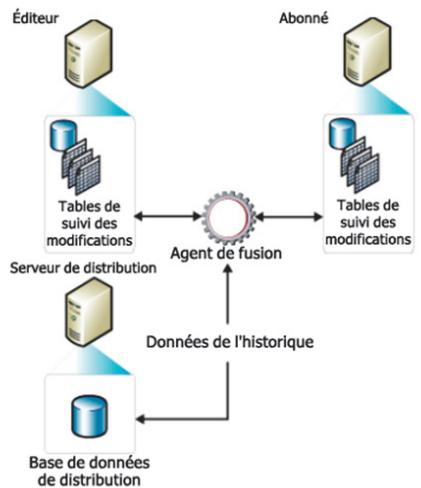
Comme exemple classique, une société peut avoir des serveurs à Londres, à Singapour et à Mumbai. Chaque ligne d'une table indiquerait l'origine des données. Toutes les lignes sont répliquées sur chaque serveur mais une seule ligne n'est jamais modifiée en plus d'un emplacement.

La réPLICATION TRANSACTIONNELLE D'ÉGAL À ÉGAL est utile dans les scénarios suivants :

- quand il existe un partitionnement naturel des données ;
- quand le partitionnement des données correspond à la topologie des serveurs ;
- quand des conflits sont peu susceptibles de survenir, comme par exemple quand chaque ligne de données dans une table appartient à un serveur unique.

RéPLICATION de fusion

- Utilisée en général pour les applications client-serveur simples
- Prend en charge les nœuds de serveur non SQL
- Non cohérent d'un point de vue transactionnel
- Les performances sont affectées avec
 - Volumes importants de données
 - Nombre élevé d'abonnés



Points clés

La réPLICATION de fusion est généralement utilisée dans les applications client-serveur simples.

La réPLICATION de fusion fonctionne par l'application d'un jeu de déclencheurs aux tables sources et par l'utilisation des déclencheurs pour insérer des données dans les tables de suivi des modifications à chaque modification des données. L'agent de fusion sert à copier les modifications sur les différents abonnés lors de chaque synchronisation.

La réPLICATION de fusion n'est pas transactionnellement cohérente mais prend en charge les nœuds non-SQL Server dans le cadre de la topologie de réPLICATION. En général, les performances de la réPLICATION de fusion pâtissent de l'augmentation du volume des données ou du nombre d'abonnés.

La réPLICATION de fusion est utile dans les situations suivantes :

- plusieurs abonnés peuvent mettre à jour leurs données en même temps. La réPLICATION de fusion fournit la détection de conflit et la correction de conflit ;
- les abonnés doivent modifier leurs données lorsqu'ils sont déconnectés ;
- des conflits surviennent lors de la modification de données chez plusieurs abonnés ;
- les applications ne dépendent pas de la cohérence transactionnelle des tables.

Choix parmi les options de réPLICATION

Aspect	Capture instantanée	Transactionnelle	Égal à égal	Fusion
Transactionnelle	Non	Oui	Oui	Non
Complexité	Faible	Moyenne	Élevée	Moyenne
Les abonnés peuvent mettre à jour	Non	Avec options limitées	Avec données partitionnées	Oui
Détection de conflit	Non	Non	Oui	Oui
Résolution des conflits	Non	Non	Non	Oui
Performances	Moyennes	Élevées	Élevées	Faibles

Points clés

Le tableau ci-dessus fournit une comparaison d'un certain nombre d'aspects de chaque option de réPLICATION, tels que décrits dans les quatre rubriques qui précédent.

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

Atelier pratique 11 : Autorisation de l'accès des utilisateurs aux ressources

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_11_PRJ\22462A_11_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 : Attribuer des autorisations au niveau du schéma

► **Tâche 1 : Examiner la configuration de sécurité requise qui a été mise à jour à partir du module précédent**

1. Prenez connaissance des spécifications de sécurité fournies dans la documentation.
2. Déterminez les autorisations qui doivent être attribuées au niveau du schéma.
(Remarque : un exemple de solution est affiché dans la tâche 2).

► **Tâche 2 : Attribuer les autorisations 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 : Attribuer des autorisations au niveau de l'objet

► Tâche 1 : Examiner la configuration requise pour la sécurité

1. Prenez connaissance des spécifications de sécurité fournies dans la documentation.
2. Déterminez les autorisations qui doivent être attribuées au niveau de l'objet.
(Remarque : un exemple de solution est affiché dans la tâche 2).

► Tâche 2 : Attribuer les autorisations requises

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 difficile 3 : Tester les autorisations (si le temps le permet)

► Tâche 1 : Concevoir et exécuter un test

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **71 - Lab Exercise 3a.sql** pour l'ouvrir.
2. Examinez le script T-SQL.
3. Dans la **barre d'outils**, cliquez sur **Exécuter**.
4. Dans l'Explorateur de solutions, double-cliquez sur le fichier **72 - Lab Exercise 3b.sql** pour l'ouvrir.
5. Examinez le script T-SQL.
6. Dans la **barre d'outils**, cliquez sur **Exécuter**.



Remarque Une erreur est retournée lorsque April.Reagan n'a pas l'autorisation de sélectionner les lignes de la table DirectMarketing.Competitor.

Module 12 : Audit des environnements SQL Server

Atelier pratique 12 : Audit des environnements 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, sélectionnez le projet **D:\22462A_Labs\22462A_12_PRJ\22462A_12_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 configuration d'audit et créer l'audit

► Tâche 1 : Examiner la configuration requise

- Prenez connaissance des éléments requis fournis dans la documentation associée à l'exercice, en tenant compte des spécifications liées aux audits de serveur.

► Tâche 2 : Créer l'audit du serveur

- Déterminez la configuration de l'audit du serveur requis.
- À l'aide de Windows Explorer, créez un dossier **C:\Audit**.
- À l'aide de Windows Explorer, créez un dossier **C:\Audit\AuditLog**.
- Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\administrateur)** et développez **Sécurité**.
- Cliquez avec le bouton droit sur **Audits**, puis cliquez sur **Nouvel audit**.
- Dans la fenêtre Créez un audit, tapez **Audit de la conformité de Proseware** dans la zone de texte **Nom de l'audit**.
- Dans la zone de texte **Délai de file d'attente (en millisecondes)**, tapez **2000**.
- Pour **En cas de défaillance du journal d'audit**, sélectionnez l'option **Arrêter le serveur**.
- Dans la zone de texte **Chemin d'accès au fichier**, tapez **C:\Audit\AuditLog**.
- Désactivez la case à cocher **Illimité** pour **Taille de fichier maximale**.
- Dans la zone de liste déroulante **Taille de fichier maximale**, tapez **1**.
- Cliquez sur le bouton d'option **Go** pour **Taille de fichier maximale**, puis cliquez sur **OK**.
- Dans l'Explorateur d'objets, développez **Audits**, cliquez avec le bouton droit sur **Audit de la conformité de Proseware**, puis cliquez sur **Activer l'audit**.
- Dans la fenêtre Activer l'audit, cliquez sur **Fermer**.

Exercice 2 : Créer les spécifications de l'audit du serveur

► Tâche 1 : Examiner la configuration requise

- Prenez connaissance des éléments requis fournis dans la documentation associée à l'exercice, en tenant compte des spécifications de l'audit du serveur.

► Tâche 2 : Créer les spécifications de l'audit du serveur

- Déterminez la configuration des spécifications requises de l'audit du serveur.
- Dans l'Explorateur d'objets, cliquez avec le bouton droit **Spécifications de l'audit du serveur**, puis cliquez sur **Nouvelle spécification de l'audit du serveur**.
- Dans la fenêtre Créez la spécification de l'audit du serveur, tapez **Spécification de l'audit du serveur de conformité Proseware** dans la zone de texte **Nom**.
- Dans la liste déroulante **Audit**, sélectionnez **Audit de la conformité de Proseware**.
- Dans la liste **Actions**, à la ligne 1, cliquez sur le menu déroulant **Type d'action de l'audit**, puis cliquez sur **FAILED_LOGIN_GROUP**.
- Dans la liste **Actions**, à la ligne 2, cliquez sur le menu déroulant **Type d'action de l'audit**, puis cliquez sur **SERVER_PRINCIPAL_CHANGE_GROUP**.
- Dans la liste **Actions**, à la ligne 3, cliquez sur le menu déroulant **Type d'action de l'audit**, puis cliquez sur **SERVER_ROLE_MEMBER_CHANGE_GROUP**.
- Dans la liste **Actions**, à la ligne 4, cliquez sur le menu déroulant **Type d'action de l'audit**, cliquez sur **LOGIN_CHANGE_PASSWORD_GROUP**, puis cliquez sur **OK**.
- Dans l'Explorateur d'objets, développez **Spécifications de l'audit du serveur**, cliquez avec le bouton droit sur **Spécification de l'audit du serveur de conformité de Proseware**, puis cliquez sur **Activer la spécification de l'audit du serveur**.
- Dans la fenêtre Activer la spécification de l'audit du serveur, cliquez sur **Fermer**.

Exercice 3 : Créer les spécifications de l'audit de la base de données

► Tâche 1 : Examiner la configuration requise

- Prenez connaissance des éléments requis fournis dans la documentation associée à l'exercice, en tenant compte des spécifications de l'audit de la base de données.

