Microsoft

OFFICIAL MICROSOFT LEARNING PRODUCT

10775A

Administering Microsoft® SQL Server® 2012 Database

Product Number: 10775A

Part Number: X18-29125

Released: 05/2012

MICROSOFT LICENSE TERMS
OFFICIAL MICROSOFT LEARNING PRODUCTS
MICROSOFT OFFICIAL COURSE Pre-Release and Final Release Versions

---

These license terms are an agreement between Microsoft Corporation and you. Please read them. They apply to the Licensed Content named above, which includes the media on which you received it, if any. These license terms also apply to any updates, supplements, internet based services and support services for the Licensed Content, unless other terms accompany those items. If so, those terms apply.

BY DOWNLOADING OR USING THE LICENSED CONTENT, YOU ACCEPT THESE TERMS. IF YOU DO NOT ACCEPT THEM, DO NOT DOWNLOAD OR USE THE LICENSED CONTENT.

If you comply with these license terms, you have the rights below.

---

1. **DEFINITIONS.**

   a. "Authorized Learning Center" means a Microsoft Learning Competency Member, Microsoft IT Academy Program Member, or such other entity as Microsoft may designate from time to time.

   b. "Authorized Training Session" means the Microsoft-authorized instructor-led training class using only MOC Courses that are conducted by a MCT at or through an Authorized Learning Center.

   c. "Classroom Device" means one (1) dedicated, secure computer that you own or control that meets or exceeds the hardware level specified for the particular MOC Course located at your training facilities or primary business location.

   d. "End User" means an individual who is (i) duly enrolled for an Authorized Training Session or Private Training Session, (ii) an employee of a MPN Member, or (iii) a Microsoft full-time employee.

   e. "Licensed Content" means the MOC Course and any other content accompanying this agreement. Licensed Content may include (i) Trainer Content, (ii) sample code, and (iii) associated media.

   f. "Microsoft Certified Trainer" or "MCT" means an individual who is (i) engaged to teach a training session to End Users on behalf of an Authorized Learning Center or MPN Member, (ii) currently certified as a Microsoft Certified Trainer under the Microsoft Certification Program, and (iii) holds a Microsoft Certification in the technology that is the subject of the training session.

   g. "Microsoft IT Academy Member" means a current, active member of the Microsoft IT Academy Program.

   h. "Microsoft Learning Competency Member" means a Microsoft Partner Network Program Member in good standing that currently holds the Learning Competency status.

   i. "Microsoft Official Course" or "MOC Course" means the Official Microsoft Learning Product instructor-led courseware that educates IT professionals or developers on Microsoft technologies.

j. "Microsoft Partner Network Member" or "MPN Member" means a silver or gold-level Microsoft Partner Network program member in good standing.

k. "Personal Device" means one (1) device, workstation or other digital electronic device that you personally own or control that meets or exceeds the hardware level specified for the particular MOC Course.

l. "Private Training Session" means the instructor-led training classes provided by MPN Members for corporate customers to teach a predefined learning objective. These classes are not advertised or promoted to the general public and class attendance is restricted to individuals employed by or contracted by the corporate customer.

m. "Trainer Content" means the trainer version of the MOC Course and additional content designated solely for trainers to use to teach a training session using a MOC Course. Trainer Content may include Microsoft PowerPoint presentations, instructor notes, lab setup guide, demonstration guides, beta feedback form and trainer preparation guide for the MOC Course. To clarify, Trainer Content does not include virtual hard disks or virtual machines.

2. **INSTALLATION AND USE RIGHTS**. The Licensed Content is licensed not sold. The Licensed Content is licensed on a one copy per user basis, such that you must acquire a license for each individual that accesses or uses the Licensed Content.

2.1 Below are four separate sets of installation and use rights. Only one set of rights apply to you.

a. **If you are a Authorized Learning Center:**

i. If the Licensed Content is in digital format for each license you acquire you may either:

1. install one (1) copy of the Licensed Content in the form provided to you on a dedicated, secure server located on your premises where the Authorized Training Session is held for access and use by one (1) End User attending the Authorized Training Session, or by one (1) MCT teaching the Authorized Training Session, or
2. install one (1) copy of the Licensed Content in the form provided to you on one (1) Classroom Device for access and use by one (1) End User attending the Authorized Training Session, or by one (1) MCT teaching the Authorized Training Session.

ii. You agree that:

1. you will acquire a license for each End User and MCT that accesses the Licensed Content,
2. each End User and MCT will be presented with a copy of this agreement and each individual will agree that their use of the Licensed Content will be subject to these license terms prior to their accessing the Licensed Content. Each individual will be required to denote their acceptance of the EULA in a manner that is enforceable under local law prior to their accessing the Licensed Content,
3. for all Authorized Training Sessions, you will only use qualified MCTs who hold the applicable competency to teach the particular MOC Course that is the subject of the training session,
4. you will not alter or remove any copyright or other protective notices contained in the Licensed Content,

5. you will remove and irretrievably delete all Licensed Content from all Classroom Devices and servers at the end of the Authorized Training Session,
6. you will only provide access to the Licensed Content to End Users and MCTs,
7. you will only provide access to the Trainer Content to MCTs, and
8. any Licensed Content installed for use during a training session will be done in accordance with the applicable classroom set-up guide.

b. **If you are a MPN Member.**

i. If the Licensed Content is in digital format for each license you acquire you may either:

1. install one (1) copy of the Licensed Content in the form provided to you on (A) one (1) Classroom Device, or (B) one (1) dedicated, secure server located at your premises where the training session is held for use by one (1) of your employees attending a training session provided by you, or by one (1) MCT that is teaching the training session, **or**
2. install one (1) copy of the Licensed Content in the form provided to you on one (1) Classroom Device for use by one (1) End User attending a Private Training Session, or one (1) MCT that is teaching the Private Training Session.

ii. You agree that:

1. you will acquire a license for each End User and MCT that accesses the Licensed Content,
2. each End User and MCT will be presented with a copy of this agreement and each individual will agree that their use of the Licensed Content will be subject to these license terms prior to their accessing the Licensed Content. Each individual will be required to denote their acceptance of the EULA in a manner that is enforceable under local law prior to their accessing the Licensed Content,
3. for all training sessions, you will only use qualified MCTs who hold the applicable competency to teach the particular MOC Course that is the subject of the training session,
4. you will not alter or remove any copyright or other protective notices contained in the Licensed Content,
5. you will remove and irretrievably delete all Licensed Content from all Classroom Devices and servers at the end of each training session,
6. you will only provide access to the Licensed Content to End Users and MCTs,
7. you will only provide access to the Trainer Content to MCTs, and
8. any Licensed Content installed for use during a training session will be done in accordance with the applicable classroom set-up guide.

c. **If you are an End User:**

You may use the Licensed Content solely for your personal training use. If the Licensed Content is in digital format, for each license you acquire you may (i) install one (1) copy of the Licensed Content in the form provided to you on one (1) Personal Device and install another copy on another Personal Device as a backup copy, which may be used only to reinstall the Licensed Content; or (ii) print one (1) copy of the Licensed Content. You may not install or use a copy of the Licensed Content on a device you do not own or control.

d. **If you are a MCT.**

i. For each license you acquire, you may use the Licensed Content solely to prepare and deliver an Authorized Training Session or Private Training Session. For each license you acquire, you may install and use one (1) copy of the Licensed Content in the form provided to you on one (1) Personal Device and install one (1) additional copy on another Personal Device as a backup copy, which may be used only to reinstall the Licensed Content. You may not install or use a copy of the Licensed Content on a device you do not own or control.

ii. **Use of Instructional Components in Trainer Content.** You may customize, in accordance with the most recent version of the MCT Agreement, those portions of the Trainer Content that are logically associated with instruction of a training session. If you elect to exercise the foregoing rights, you agree: (a) that any of these customizations will only be used for providing a training session, (b) any customizations will comply with the terms and conditions for Modified Training Sessions and Supplemental Materials in the most recent version of the MCT agreement and with this agreement. For clarity, any use of "*customize"* refers only to changing the order of slides and content, and/or not using all the slides or content, it does not mean changing or modifying any slide or content.

2.2 **Separation of Components.** The Licensed Content components are licensed as a single unit and you may not separate the components and install them on different devices.

2.3 **Reproduction/Redistribution Licensed Content**. Except as expressly provided in the applicable installation and use rights above, you may not reproduce or distribute the Licensed Content or any portion thereof (including any permitted modifications) to any third parties without the express written permission of Microsoft.

2.4 **Third Party Programs**. The Licensed Content may contain third party programs or services. These license terms will apply to your use of those third party programs or services, unless other terms accompany those programs and services.

2.5 **Additional Terms**. Some Licensed Content may contain components with additional terms, conditions, and licenses regarding its use. Any non-conflicting terms in those conditions and licenses also apply to that respective component and supplements the terms described in this Agreement.

3. **PRE-RELEASE VERSIONS.** If the Licensed Content is a pre-release ("**beta**") version, in addition to the other provisions in this agreement, then these terms also apply:

a. **Pre-Release Licensed Content.** This Licensed Content is a pre-release version. It may not contain the same information and/or work the way a final version of the Licensed Content will. We may change it for the final version. We also may not release a final version. Microsoft is under no obligation to provide you with any further content, including the final release version of the Licensed Content.

b. **Feedback.** If you agree to give feedback about the Licensed Content to Microsoft, either directly or through its third party designee, you give to Microsoft without charge, the right to use, share and commercialize your feedback in any way and for any purpose. You also give to third parties, without charge, any patent rights needed for their products, technologies and services to use or interface with any specific parts of a Microsoft software, Microsoft product, or service that includes the feedback. You will not give feedback that is subject to a license that requires Microsoft to license its software, technologies, or products to third parties because we include your feedback in them. These rights

survive this agreement.

c. **Term**. If you are an Authorized Training Center, MCT or MPN, you agree to cease using all copies of the beta version of the Licensed Content upon (i) the date which Microsoft informs you is the end date for using the beta version, or (ii) sixty (60) days after the commercial release of the Licensed Content, whichever is earliest ("**beta term**"). Upon expiration or termination of the beta term, you will irretrievably delete and destroy all copies of same in the possession or under your control.

4. **INTERNET-BASED SERVICES**. Classroom Devices located at Authorized Learning Center's physical location may contain virtual machines and virtual hard disks for use while attending an Authorized Training Session. You may only use the software on the virtual machines and virtual hard disks on a Classroom Device solely to perform the virtual lab activities included in the MOC Course while attending the Authorized Training Session. Microsoft may provide Internet-based services with the software included with the virtual machines and virtual hard disks. It may change or cancel them at any time. If the software is pre-release versions of software, some of its Internet-based services may be turned on by default. The default setting in these versions of the software do not necessarily reflect how the features will be configured in the commercially released versions. If Internet-based services are included with the software, they are typically simulated for demonstration purposes in the software and no transmission over the Internet takes place. However, should the software be configured to transmit over the Internet, the following terms apply:

a. **Consent for Internet-Based Services.** The software features described below connect to Microsoft or service provider computer systems over the Internet. In some cases, you will not receive a separate notice when they connect. You may switch off these features or not use them. By using these features, you consent to the transmission of this information. Microsoft does not use the information to identify or contact you.

b. **Computer Information.** The following features use Internet protocols, which send to the appropriate systems computer information, such as your Internet protocol address, the type of operating system, browser and name and version of the software you are using, and the language code of the device where you installed the software. Microsoft uses this information to make the Internet-based services available to you.

- **Accelerators**. When you use click on or move your mouse over an Accelerator, the title and full web address or URL of the current webpage, as well as standard computer information, and any content you have selected, might be sent to the service provider. If you use an Accelerator provided by Microsoft, the information sent is subject to the Microsoft Online Privacy Statement, which is available at go.microsoft.com/fwlink/?linkid=31493. If you use an Accelerator provided by a third party, use of the information sent will be subject to the third party's privacy practices.

- **Automatic Updates.** This software contains an Automatic Update feature that is on by default. For more information about this feature, including instructions for turning it off, see go.microsoft.com/fwlink/?LinkId=178857. You may turn off this feature while the software is running ("opt out"). Unless you expressly opt out of this feature, this feature will (a) connect to Microsoft or service provider computer systems over the Internet, (b) use Internet protocols to send to the appropriate systems standard computer information, such as your computer's Internet protocol address, the type of operating system, browser and name and version of the software you are using, and the language code of the device where you installed the software, and (c) automatically download and install, or prompt you to download and/or install, current Updates to the software. In some cases, you will not receive a separate notice before this feature takes effect.

By installing the software, you consent to the transmission of standard computer information and the automatic downloading and installation of updates.

- **Auto Root Update**. The Auto Root Update feature updates the list of trusted certificate authorities. you can switch off the Auto Root Update feature.
- **Customer Experience Improvement Program (CEIP), Error and Usage Reporting; Error Reports**. This software uses CEIP and Error and Usage Reporting components enabled by default that automatically send to Microsoft information about your hardware and how you use this software. This software also automatically sends error reports to Microsoft that describe which software components had errors and may also include memory dumps. You may choose not to use these software components. For more information please go to <http://go.microsoft.com/fwlink/?LinkID=196910>.
- **Digital Certificates**. The software uses digital certificates. These digital certificates confirm the identity of Internet users sending X.509 standard encrypted information. They also can be used to digitally sign files and macros, to verify the integrity and origin of the file contents. The software retrieves certificates and updates certificate revocation lists. These security features operate only when you use the Internet.
- **Extension Manager**. The Extension Manager can retrieve other software through the internet from the Visual Studio Gallery website. To provide this other software, the Extension Manager sends to Microsoft the name and version of the software you are using and language code of the device where you installed the software. This other software is provided by third parties to Visual Studio Gallery. It is licensed to users under terms provided by the third parties, not from Microsoft. Read the Visual Studio Gallery terms of use for more information.
- **IPv6 Network Address Translation (NAT) Traversal service (Teredo)**. This feature helps existing home Internet gateway devices transition to IPv6. IPv6 is a next generation Internet protocol. It helps enable end-to-end connectivity often needed by peer-to-peer applications. To do so, each time you start up the software the Teredo client service will attempt to locate a public Teredo Internet service. It does so by sending a query over the Internet. This query only transfers standard Domain Name Service information to determine if your computer is connected to the Internet and can locate a public Teredo service. If you

    · use an application that needs IPv6 connectivity or
    · configure your firewall to always enable IPv6 connectivity

  by default standard Internet Protocol information will be sent to the Teredo service at Microsoft at regular intervals. No other information is sent to Microsoft. You can change this default to use non-Microsoft servers. You can also switch off this feature using a command line utility named “netsh”.
- **Malicious Software Removal**. During setup, if you select “Get important updates for installation”, the software may check and remove certain malware from your device. “Malware” is malicious software. If the software runs, it will remove the Malware listed and updated at www.support.microsoft.com/?kbid=890830. During a Malware check, a report will be sent to Microsoft with specific information about Malware detected, errors, and other information about your device. This information is used to improve the software and other Microsoft products and services. No information included in these reports will be used to identify or contact you. You may disable the software’s reporting functionality by following the instructions found at

www.support.microsoft.com/?kbid=890830. For more information, read the Windows Malicious Software Removal Tool privacy statement at go.microsoft.com/fwlink/?LinkId=113995.