► Tâche 2 : Créer les spécifications de l'audit de la base de données

- Déterminez la configuration des spécifications requises de l'audit de la base de données.
- Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\administrateur)**, développez **Bases de données**, développez la base de données **MarketDev**, développez **Sécurité**, puis développez **Spécifications de l'audit de la base de données**.
- Cliquez avec le bouton droit sur **Spécifications de l'audit de la base de données**, puis cliquez sur **Nouvelles spécifications de l'audit de la base de données**.
- Dans la fenêtre Créez la spécification de l'audit de la base de données, tapez **Spécification de l'audit de MarketDev de conformité Proseware** dans la zone de texte **Nom**.
- Dans la liste déroulante **Audit**, sélectionnez **Audit de la conformité de Proseware**.
- Dans la liste **Actions**, à la ligne 1, cliquez sur le menu déroulant **Type d'action de l'audit**, puis cliquez sur **BACKUP_RESTORE_GROUP**.
- Dans la liste **Actions**, à la ligne 2, cliquez sur le menu déroulant **Type d'action de l'audit**, puis cliquez sur **DATABASE_OWNERSHIP_CHANGE_GROUP**.
- Dans la liste **Actions**, à la ligne 3, cliquez sur le menu déroulant **Type d'action de l'audit**, puis cliquez sur **DATABASE_PERMISSION_CHANGE_GROUP**.
- Dans la liste **Actions**, à la ligne 4, cliquez sur le menu déroulant **Type d'action de l'audit**, puis cliquez sur **DATABASE_PRINCIPAL_CHANGE_GROUP**.
- Dans la liste **Actions**, à la ligne 5, cliquez sur le menu déroulant **Type d'action de l'audit**, puis cliquez sur **DATABASE_ROLE_MEMBER_CHANGE_GROUP**.
- Dans la liste **Actions**, à la ligne 6, cliquez sur le menu déroulant **Type d'action de l'audit**, puis cliquez sur **EXECUTE**.
- À la ligne 6, cliquez sur la liste déroulante **Classe d'objets**, puis cliquez sur **OBJECT**.
- À la ligne 6, cliquez sur les **points de suspension** dans la colonne **Nom de l'objet**.
- Dans la fenêtre Sélectionner des objets, cliquez sur **Parcourir**.
- Dans le volet **Objets correspondants**, activez la case à cocher en regard de **[Marketing].[MoveCampaignBalance]**, puis cliquez sur **OK**.
- Dans la fenêtre Sélectionner des objets, cliquez sur **OK**.
- À la ligne 6, cliquez sur les **points de suspension** dans la colonne **Nom du principal**.
- Dans la fenêtre Sélectionner des objets, cliquez sur **Parcourir**.
- Dans le volet **Objets correspondants**, activez la case à cocher en regard de **[public]**, puis cliquez sur **OK**.

20. Dans la fenêtre Sélectionner des objets, cliquez sur **OK**.
21. Dans la liste **Actions**, à la ligne 7, cliquez sur le menu déroulant **Type d'action de l'audit**, puis cliquez sur **UPDATE**.
22. À la ligne 7, cliquez sur la liste déroulante **Classe d'objets**, puis cliquez sur **OBJECT**.
23. À la ligne 7, cliquez sur les **points de suspension** dans la colonne **Nom de l'objet**.
24. Dans la fenêtre Sélectionner des objets, cliquez sur **Parcourir**.
25. Dans le volet **Objets correspondants**, activez la case à cocher en regard de **[Marketing].[CampaignBalance]**, puis cliquez sur **OK**.
26. Dans la fenêtre Sélectionner des objets, cliquez sur **OK**.
27. À la ligne 7, cliquez sur les points de suspension dans la colonne **Nom du principal**.
28. Dans la fenêtre Sélectionner des objets, cliquez sur **Parcourir**.
29. Dans le volet **Objets correspondants**, activez la case à cocher en regard de **[public]**, puis cliquez sur **OK**.
30. Dans la fenêtre Sélectionner des objets, cliquez sur **OK**.
31. Dans la fenêtre Créer la spécification de l'audit de la base de données, cliquez sur **OK**.
32. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur **Spécification de l'audit de MarketDev de conformité Proseware**, puis cliquez sur **Activer la spécification de l'audit de la base de données**.
33. Dans la fenêtre Activer la spécification de l'audit de la base de données, cliquez sur **Fermer**.

Exercice difficile 4 : Tester les fonctionnalités d'audit (si le temps le permet)

► Tâche 1 : Exécuter le script de charge de travail

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **81 - Lab Exercise 4a.sql** pour l'ouvrir.
2. Examinez le script T-SQL. Notez toutes les actions que vous envisageriez de faire auditer.
Dans la tâche 2, vous verrez la liste réelle des actions auditées.
3. Dans la **barre d'outils**, cliquez sur **Exécuter**.

► Tâche 2 : Examiner les informations d'audit capturées

1. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\administrateur)**, puis **Sécurité**, puis **Audits**.
2. Cliquez avec le bouton droit sur **Audit de la conformité de Proseware**, puis cliquez sur **Afficher les journaux d'audit**.
3. Dans la fenêtre Visionneuse du fichier journal – Proseware, examinez les événements capturés et comparez-les à votre liste d'événements attendues de la tâche 1.

 **Remarque** Veillez à parcourir la liste des événements vers la droite pour afficher toutes les colonnes disponibles.

4. Dans la fenêtre Visionneuse du fichier journal – Proseware, cliquez sur **Fermer**.
5. Dans l'Explorateur de solutions, double-cliquez sur le fichier **82 - Lab Exercise 4b.sql** pour l'ouvrir.
6. Examinez le script T-SQL.
7. Dans la **barre d'outils**, cliquez sur **Exécuter**. Une liste d'événements audités est retournée.

Module 13 : Automatisation de la gestion de SQL Server 2012

Atelier pratique 13 : Automatisation de la gestion de 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, sélectionnez le projet **D:\22462A_Labs\22462A_13_PRJ\22462A_13_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 : Créer un travail d'extraction de données

► Tâche 1 : Créer le travail requis

1. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\administrateur)**, développez **Agent SQL Server**, cliquez avec le bouton droit sur **Travaux**, puis cliquez sur **Nouveau travail**.
2. Dans la fenêtre Nouveau travail, dans la zone de texte **Nom**, tapez **Extract Uncontacted Prospects**.
3. Dans le volet **Sélectionner une page**, cliquez sur **Étapes**, puis cliquez sur **Nouveau**.
4. Dans la zone de texte **Nom de l'étape**, tapez **Exécuter le package d'extraction de prospects**.
5. Dans la liste déroulante **Type**, sélectionnez **Package SQL Server Integration Services**.
6. Dans l'onglet **Général**, dans le menu déroulant **Source du package**, sélectionnez **SQL Server**.
7. Dans la zone de texte **Serveur**, tapez **Proseware**, puis cliquez sur les **points de suspension** en regard de la zone de texte **Package**.
8. Dans la fenêtre Sélectionner un package SSIS, cliquez sur **Weekly Extract of Prospects to Contact**, puis cliquez sur **OK**.
9. Dans la fenêtre Nouvelle étape du travail, cliquez sur **OK**.
10. Dans la fenêtre Nouveau travail, cliquez sur **OK**.

► Tâche 2 : Vérifier que le travail s'exécute sans erreur

1. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\administrateur)**, développez **Agent SQL Server**, puis développez **Travaux**.
2. Cliquez avec le bouton droit sur **Travaux**, puis cliquez sur **Actualiser**.
3. Cliquez avec le bouton droit sur **Extract Uncontacted Prospects**, puis cliquez sur **Démarrer le travail à l'étape**.
4. Dans la fenêtre Démarrer les travaux - Proseware, vérifiez que le travail s'effectue correctement, puis cliquez sur **Fermer**.

Exercice 2 : Planifier le travail d'extraction de données

► Tâche 1 : Planifier le travail d'extraction de données

1. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\administrateur)**, développez **Agent SQL Server**, puis développez **Travaux**.
2. Cliquez avec le bouton droit sur **Travaux**, puis cliquez sur **Actualiser**.
3. Cliquez avec le bouton droit sur **Extract Uncontacted Prospects**, puis cliquez sur **Propriétés**.
4. Dans le volet **Sélectionner une page**, cliquez sur **Planifications**, puis cliquez sur **Nouvelle**.
5. Dans la fenêtre Nouvelle planification du travail, dans la zone de texte **Nom**, tapez **Chaque lundi matin à 08 h 30**.
6. Activez la case à cocher **Lundi** ; désactivez la case à cocher **Dimanche**.
7. Dans la zone de texte **Une fois le**, modifiez l'heure à **08:30:00** et appuyez sur la touche TAB. Notez le contenu de la zone de texte **Description**, puis cliquez sur **OK**.
8. Dans la fenêtre Propriétés du travail – Extract Uncontacted Prospects, cliquez sur **Nouvelle**.
9. Dans la fenêtre Nouvelle planification du travail, dans la zone de texte **Nom**, tapez **Chaque mardi soir à 18 h 30**.
10. Activez la case à cocher **Mardi** ; désactivez la case à cocher **Dimanche**.
11. Dans la zone de texte **Une fois le**, modifiez l'heure à **18:30:00** et appuyez sur la touche TAB. Notez le contenu de la zone de texte **Description**, puis cliquez sur **OK**.
12. Dans la fenêtre Propriétés du travail – Extract Uncontacted Prospects, cliquez sur **OK**.

Exercice difficile 3 : Résoudre les problèmes liés à un travail défaillant (si le temps le permet)

► Tâche 1 : Résoudre les problèmes liés au travail défaillant

1. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\administrateur)**, développez **Agent SQL Server**, puis développez **Travaux**.
2. Cliquez avec le bouton droit sur **Travaux**, puis cliquez sur **Actualiser**.
3. Cliquez avec le bouton droit sur **Extract Long Page Loads**, puis cliquez sur **Afficher l'historique**.
4. Dans la fenêtre Visionneuse du fichier journal – Proseware, cliquez sur le signe **plus** d'une ligne qui comporte une exécution manquante pour développer les étapes de travail. Notez que l'ID de l'étape **2** a échoué.
5. Cliquez sur la ligne de l'ID de l'étape **2** et parcourez le volet **Détails de la ligne sélectionnée** pour rechercher l'erreur. Notez que l'erreur est provoquée par une référence à un nom d'objet valide **Marketing.RecentLongPageLoads**.
6. Cliquez sur **Fermer** pour fermer la fenêtre Visionneuse du fichier journal – Proseware.
7. Cliquez avec le bouton droit sur **Extract Long Page Loads**, puis cliquez sur **Propriétés**.
8. Dans le volet **Sélectionner une page**, cliquez sur **Étapes**.
9. Dans la liste **Propriétés du travail - Extract Long Page Loads**, cliquez sur la ligne de l'Étape 2, puis cliquez sur **Modifier**.
10. Examinez la commande exécutée et notez que **Marketing.RecentLongPageLoads** semble être une table ou une vue dans laquelle les lignes du journal Web sont insérées.
11. Dans l'Explorateur d'objets, développez **Bases de données**, **MarketDev**, puis **Tables**. Notez que le nom de la table doit être **Marketing.RecentLongPageLoad**.
12. Dans la fenêtre Propriétés de l'étape du travail – Copier les longues charges de page récentes, modifiez le nom de la table dans la zone de texte **Commande** de **Marketing.RecentLongPageLoads** en **Marketing.RecentLongPageLoad**, puis cliquez sur **OK**.
13. Dans la fenêtre Propriétés du travail – Extract Long Page Loads, cliquez sur **OK**.
14. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur le travail **Extract Long Page Loads**, puis cliquez sur **Démarrer le travail à l'étape**.
15. Dans Démarrer les travaux- Proseware, cliquez sur **Démarrer**.
16. Dans Démarrer les travaux- Proseware, vérifiez que le travail a été exécuté avec succès, puis cliquez sur **Fermer**.
17. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur **Extract Long Page Loads**, puis cliquez sur **Propriétés**.
18. Dans le volet **Sélectionner une page**, cliquez sur **Planifications** et notez la différence entre le nom de la planification et la description de la planification, puis cliquez sur **Modifier**.
19. Dans la zone de liste déroulante **Périodicité**, cliquez sur **Hebdomadaire**.