- **Microsoft Digital Rights Management.** If you use the software to access content that has been protected with Microsoft Digital Rights Management (DRM), then, in order to let you play the content, the software may automatically request media usage rights from a rights server on the Internet and download and install available DRM updates. For more information, see go.microsoft.com/fwlink/?LinkId=178857.
- **Microsoft Telemetry Reporting Participation**. If you choose to participate in Microsoft Telemetry Reporting through a "basic" or "advanced" membership, information regarding filtered URLs, malware and other attacks on your network is sent to Microsoft. This information helps Microsoft improve the ability of Forefront Threat Management Gateway to identify attack patterns and mitigate threats. In some cases, personal information may be inadvertently sent, but Microsoft will not use the information to identify or contact you. You can switch off Telemetry Reporting. For more information on this feature, see http://go.microsoft.com/fwlink/?LinkId=130980.
- **Microsoft Update Feature**. To help keep the software up-to-date, from time to time, the software connects to Microsoft or service provider computer systems over the Internet. In some cases, you will not receive a separate notice when they connect. When the software does so, we check your version of the software and recommend or download updates to your devices. You may not receive notice when we download the update. You may switch off this feature.
- **Network Awareness**. This feature determines whether a system is connected to a network by either passive monitoring of network traffic or active DNS or HTTP queries. The query only transfers standard TCP/IP or DNS information for routing purposes. You can switch off the active query feature through a registry setting.
- **Plug and Play and Plug and Play Extensions**. You may connect new hardware to your device, either directly or over a network. Your device may not have the drivers needed to communicate with that hardware. If so, the update feature of the software can obtain the correct driver from Microsoft and install it on your device. An administrator can disable this update feature.
- **Real Simple Syndication ("RSS") Feed**. This software start page contains updated content that is supplied by means of an RSS feed online from Microsoft.
- **Search Suggestions Service**. When you type a search query in Internet Explorer by using the Instant Search box or by typing a question mark (?) before your search term in the Address bar, you will see search suggestions as you type (if supported by your search provider). Everything you type in the Instant Search box or in the Address bar when preceded by a question mark (?) is sent to your search provider as you type it. In addition, when you press Enter or click the Search button, all the text that is in the search box or Address bar is sent to the search provider. If you use a Microsoft search provider, the information you send is subject to the Microsoft Online Privacy Statement, which is available at go.microsoft.com/fwlink/?linkid=31493. If you use a third-party search provider, use of the information sent will be subject to the third party's privacy practices. You can turn search suggestions off at any time in Internet Explorer by using Manage Add-ons under the Tools button. For more information about the search suggestions service, see go.microsoft.com/fwlink/?linkid=128106.
- **SQL Server Reporting Services Map Report Item.** The software may include features that retrieve content such as maps, images and other data through the Bing Maps (or successor branded)

application programming interface (the "Bing Maps APIs"). The purpose of these features is to create reports displaying data on top of maps, aerial and hybrid imagery. If these features are included, you may use them to create and view dynamic or static documents. This may be done only in conjunction with and through methods and means of access integrated in the software. You may not otherwise copy, store, archive, or create a database of the content available through the Bing Maps APIs. you may not use the following for any purpose even if they are available through the Bing Maps APIs:

- Bing Maps APIs to provide sensor based guidance/routing, or
- Any Road Traffic Data or Bird's Eye Imagery (or associated metadata).

Your use of the Bing Maps APIs and associated content is also subject to the additional terms and conditions at http://www.microsoft.com/maps/product/terms.html.

- **URL Filtering**. The URL Filtering feature identifies certain types of web sites based upon predefined URL categories, and allows you to deny access to such web sites, such as known malicious sites and sites displaying inappropriate or pornographic materials. To apply URL filtering, Microsoft queries the online Microsoft Reputation Service for URL categorization. You can switch off URL filtering. For more information on this feature, see http://go.microsoft.com/fwlink/?LinkId=130980
- **Web Content Features**. Features in the software can retrieve related content from Microsoft and provide it to you. To provide the content, these features send to Microsoft the type of operating system, name and version of the software you are using, type of browser and language code of the device where you run the software. Examples of these features are clip art, templates, online training, online assistance and Appshelp. You may choose not to use these web content features.
- **Windows Media Digital Rights Management**. Content owners use Windows Media digital rights management technology (WMDRM) to protect their intellectual property, including copyrights. This software and third party software use WMDRM to play and copy WMDRM-protected content. If the software fails to protect the content, content owners may ask Microsoft to revoke the software's ability to use WMDRM to play or copy protected content. Revocation does not affect other content. When you download licenses for protected content, you agree that Microsoft may include a revocation list with the licenses. Content owners may require you to upgrade WMDRM to access their content. Microsoft software that includes WMDRM will ask for your consent prior to the upgrade. If you decline an upgrade, you will not be able to access content that requires the upgrade. You may switch off WMDRM features that access the Internet. When these features are off, you can still play content for which you have a valid license.
- **Windows Media Player**. When you use Windows Media Player, it checks with Microsoft for
    - compatible online music services in your region;
    - new versions of the player; and
    - codecs if your device does not have the correct ones for playing content.

  You can switch off this last feature. For more information, go to www.microsoft.com/windows/windowsmedia/player/11/privacy.aspx.
- **Windows Rights Management Services.** The software contains a feature that allows you to create content that cannot be printed, copied or sent to others without your permission. For more information, go to www.microsoft.com/rms. you may choose not to use this feature

- **Windows Time Service**. This service synchronizes with time.windows.com once a week to provide your computer with the correct time. You can turn this feature off or choose your preferred time source within the Date and Time Control Panel applet. The connection uses standard NTP protocol.
- **Windows Update Feature**. You may connect new hardware to the device where you run the software. Your device may not have the drivers needed to communicate with that hardware. If so, the update feature of the software can obtain the correct driver from Microsoft and run it on your device. You can switch off this update feature.

c. **Use of Information.** Microsoft may use the device information, error reports, and malware reports to improve our software and services. We may also share it with others, such as hardware and software vendors. They may use the information to improve how their products run with Microsoft software.

d. **Misuse of Internet-based Services.** You may not use any Internet-based service in any way that could harm it or impair anyone else's use of it. You may not use the service to try to gain unauthorized access to any service, data, account or network by any means.

5. **SCOPE OF LICENSE**. The Licensed Content is licensed, not sold. This agreement only gives you some rights to use the Licensed Content. Microsoft reserves all other rights. Unless applicable law gives you more rights despite this limitation, you may use the Licensed Content only as expressly permitted in this agreement. In doing so, you must comply with any technical limitations in the Licensed Content that only allows you to use it in certain ways. Except as expressly permitted in this agreement, you may not:
   - install more copies of the Licensed Content on devices than the number of licenses you acquired;
   - allow more individuals to access the Licensed Content than the number of licenses you acquired;
   - publicly display, or make the Licensed Content available for others to access or use;
   - install, sell, publish, transmit, encumber, pledge, lend, copy, adapt, link to, post, rent, lease or lend, make available or distribute the Licensed Content to any third party, except as expressly permitted by this Agreement.
   - reverse engineer, decompile, remove or otherwise thwart any protections or disassemble the Licensed Content except and only to the extent that applicable law expressly permits, despite this limitation;
   - access or use any Licensed Content for which you are not providing a training session to End Users using the Licensed Content;
   - access or use any Licensed Content that you have not been authorized by Microsoft to access and use; or
   - transfer the Licensed Content, in whole or in part, or assign this agreement to any third party.

6. **RESERVATION OF RIGHTS AND OWNERSHIP**. Microsoft reserves all rights not expressly granted to you in this agreement. The Licensed Content is protected by copyright and other intellectual property laws and treaties. Microsoft or its suppliers own the title, copyright, and other intellectual property rights in the Licensed Content. You may not remove or obscure any copyright, trademark or patent notices that appear on the Licensed Content or any components thereof, as delivered to you.

7. **EXPORT RESTRICTIONS**. The Licensed Content is subject to United States export laws and regulations. You must comply with all domestic and international export laws and regulations that apply to the Licensed Content. These laws include restrictions on destinations, End Users and end use. For additional information, see www.microsoft.com/exporting.

8. **LIMITATIONS ON SALE, RENTAL, ETC. AND CERTAIN ASSIGNMENTS**. You may not sell, rent, lease, lend or sublicense the Licensed Content or any portion thereof, or transfer or assign this agreement.

9. **SUPPORT SERVICES**. Because the Licensed Content is "as is", we may not provide support services for it.

10. **TERMINATION.** Without prejudice to any other rights, Microsoft may terminate this agreement if you fail to comply with the terms and conditions of this agreement. Upon any termination of this agreement, you agree to immediately stop all use of and to irretrievable delete and destroy all copies of the Licensed Content in your possession or under your control.

11. **LINKS TO THIRD PARTY SITES**. You may link to third party sites through the use of the Licensed Content. The third party sites are not under the control of Microsoft, and Microsoft is not responsible for the contents of any third party sites, any links contained in third party sites, or any changes or updates to third party sites. Microsoft is not responsible for webcasting or any other form of transmission received from any third party sites. Microsoft is providing these links to third party sites to you only as a convenience, and the inclusion of any link does not imply an endorsement by Microsoft of the third party site.

12. **ENTIRE AGREEMENT.** This agreement, and the terms for supplements, updates and support services are the entire agreement for the Licensed Content.

13. **APPLICABLE LAW.**
    a. United States. If you acquired the Licensed Content in the United States, Washington state law governs the interpretation of this agreement and applies to claims for breach of it, regardless of conflict of laws principles. The laws of the state where you live govern all other claims, including claims under state consumer protection laws, unfair competition laws, and in tort.

    b. Outside the United States. If you acquired the Licensed Content in any other country, the laws of that country apply.

14. **LEGAL EFFECT**. This agreement describes certain legal rights. You may have other rights under the laws of your country. You may also have rights with respect to the party from whom you acquired the Licensed Content. This agreement does not change your rights under the laws of your country if the laws of your country do not permit it to do so.

15. **DISCLAIMER OF WARRANTY. THE LICENSED CONTENT IS LICENSED "AS-IS," "WITH ALL FAULTS," AND "AS AVAILABLE." YOU BEAR THE RISK OF USING IT. MICROSOFT CORPORATION AND ITS RESPECTIVE AFFILIATES GIVE NO EXPRESS WARRANTIES, GUARANTEES, OR CONDITIONS UNDER OR IN RELATION TO THE LICENSED CONTENT. YOU MAY HAVE ADDITIONAL CONSUMER RIGHTS UNDER YOUR LOCAL LAWS WHICH THIS AGREEMENT CANNOT CHANGE. TO THE EXTENT PERMITTED UNDER YOUR LOCAL LAWS, MICROSOFT CORPORATION AND ITS RESPECTIVE AFFILIATES EXCLUDE ANY IMPLIED WARRANTIES OR CONDITIONS, INCLUDING THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT.**

16. **LIMITATION ON AND EXCLUSION OF REMEDIES AND DAMAGES. TO THE EXTENT NOT PROHIBITED BY LAW, YOU CAN RECOVER FROM MICROSOFT CORPORATION AND ITS SUPPLIERS ONLY DIRECT DAMAGES UP TO USD$5.00. YOU AGREE NOT TO SEEK TO RECOVER ANY OTHER DAMAGES, INCLUDING CONSEQUENTIAL, LOST PROFITS, SPECIAL, INDIRECT OR INCIDENTAL DAMAGES FROM MICROSOFT CORPORATION AND ITS RESPECTIVE SUPPLIERS.**

This limitation applies to

- anything related to the Licensed Content, services made available through the Licensed Content, or content (including code) on third party Internet sites or third-party programs; and
- claims for breach of contract, breach of warranty, guarantee or condition, strict liability, negligence, or other tort to the extent permitted by applicable law.

It also applies even if Microsoft knew or should have known about the possibility of the damages. The above limitation or exclusion may not apply to you because your country may not allow the exclusion or limitation of incidental, consequential or other damages.

Please note: As this Licensed Content is distributed in Quebec, Canada, some of the clauses in this agreement are provided below in French.

Remarque : Ce le contenu sous licence étant distribué au Québec, Canada, certaines des clauses dans ce contrat sont fournies ci-dessous en français.

EXONÉRATION DE GARANTIE. Le contenu sous licence visé par une licence est offert « tel quel ». Toute utilisation de ce contenu sous licence est à votre seule risque et péril. Microsoft n'accorde aucune autre garantie expresse. Vous pouvez bénéficier de droits additionnels en vertu du droit local sur la protection dues consommateurs, que ce contrat ne peut modifier. La ou elles sont permises par le droit locale, les garanties implicites de qualité marchande, d'adéquation à un usage particulier et d'absence de contrefaçon sont exclues.

LIMITATION DES DOMMAGES-INTÉRÊTS ET EXCLUSION DE RESPONSABILITÉ POUR LES DOMMAGES. Vous pouvez obtenir de Microsoft et de ses fournisseurs une indemnisation en cas de dommages directs uniquement à hauteur de 5,00 $ US. Vous ne pouvez prétendre à aucune indemnisation pour les autres dommages, y compris les dommages spéciaux, indirects ou accessoires et pertes de bénéfices.
Cette limitation concerne:

- tout ce qui est relié au le contenu sous licence , aux services ou au contenu (y compris le code) figurant sur des sites Internet tiers ou dans des programmes tiers ; et
- les réclamations au titre de violation de contrat ou de garantie, ou au titre de responsabilité stricte, de négligence ou d'une autre faute dans la limite autorisée par la loi en vigueur.

Elle s'applique également, même si Microsoft connaissait ou devrait connaître l'éventualité d'un tel dommage. Si votre pays n'autorise pas l'exclusion ou la limitation de responsabilité pour les dommages indirects, accessoires ou de quelque nature que ce soit, il se peut que la limitation ou l'exclusion ci-dessus ne s'appliquera pas à votre égard.

EFFET JURIDIQUE. Le présent contrat décrit certains droits juridiques. Vous pourriez avoir d'autres droits prévus par les lois de votre pays. Le présent contrat ne modifie pas les droits que vous confèrent les lois de votre pays si celles-ci ne le permettent pas.

Revised March 2012

Welcome!

Thank you for taking our training! We've worked together with our Microsoft Certified Partners for Learning Solutions and our Microsoft IT Academies to bring you a world-class learning experience—whether you're a professional looking to advance your skills or a student preparing for a career in IT.

- **Microsoft Certified Trainers and Instructors**—Your instructor is a technical and instructional expert who meets ongoing certification requirements. And, if instructors are delivering training at one of our Certified Partners for Learning Solutions, they are also evaluated throughout the year by students and by Microsoft.

- **Certification Exam Benefits**—After training, consider taking a Microsoft Certification exam. Microsoft Certifications validate your skills on Microsoft technologies and can help differentiate you when finding a job or boosting your career. In fact, independent research by IDC concluded that 75% of managers believe certifications are important to team performance[1]. Ask your instructor about Microsoft Certification exam promotions and discounts that may be available to you.

- **Customer Satisfaction Guarantee**—Our Certified Partners for Learning Solutions offer a satisfaction guarantee and we hold them accountable for it. At the end of class, please complete an evaluation of today's experience. We value your feedback!

We wish you a great learning experience and ongoing success in your career!

Sincerely,

Microsoft Learning
www.microsoft.com/learning

Microsoft | Learning

[1] *IDC,* Value of Certification: Team Certification and Organizational Performance, *November 2006*

Acknowledgments

Microsoft Learning would like to acknowledge and thank the following for their contribution towards developing this title. Their effort at various stages in the development has ensured that you have a good classroom experience.

Design and Development

This course was designed and developed by SolidQ. SolidQ is a global provider of consulting, mentoring and training services for Microsoft Data Management, Business Intelligence and Collaboration platforms.

Greg Low – Lead Developer

Dr Greg Low is a SQL Server MVP, an MCT, and a Microsoft Regional Director for Australia. Greg has worked with SQL Server since version 4.2 as an active mentor, consultant and trainer. Greg describes himself as a SQL Server junkie and also describes himself as having been involved in development since dinosaurs roamed the Earth. He has been an instructor in the Microsoft SQL Server Masters certification program for several years and was one of the first two people to achieve the SQL Server 2008 Master certification. He is the author of a number whitepapers on the Microsoft MSDN and TechNet web sites and the author of a number of SQL Server related books. Greg is based in Melbourne Australia.

Herbert Albert – Course Developer

Herbert Albert started his career in 1994. He works as a trainer, consultant, and author focusing on SQL Server technologies. Herbert is a mentor and the Central European CEO for SolidQ. He is based in Vienna, Austria. He has several Microsoft certifications including MCT which he has held since 1997. Herbert is a regular speaker at conferences and co-author of the SQL Server 2012 Upgrade Technical Reference Guide and SQL Server 2005 Step-by-Step Applied Techniques. Together with Gianluca Hotz, Herbert writes a regular column at the SolidQ Journal.

Mark Hions – Technical Reviewer

Mark's passion for computing and skill as a communicator were well suited to his position as instructor at Honeywell Canada, where he started working with minicomputers, mainframes and mature students in 1984. He first met Microsoft SQL Server when it ran on OS/2, and has delivered training on every version since. An independent MCT and consultant for many years, he is a highly-rated presenter at TechEd, has designed SQL Server exams for Microsoft, and has delivered deep dive courses through the Microsoft Partner Channel. Mark is now the Principal SQL Server Instructor and Consultant at DesTech, which is the largest provider of SQL Server training in the Toronto area.

Chris Barker – Technical Reviewer

Chris Barker is an MCT working in the New Zealand market and currently employed as a staff trainer at Auldhouse, one of New Zealand's major CPLS training centers in Wellington. Chris' background includes programming from the early 1970s—his first program was written in assembly language and debugged in binary (literally)! While focusing training on programming (mostly .NET) and databases (mostly Microsoft SQL Server) Chris has also been an infrastructure trainer and has both Novell and Microsoft networking qualifications.

Contents

Module 1: Introduction to SQL Server 2012 and Its Toolset

Lesson 1: Introduction to the SQL Server Platform 1-3
Lesson 2: Working with SQL Server Tools 1-14
Lesson 3: Configuring SQL Server Services 1-26
Lab 1: Introduction to SQL Server and Its Toolset 1-36

Module 2: Preparing Systems for SQL Server 2012

Lesson 1: Overview of SQL Server Architecture 2-3
Lesson 2: Planning Server Resource Requirements 2-17
Lesson 3: Pre-installation Testing for SQL Server 2-29
Lab 2: Preparing Systems for SQL Server 2-35

Module 3: Installing and Configuring SQL Server 2012

Lesson 1: Preparing to Install SQL Server 3-3
Lesson 2: Installing SQL Server 3-16
Lesson 3: Upgrading and Automating Installation 3-24
Lab 3: Installing and Configuring SQL Server 3-32

Module 4: Working with Databases

Lesson 1: Overview of SQL Server Databases 4-3
Lesson 2: Working with Files and Filegroups 4-15
Lesson 3: Moving Database Files 4-29
Lab 4: Working with Databases 4-39

Module 5: Understanding SQL Server 2012 Recovery Models

Lesson 1: Backup Strategies 5-3
Lesson 2: Understanding SQL Server Transaction Logging 5-12
Lesson 3: Planning a SQL Server Backup Strategy 5-22
Lab 5: Understanding SQL Server Recovery Models 5-32

Module 6: Backup of SQL Server 2012 Databases

Lesson 1: Backing up Databases and Transaction Logs 6-3
Lesson 2: Managing Database Backups 6-14
Lesson 3: Working with Backup Options 6-20
Lab 6: Backup of SQL Server Databases 6-26

Module 7: Restoring SQL Server 2012 Databases

Lesson 1: Understanding the Restore Process 7-3
Lesson 2: Restoring Databases 7-8
Lesson 3: Working with Point-in-time recovery 7-19
Lesson 4: Restoring System Databases and Individual Files 7-27
Lab 7: Restoring SQL Server 2012 Databases 7-34

Module 8: Importing and Exporting Data

Lesson 1: Transferring Data To/From SQL Server 8-3
Lesson 2: Importing & Exporting Table Data 8-15
Lesson 3: Inserting Data in Bulk 8-20
Lab 8: Importing and Exporting Data 8-29

Module 9: Authenticating and Authorizing Users

Lesson 1: Authenticating Connections to SQL Server 9-3
Lesson 2: Authorizing Logins to Access Databases 9-13
Lesson 3: Authorization Across Servers 9-22
Lab 9: Authenticating and Authorizing Users 9-30

Module 10: Assigning Server and Database Roles

Lesson 1: Working with Server Roles 10-3
Lesson 2: Working with Fixed Database Roles 10-12
Lesson 3: Creating User-defined Database Roles 10-18
Lab 10: Assigning Server and Database Roles 10-26

Module 11: Authorizing Users to Access Resources

Lesson 1: Authorizing User Access to Objects 11-3
Lesson 2: Authorizing Users to Execute Code 11-12
Lesson 3: Configuring Permissions at the Schema Level 11-21
Lab 11: Authorizing Users to Access Resources 11-28

Module 12: Auditing SQL Server Environments

Lesson 1: Options for Auditing Data Access in SQL 12-3
Lesson 2: Implementing SQL Server Audit 12-12
Lesson 3: Managing SQL Server Audit 12-26
Lab 12: Auditing SQL Server Environments 12-31

Module 13: Automating SQL Server 2012 Management

Lesson 1: Automating SQL Server Management 13-3
Lesson 2: Working with SQL Server Agent 13-11
Lesson 3: Managing SQL Server Agent Jobs 13-19
Lab 13: Automating SQL Server Management 13-26

Module 14: Configuring Security for SQL Server Agent

Lesson 1: Understanding SQL Server Agent Security **14-3**
Lesson 2: Configuring Credentials **14-13**
Lesson 3: Configuring Proxy Accounts **14-18**
Lab 14: Configuring Security for SQL Server Agent **14-24**

Module 15: Monitoring SQL Server 2012 with Alerts and Notifications

Lesson 1: Configuration of Database Mail **15-3**
Lesson 2: Monitoring SQL Server Errors **15-11**
Lesson 3: Configuring Operators, Alerts and Notifications **15-18**
Lab 15: Monitoring SQL Agent Jobs with Alerts and Notifications **15-30**

Module 16: Performing Ongoing Database Maintenance

Lesson 1: Ensuring Database Integrity **16-3**
Lesson 2: Maintaining Indexes **16-12**
Lesson 3: Automating Routine Database Maintenance **16-26**
Lab 16: Performing Ongoing Database Maintenance **16-30**

Module 17: Tracing Access to SQL Server 2012

Lesson 1: Capturing Activity using SQL Server Profiler and Extended Events Profiler **17-3**
Lesson 2: Improving Performance with the Database Engine Tuning Advisor **17-17**
Lesson 3: Working with Tracing Options **17-25**
Lab 17: Tracing Access to SQL Server 2012 **17-36**

Module 18: Monitoring SQL Server 2012

Lesson 1: Monitoring Activity **18-3**
Lesson 2: Capturing and Managing Performance Data **18-15**
Lesson 3: Analyzing Collected Performance Data **18-23**
Lab 18: Monitoring SQL Server 2012 **18-32**

Module 19: Managing Multiple Servers

Lesson 1: Working with Multiple Servers **19-3**
Lesson 2: Virtualizing SQL Server **19-9**
Lesson 3: Deploying and Upgrading Data-tier Applications **19-15**
Lab 19: Managing Multiple Servers **19-22**

Module 20: Troubleshooting Common SQL Server 2012 Administrative Issues

Lesson 1: SQL Server Troubleshooting Methodology **20-3**
Lesson 2: Resolving Service-related Issues **20-7**
Lesson 3: Resolving Login and Connectivity Issues **20-13**
Lesson 4: Resolving Concurrency Issues **20-17**
Lab 20: Troubleshooting Common Issues **20-25**

Appendix A: Core Concepts in SQL Server High Availability and Replication

Lesson 1: Core Concepts in High Availability **A-3**
Lesson 2: Core Concepts in Replication **A-11**

Appendix: Lab Answer Keys

Module 1 Lab: Introduction to SQL Server and Its Toolset **L1-1**
Module 2 Lab: Preparing Systems for SQL Server **L2-5**
Module 3 Lab: Installing and Configuring SQL Server **L3-11**
Module 4 Lab: Working with Databases **L4-17**
Module 5 Lab: Understanding SQL Server Recovery Models **L5-23**
Module 6 Lab: Backup of SQL Server Databases **L6-27**
Module 7 Lab: Restoring SQL Server 2012 Databases **L7-31**
Module 8 Lab: Importing and Exporting Data **L8-35**
Module 9 Lab: Authenticating and Authorizing Users **L9-39**
Module 10 Lab: Assigning Server and Database Roles **L10-41**
Module 11 Lab: Authorizing Users to Access Resources **L11-43**
Module 12 Lab: Auditing SQL Server Environments **L12-45**
Module 13 Lab: Automating SQL Server Management **L13-49**
Module 14 Lab: Configuring Security for SQL Server Agent **L14-53**
Module 15 Lab: Monitoring SQL Server 2012 with Alerts and Notifications **L15-57**
Module 16 Lab: Performing Ongoing Database Maintenance **L16-63**
Module 17 Lab: Tracing Access to SQL Server **L17-67**
Module 18 Lab: Monitoring SQL Server 2012 **L18-71**
Module 19 Lab: Managing Multiple Servers **L19-75**
Module 20 Lab: Troubleshooting Common Issues **L20-79**

Module 12

Auditing SQL Server Environments

Contents:

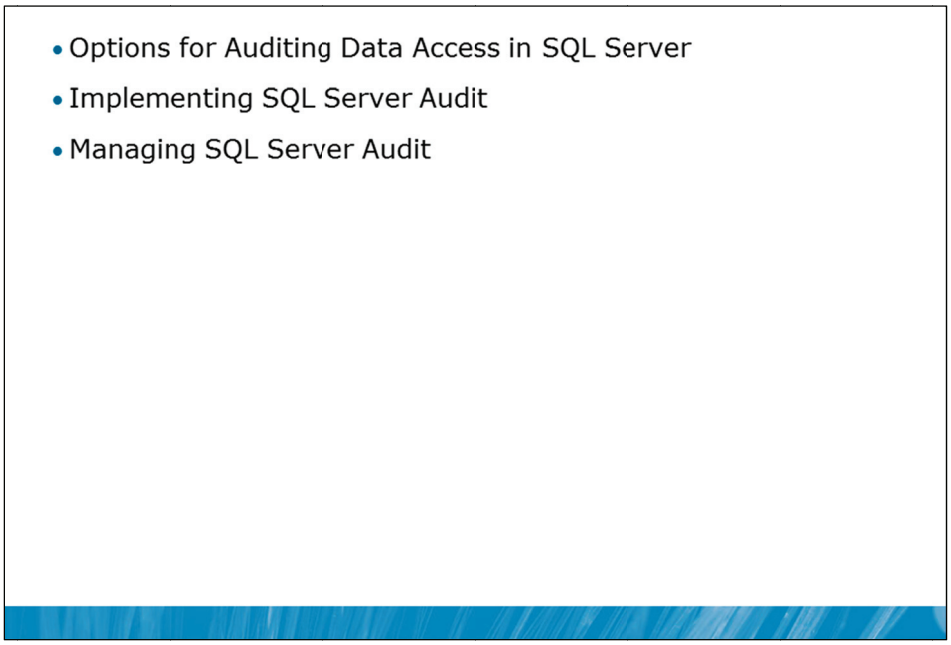
Lesson 1: Options for Auditing Data Access in SQL **12-3**

Lesson 2: Implementing SQL Server Audit **12-12**

Lesson 3: Managing SQL Server Audit **12-26**

Lab 12: Auditing SQL Server Environments **12-31**

Module Overview



One of the most important aspects of configuring security for Microsoft® SQL Server® systems is ensuring that auditing and compliance requirements are met. Organizations may need to meet a variety of compliance goals. Choosing the appropriate configuration for SQL Server will often be a key component in meeting those goals.

SQL Server 2008 introduced the SQL Server Audit feature in the Enterprise edition. (Some audit features are part of all editions of SQL Server 2012). The audit feature provides that ability to perform types of auditing that were not possible in earlier versions of the product. In this module, you will see the available options for auditing within SQL Server, see how to implement the SQL Server Audit feature and learn to manage that feature.

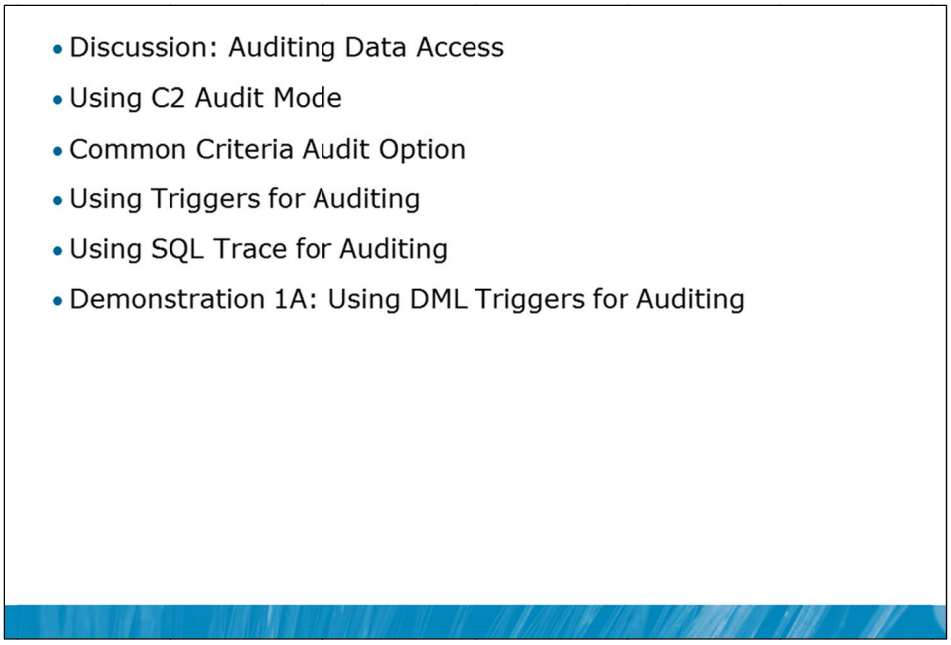
Objectives

After completing this lesson, you will be able to:

- Describe the options for auditing data access in SQL Server.
- Implement SQL Server Audit.
- Manage SQL Server Audit.

Lesson 1

Options for Auditing Data Access in SQL Server



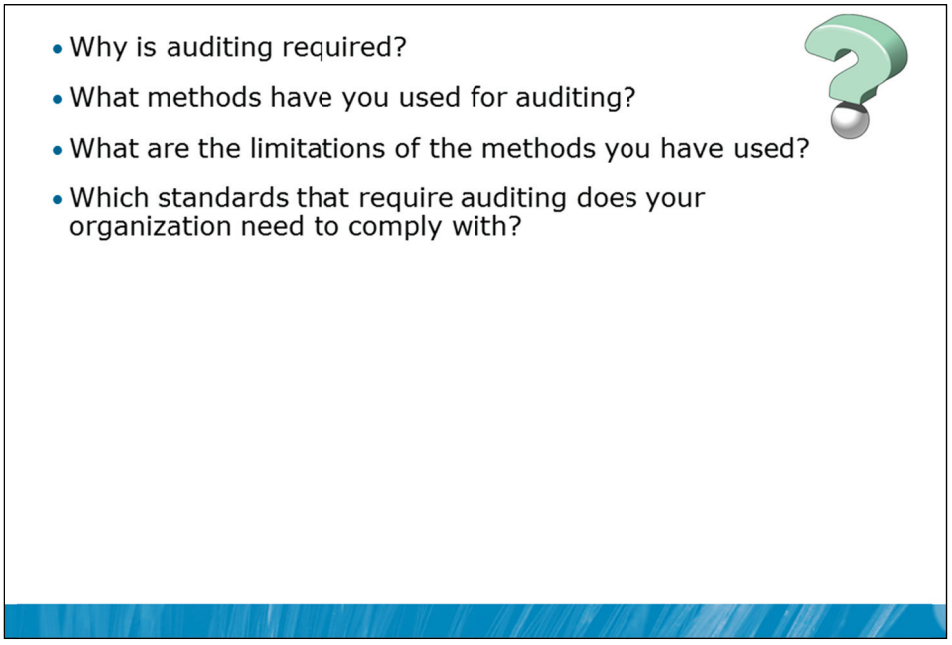
Prior to SQL Server 2008, a variety of auditing options were available in SQL Server. These options are still available in SQL Server 2012 and may form part of your auditing strategy. In general, no one feature provides all possible auditing requirements and a combination of features often needs to be used. In this lesson, you will learn about the options available.