20. Activez la case à cocher **Lundi** et vérifiez les paramètres restants, puis cliquez sur **OK**.
21. Dans la liste **Planification**, notez que le nom de la planification est liée à la description de la planification.
22. Dans la fenêtre Propriétés du travail – Extract Long Page Loads, cliquez sur **OK**.

Module 14 : Configuration de la sécurité pour SQL Server Agent

Atelier pratique 14 : Configuration de la sécurité pour SQL Server Agent

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_14_PRJ\22462A_14_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 : Résoudre la défaillance d'exécution du travail

► Tâche 1 : Résoudre la défaillance d'exécution du travail

1. Examinez les actions effectuées précédemment par Terry Adams qui sont détaillées dans la documentation de l'exercice.
2. Dans l'Explorateur d'objets, dans le serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\administrateur)**, développez **Agent SQL Server**, développez **Travaux**, cliquez avec le bouton droit sur **Extract Uncontacted Prospects**, puis cliquez sur **Afficher l'historique**.
3. Dans la fenêtre Visionneuse du fichier journal – Proseware, cliquez sur le signe **plus** en regard d'une ligne d'échec pour développer les étapes de travail.
4. Cliquez sur la ligne de l'ID de l'étape **1**. Dans le volet **Détails de la ligne sélectionnée**, faites défiler le volet jusqu'à ce que vous localisiez l'erreur.



Remarque L'erreur indique que les non-administrateurs système se sont vus refuser l'autorisation d'exécuter les étapes de travail d'exécution DTS sans compte proxy. L'étape a échoué.

5. Vous avez déterminé que le compte proxy est nécessaire pour l'étape de travail. Fermez la fenêtre Visionneuse du fichier journal - Proseware.

Exercice 2 : Résoudre le problème de sécurité

► Tâche 1 : Créer et affecter un compte proxy

1. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\administrateur)**, développez **Sécurité**, puis **Informations d'identification**.
2. Cliquez avec le bouton droit sur **Informations d'identification** et cliquez sur **Nouvelles informations d'identification**.
3. Dans la fenêtre Nouvelles informations d'identification, dans la zone de texte **Nom d'identification**, tapez **ExtractIdentity**.
4. Dans la zone de texte **Identité**, tapez **MIA-SQL1\ExtractUser**.
5. Dans les zones de texte **Mot de passe** et **Confirmer le mot de passe**, tapez **Pa\$\$w0rd**, puis cliquez sur **OK**.
6. Dans l'Explorateur d'objets, développez **Agent SQL Server**, cliquez avec le bouton droit sur **Proxies** et cliquez sur **Nouveau proxy**.
7. Dans la fenêtre Nouveau compte proxy, dans la zone **Nom de proxy**, tapez **ExtractionProxy**.
8. Dans la zone de texte **Nom d'identification**, tapez **ExtractIdentity**.
9. Dans la liste **Actif pour les sous-systèmes suivants**, activez la case à cocher en regard de **Package SQL Server Integration Services**.
10. Dans le volet **Sélectionner une page**, cliquez sur **Principaux**, puis cliquez sur **Ajouter**.
11. Dans la liste **Principaux disponibles**, activez la case à cocher en regard de la connexion **PromoteApp**, puis cliquez sur **OK**.
12. Dans une fenêtre Nouveau compte proxy, cliquez sur **OK**.
13. Dans l'Explorateur d'objets, développez **Travaux**, cliquez avec le bouton droit sur **Extract Uncontacted Prospects**, puis cliquez sur **Propriétés**.
14. Dans la fenêtre Propriétés du travail - Extract Uncontacted Prospects, dans le volet **Sélectionner une page**, cliquez sur **Étapes**, puis cliquez sur **Modifier**.
15. Dans la liste déroulante **Exécuter en tant que**, cliquez sur **ExtractionProxy**, puis cliquez sur **OK**.
16. Dans la fenêtre Propriétés du travail – Extract Uncontacted Prospects, cliquez sur **OK**.

► Tâche 2 : Effectuer des tests pour voir si tous les problèmes ont été résolus

1. Dans l'Explorateur d'objets, dans le serveur **Proseware**, développez **Agent SQL Server**, puis développez **Travaux**. Cliquez avec le bouton droit sur le travail **Extract Uncontacted Prospects**, puis cliquez sur **Démarrer le travail à l'étape**.
2. Dans la fenêtre Démarrer les travaux - Proseware, notez que le travail se solde toujours par un échec et cliquez sur **Fermer**.
3. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur le travail **Extract Uncontacted Prospects**, puis cliquez sur **Afficher l'historique**.
4. Dans la fenêtre Visionneuse du fichier journal - Proseware, cliquez sur le signe **plus** en regard de la première entrée de la liste pour développer les étapes de travail.

5. Cliquez sur la ligne de l'ID de l'étape **1**, dans le volet **Détails de la ligne sélectionnée**, faites défiler l'écran vers le bas pour rechercher l'erreur.

 **Remarque** Une instruction SQL est désormais en échec à cause d'un problème de connexion.

6. Vous avez résolu le problème d'origine. Si vous disposez de suffisamment de temps, passez à l'exercice 3 pour résoudre le problème restant.

Exercice difficile 3 : Dépannage plus avancé (si le temps le permet)

► Tâche 1 : Dépannage plus avancé

1. Dans la fenêtre Visionneuse du fichier journal - Proseware, dans le volet **Détails de la ligne sélectionnée**, lisez le détail du message d'erreur, déterminez la cause de l'erreur et fermez la fenêtre.

Remarque L'erreur la plus importante est l'échec de la connexion de l'utilisateur « MIA-SQL1\ExtractUser ». Bien que les informations d'identification Windows sont requises pour le travail SSIS en vue d'accéder au système de fichiers et écrire dans le fichier extrait, les informations d'identification doivent également permettre de se connecter à SQL Server pour récupérer les données de la table Marketing.Prospects. Vous devez créer une connexion pour l'utilisateur Windows, créer un utilisateur de base de données pour la connexion, puis affecter une autorisation SELECT sur la table Marketing.Prospects applicable aux informations d'identification.

2. Dans l'Explorateur de solutions, double-cliquez sur le fichier **71 - Lab Exercise 3.sql** pour l'ouvrir.
3. Examinez le script T-SQL.
4. Dans la **barre d'outils**, cliquez sur **Exécuter**.
5. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur le travail **Extract Uncontacted Prospects**, puis cliquez sur **Démarrer le travail à l'étape**.
7. Dans la fenêtre Démarrer les travaux - Proseware, notez que le travail se solde toujours par un échec et cliquez sur **Fermer**.
8. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur le travail **Extract Uncontacted Prospects**, puis cliquez sur **Afficher l'historique**.
9. Dans la fenêtre Visionneuse du fichier journal - Proseware, cliquez sur le signe **plus** en regard de la première entrée de la liste pour développer les étapes de travail.
10. Cliquez sur la ligne de l'ID de l'étape **1**, dans le volet **Détails de la ligne sélectionnée**, faites défiler l'écran vers le bas pour rechercher l'erreur.

Remarque ExtractUser ne dispose pas d'autorisations d'accès au dossier D:\MKTG.

11. Dans l'Explorateur Windows, naviguez jusqu'au dossier **D:\MKTG**, cliquez avec le bouton droit sur le dossier **MKTG**, puis cliquez sur **Propriétés**.
12. Dans la fenêtre Propriétés de : MKTG, cliquez sur l'onglet **Sécurité**, puis sur **Modifier**.
13. Dans la fenêtre Autorisations pour MKTG, cliquez sur **Ajouter**.
14. Dans la boîte de dialogue Sélectionner les utilisateurs, des ordinateurs, des comptes de service ou des groupes, dans la zone de texte **Entrez les noms des objets à sélectionner (exemples)**, tapez **MIA-SQL1\ExtractUser**, puis cliquez sur **OK**.
15. Dans la fenêtre Autorisations pour MKTG, activez la case **Autoriser** de la ligne **Modification**, puis cliquez sur **OK**.

16. Dans la fenêtre Propriétés de : MKTG, cliquez sur **OK**.
17. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur le travail **Extract Uncontacted Prospects**, puis cliquez sur **Démarrer le travail à l'étape**.
18. Dans la fenêtre Démarrer les travaux - Proseware, notez que le travail s'est maintenant terminé correctement et cliquez sur **Fermer**.