Objectives

After completing this lesson, you will be able to:

- Describe the need for auditing.
- Use C2 audit mode.
- Describe the Common Criteria Audit Option.
- Configure triggers for auditing.
- Describe the use of SQL trace for auditing.

Discussion: Auditing Data Access



Key Points

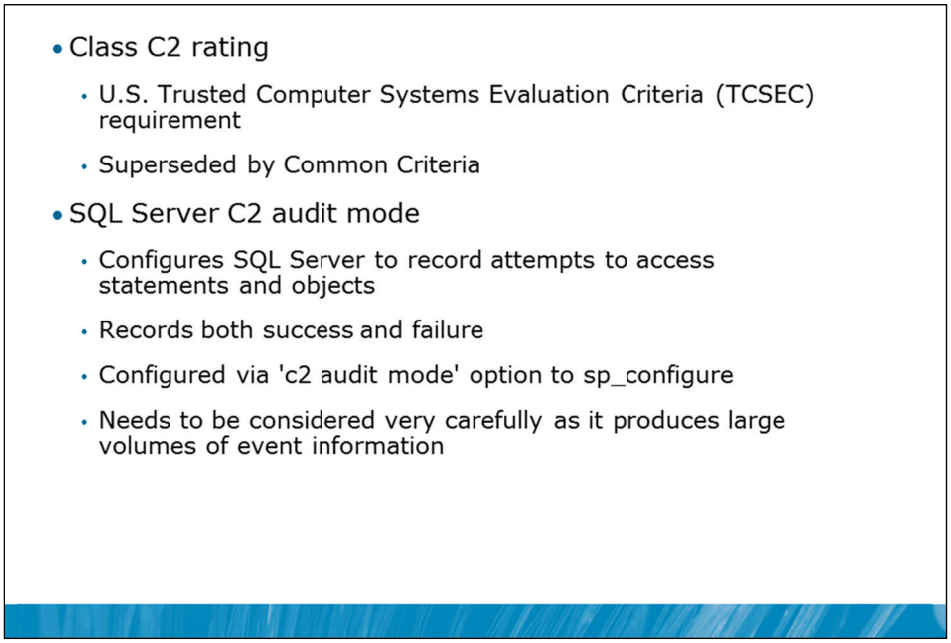
Question: Why is auditing required?

Question: What methods have you used for auditing?

Question: What are the limitations of the methods you have used?

Question: Which standards that require auditing does your organization need to comply with?

Using C2 Audit Mode



Key Points

C2 refers to a set of security policies that define how a secure system operates. Certification applies to a particular installation, including the hardware, software, and the environment that the system operates in. Products do not become C2 certified. Sites become C2 certified.

C2 Certification

The Windows NT® operating system (server and workstation) were first listed on the National Security Agency (NSA) Evaluated Products List (EPL) in 1995. This means that it was acknowledged that those products could be configured in a way that would enable sites using them to become certified. Windows systems have a long history of being able to be configured in a compliant manner.

C2 is one of a series of security ratings, involving A, B, C, and D level secure products. These ratings were published by the National Computer Security Center (NCSC) in a document called the Trusted Computer System Evaluation Criteria (TCSEC). This document was commonly referred to in the industry as the "orange book".

Note As a curious side-note, the "orange book" was part of the "rainbow series".

The security policy in C2 is known as Discretionary Access Control. Users of the system:

- Own objects.
- Have control over the protection of the objects they own.
- Are accountable for all their access-related actions.

By today's standards, the C2 requirements are relatively easy for sites to attain.

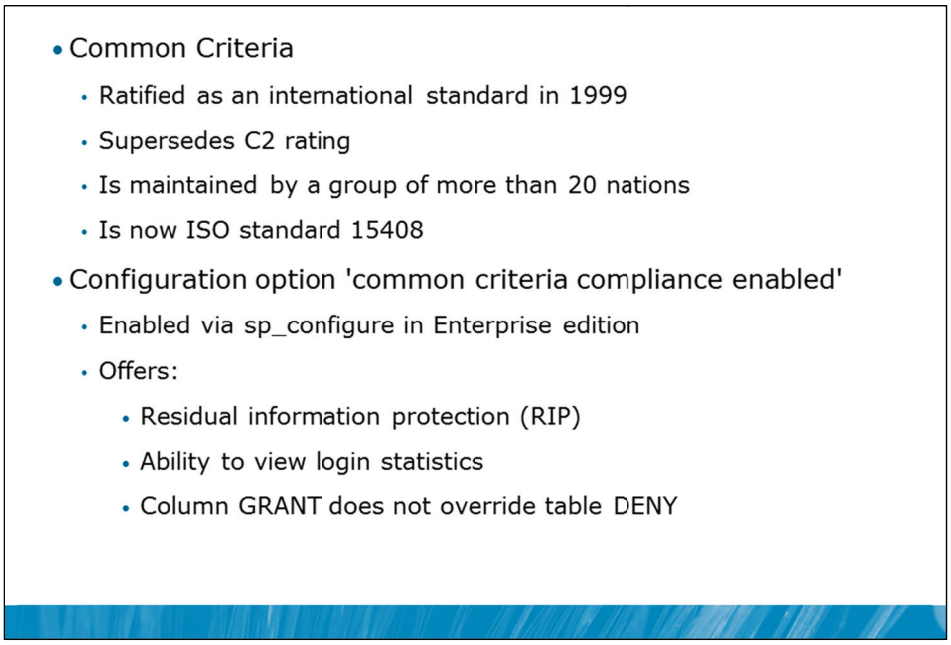
SQL Server and C2

SQL Server can be configured to meet C2 requirements. A system configuration option 'c2 audit mode' can be enabled using the sp_configure system stored procedure. While easy to configure, this option is now rarely used or even appropriate to use. Enabling the C2 audit option can have a negative performance impact on the server through the generation of large volumes of event information.

Most customers that configure this option do so without realizing what they are enabling and eventually (sometimes sooner rather than later) end up running out of disk space.

For practical purposes, C2 has now been superseded by Common Criteria compliance, which is described in the next topic.

Common Criteria Audit Option



Key Points

C2 ratings were U.S. based. Common Criteria is an international standard that was ratified by more than twenty nations in 1999 and has superseded C2 rating as a requirement in most standards.

It is also maintained on an ongoing basis by over twenty countries and was adopted by the International Standards Organization (ISO) as standard 15408.

'common criteria compliance enabled' Option

SQL Server provides a server option 'common criteria compliance enabled' that can be set using the sp_configure system stored procedure. It is available in the Enterprise edition for production use. (It is also available in the Developer and Evaluation editions for non-production use). In addition to enabling the common criteria compliance enabled option, you also must download and run a script that finishes configuring SQL Server to comply with Common Criteria Evaluation Assurance Level 4+ (EAL4+). You can download this script from the Microsoft SQL Server Common Criteria Web site.

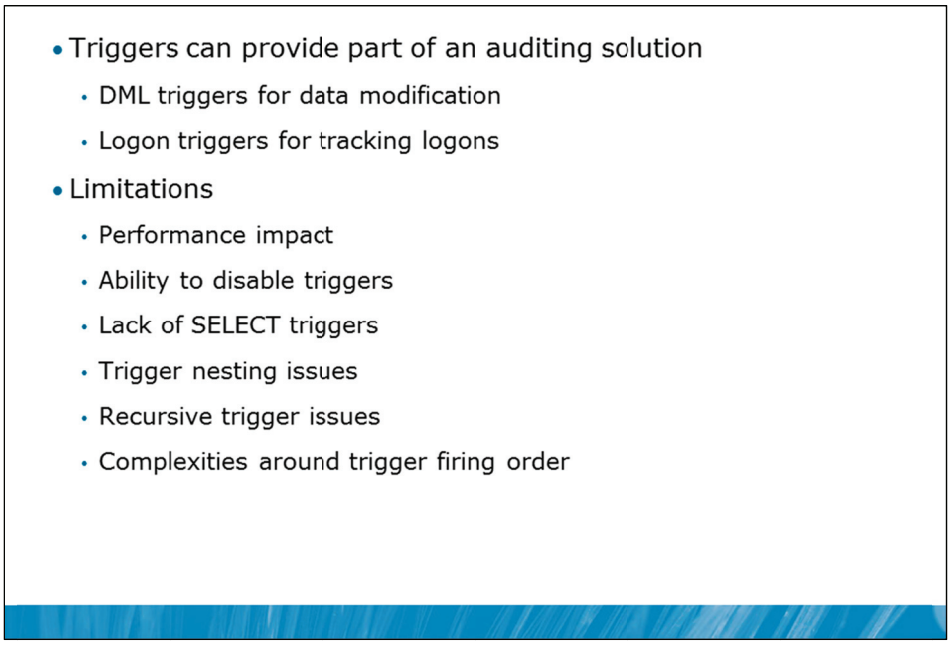
When this option is enabled, three changes occur to how SQL Server operates:

Issue	Description
Residual Information Protection (RIP)	Memory is always overwritten with a known bit pattern before being reused
Ability to view login statistics	Auditing of logins is automatically enabled
Column GRANT does not override table DENY	Changes the behavior of the permission system

The implementation of RIP increases security but will negatively impact on the performance of the system.

Question: Why is there a need to make a change for GRANT as the column-level overriding DENY at the table-level?

Using Triggers for Auditing



Key Points

Triggers can play an important role in auditing. Prior to SQL Server 2008, many actions could only be audited via triggers.

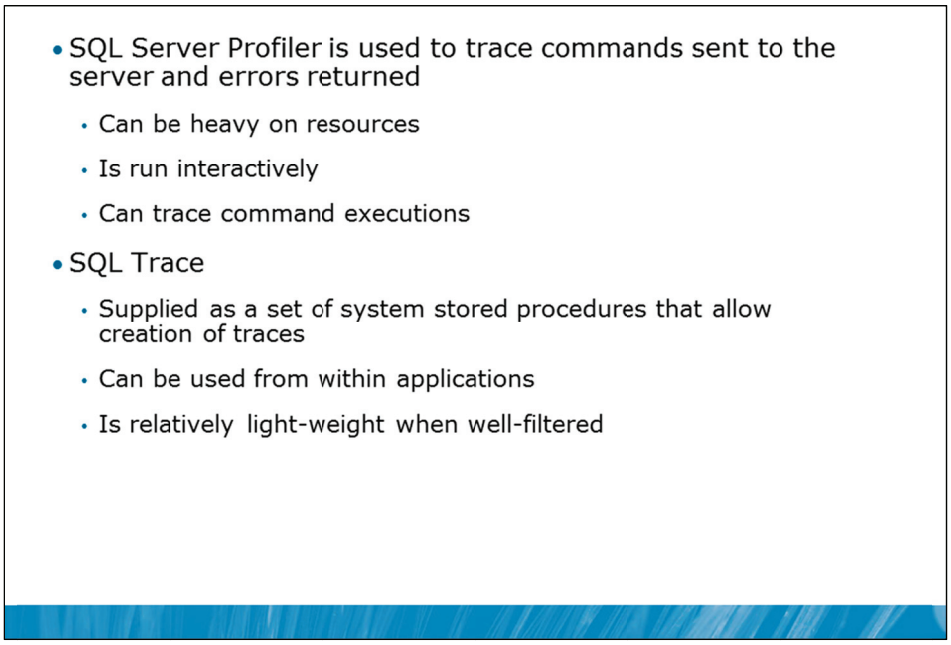
SQL Server 2005 SP2 introduced logon triggers. These allowed tracking more details of logons and also allowed rolling back logons based on business or administrative logic.

Triggers are not perfect though:

- Performance of the system is impacted by triggers. Prior to SQL Server 2005, the inserted and deleted virtual tables in triggers were implemented in a similar manner to views over the transaction log. They did not offer high performance. From SQL Server 2005 onwards, the internal structure of these internal tables changed. They are now based on a row version table that resides in the tempdb database. Triggers that use these tables operate much more quickly than in prior versions but there can be a significant impact on the performance of the tempdb database that needs to be considered.
- Triggers can be disabled. This is a significant issue for auditing.
- Users were requesting SELECT triggers. They did not only want to track data modifications. Many users in high security environments wanted to see not only the commands that were executed to retrieve data but also the data that was retrieved.
- Triggers have a nesting limit of thirty-two levels beyond which they do not work.
- Recursive triggers can be disabled.
- Only limited ability to control trigger firing order is provided. Auditing would normally need to be the last trigger that fires to make sure that it captures all the changes made by other triggers.

Question: What would you imagine that the term "recursive trigger" might refer to?

Using SQL Trace for Auditing



Key Points

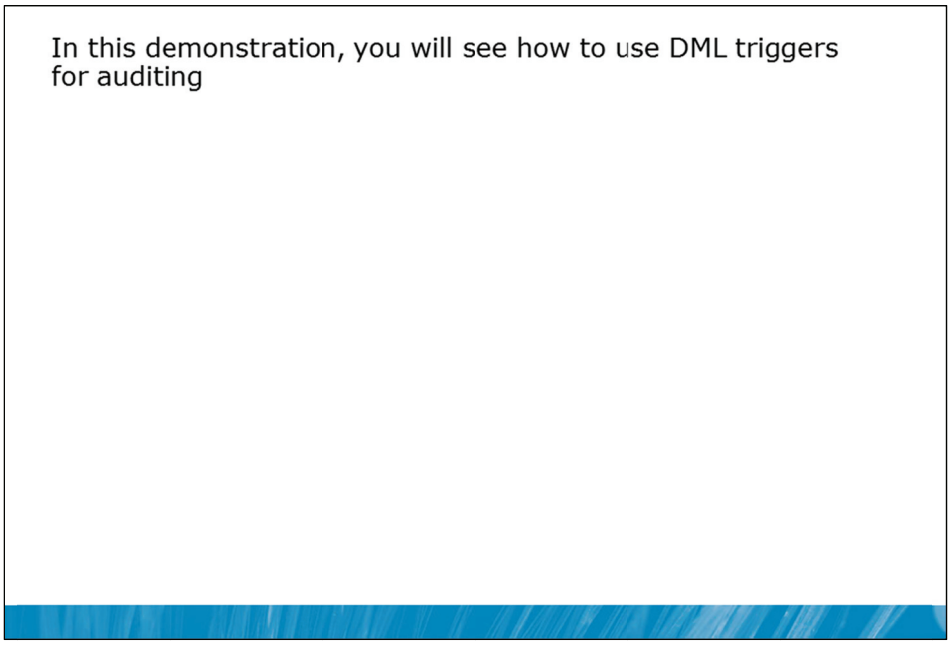
Many users have attempted to use SQL Server Profiler for auditing because it allows tracing commands that are sent to SQL Server and tracing the errors that are returned. SQL Server Profiler can have a significantly negative performance impact when it is run interactively on production systems.

SQL Trace is a set of system stored procedures that are utilized by SQL Server Profiler. Executing these procedures to manage tracing offers a much more lightweight method of tracing, particularly when the events are well-filtered.

SQL Trace can then have a role in auditing. Because it has the ability to capture commands that are sent to the server, it can be used to audit those commands.