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

Atelier pratique 15 : Surveillance des travaux de l'agent SQL avec alertes et notifications

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_15_PRJ\22462A_15_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 : Configurer la messagerie de base de données

► Tâche 1 : Configurer la messagerie de base de données

1. Vérifiez les paramètres de configuration de la messagerie de base de données dans la documentation associée à l'exercice.
2. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\administrateur)**, développez **Gestion**, cliquez avec le bouton droit sur **Messagerie de base de données**, puis cliquez sur **Configurer la messagerie de base de données**.
3. Dans la fenêtre Bienvenue dans l'Assistant Configuration de la messagerie de base de données, cliquez sur **Suivant**.
4. Dans la fenêtre Sélectionner une tâche de configuration, cliquez sur **Suivant**.
5. Dans la fenêtre Microsoft SQL Server Management Studio, cliquez sur **Oui**.
6. Dans la fenêtre Nouveau profil, dans la zone de texte **Nom de profil**, tapez **Profil de SQL Server Agent de Proseware**, puis cliquez sur **Ajouter**.
7. Dans la fenêtre Nouveau compte de messagerie de base de données, dans la zone de texte **Nom du compte**, tapez **Administrateur Proseware**.
8. Dans la zone de texte **Adresse de messagerie**, tapez **prosewaresqladmin@adventureworks.com**.
9. Dans la zone de texte **Nom complet**, tapez **Administrateur SQL Server Proseware**.
10. Dans la zone de texte **Répondre au courrier**, tapez **prosewaresqladmin@adventureworks.com**.
11. Dans la zone de texte **Nom du serveur**, tapez **mailserver.adventureworks.com**, puis cliquez sur **OK**.
12. Dans la fenêtre Nouveau profil, cliquez sur **Ajouter**.
13. Dans la fenêtre Nouveau compte de messagerie de base de données, dans la zone de texte **Nom du compte**, tapez **Administrateur AdventureWorks**.
14. Dans la zone de texte **Adresse de messagerie**, tapez **adventureworkssqladmin@adventureworks.com**.
15. Dans la zone de texte **Nom complet**, tapez **Administrateur SQL Server AdventureWorks**.
16. Dans la zone de texte **Répondre au courrier**, tapez **adventureworkssqladmin@adventureworks.com**.
17. Dans la zone de texte **Nom du serveur**, tapez **mailserver.adventureworks.com**, puis cliquez sur **OK**.
18. Dans la fenêtre Nouveau profil, cliquez sur **Suivant**.
19. Dans la fenêtre Gérer la sécurité des profils, activez la case à cocher dans la colonne **Public** et remplacez la valeur actuelle de la colonne **Profil par défaut** par **Oui**.
20. Cliquez sur l'onglet **Profils privés**, puis dans la liste déroulante **Nom d'utilisateur**, sélectionnez **ADVENTUREWORKS\pwservice**.
21. Activez la case à cocher dans la colonne **Accès** et changez la valeur de la colonne **Profil par défaut** en appliquant **Oui**, puis cliquez sur **Suivant**.

22. Dans la fenêtre Configurer les paramètres du système, changez la **Taille de fichier maximale (octets)** en appliquant **4194304**, puis cliquez sur **Suivant**.

23. Dans la fenêtre Terminer l'Assistant, cliquez sur **Terminer**.

24. Dans la fenêtre **Configuration en cours**, cliquez sur **Fermer**.

► **Tâche 2 : Tester le fonctionnement de la messagerie de base de données**

1. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur **Messagerie de base de données**, puis cliquez sur **Envoyer un message électronique de test**.
2. Dans la fenêtre Envoyer un message électronique de test à partir de MIA-SQL1\MKTG, dans la zone de texte **A**, tapez **prosewaresqladmin@adventureworks.com**, puis cliquez sur **Envoyer un message électronique de test**.
3. Dans la fenêtre Message électronique de test de la messagerie de base de données, notez la description et cliquez sur **OK**.
4. Dans l'Explorateur de solutions, double-cliquez sur le fichier **51 - Lab Exercise 1.sql** pour l'ouvrir.
5. Examinez le script T-SQL.
6. Dans la **barre d'outils**, cliquez sur **Exécuter**.



Remarque Un message électronique avec l'objet Test de la messagerie de base de données doit être retourné par la requête.

Exercice 2 : Implémenter des notifications

► Tâche 1 : Examiner la configuration requise

- Prenez connaissance des éléments requis fournis dans la documentation associée de l'exercice. En particulier, notez tous les opérateurs requis.

► Tâche 2 : Configurer les opérateurs requis

- Dans l'Explorateur d'objets, développez **Agent SQL Server**, cliquez avec le bouton droit sur **Opérateurs** et cliquez sur **Nouvel opérateur**.
- Dans la fenêtre Nouvel opérateur, dans la zone de texte **Nom**, tapez **Opérateur de prévention de défaillance du support informatique**.
- Dans **Nom de l'adresse de radiomessagerie**, tapez **itsupport.page@adventureworks.com**.
- Dans **Planification de la radiomessagerie active**, activez la case à cocher **Lundi, Mardi, Mercredi, Jeudi, Vendredi, Samedi et Dimanche**. Changez l'heure dans **Début de journée** en appliquant **12:00:00** et l'heure de **Fin de journée** en appliquant **23:59:59** pour chaque ligne, puis cliquez sur **OK**.
- Dans l'Explorateur d'objets, cliquez avec le bouton droit sur **Opérateurs** et cliquez sur **Nouvel opérateur**.
- Dans la fenêtre Nouvel opérateur, dans la zone de texte **Nom**, tapez **Jeff Hay**.
- Dans **Nom de l'adresse de radiomessagerie**, tapez **jeff.hay.page@adventureworks.com**.
- Dans **Planification de la radiomessagerie active**, activez la case à cocher **Lundi, Mardi, Mercredi, Jeudi, Vendredi, Samedi et Dimanche**. Changez l'heure dans **Début de journée** en appliquant **12:00:00** et l'heure de **Fin de journée** en appliquant **23:59:59** pour chaque ligne, puis cliquez sur **OK**.
- Dans l'Explorateur d'objets, cliquez avec le bouton droit sur **Opérateurs** et cliquez sur **Nouvel opérateur**.
- Dans la fenêtre Nouvel opérateur, dans la zone de texte **Nom**, tapez **Palle Petersen**.
- Dans la page **Nom de l'adresse de radiomessagerie**, tapez **palle.petersen.page@adventureworks.com**.
- Dans **Planification de la radiomessagerie active**, activez la case à cocher **Lundi, Mardi, Mercredi, Jeudi, Vendredi, Samedi et Dimanche**. Changez l'heure dans **Début de journée** en appliquant **12:00:00** et l'heure de **Fin de journée** en appliquant **23:59:59** pour chaque ligne, puis cliquez sur **OK**.

► Tâche 3 : Configurer la messagerie SQL Server Agent

- Dans l'Explorateur d'objets, cliquez avec le bouton droit sur **Agent SQL Server** et cliquez sur **Propriétés**.
- Dans le volet **Sélectionner une page**, cliquez sur **Système d'alerte**. Dans les options **Session de messagerie**, cliquez sur **Activer le profil de messagerie**.
- Dans la liste déroulante **Profil de la messagerie**, sélectionnez **Profil de SQL Server Agent de Proseware**.

4. Dans les options **Opérateur de prévention de défaillance**, activez la case à cocher **Activer l'opérateur de prévention de défaillance**.
 5. Dans la liste déroulante **Opérateur**, sélectionnez **Opérateur de prévention de défaillance du support informatique**.
 6. Dans **Notifier en utilisant des options**, activez la case **Radiomessagerie**, puis cliquez sur **OK**.
 7. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur **Agent SQL Server** et cliquez sur **Redémarrer**.
 8. Dans la fenêtre Microsoft SQL Server Management Studio, cliquez sur **Oui**.
- **Tâche 4 : Configurer et tester les notifications dans les travaux de SQL Server Agent**
1. Dans l'Explorateur d'objets, développez **Agent SQL Server**, développez **Travaux**, cliquez avec le bouton droit sur **Backup Log TestAlertDB**, puis cliquez sur **Propriétés**.
 2. Dans le volet **Sélectionner une page**, cliquez sur **Notifications**.
 3. Dans l'option **Actions à effectuer lorsque le travail se termine**, activez la case **Page**.
 4. Dans la liste déroulante **Page**, sélectionnez **Jeff Hay**.
 5. Dans la liste déroulante des actions **Page**, sélectionnez **Lorsque le travail est terminé**, puis cliquez sur **OK**.
 6. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur **Job that Fails**, puis cliquez sur **Propriétés**.
 7. Dans le volet **Sélectionner une page**, cliquez sur **Notifications**.
 8. Dans l'option **Actions à effectuer lorsque le travail se termine**, activez la case **Page**.
 9. Dans la liste déroulante **Page**, sélectionnez **Palle Petersen**.
 10. Dans la liste déroulante des actions **Page**, sélectionnez **Lors de l'échec du travail**, puis cliquez sur **OK**.
 11. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur **Job that Succeeds**, puis cliquez sur **Propriétés**.
 12. Dans le volet **Sélectionner une page**, cliquez sur **Notifications**.
 13. Dans l'option **Actions à effectuer lorsque le travail se termine**, activez la case **Page**.
 14. Dans la liste déroulante **Page**, sélectionnez **Palle Petersen**.
 15. Dans la liste déroulante des actions **Page**, sélectionnez **Lors de l'échec du travail**, puis cliquez sur **OK**.
 16. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur le **Backup Log TestAlertDB**, puis cliquez sur **Démarrer le travail à l'étape**.
 17. Dans la fenêtre Démarrer les travaux - Proseware, cliquez sur **Fermer**.
 18. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur **Job That Fails**, puis cliquez sur **Démarrer le travail à l'étape**.
 19. Dans la fenêtre Démarrer les travaux - Proseware, notez que le travail a échoué et cliquez sur **Fermer**.

20. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur **Job That Succeeds**, puis cliquez sur **Démarrer le travail à l'étape**.
21. Dans la fenêtre Démarrer les travaux - Proseware, cliquez sur **Fermer**.
22. Dans l'Explorateur de solutions, double-cliquez sur le fichier **51 - Lab Exercise 1.sql** pour l'ouvrir.
23. Examinez le script T-SQL.
24. Dans la **barre d'outils**, cliquez sur **Exécuter**.



Remarque Deux messages électroniques supplémentaires doivent maintenant apparaître dans la liste, un courrier électronique du travail de sauvegarde envoyé à Jeff Hay et un courrier électronique du travail ayant échoué envoyé à Palle Petersen. Aucun message ne doit être envoyé pour le travail appelé « Le travail a réussi ».