Since SQL Server 2005, SQL Trace uses a server-side tracing mechanism that guarantees that no events are lost, as long as there is space available on the disk and that no write errors occur. If the disk fills or write errors occur, the trace stops. SQL Server continues unless c2 audit mode is also enabled. The possibility of missing events needs to be considered when evaluating the use of SQL Trace for auditing purposes.

Demonstration 1A: Using DML Triggers for Auditing

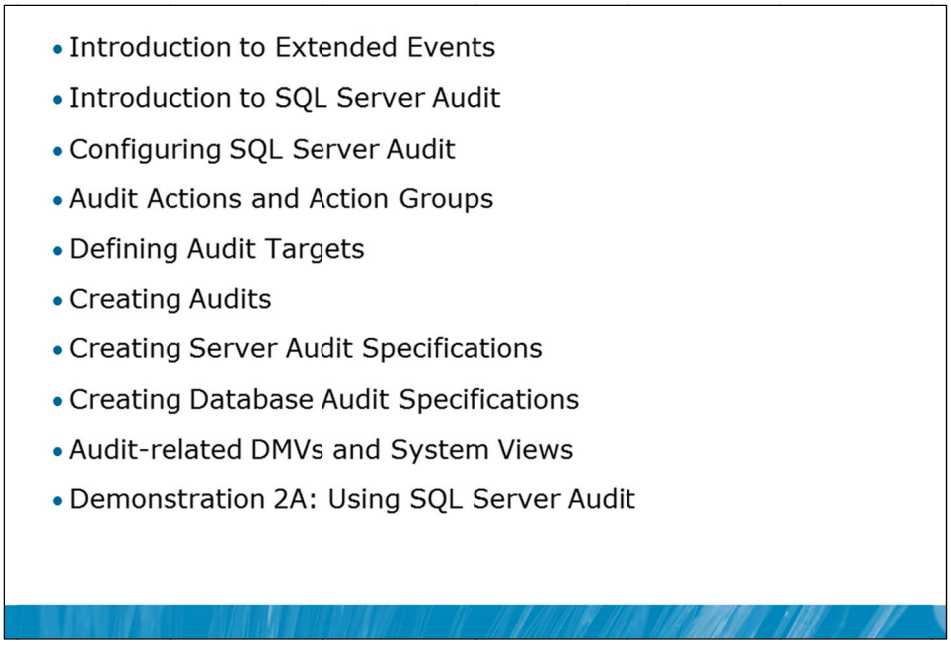


Demonstration Steps

1. Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
2. In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, click **SQL Server Management Studio**. In the Connect to Server window, type **Proseware** and click **Connect**. From the **File** menu, click **Open**, click **Project/Solution**, navigate to **D:\10775A_Labs\10775A_12_PRJ\10775A_12_PRJ.ssmssln** and click **Open**.
3. From the **View** menu, click **Solution Explorer**. Open and execute the **00 – Setup.sql** script file from within Solution Explorer.
4. Open the **11 – Demonstration 1A.sql** script file.
5. Follow the instructions contained within the comments of the script file.

   Question: Can DML triggers be used to audit the reading of data in a table?

Lesson 2
Implementing SQL Server Audit



SQL Server 2008 introduced the SQL Server Audit feature. It was based on a new eventing engine called Extended Events. In this lesson, you will learn about the core terminology used by Extended Events and how SQL Server Audit has been created as a specific package within Extended Events.

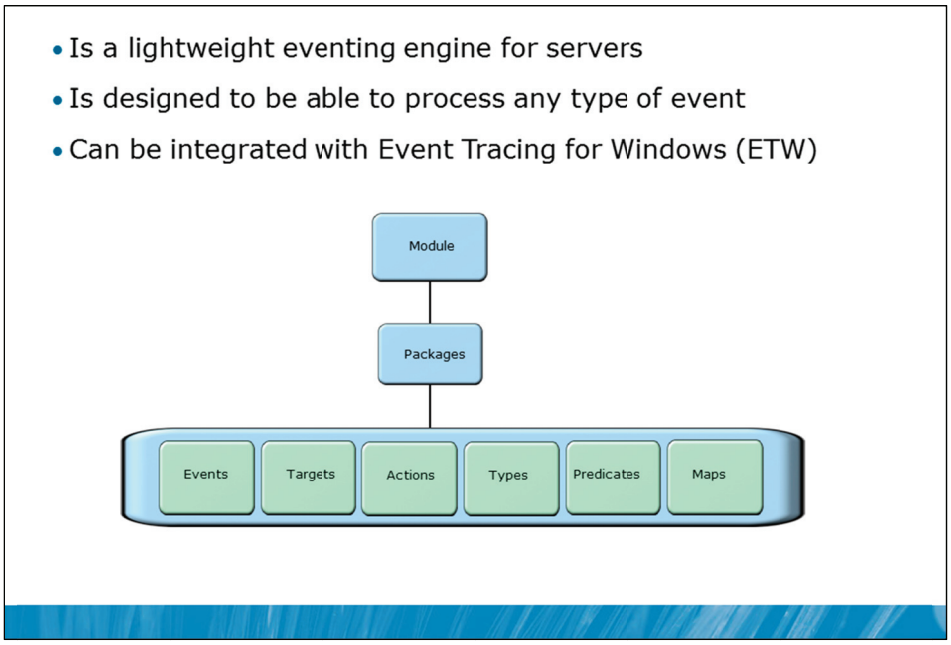
Preparing SQL Server Audit for use requires the configuration of a number of objects. In this lesson, each of these objects is introduced, along with details of how they are configured. Finally, you will see the dynamic management views (DMVs) and system views that have been introduced to support the SQL Server Audit feature.

Objectives

After completing this lesson, you will be able to:

- Describe the Extended Events infrastructure.
- Describe SQL Server Audit.
- Configure SQL Server Audit.
- Detail the roles of audit actions and action groups.
- Define audit targets.
- Create audits.
- Create server audit specifications.
- Create database audit specifications.
- Use audit-related DMVs and system views.

Introduction to Extended Events



Key Points

A wide variety of events occurs within the SQL Server database engine. For example, a user could execute a query, the database engine could need to request additional memory, or permissions could be checked. SQL Server 2008 introduced a new feature called Extended Events that allows you to define actions that should be taken when events occur. As SQL Server executes its internal code, it checks to see if an external user has defined an action that should be taken at that point in the code. If an action is defined, an event is fired and details of the event are sent to a target location. Targets can be operating system files, memory-based ring buffers or Windows event logs.

Extended Events is considered to be a lightweight eventing engine as it has very little performance impact on the database engine that it is monitoring. Extended events can be used for many purposes that SQL Trace is currently used for.

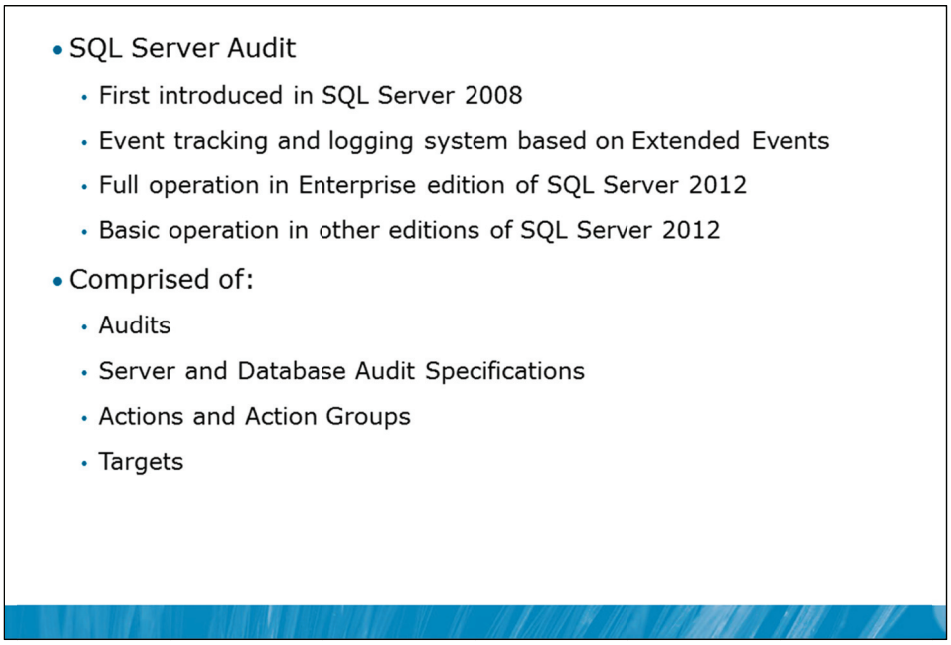
Extended Events are important as SQL Server Audit is based on the extended events infrastructure. The eventing engine that is provided by Extended Events is not tied to particular types of events. The engine is written in such a way that it can process any type of event.

Configurations of Extended Events are shipped in .exe or .dll files that are called "packages". Packages are the unit of deployment and installation for Extended Events. A package is a container for all objects that are part of a particular Extended Events configuration. SQL Server Audit is a special package within Extended Events. You cannot change how it is internally configured. You can change other packages.

Extended Events uses specific terminology for describing the objects that it uses:

Object	Description
Events	Points of interest during the execution of code
Targets	Places that the trace details are sent to (such as files)
Actions	Responses that can be made to events (for example, one type of action captures execution plans to include in the trace)
Types	Definitions of the objects that Extended Events works with
Predicates	Dynamic filters that are applied to the event capture
Maps	Mapping of values to strings. (An example would be the mapping of codes to descriptions)

Introduction to SQL Server Audit



Key Points

SQL Server Audit was introduced in SQL Server 2008 to address compliance issues. The Enterprise edition of SQL Server 2012 provides full functionality and other editions provide basic functionality and are limited to defining server audit specifications.

SQL Server Audit

It is important to be aware that SQL Server Audit is the name of the feature and the name of one of the objects that are part of the feature.

An audit is a definition of where the results of the auditing process are sent. This might seem counter-intuitive at first, given that the name sounds like an action that you perform, not a location for the results of the action. An audit is created at the instance level and multiple audits can be created per instance.

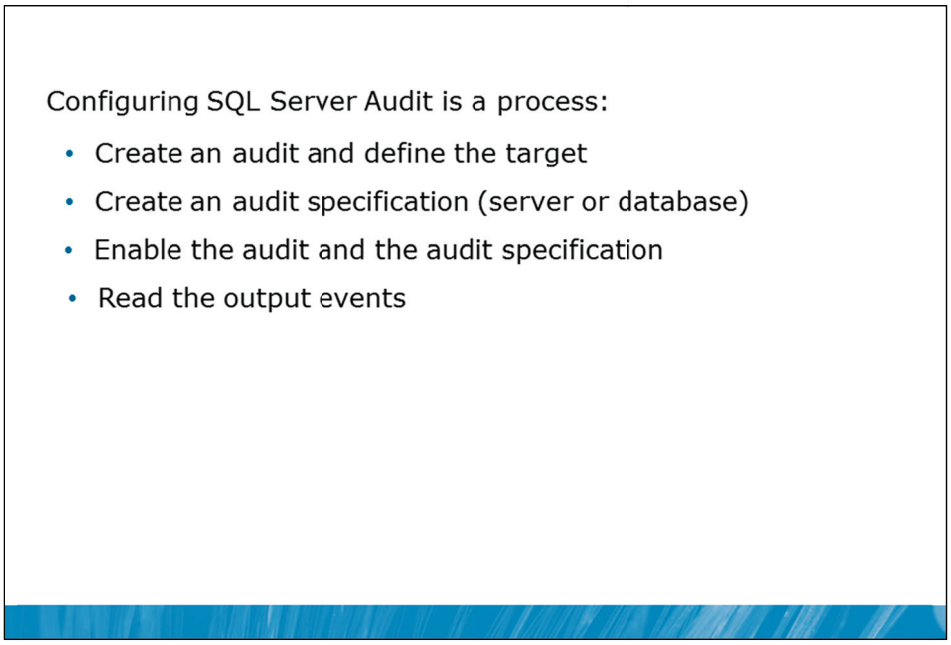
The results of an audit are sent to a target.

Note The term "target" has the same meaning for SQL Server Audit as it does for Extended Events.

Audit Specifications

Server and database audit specifications determine the actions to audit. There are predefined sets of actions called "action groups". The use of these action groups avoids the need to configure large number of individual audit actions.

Configuring SQL Server Audit



Key Points

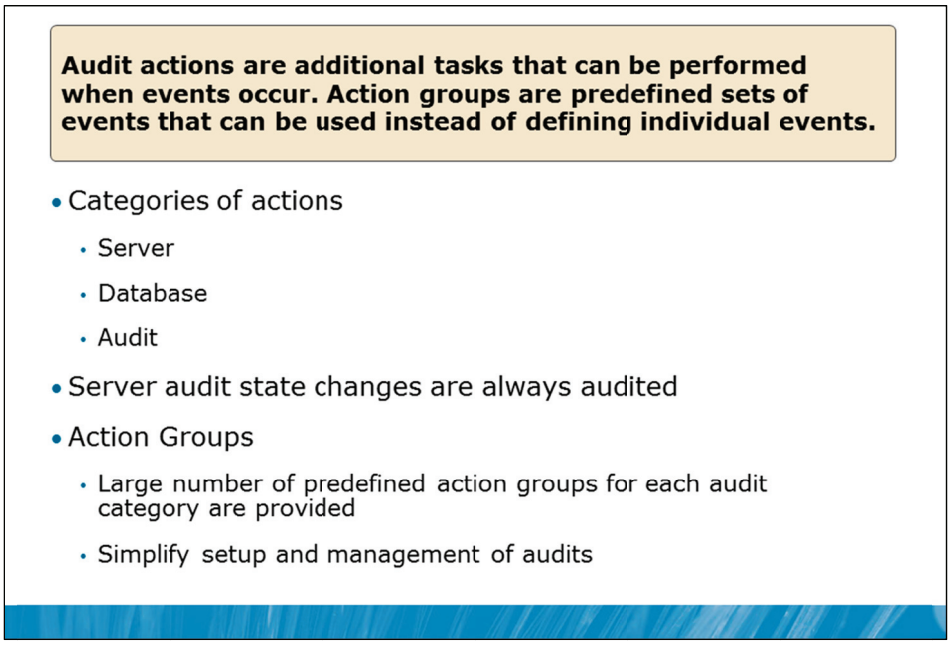
Configuring SQL Server Audit is a multi-step process:

Step	Description
Creating an audit	Determines how the results will be processed. For example, when configuring an audit, you will decide what to do if the disk space runs out. You will also decide how long SQL Server can buffer audit results before writing them to the target.
Defining the target	Determines where the output will be sent.
Creating an audit specification	Determines the actions to be audited. These actions can be at the server or database level.
Enabling the audit and audit specification	Is the step where the objects are enabled. (Audits are created in a disabled state and audit specifications are created in a disabled state by default).
Read the output events	Relates to extracting the output details from the audits.

Several options exist for reading the output events after they are captured:

- Windows event/log file viewers allow reading event log details
- The sys.fn_get_audit_file function returns file-based output as a table that can be queried in T-SQL.

Audit Actions and Action Groups



Key Points

Actions are the events that occur that are of interest to the audit. Actions can occur at three levels: server, database, and audit.

Action Groups

To avoid the need for many individual actions, action groups are provided. This makes setup and management of audits easier as it avoids the need to set up large numbers of individual actions for auditing.