Exercice difficile 3 : Implémenter des alertes (si le temps le permet)

► Tâche 1 : Configurer et tester les alertes

1. Examinez la documentation associée pour prendre connaissance des spécifications d'alertes.
2. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur **Alertes** et cliquez sur **Nouvelle alerte**.
3. Dans la fenêtre Nouvelle alerte, dans la zone de texte **Nom**, tapez **Alerte gravité d'erreur 17**.
4. Dans la liste déroulante **Gravité**, cliquez sur **017 - Ressources insuffisantes**.
5. Dans le volet **Sélectionner une page**, cliquez sur **Réponse**.
6. Activez la case **Notifier les opérateurs**, activez toutes les cases à cocher de la colonne **Radiomessagerie**, puis cliquez sur **OK**.
7. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur **Alertes** et cliquez sur **Nouvelle alerte**.
8. Dans la fenêtre Nouvelle alerte, dans la zone de texte **Nom**, tapez **Alerte gravité d'erreur 18**.
9. Dans la liste déroulante **Gravité**, cliquez sur **018 - Erreur interne récupérable**.
10. Dans le volet **Sélectionner une page**, cliquez sur **Réponse**.
11. Activez la case **Notifier les opérateurs**, activez toutes les cases à cocher de la colonne **Radiomessagerie**, puis cliquez sur **OK**.
12. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur **Alertes** et cliquez sur **Nouvelle alerte**.
13. Dans la fenêtre Nouvelle alerte, dans la zone de texte **Nom**, tapez **Alerte journal des transactions plein**.
14. Cliquez sur l'option **Numéro d'erreur**, et dans la zone de texte **Numéro d'erreur**, tapez **9002**.
15. Dans le volet **Sélectionner une page**, cliquez sur **Réponse**.
16. Activez la case **Notifier les opérateurs**, activez toutes les cases à cocher de la colonne **Radiomessagerie**, puis cliquez sur **OK**.
17. Dans l'Explorateur de solutions, double-cliquez sur le fichier **71 - Lab Exercise 3.sql** pour l'ouvrir.
18. Examinez le script T-SQL.
19. Dans la **barre d'outils**, cliquez sur **Exécuter**.



Remarque L'exécution de ce script génère le message d'erreur 9002.

20. Dans l'Explorateur de solutions, double-cliquez sur le fichier **51 - Lab Exercise 1.sql** pour l'ouvrir.
21. Examinez le script T-SQL.
22. Dans la **barre d'outils**, cliquez sur **Exécuter**.



Remarque Des messages électroniques supplémentaires relatifs au système d'alerte de SQL Server doivent être répertoriés.

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

Atelier pratique 16 : Exécution de la maintenance régulière des bases 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_16_PRJ\22462A_16_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 : Contrôler l'intégrité de la base de données avec DBCC CHECKDB

► Tâche 1 : Vérifier la cohérence des bases de données sur l'instance Proseware

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **51 - Lab Exercise 1a.sql** pour l'ouvrir.
2. Examinez le script T-SQL.
3. Dans la **barre d'outils**, cliquez sur **Exécuter**.



Remarque Les erreurs sont retournées de la base de données CoreAdmin.

► Tâche 2 : Corriger les problèmes détectés

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **52 - Lab Exercise 1b.sql** pour l'ouvrir.
2. Examinez le script T-SQL.
3. Dans la **barre d'outils**, cliquez sur **Exécuter**.



Remarque Les erreurs sont retournées de la base de données CoreAdmin pendant ce processus.

Exercice 2 : Corriger la fragmentation des index

► Tâche 1 : Examiner la fragmentation des index de la base de données MarketDev pour déterminer les index à défragmenter et les index à reconstruire

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **61 - Lab Exercise 2a.sql** pour l'ouvrir.
2. Examinez le script T-SQL.
3. Dans la **barre d'outils**, cliquez sur **Exécuter**.

► Tâche 2 : Défragmenter les index comme déterminé

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **62 - Lab Exercise 2b.sql** pour l'ouvrir.
2. Examinez le script T-SQL.
3. Dans la **barre d'outils**, cliquez sur **Exécuter**.

► Tâche 3 : Reconstruire les index comme déterminé

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **63 - Lab Exercise 2c.sql** pour l'ouvrir.
2. Examinez le script T-SQL.
3. Dans la **barre d'outils**, cliquez sur **Exécuter**.

Exercice 3 : Créer un plan de maintenance de la base de données

► Tâche 1 : Créer un plan de maintenance de base de données requis

1. Dans l'Explorateur d'objets, développez **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\administrateur)**, développez **Gestion**, cliquez avec le bouton droit sur **Plans de maintenance**, puis cliquez sur **Assistant Plan de maintenance**.
2. Dans la fenêtre Assistant Plan de maintenance, cliquez sur **Suivant**.
3. Dans la fenêtre Sélectionner les propriétés de plan, dans la zone de texte **Nom**, tapez **Maintenance hebdomadaire Proseware**. Notez les options de planification disponibles et cliquez sur **Modifier**.
4. Dans la fenêtre Nouvelle planification du travail, dans la zone de texte **Nom**, tapez **Chaque dimanche à 18h00**.
5. Dans la zone de texte **Une fois le**, modifiez l'heure à **18:00:00** et cliquez sur **OK**.
6. Dans la fenêtre Sélectionner les propriétés de plan, cliquez sur **Suivant**.
7. Examinez les options disponibles et sélectionnez **Reconstruire l'index** et **Vérifier l'intégrité de la base de données**, puis cliquez sur **Suivant**.
8. Dans la fenêtre Sélectionner l'ordre des tâches de maintenance, cliquez sur **Suivant**.
9. Dans la fenêtre Définir la tâche Vérifier l'intégrité de la base de données, dans la liste déroulante **Bases de données**, sélectionnez **Toutes les bases de données**, puis cliquez sur **OK**, puis **Suivant**.
10. Dans la fenêtre Définir la tâche Reconstruire l'index, dans la liste déroulante **Bases de données**, sélectionnez la base de données **MarketDev**, puis cliquez sur **OK**.
11. Dans les **Options relatives à l'espace libre**, cliquez sur **Modifier l'espace disponible par page de :**
12. Dans la zone de texte **Modifier l'espace disponible par page de :**, tapez **10**.
13. Sélectionnez **Conserver l'index en ligne lors de la réindexation**, puis cliquez sur **Suivant**.
14. Dans la fenêtre Sélectionner des options de rapport, dans la zone de texte **Emplacement du dossier**, tapez **L:\MKTG**, puis cliquez sur **Suivant**.
15. Dans la fenêtre Terminer l'Assistant, cliquez sur **Terminer**.
16. Dans la fenêtre Progression de l'Assistant Plan de maintenance, cliquez sur **Fermer**.

Exercice difficile 4 : Étudier les performances du verrou de table (uniquement si le temps le permet)

► Tâche 1 : Exécuter DBCC CHECKDB avec des captures instantanées de base de données

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **81 - Lab Exercise 4a.sql** pour l'ouvrir.
2. Examinez le script T-SQL.
3. Dans la **barre d'outils**, cliquez sur **Exécuter**.



Remarque Notez la durée nécessaire pour l'exécution. Une durée classique serait de 42 secondes.

► Tâche 2 : Exécuter DBCC CHECKDB avec les verrous de table

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **82 - Lab Exercise 4b.sql** pour l'ouvrir.
2. Examinez le script T-SQL.
3. Dans la **barre d'outils**, cliquez sur **Exécuter**.



Remarque Notez la durée nécessaire pour l'exécution. Une durée classique serait de 7 secondes.

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

Atelier pratique 17 : Trace de l'accès à 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, ouvrez le projet **D:\22462A_Labs\22462A_17_PRJ\22462A_17_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 : Capturer une trace avec SQL Server Profiler

► Tâche 1 : Créer et démarrer une trace SQL Server Profiler appropriée

1. Dans 22462A_17_PRJ-Microsoft SQL Server Management Studio (Administrateur), dans le menu **Outils**, cliquez sur **SQL Server Profiler**.
2. Dans la fenêtre Se connecter au serveur, cliquez sur **Se conn.**
3. Dans **Nom de la trace**, tapez **Trace de Proseware**.
4. Dans le menu déroulant **Utiliser le modèle**, sélectionnez **Tuning**.
5. Activez l'option **Enregistrer dans le fichier**.
6. Dans la fenêtre Enregistrer sous, accédez au bureau, puis cliquez sur **Enregistrer**.
7. Désactivez l'option **Activer la substitution de fichier**.
8. Modifiez l'option **Définir la taille de fichier maximale (Mo)** en appliquant la valeur **500**.
9. Cliquez sur l'onglet **Sélection des événements**. Notez les événements sélectionnés.
10. Cliquez sur le titre de colonne **Database Name** pour créer un filtre.
11. Développez le nœud **Comme** et entrez **MarketDev**, puis cliquez sur **OK**.
12. Dans la fenêtre Propriétés de la trace, cliquez sur **Exécuter**.
13. Dans le menu **Fenêtre**, désactivez **Défilement automatique**.

► Tâche 2 : Exécuter la charge de travail

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 le menu **Requête**, sélectionnez **Options de requête**.
4. Dans la fenêtre Options de requête, cliquez sur **Grille**. Dans les options **Grille**, activez la case **Ignorer les résultats après l'exécution**.
5. Cliquez sur **OK** pour fermer la fenêtre Options de requête.
6. Dans la **barre d'outils**, cliquez sur **Exécuter**.
7. Attendez que la requête se termine.



Remarque La requête est terminée lorsqu'elle s'est exécutée avec succès et s'affiche sous l'onglet Messages. Aucun message de sortie ne s'affiche.

► Tâche 3 : Arrêter la trace

- Dans SQL Server Profiler, dans la **Barre d'outils**, cliquez sur l'icône **Arrêter la trace sélectionnée**, puis fermez **SQL Server Profiler**.

Exercice 2 : Analyser une trace avec l'Assistant Paramétrage du moteur de base de données

► Tâche 1 : Analyser la trace capturée dans l'Assistant Paramétrage du moteur de base de données

1. Dans 22462A_17_PRJ-Microsoft SQL Server Management Studio (Administrateur), dans le menu **Outils**, cliquez sur **Assistant Paramétrage du moteur de base de données**.
2. Dans la fenêtre Se connecter au serveur, cliquez sur **Se conn.**
3. Agrandissez la fenêtre de l'Assistant Paramétrage du moteur de base de données.
4. Dans la zone de groupe **Charge de travail**, vérifiez que **Fichier** est sélectionné, puis cliquez sur le bouton **Rechercher un fichier de charge de travail**.
5. Accédez au dossier **Bureau**, sélectionnez le fichier **Trace de Proseware.trc**, puis cliquez sur **Ouvrir**.
6. Dans le menu déroulant **Base de données pour l'analyse de la charge de travail**, sélectionnez **MarketDev**.
7. Dans la liste **Sélectionnez les bases de données et tables à analyser**, sélectionnez la base de données **MarketDev**.
8. Dans la **barre d'outils**, cliquez sur le bouton **Démarrer l'analyse** pour démarrer l'analyse du paramétrage.

► Tâche 2 : Examiner les modifications suggérées

1. Observez la Progression du paramétrage et une fois l'analyse terminée, notez les recommandations. Les recommandations varient mais il est probable que des recommandations relatives aux index et statistiques s'affichent.
2. Faites défiler les recommandations qui apparaissent à droite et notez les liens hypertexte à l'exemple de code.
3. Cliquez sur chacune des recommandations et notez les structures d'index ou de statistiques suggérées.
4. Fermez l'Assistant Paramétrage du moteur de base de données.

Exercice difficile 3 : Configurer Trace SQL (si le temps le permet)

► Tâche 1 : Créer un script qui utilise les procédures de Trace SQL pour implémenter le même type de capture que vous avez effectué dans l'exercice 1 mais avec un nom de trace différent

1. Dans 22462A_17_PRJ-Microsoft SQL Server Management Studio (Administrateur), dans le menu **Outils**, cliquez sur **SQL Server Profiler**.
2. Dans la fenêtre Se connecter au serveur, cliquez sur **Se conn.**
3. Dans **Nom de la trace**, tapez **ProsewareTrace2**.
4. Dans le menu déroulant **Utiliser le modèle**, sélectionnez **Tuning**.
5. Activez l'option **Enregistrer dans le fichier**.
6. Dans la fenêtre Enregistrer sous, accédez au bureau, puis cliquez sur **Enregistrer**.
7. Désactivez l'option **Activer la substitution de fichier**.
8. Modifiez l'option **Définir la taille de fichier maximale (Mo)** en appliquant la valeur **500**.
9. Cliquez sur l'onglet **Sélection des événements**. Notez les événements sélectionnés.
10. Cliquez sur le titre de colonne **Database Name** pour créer un filtre.
11. Développez le nœud **Comme** et entrez **MarketDev**, puis cliquez sur **OK**.
12. Dans la fenêtre Propriétés de la trace, cliquez sur **Exécuter**.
13. Dans la **barre d'outils**, cliquez sur l'icône **Arrêter la trace sélectionnée**.
14. Dans le menu **Fichier**, cliquez sur **Exporter**, cliquez sur **Générer un script de la définition de la trace**, cliquez sur **Pour SQL Server 2005 – SQL11**.
15. Naviguez jusqu'au **Bureau** et dans la zone de texte **Nom du fichier**, tapez **ProsewareTrace2**, puis cliquez sur **Enregistrer**.
16. Dans la fenêtre SQL Server Profiler, cliquez sur **OK**.
17. Fermez SQL Server Profiler.

► Tâche 2 : Tester que le script s'exécute comme prévu en utilisant la même charge de travail

1. Dans l'explorateur de solutions, dans le menu **Fichier**, cliquez sur **Ouvrir**, cliquez sur **Fichier**.
2. Dans la fenêtre Ouvrir un fichier, accédez au **Bureau**, cliquez sur **ProsewareTrace2.sql**, puis cliquez sur **Ouvrir**.
3. Sur la première ligne qui commence par une commande exec, modifiez la chaîne **InsertFileNameHere** en appliquant **D:\MKTG\ProsewareTrace2**.
4. Dans la **barre d'outils**, cliquez sur **Exécuter** pour démarrer la trace.



Remarque Enregistrez la valeur d'ID de trace retornée.

5. Dans l'Explorateur de solutions, double-cliquez sur le fichier **51 - Lab Exercise 1.sql** pour l'ouvrir.

6. Examinez le script T-SQL.
7. Dans le menu **Requête**, sélectionnez **Options de requête**.
8. Dans la fenêtre Options de requête, cliquez sur **Grille**. Dans les options **Grille**, activez la case **Ignorer les résultats après l'exécution**.
9. Cliquez sur **OK** pour fermer la fenêtre Options de requête.
10. Dans la **barre d'outils**, cliquez sur **Exécuter**.
11. Attendez que la requête se termine.

 **Remarque** La requête est terminée lorsqu'elle s'est exécutée avec succès et s'affiche sous l'onglet Messages. Aucun message de sortie ne s'affiche.

12. Dans l'Explorateur de solutions, double-cliquez sur le fichier **71 - Lab Exercise 3.sql** pour l'ouvrir.
13. Examinez le script T-SQL et remplacez la valeur de la variable **@TraceID** par la valeur que vous avez enregistrée précédemment dans cette tâche.
14. Dans la **barre d'outils**, cliquez sur **Exécuter**.

► **Tâche 3 : Analyser la nouvelle sortie capturée et noter si les modifications suggérées sont identiques à celles suggérées dans l'exercice 2**

1. Dans 22462A_17_PRJ-Microsoft SQL Server Management Studio (Administrateur), dans le menu **Outils**, cliquez sur **Assistant Paramétrage du moteur de base de données**.
2. Dans la fenêtre Se connecter au serveur, cliquez sur **Se conn.**
3. Agrandissez la fenêtre de l'Assistant Paramétrage du moteur de base de données.
4. Dans la zone de groupe **Charge de travail**, vérifiez que **Fichier** est sélectionné, puis cliquez sur le bouton **Rechercher un fichier de charge de travail**.
5. Accédez au dossier **D:\MKTG**, sélectionnez le fichier **ProsewareTrace2.trc**, puis cliquez sur **Ouvrir**.
6. Dans le menu déroulant **Base de données pour l'analyse de la charge de travail**, sélectionnez **MarketDev**.
7. Dans la liste **Sélectionnez les bases de données et tables à analyser**, sélectionnez la base de données **MarketDev**.
8. Dans la **barre d'outils**, cliquez sur le bouton **Démarrer l'analyse** pour démarrer l'analyse du paramétrage.
9. Une fois l'analyse terminée, comparez les résultats avec ceux que vous avez précédemment obtenus dans l'exercice 2.

 **Remarque** Les résultats doivent être identiques.

Module 18 : Surveillance de SQL Server 2012

Atelier pratique 18 : Surveillance de 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_18_PRJ\22462A_18_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 : Étudier les vues de gestion dynamique

► **Tâche 1 : Étudier l'utilisation des vues et des fonctions de gestion dynamique**

1. Dans l'Explorateur de solutions, double-cliquez sur le fichier **51 - Lab Exercise 1.sql** pour l'ouvrir.
2. Suivez les instructions contenues dans le script T-SQL.

Exercice 2 : Configurer l'entrepôt de données de gestion

- Tâche 1 : Créer un entrepôt de données de gestion pour la collection centrale des données de performances
1. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\administrateur)**, développez **Gestion**, cliquez avec le bouton droit sur **Collecte de données**, puis cliquez sur **Configurer l'entrepôt de données de gestion**.
 2. Dans la fenêtre Assistant Configuration de l'entrepôt de données de gestion, cliquez sur **Suivant**.
 3. Dans la fenêtre Sélectionner une tâche de configuration, cliquez sur **Suivant**.
 4. Dans la fenêtre Configuration du stockage de l'entrepôt de données de gestion, dans la zone de texte **Nom du serveur**, vérifiez que **Proseware** a été saisi, puis cliquez sur **Nouvelle**.
 5. Dans la fenêtre Nouvelle base de données, dans la zone de texte **Nom de la base de données**, tapez **MDW**, puis cliquez sur **OK**.
 6. Dans la fenêtre Configuration du stockage de l'entrepôt de données de gestion, cliquez sur **Suivant**.
 7. Dans la fenêtre Mapper les connexions et les utilisateurs, passez en revue les options et cliquez sur **Suivant**.
 8. Dans la fenêtre Terminer l'Assistant, cliquez sur **Terminer**.
 9. Dans la fenêtre Progression de l'Assistant Configuration de la collecte de données, cliquez sur **Fermer**.

Exercice 3 : Configurer les instances de la collecte de données

► Tâche 1 : Configurer la collecte de données sur chaque instance

1. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\administrateur)**, développez **Gestion**, cliquez avec le bouton droit sur **Collecte de données**, puis cliquez sur **Configurer l'entrepôt de données de gestion**.
2. Dans la fenêtre Assistant Configuration de l'entrepôt de données de gestion, cliquez sur **Suivant**.
3. Dans la fenêtre Sélectionner une tâche de configuration, cliquez sur **Configurer la collecte de données**, puis cliquez sur **Suivant**.
4. Dans la fenêtre Configuration du stockage de l'entrepôt de données de gestion, cliquez sur les **points de suspension** en regard de la zone de texte **Nom du serveur**.
5. Dans la fenêtre Se connecter au serveur, dans la zone de texte **Nom du serveur**, tapez **Proseware**, puis cliquez sur **Se conn.**
6. Dans la liste déroulante **Nom de la base de données**, sélectionnez **MDW**, puis cliquez sur **Suivant**.
7. Dans la fenêtre Terminer l'Assistant, cliquez sur **Terminer**.
8. Dans la fenêtre Progression de l'Assistant Configuration de la collecte de données, cliquez sur **Fermer**.
9. Dans l'Explorateur d'objets, cliquez sur **Connecter**, puis sur **Moteur de base de données**. Dans la fenêtre Se connecter au serveur, dans la zone de texte **Nom du serveur**, tapez **AdventureWorks**, puis cliquez sur **Se conn.**
10. Dans l'Explorateur d'objets, développez le serveur **AdventureWorks (SQL Server 11.0.2100-AdventureWorks\administrateur)**, développez **Gestion**, cliquez avec le bouton droit sur **Collecte de données**, puis cliquez sur **Configurer l'entrepôt de données de gestion**.
11. Dans la fenêtre Assistant Configuration de l'entrepôt de données de gestion, cliquez sur **Suivant**.
12. Dans la fenêtre Sélectionner une tâche de configuration, cliquez sur **Configurer la collecte de données**, puis cliquez sur **Suivant**.
13. Dans la fenêtre Configuration du stockage de l'entrepôt de données de gestion, cliquez sur les **points de suspension** en regard de la zone de texte **Nom du serveur**.
14. Dans la fenêtre Se connecter au serveur, dans la zone de texte **Nom du serveur**, tapez **Proseware**, puis cliquez sur **Se conn.**
15. Dans la liste déroulante **Nom de la base de données**, sélectionnez **MDW**, puis cliquez sur **Suivant**.
16. Dans la fenêtre Terminer l'Assistant, cliquez sur **Terminer**.
17. Dans la fenêtre Progression de l'Assistant Configuration de la collecte de données, cliquez sur **Fermer**.