Examples of action groups are:

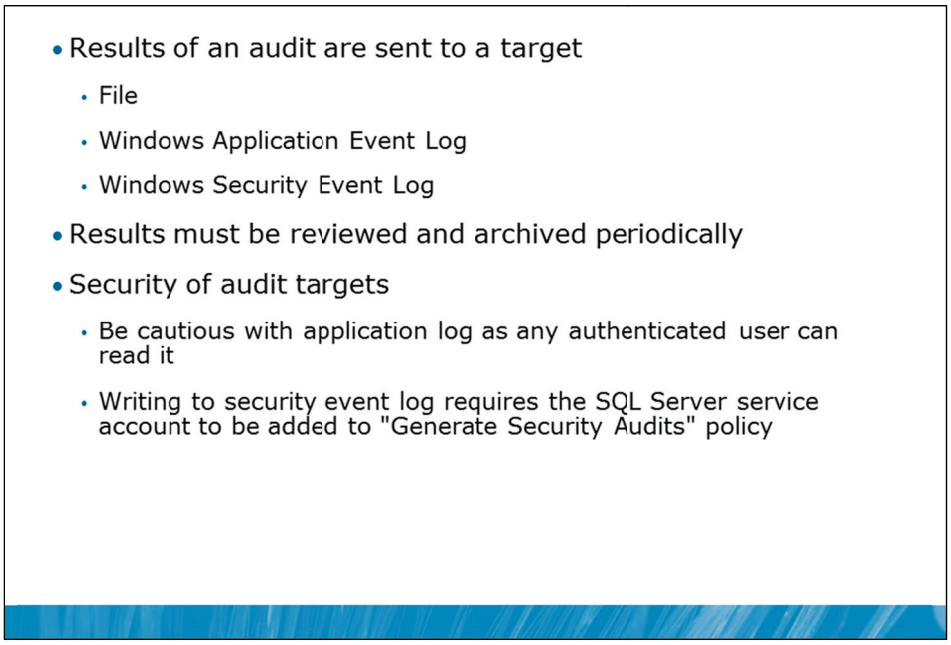
- BACKUP_RESTORE_GROUP
- DATABASE_MIRRORING_LOGIN_GROUP
- DATABASE_OBJECT_ACCESS_GROUP
- DBCC_GROUP
- FAILED_LOGIN_GROUP
- LOGIN_CHANGE_PASSWORD_GROUP

Note that a state change of any audit is always audited. This cannot be disabled.

SQL Server 2012 introduced a new group called USER_DEFINED_AUDIT_GROUP. Applications can cause audit events to be written to that group by calling the sp_audit_write system stored procedure.

Question: Why would no option exist for disabling the auditing of audit changes?

Defining Audit Targets



Key Points

Audits can be sent to three targets in the current version.

- Results can be sent to a file. File output provides the highest performance and is the easiest option to configure.
- Results can be sent to the Windows application event log. Avoid sending too much detail to this log as network administrators tend to dislike applications that write too much content to any of the event logs.

Note Be cautious about using the Windows application event log as an output target for sensitive information as any authenticated user can read the contents of that log.

- Results can be sent to the Windows security event log. The security event log is a secure output option but requires the SQL Server service account to be added to the "Generate Security Audits" policy before it can be used.

Note If it is important for SQL Server administrators to have access to the contents of the audit, consider whether the use of the security event log is appropriate.

Question: Why would many SQL Server DBAs have difficulty working with audit entries in the Windows security event log?

Creating Audits

- Creating an audit requires a number of configurations:

Configuration	Comment
Audit name	Name for the audit
Queue delay (in milliseconds)	Amount in time before audit actions must be processed
Shut down server on audit failure	Indicates that SQL Server cannot continue if audit is not working
Audit destination	Audit Target
Maximum rollover files	Maximum number of files to retain (only for files)
Maximum file size (MB)	Maximum size of each audit file
Reserve disk space	Indicates whether disk space for the audit files should be reserved in advance
Maximum files	Caps the number of audit files

Key Points

When you create an audit, you make decisions about how SQL Server will process the results that are sent to the audit target. Audits can be created using the GUI in SSMS or via the CREATE SERVER AUDIT command in T-SQL.

Audit Configuration Options

The name of an audit will often relate to details of what the audit will contain, or the date and time when the audit was created or a combination of both.

After configuring a name, configuring a queue delay is particularly important. The queue delay indicates (in milliseconds) how long SQL Server can buffer the audit results before flushing them to the target.

Note The value chosen for queue delay is a trade-off between security and performance. If a server failure occurs, results that are in the buffer and not yet flushed to the target can be lost. A value of zero for queue delay will cause synchronous writes as events occur. This avoids the chance of losing events on failure but can impact performance significantly.

For serious production auditing, the option to shut down the server on audit failure should be selected. SQL Server 2012 introduced a new option to fail the operation that fired the audit, rather than shutting down the entire server instance.

Note If the shutdown option is chosen, SQL Server may fail to initiate if auditing cannot function. In the next lesson, you will see how to deal with this situation.

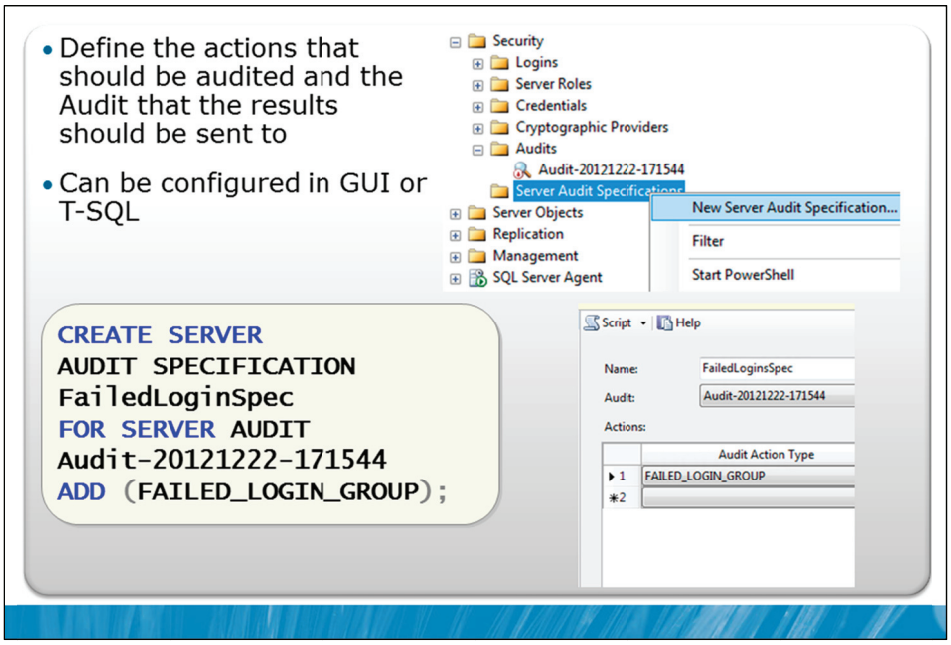
You need to choose a target for the output of your audit. The available audit targets were discussed in a previous topic.

Note On Windows® XP, the security event log is not available as a destination.

Note Each audit can be the target of at most one server audit specification and one database audit specification.

Question: Why is it recommended to select the option to shut down server on audit failure?

Creating Server Audit Specifications



Key Points

Creating a server audit specification can be performed with the GUI or T-SQL. Server audit specifications are created in a disabled state by default. Audit objects, including audit specifications, are usually left disabled until all audit objects have been created.

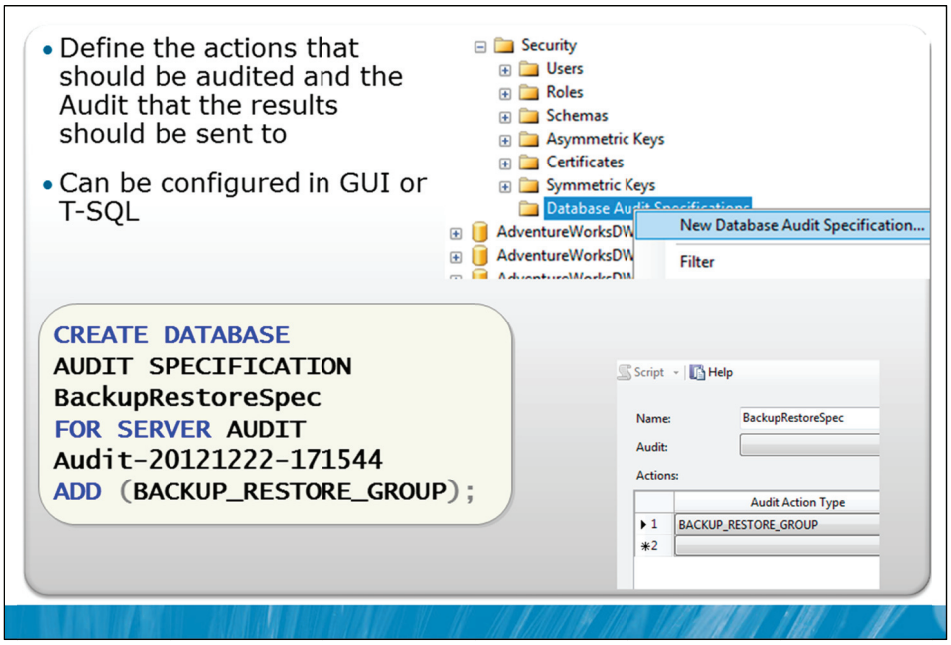
Server Audit Specification

A server audit specification details the actions to be audited. You can choose either action groups or individual actions and objects. In the example shown in the slide, a server audit specification is being created using the CREATE SERVER AUDIT SPECIFICATION statement. The configuration of the same specification using the GUI is also shown.

The name of the specification is FailedLoginSpec and the data collected from the specification will be sent to the Audit-20121222-171544 audit. The action group to be audited is the FAILED_LOGIN_GROUP.

Question: Why would enabling logging of failed logins have potential risks to availability?

Creating Database Audit Specifications



Key Points

Creating a database audit specification can also be performed with the GUI or T-SQL. Database audit specifications are also created in a disabled state by default. Database audit specifications can only be created in Enterprise edition.

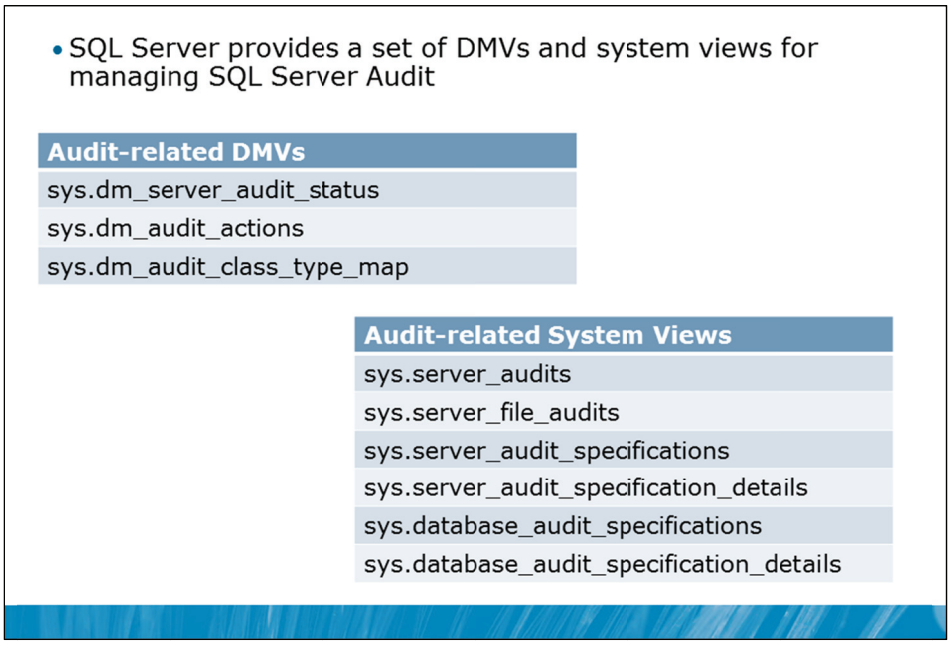
Database Audit Specification

A server audit specification details the actions to be audited. You can choose either action groups or individual actions and objects.

In the example shown in the slide, a database audit specification is being created using the CREATE DATABASE AUDIT SPECIFICATION statement. The configuration of the same specification using the GUI is also shown.

The name of the specification is BackupRestoreSpec and the data collected from the specification will be sent to the Audit-20121222-171544 audit. The action group to be audited is the BACKUP_RESTORE_GROUP.

Audit-related DMVs and System Views



Key Points

SQL Server provides a number of dynamic management views (DMVs) and system views that can help you manage SQL Server Audit.

The following DMVs and system views are available:

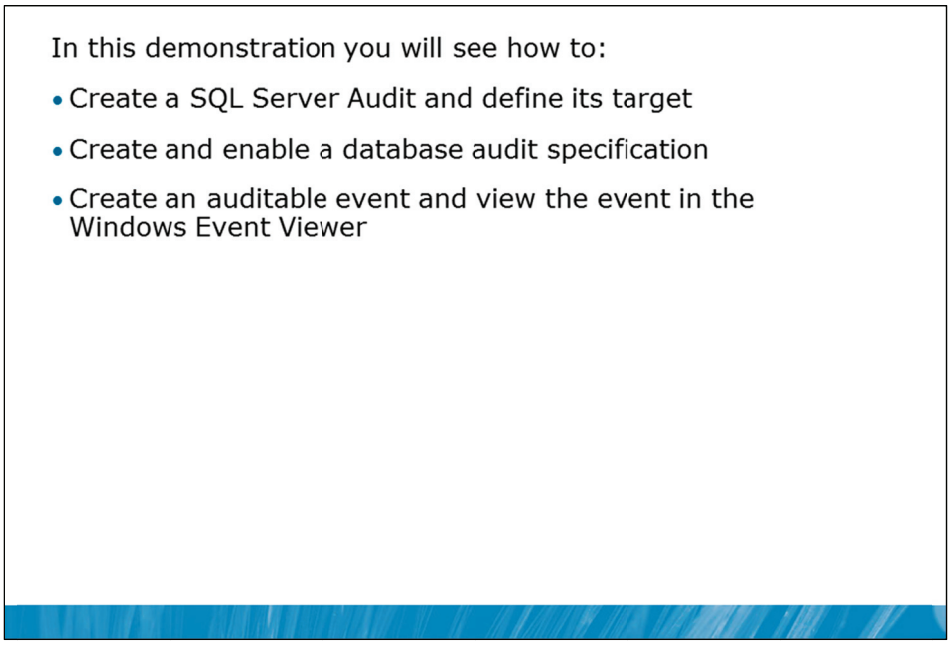
DMV/View	Description
sys.dm_server_audit_status	Returns a row for each server audit indicating the current state of the audit
sys.dm_audit_actions	Returns a row for every audit action that can be reported in the audit log and every action group that can be configured as part of an audit
sys.dm_audit_class_type_map	Returns a table that maps the class types to class descriptions
sys.server_audits	Contains one row for each SQL Server audit in a server instance
sys.server_file_audits	Contains extended information about the file audit type in a SQL Server audit
sys.server_audit_specifications	Contains information about the server audit specifications in a SQL Server audit

(continued)

DMV/View	Description
sys.server_audit_specification_details	Contains information about the server audit specification details (actions) in a SQL Server audit
Sys.database_audit_specifications	Contains information about the database audit specifications in a SQL Server audit
Sys.database_audit_specification_details	Contains information about the database audit specifications in a SQL Server audit

A number of the system views will be used in the upcoming demonstrations.

Demonstration 2A: Using SQL Server Audit

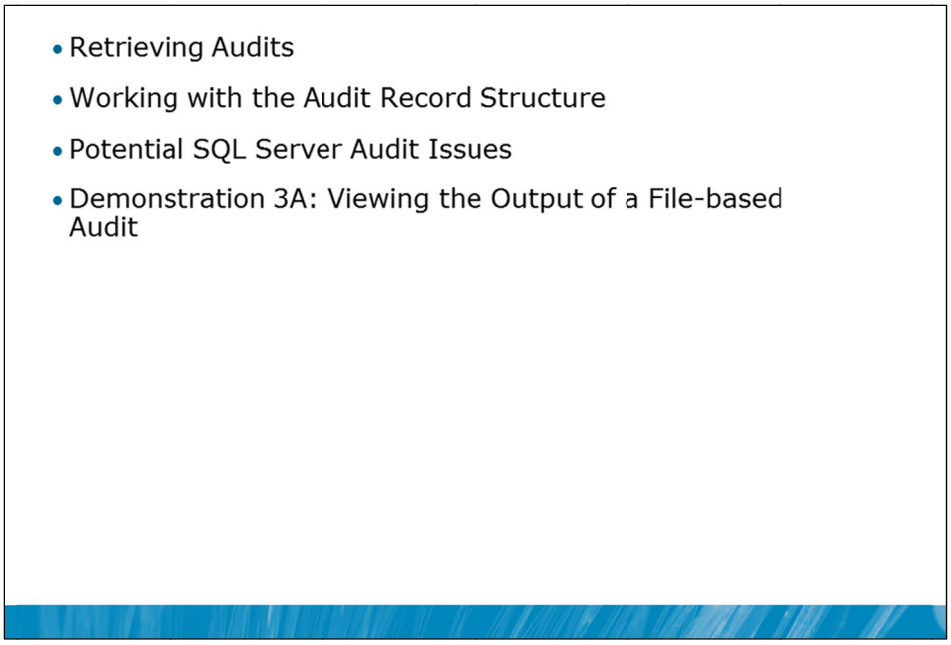


Demonstration Steps

1. If Demonstration 1A was not performed:
   - Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
   - In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, click **SQL Server Management Studio**. In the Connect to Server window, type **Proseware** and click **Connect**. From the **File** menu, click **Open**, click **Project/Solution**, navigate to **D:\10775A_Labs\10775A_12_PRJ\10775A_12_PRJ.ssmssln** and click **Open**.
   - From the **View** menu, click **Solution Explorer**. Open and execute the **00 – Setup.sql** script file from within Solution Explorer.
2. Open the **21 – Demonstration 2A.sql** script file.
3. Follow the instructions contained within the comments of the script file.

   Question: What are the three possible event targets for SQL Server Audit?

Lesson 3
Managing SQL Server Audit



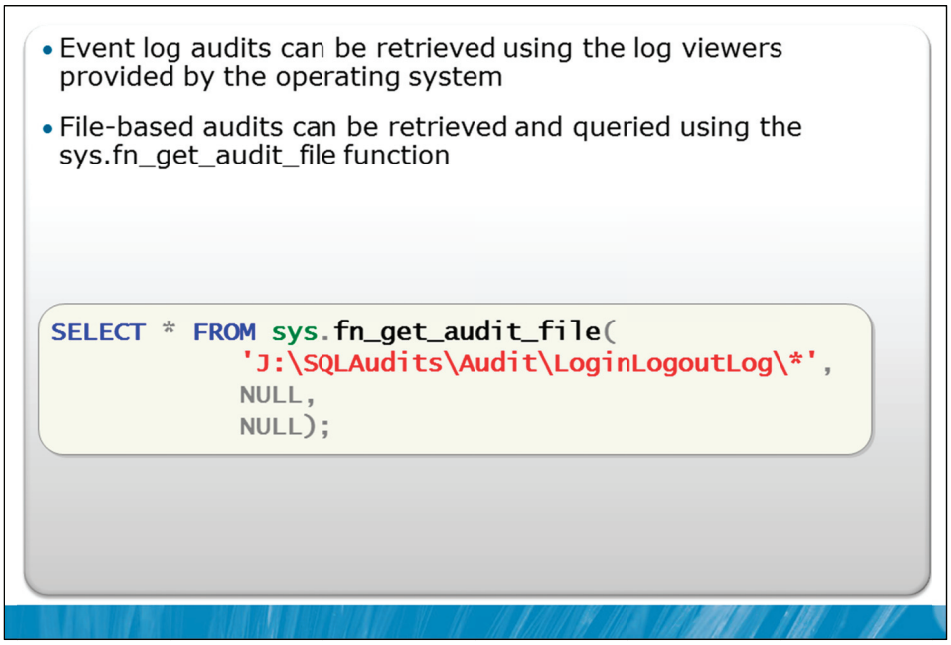
It is important to be able to retrieve the results from the audits and to understand a few aspects of ongoing management of SQL Server Audit. In particular you will investigate issues related to migrating databases between servers and see how to restart servers if SQL Server refuses to start due to an audit failure.

Objectives

After completing this lesson, you will be able to:

- Retrieve audits.
- Work with the audit record structure.
- Identify potential SQL Server audit issues.

Retrieving Audits



Key Points

No special configuration is needed to view audits sent to event logs. SSMS provides a log reader for these targets.

Retrieving File Output

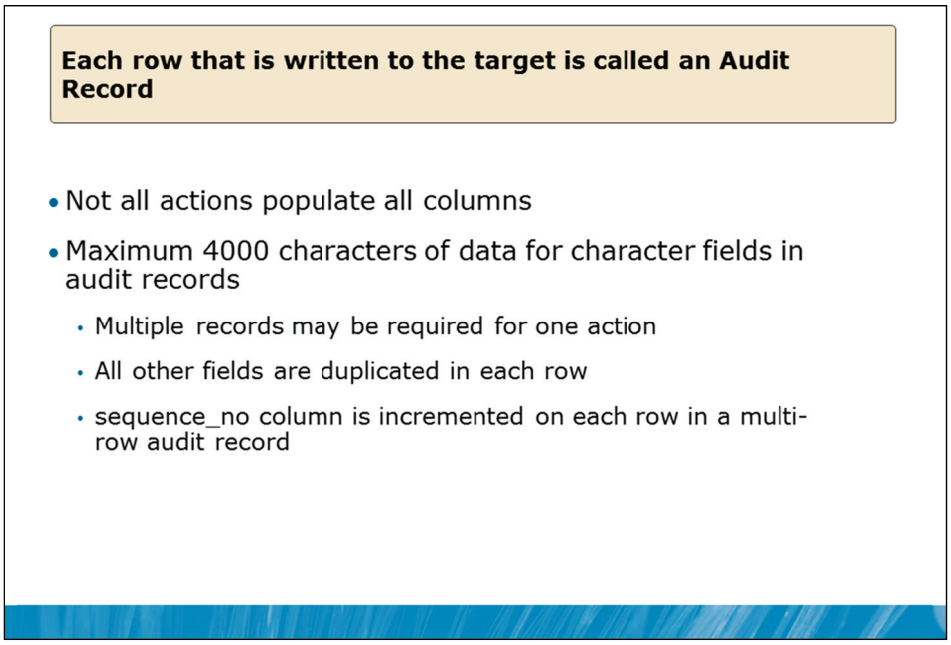
For logs that are sent to a file, SQL Server provides a function that returns the contents of the file-based logs as a table that you can query with T-SQL.

Note The filename that you provide to the FILEPATH parameter when creating a server audit is actually the name of a folder.

The folder that contains the audit logs often contains multiple audit files. The sys.fn_get_audit_file function is used to retrieve those files. It takes three parameters: the file_pattern, the initial_file_name, and the audit_record_offset. The file_pattern provided can be in one of three formats:

Format	Description
<path>*	Collects all audit files in the specified location
<path>\LoginsAudit_{GUID}	Collect all audit files that have the specified name and GUID pair
<path>\LoginsAudit_{GUID}_00_29384.sqlaudit	Collect a specific audit file

Working with the Audit Record Structure



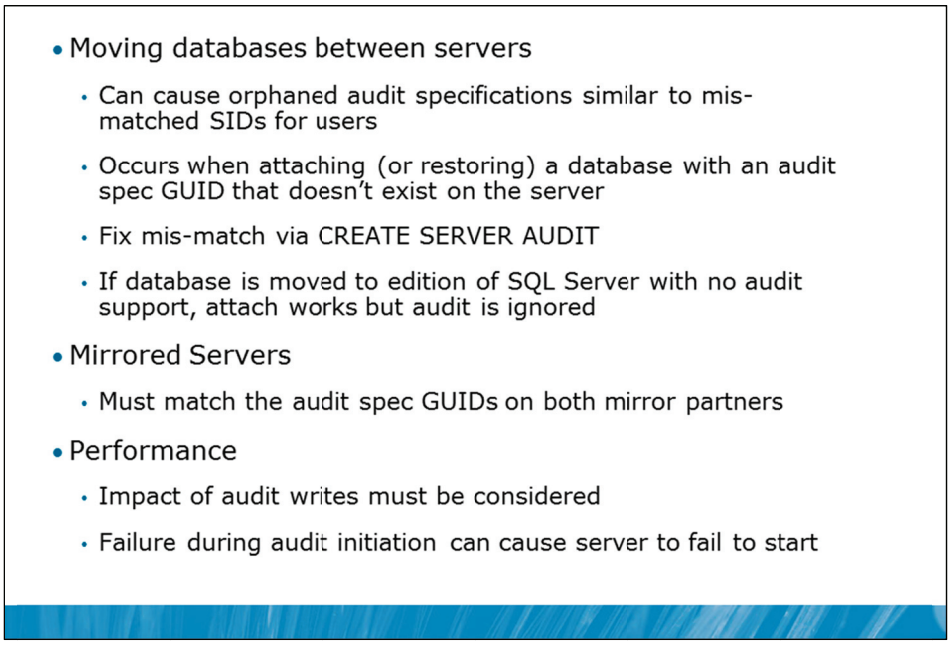
Key Points

The audit record structure is detailed in Books Online under the topic "sys.fn_get_audit_file (Transact-SQL)".

Audit records need to be able to be stored in system event logs as well as in files. Because of this requirement, the record format is limited in size by the rules related to those event logging systems. Character fields will be split into 4000 character chunks and the chunks will be spread across a number of entries.

This means that a single event could generate multiple audit entries. A sequence_no column is provided to indicate the order of multiple row entries.

Potential SQL Server Audit Issues



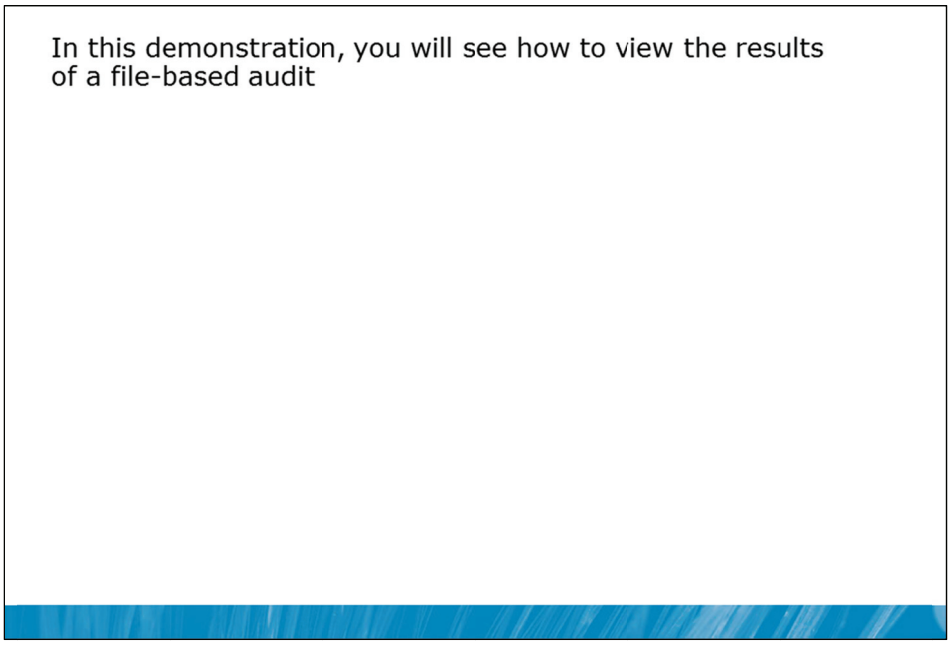
Key Points

There are a number of potential issues to consider with SQL Server audit.

- Each audit is identified by a GUID. When a database is restored or attached on a server, an attempt is made to match the GUID in the database with the GUID of the audit on the server. If no match occurs, auditing will not work until the situation is corrected by executing the CREATE SERVER AUDIT command to set the appropriate GUID.
- If databases are attached to editions of SQL Server that do not support the same level of audit capability, the attach works but the audit is ignored.
- Mirrored servers introduce a similar issue to mis-matched GUIDs. The mirror partner must have a server audit with the same GUID. You can create this by using the CREATE SERVER AUDIT command and supplying the GUID value to match the value on the primary server.
- In general, the performance impact of audit writes must be considered. If disk space fills up, SQL Server may not start. If so, you may need to force entry to it via a single user startup and the –f startup parameter.

Question: Why would audits be identified by a GUID as well as a name?

Demonstration 3A: Viewing the Output of a File-based Audit

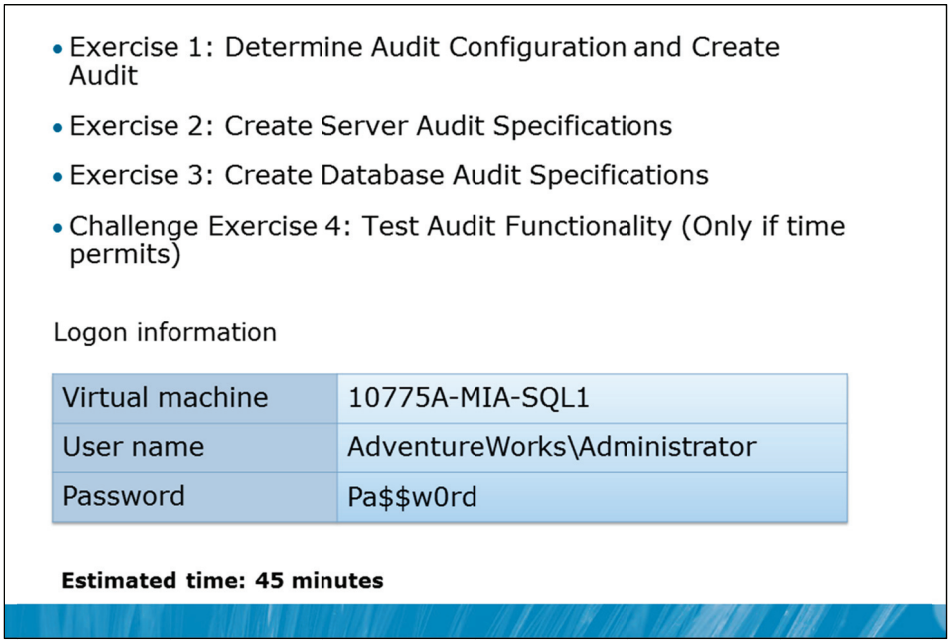


Demonstration Steps

1. If Demonstration 1A was not performed:
   - Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
   - In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, click **SQL Server Management Studio**. In the Connect to Server window, type **Proseware** and click **Connect**. From the **File** menu, click **Open**, click **Project/Solution**, navigate to **D:\10775A_Labs\10775A_12_PRJ\10775A_12_PRJ.ssmssln** and click **Open**.
   - From the **View** menu, click **Solution Explorer**. Open and execute the **00 – Setup.sql** script file from within Solution Explorer.
2. Open the **31 – Demonstration 3A.sql** script file.
3. Follow the instructions contained within the comments of the script file.

   Question: Why are there two entries in the audit log?

Lab 12: Auditing SQL Server Environments



Lab Setup

For this lab, you will use the available virtual machine environment. Before you begin the lab, you must complete the following steps:

1. Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
2. In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, and click **SQL Server Management Studio**.
3. In the Connect to Server window, type **Proseware** in the **Server name** text box.
4. In the **Authentication** drop-down list box, select **Windows Authentication** and click **Connect**.
5. In the **File** menu, click **Open**, and click **Project/Solution**.
6. In the Open Project window, open the project **D:\10775A_Labs\10775A_12_PRJ\10775A_12_PRJ.ssmssln**.
7. From the **View** menu, click **Solution Explorer**. In Solution Explorer, double-click the query **00-Setup.sql**. When the query window opens, click **Execute** on the toolbar.