Exercice difficile 4 : Utiliser les rapports du collecteur de données (uniquement si le temps le permet)

► Tâche 1 : Désactiver les collecteurs de données sur les deux instances

1. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100-AdventureWorks\administrateur)**, développez **Gestion**, cliquez avec le bouton droit sur **Collecte de données**, puis cliquez sur **Désactiver la collecte de données**. Si une fenêtre de confirmation s'affiche, cliquez sur **Fermer**.
2. Dans l'Explorateur d'objets, développez le serveur **AdventureWorks (SQL Server 11.0.2100-AdventureWorks\administrateur)**, développez **Gestion**, cliquez avec le bouton droit sur **Collecte de données**, puis cliquez sur **Désactiver la collecte de données**. Si une fenêtre de confirmation s'affiche, cliquez sur **Fermer**.

► Tâche 2 : Restaurer une sauvegarde de la base de données MDW

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 **MDW**, puis cliquez sur **Supprimer**.
2. Dans la fenêtre Supprimer un objet, activez la case à cocher **Fermer les connexions existantes**, puis cliquez sur **OK**.
3. 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**.
4. Dans la fenêtre Restaurer la base de données, dans les **options source**, cliquez sur **Périphérique**.
5. Cliquez sur les **points de suspension** en regard de la zone de texte **Périphérique**.
6. Dans la fenêtre Sélectionner les unités de sauvegarde, cliquez sur **Ajouter**.
7. Accédez au fichier **D:\22462A_Labs\22462A_18_PRJ\22462A_18_PRJ\MDW.bak**, puis cliquez sur **OK**.
8. Dans la fenêtre Sélectionner les unités de sauvegarde, cliquez sur **OK**.
9. Dans le volet **Sélectionner une page**, cliquez sur **Options**.

10. Dans **Options de restauration**, activez la case **Remplacer la base de données existante (WITH REPLACE)**, puis cliquez sur **OK**.

11. Dans la fenêtre Microsoft SQL Server Management Studio, cliquez sur **OK**.

► Tâche 3 : Examiner les rapports disponibles

1. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\administrateur)**, puis développez **Bases de données**.
2. Cliquez avec le bouton droit sur la base de données **MDW**, cliquez sur **Rapports**, cliquez sur **Entrepôt de données de gestion**, puis cliquez sur **Vue d'ensemble de l'entrepôt de données de gestion**.
3. Examinez le **rapport d'Utilisation du disque** relatif à **MIA-SQL1\MKTG**.

4. Fermez la fenêtre du rapport.
5. Cliquez avec le bouton droit sur la base de données **MDW**, cliquez sur **Rapports**, cliquez sur **Entrepôt de données de gestion**, puis cliquez sur **Vue d'ensemble de l'entrepôt de données de gestion**.
6. Examinez le **rapport d'Activité du serveur** relatif à **MIA-SQL1\MKTG**.
7. Fermez la fenêtre du rapport. Cliquez avec le bouton droit sur la base de données **MDW**, cliquez sur **Rapports**, cliquez sur **Entrepôt de données de gestion**, puis cliquez sur **Vue d'ensemble de l'entrepôt de données de gestion**.
8. Examinez le **rapport Statistiques sur les requêtes** relatif à **MIA-SQL1\MKTG**.
9. Fermez la fenêtre du rapport.

Module 19 : Gestion de plusieurs serveurs

Atelier pratique 19 : Gestion de plusieurs serveurs

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_19_PRJ\22462A_19_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 : Configurer CMS et exécuter des requêtes multiserveurs

► Tâche 1 : Créer un serveur de gestion centralisée

1. Dans le menu **Affichage**, dans **Microsoft SQL Server Management Studio (Administrateur)**, cliquez sur **Serveurs inscrits**.
2. Dans la fenêtre Serveurs inscrits qui s'ouvre, développez **Moteur de base de données**.
3. Cliquez avec le bouton droit sur **Serveurs de gestion centralisée**, puis cliquez sur **Inscrire le serveur de gestion centralisée**.
4. Dans la fenêtre Nouvelle inscription de serveur, dans la zone de texte **Nom du serveur**, tapez **Proseware**, puis cliquez sur **Enregistrer**.

► Tâche 2 : Créer un groupe de serveurs sur le serveur de gestion centralisée

1. Dans Microsoft SQL Server Management Studio (Administrateur), dans **Serveurs inscrits**, développez **Serveurs de gestion centralisée**.
2. Cliquez avec le bouton droit sur **Proseware**, puis cliquez sur **Nouveau groupe de serveurs**.
3. Dans la fenêtre Propriétés du nouveau groupe de serveurs, dans la zone de texte **Nom du groupe**, tapez **Serveurs de base**, puis cliquez sur **OK**.
4. Dans **Serveurs inscrits**, cliquez avec le bouton droit sur **Serveurs de base**, puis cliquez sur **Nouvelle inscription de serveur**.
5. Dans la fenêtre Nouvelle inscription de serveur, dans la zone de texte **Nom du serveur**, tapez **Proseware**, puis cliquez sur **Enregistrer**.
6. Dans **Serveurs inscrits**, cliquez avec le bouton droit sur **Serveurs de base**, puis cliquez sur **Nouvelle inscription de serveur**.
7. Dans la fenêtre Nouvelle inscription de serveur, dans la zone de texte **Nom du serveur**, tapez **AdventureWorks**, puis cliquez sur **Enregistrer**.

► Tâche 3 : Exécuter une commande pour rechercher toutes les bases de données sur n'importe quel serveur de base en mode de restauration complète

1. Dans **Serveurs inscrits**, cliquez avec le bouton droit sur **Serveurs de base**, puis cliquez sur **Nouvelle requête**.
2. Tapez le code d'instruction suivant dans la nouvelle fenêtre de requête :

```
SELECT * FROM sys.databases WHERE recovery_model_desc = 'FULL';
```

3. Dans la **barre d'outils**, cliquez sur **Exécuter**.



Remarque Une liste de bases de données en mode de restauration complète est retournée.

Exercice 2 : Déployer une application de la couche Données

► Tâche 1 : Déployer l'application de la couche Données à l'aide de SSMS

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 **Déployer une application de la couche Données**.
2. Dans la fenêtre Introduction, cliquez sur **Suivant**.
3. Dans la fenêtre Sélectionner un package, entrez ou sélectionnez le chemin d'accès **D:\22462A_Labs\22462A_19_PRJ\22462A_19_PRJ\CityCode.dacpac**, puis cliquez sur **Suivant**.
4. Dans la fenêtre Mettre à jour la configuration, cliquez sur **Suivant**.
5. Dans la fenêtre Résumé, cliquez sur **Suivant**.
6. Dans la fenêtre Déployer une DAC, cliquez sur **Terminer**.
7. Dans l'Explorateur de solutions, double-cliquez sur le fichier **61 - Lab Exercise 2.sql** pour l'ouvrir.
8. Examinez le script T-SQL.
9. Dans la **barre d'outils**, cliquez sur **Exécuter**.

Exercice 3 : Enregistrer et extraire une application de la couche Données

► Tâche 1 : Enregistrer la base de données existante en tant qu'application de la couche Données

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 **Research**, cliquez sur **Tâches**, puis cliquez sur **Inscrire en tant qu'application de la couche Données**.
2. Dans la fenêtre Introduction, cliquez sur **Suivant**.
3. Dans la fenêtre Définir les propriétés, cliquez sur **Suivant**.
4. Dans la fenêtre Validation et résumé, cliquez sur **Suivant**.
5. Dans la fenêtre Incrire la DAC, cliquez sur **Terminer**.

► Tâche 2 : Extraire un dacpac de la base de données à envoyer à l'équipe de développement

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 **Research**, cliquez sur **Tâches**, puis cliquez sur **Extraire une application de la couche Données**.
2. Dans la fenêtre Introduction, cliquez sur **Suivant**.
3. Dans la fenêtre Définir les propriétés, dans la zone de texte **Enregistrer dans le fichier de package DAC**, tapez **D:\MKTG\Research.dacpac**, puis cliquez sur **Suivant**.
4. Dans la fenêtre Validation et résumé, cliquez sur **Suivant**.
5. Dans la fenêtre Générer le package, cliquez sur **Terminer**.

Exercice difficile 4 : Mettre à niveau une application de la couche Données (uniquement si le temps le permet)

► Tâche 1 : Tenter de déployer la mise à niveau v2

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 **Actualiser**.
2. Développez **Bases de données**, cliquez avec le bouton droit sur la base de données **CityCode**, cliquez sur **Tâches**, puis cliquez sur **Mettre à niveau l'application de la couche Données**.
3. Dans la fenêtre Introduction, cliquez sur **Suivant**.
4. Dans la fenêtre Sélectionner un package, entrez ou sélectionnez le chemin d'accès **D:\22462A_Labs\22462A_19_PRJ\22462A_19_PRJ\CityCode_v2.dacpac**, puis cliquez sur **Suivant**.
5. Dans la fenêtre Détecter les modifications, cliquez sur **Suivant**.
6. Dans la fenêtre Options, activez la case **Restauration en cas d'échec**, puis cliquez sur **Suivant**.
7. Dans la fenêtre Vérifier le plan de mise à niveau, activez la case à cocher **Poursuivre malgré le risque de perte de données**, puis cliquez sur **Suivant**.
8. Dans la fenêtre Résumé, cliquez sur **Suivant**.
9. Dans la fenêtre Microsoft SQL Server Management Studio, cliquez sur **OK**.
10. Dans la fenêtre Mettre à niveau la DAC, notez les échecs et cliquez sur **Terminer**.
11. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur **Bases de données**, puis cliquez sur **Actualiser**.



Remarque Toutes les bases de données et les applications de la couche Données ont été conservés inchangées.

► Tâche 2 : Déployer la mise à niveau v3

1. Dans l'Explorateur d'objets, développez **Bases de données**, cliquez avec le bouton droit sur la base de données **CityCode**, cliquez sur **Tâches**, puis cliquez sur **Mettre à niveau l'Application de la couche Données**.
2. Dans la fenêtre Introduction, cliquez sur **Suivant**.
3. Dans la fenêtre Sélectionner un package, entrez ou sélectionnez le chemin d'accès **D:\22462A_Labs\22462A_19_PRJ\22462A_19_PRJ\CityCode_v3.dacpac**, puis cliquez sur **Suivant**.
4. Dans la fenêtre Détecter les modifications, cliquez sur **Suivant**.
5. Dans la fenêtre Options, activez la case **Restauration en cas d'échec**, puis cliquez sur **Suivant**.
6. Dans la fenêtre Vérifier le plan de mise à niveau, cliquez sur **Suivant**.
7. Dans la fenêtre Résumé, cliquez sur **Suivant**.

8. Dans la fenêtre Mettre à niveau la DAC, cliquez sur **Terminer**.
9. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur **Bases de données**, puis cliquez sur **Actualiser**.



Remarque L'application de la couche Données est mise à niveau.

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

Atelier pratique 20 : Résolution des problèmes courants

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_20_PRJ\22462A_20_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.

Exercices 1 à 5 : Dépanner et résoudre les problèmes liés à l'administration SQL Server

Exercice 1 – Problème 1

► **Tâche 1 : Lire la documentation associée à l'exercice**

- Lisez la documentation associée à l'exercice.

► **Tâche 2 : Dépanner et résoudre le problème**

1. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100 – ADVENTUREWORKS\administrateur)**, développez **Sécurité**, puis **Connexions**.
2. En regard de l'utilisateur **PromoteApp**, remarquez l'icône avec une flèche rouge orientée vers le bas, elle indique que la connexion est désactivée.
3. Cliquez avec le bouton droit sur **PromoteApp**, puis cliquez sur **Propriétés**.
4. Dans le volet **Sélectionner une page**, cliquez sur **État**, cliquez sur **Activé**, puis cliquez sur **OK**.
5. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur **Connexions** et cliquez sur **Actualiser**. Notez que la connexion est maintenant activée.

Exercice 2 – Problème 2

► **Tâche 1 : Lire la documentation associée à l'exercice**

- Lisez la documentation associée à l'exercice.

► **Tâche 2 : Dépanner et résoudre le problème**

1. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100 – ADVENTUREWORKS\administrateur)**, développez **Bases de données**, puis notez l'état de la base de données **AdminDB (Restauration...)**. La base de données est en état de restauration, l'administrateur de base de données junior peut avoir effectué une sauvegarde de la fin du journal.
2. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur la base de données **AdminDB**, cliquez sur **Tâches**, cliquez sur **Restaurer**, puis sur **Journal des transactions**.
3. Dans la fenêtre Restaurer le journal des transactions – AdminDB, cliquez sur **À partir d'un fichier ou d'une bande**, puis cliquez sur **OK**.
4. Dans la fenêtre Microsoft SQL Server Management Studio, cliquez sur **OK**.

Exercice 3 – Problème 2

► Tâche 1 : Lire la documentation associée à l'exercice

- Lisez la documentation associée à l'exercice.

► Tâche 2 : Dépanner et résoudre le problème

1. Dans l'Explorateur d'objets, développez le serveur **Proseware (SQL Server 11.0.2100-ADVENTUREWORKS\administrateur)**, développez **Agent SQL Server**, puis développez **Travaux**.
2. Cliquez avec le bouton droit sur le travail permettant **Get File List**, puis cliquez sur **Démarrer le travail à l'étape**.
3. Dans la fenêtre Démarrer les travaux – Proseware, notez l'échec, puis cliquez sur **Fermer**.
4. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur le travail permettant **Get File List**, puis cliquez sur **Afficher l'historique**.
5. Dans la fenêtre Visionneuse du fichier journal – Proseware, cliquez sur le signe plus en regard d'une ligne d'échec pour développer les étapes de travail.
6. Cliquez sur la ligne de l'ID de l'étape **1**.
7. Dans le volet **Détails de la ligne sélectionnée**, faites défiler l'écran pour rechercher l'erreur (Échec d'ouverture de session : nom d'utilisateur inconnu ou mot de passe incorrect). Notez que l'erreur se produit en utilisant le proxy de David.Alexander.
8. Dans la fenêtre Visionneuse du fichier journal – Proseware, cliquez sur **Fermer**.
9. Dans l'Explorateur d'objets, développez **Sécurité**, développez **Informations d'identification**, cliquez avec le bouton droit sur **DavidAlexanderCredential**, puis cliquez sur **Propriétés**.
10. Dans la fenêtre Propriétés des informations d'identification - DavidAlexanderCredential, dans la zone de texte **Mot de passe** et la zone de texte **Confirmer le mot de passe**, tapez **Pa\$\$w0rd**, puis cliquez sur **OK**.
11. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur le travail permettant **Get File List**, puis cliquez sur **Démarrer le travail à l'étape**.
12. Dans la fenêtre Démarrer les travaux - Proseware, notez que le travail se solde toujours par un échec et cliquez sur **Fermer**.
13. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur le travail permettant **Get File List**, puis cliquez sur **Afficher l'historique**.
14. Dans la fenêtre Visionneuse du fichier journal – Proseware, cliquez sur le signe plus en regard d'une ligne d'échec pour développer les étapes de travail.
15. Cliquez sur la ligne de l'ID de l'étape **1**.
16. Dans le volet **Détails de la ligne sélectionnée**, faites défiler l'écran pour rechercher l'erreur (accès refusé).
17. Dans la fenêtre Visionneuse du fichier journal – Proseware, cliquez sur **Fermer**. Nous devons étudier l'étape de travail afin de déterminer pourquoi l'erreur d'accès refusé est retournée.
18. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur le travail permettant **Get File List**, puis cliquez sur **Propriétés**.

19. Dans le volet **Sélectionner une page**, cliquez sur **Étapes**, puis cliquez sur **Modifier**.
20. Notez que la commande en cours d'exécution nécessite une autorisation pour lire les entrées de répertoire figurant dans le dossier D:\MKTG et une autorisation pour écrire dans le dossier L:\MKTG à l'aide de l'identité de David Alexander.
21. Cliquez sur **Annuler**, puis sur **Annuler**.
22. Sur la base que les autorisations d'écriture sont moins facilement accordées que les autorisations de lecture, nous étudierons les autorisations d'écriture en premier. Dans l'Explorateur Windows, cliquez avec le bouton droit sur le dossier **L:\MKTG**, puis cliquez sur **Propriétés**.
23. Dans la fenêtre Propriétés de MKTG, cliquez sur l'onglet **Sécurité** et notez que David Alexander n'a pas accès au dossier, ni aux groupes dont il est susceptible d'être membre.
24. Cliquez sur **Modifier**, puis dans la fenêtre Autorisations de MKTG, cliquez sur **Ajouter**.
25. Vérifiez que la valeur **À partir de cet emplacement** est définie pour **AdventureWorks.msft**, puis dans la zone de texte **Entrez les noms d'objets à sélectionner (exemples)**, tapez **David.Alexander**, puis cliquez sur **OK**.
26. Dans la fenêtre Autorisations pour MKTG, cliquez sur **la case Autoriser de la ligne Modifier**, puis cliquez sur **OK**.
27. Dans la fenêtre Propriétés de : MKTG, cliquez sur **OK**.
28. Maintenant, nous testerons si le travail s'exécute.
29. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur le travail permettant **Get File List**, puis cliquez sur **Démarrer le travail à l'étape**.
30. Dans la fenêtre Démarrer les travaux - Proseware, notez que le travail se solde toujours par un échec et cliquez sur **Fermer**.
31. Maintenant, nous devons vérifier les autorisations de lecture.
32. Dans l'Explorateur Windows, cliquez avec le bouton droit sur le dossier **D:\MKTG**, puis cliquez sur **Propriétés**.
33. Dans la fenêtre Propriétés de: MKTG, cliquez sur l'onglet **Sécurité** et notez que David Alexander n'a pas accès au dossier, ni aux groupes dont il est susceptible d'être membre.
34. Cliquez sur **Modifier**, puis dans la fenêtre Autorisations de MKTG, cliquez sur **Ajouter**.
35. Vérifiez que la valeur **À partir de cet emplacement** est définie pour **AdventureWorks.msft**, puis dans la zone de texte **Entrez les noms d'objets à sélectionner (exemples)**, tapez **David.Alexander**, puis cliquez sur **OK**.
36. Dans la fenêtre Autorisations pour MKTG, cliquez sur **OK**.
37. Dans la fenêtre Propriétés de : MKTG, cliquez sur **OK**.
38. Maintenant, nous testerons à nouveau si le travail s'exécute.
39. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur le travail permettant **Get File List**, puis cliquez sur **Démarrer le travail à l'étape**.
40. Dans la fenêtre Démarrer les travaux - Proseware, notez que le travail s'exécute désormais correctement ; cliquez sur **Fermer**.

Exercice 4 – Problème 2

► Tâche 1 : Lire la documentation associée à l'exercice

- Lisez la documentation associée à l'exercice.

► Tâche 2 : Dépanner et résoudre le problème

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 **AdminDB**, puis cliquez sur **Propriétés**.
2. Dans le volet **Sélectionner une page**, cliquez sur **Options**.
3. Vérifiez les options associées aux statistiques.
4. Changez la **Création automatique des statistiques** en appliquant la valeur **True**.
5. Changez la **Mise à jour automatique des statistiques** en appliquant la valeur **True**, puis cliquez sur **OK**.

 **Remarque** Les étapes suivantes génèrent des statistiques à mettre à jour immédiatement au lieu d'attendre les statistiques automatiques pour corriger le problème.

6. Dans l'Explorateur d'objets, cliquez avec le bouton droit sur **AdminDB**, puis cliquez sur **Nouvelle requête**.

7. Dans la fenêtre Requête, tapez la commande ci-dessous :

```
EXEC sp_updatestats;
```

8. Dans la **barre d'outils**, cliquez sur **Exécuter**.

Exercice 5 – Problème 2

► **Tâche 1 : Lire la documentation associée à l'exercice**

- Lisez la documentation associée à l'exercice.

► **Tâche 2 : Dépanner et résoudre le problème**

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 **CityDetails**, puis cliquez sur **Propriétés**.
2. Dans le volet **Sélectionner une page**, cliquez sur **Options**.
3. Vérifiez les options associées aux statistiques.
4. Modifiez l'option **Fermeture automatique** sur **False**, puis cliquez sur **OK**.