Lab Scenario

You have authorized users to access the Proseware instance. Your Compliance Department has provided you with details of the auditing requirements for both the Proseware server instance and for the MarketDev database. The auditing requirements include the need to audit the activities against tables in the MarketDev database that contain sensitive information. In this lab, you will implement a strategy to enable appropriate auditing.

If you have sufficient time, you need to test the audit strategy and write a query to extract audit records.

Supporting Documentation

Audit Requirements from the Compliance Department

1. All audit records should be written to the folder C:\Audit\AuditLog.
2. Audit records should be written as quickly as possible however a tolerance of two seconds of audit records is the maximum permitted loss in the event of failure.
3. The server instance should not continue to operate if auditing is not occurring.
4. The name of the audit should be Proseware Compliance Audit.
5. There is no limit to the number of audit files that may be created however each audit file should be limited to 1 GB in size.
6. At the server level, the following items need to be audited:
   - Failed login attempts
   - Changes to the membership of server roles
   - Any changes to server logins (principals)
   - Any changes to passwords
7. At the MarketDev database level, the following items need to be audited:
   - Changes to the membership of database roles
   - Backups and restores of the database
   - Changes to any permissions within the database
   - Any changes to database users (principals)
   - Any change of database ownership
   - Any updates to the Marketing.CampaignBalance table
   - Any executions of the Marketing.MoveCampaignBalance stored procedure

Exercise 1: Determine Audit Configuration and Create Audit

Scenario

You need to determine the configuration of a server audit, based on the business security requirements. In this exercise, you will create the required Server Audit.

The main tasks for this exercise are as follows:

1. Review the requirements.
2. Create the server audit.

▶ Task 1: Review the requirements

- Review the supplied requirements in the supporting documentation for the exercise.

▶ Task 2: Create the server audit

- Determine the configuration of the required server audit.
- Create the server audit using SQL Server Management Studio.
- Enable the server audit.

Results: After this exercise, you should have created the required server audit.

Exercise 2: Create Server Audit Specifications

Scenario

You need to determine which of the business requirements can be met via server audit specifications. You will then determine the required Server Audit Specifications and create them.

The main tasks for this exercise are as follows:

1. Review the requirements.
2. Create the server audit specifications.

▶ Task 1: Review the requirements

- Review the supplied requirements in the supporting documentation for the exercise.

▶ Task 2: Create the server audit specifications

- Determine the required server audit specifications.
- Create the server audit specifications using SQL Server Management Studio.
- Enable the server audit specifications.

Results: After this exercise, you should have created the required server audit specification.

Exercise 3: Create Database Audit Specifications

Scenario

Some of the audit requirements will require database audit specifications to be created. In this exercise, you will determine which of the audit requirements could be met by database audit specifications. You will then create those database audit specifications.

The main tasks for this exercise are as follows:

1. Review the requirements.
2. Create the database audit specifications.

▶ Task 1: Review the requirements

- Review the requirements.

▶ Task 2: Create the database audit specifications

- Determine the required database audit specifications.
- Create any required database audit specifications using SQL Server Management Studio.
- Enable any database audit specifications that you created.

Results: After this exercise, you should have created the required database audit specifications.

Challenge Exercise 4: Test Audit Functionality (Only if time permits)

Scenario

You need to check that the auditing you have set up is functioning as expected. You will execute a test workload script and then review the captured audit using both the GUI is SSMS and T-SQL.

The main tasks for this exercise are as follows:

1. Execute the workload script.
2. Review the captured audit details.

▶ Task 1: Execute the workload script

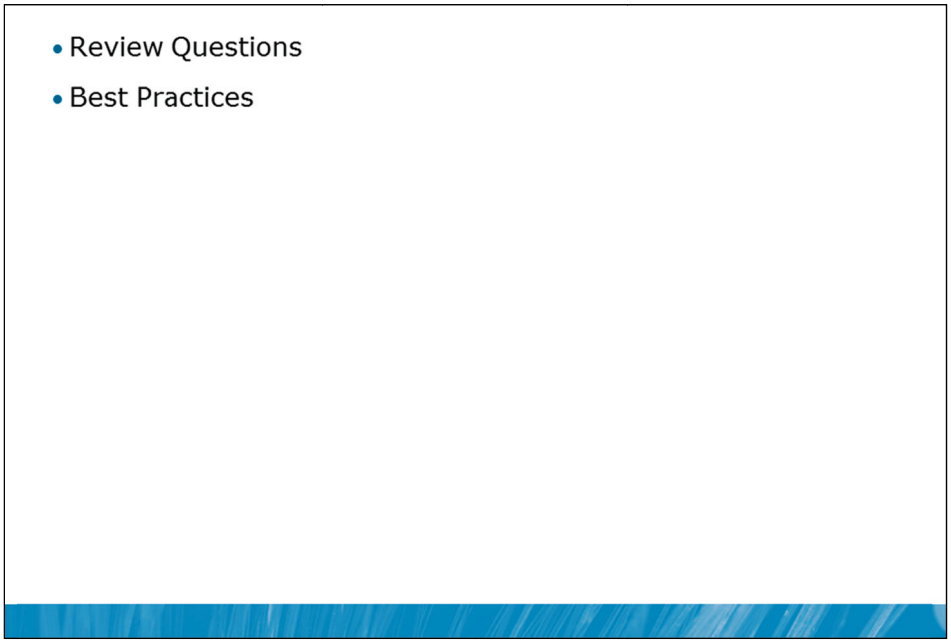
- From Solution Explorer open and execute the workload script 81 – Lab Exercise 4a.sql.

▶ Task 2: Review the captured audit details

- Review the captured audit details using the View Audit Logs option in SQL Server Management Studio. (This is a right-click option from the Server Audit).
- Write a query to retrieve the audit log details using T-SQL.

Results: After this exercise, you should have checked that the auditing works as expected.

Module Review and Takeaways



Review Questions

1. What are the three targets for SQL Server audits?
2. When common criteria compliance is enabled in SQL Server, what changes about column-level permissions?
3. You may wish to audit actions by a DBA. How would you know if the DBA stopped the audit while performing covert actions?

Best Practices

1. Choose the option to shut down SQL Server on audit failure. There is usually no point in setting up auditing and then having situations where events can occur but are not audited. This is particularly important in higher-security environments.
2. Make sure that file audits are placed on drives with large amounts of free disk space and make sure that the available disk space is monitored on a regular basis.

Module 13

Automating SQL Server 2012 Management

Contents:

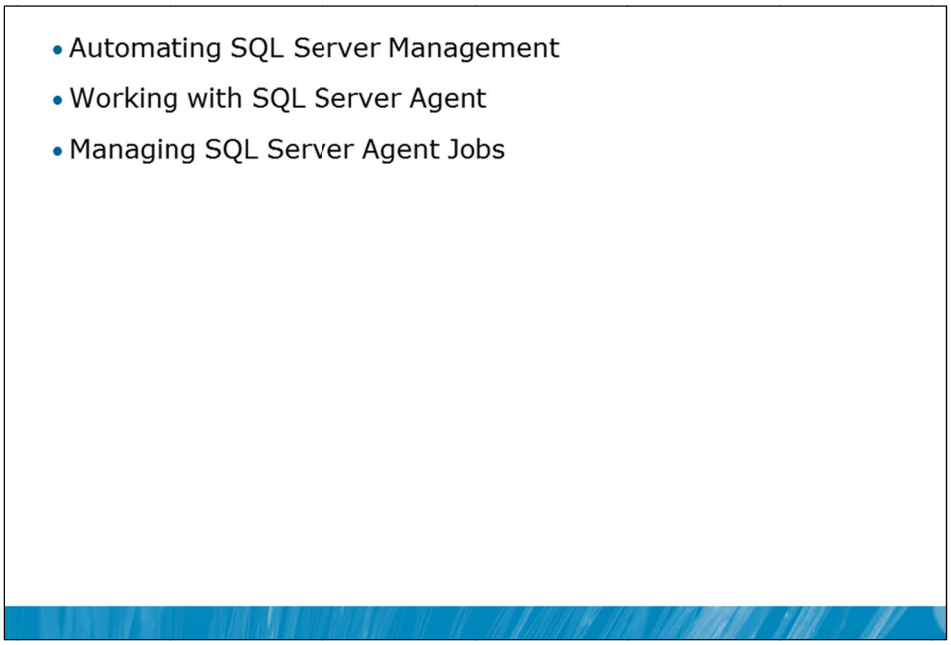
Lesson 1: Automating SQL Server Management **13-3**

Lesson 2: Working with SQL Server Agent **13-11**

Lesson 3: Managing SQL Server Agent Jobs **13-19**

Lab 13: Automating SQL Server Management **13-26**

Module Overview



The tools provided with Microsoft® SQL Server® make administration easy when compared with other database engines. Even when tasks are easy to perform though, it is common to need to repeat a task many times. Efficient database administrators learn to automate repetitive tasks. Automating tasks can help avoid situations where an administrator forgets to execute a task at the required time. Perhaps more important though, is that the automation of tasks helps to ensure that tasks are performed consistently, each time they are executed.

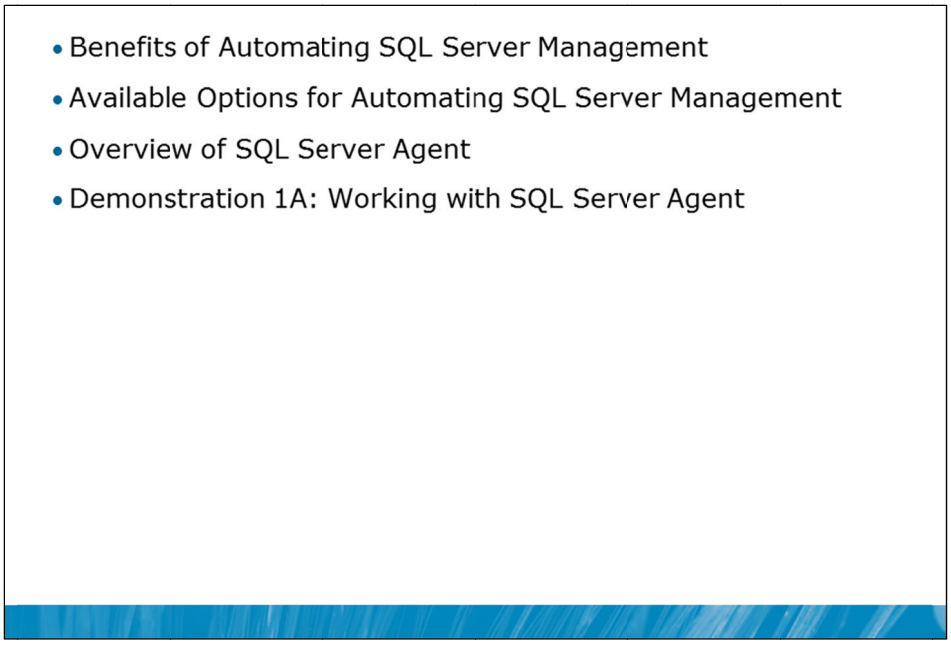
SQL Server Agent is the service that is provided in all editions of SQL Server 2012 (except SQL Server Express Edition) that is responsible for the automation of tasks. A set of tasks that need to be performed is referred to as a SQL Server Agent job. It is important to learn how to create and manage these jobs.

Objectives

After completing this lesson, you will be able to:

- Automate SQL Server Management.
- Work with SQL Server Agent.
- Manage SQL Server Agent jobs.

Lesson 1
Automating SQL Server Management



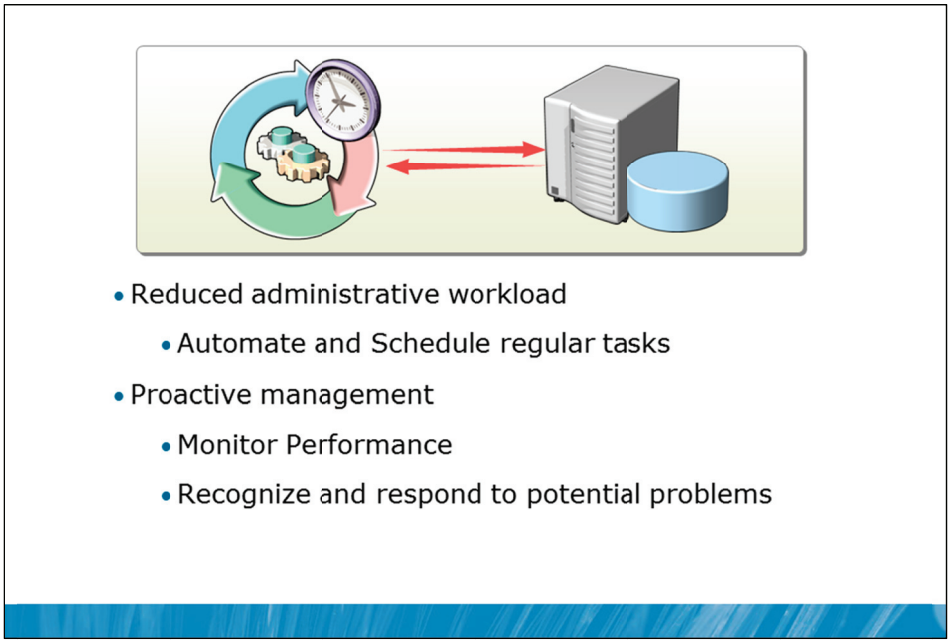
There are many benefits that can be gained from the automation of SQL Server management. Most of the benefits center on the reliable, consistent execution of routine management tasks. SQL Server is a flexible platform that provides a number of ways to automate management but the most important tool for automation of management is the SQL Server Agent. All database administrators that work with the SQL Server need to be very familiar with the configuration and ongoing management of SQL Server Agent.

Objectives

After completing this lesson, you will be able to:

- Explain the benefits of automating SQL Server management.
- Describe the available options for automating SQL Server management and the framework that is provided with SQL Server Agent.
- Describe SQL Server Agent.

Benefits of Automating SQL Server Management



Key Points

All efficient database administrators automate their routine administrative tasks. Some of the benefits that can be gained from the automation of SQL Server management are as follows:

Reduced Administrative Load

Unfortunately, some administrators that work with SQL Server, Windows®, and other tools, see their roles in terms of a constant stream of repetitive administrative tasks. For example, a Windows administrator at a University department might receive regular requests to create a large number of user accounts. The administrator might be happy to create each of these accounts one by one, using the standard tooling. A more efficient administrator would learn to write a script to create users and execute the script instead of manually creating the users.

The same sort of situation occurs with routine tasks in SQL Server. While these tasks can be performed individually or manually, efficient database administrators do not do this. They automate all their routine and repetitive tasks. Automation removes the repetitive workload from the administrators and allows the administrators to manage larger numbers of systems or to perform higher-value tasks for the organization.

Reliable Execution of Routine Tasks

When routine tasks are performed manually, there is always a chance that a vital task might be overlooked. For example, a database administrator could forget to perform database backups. Automation allows administrators to focus on exceptions that occur during the routine tasks rather than focusing on the execution of the tasks.

Consistent Execution of Routine Tasks

Another problem that can occur when routine tasks are performed manually is that the tasks may not be performed the same way each time. Imagine a situation where a database administrator is required to archive some data from a set of production tables, into a set of history tables every Monday morning. The new tables need to have the same name as the original tables with a suffix that includes the current date.

While the administrator might remember to perform this task every Monday morning, what is the likelihood that one of the following errors would occur?

- Copy the wrong tables
- Copy only some of the tables
- Forget what the correct date is when creating the suffix
- Format the date in the suffix incorrectly
- Copy data into the wrong archive table

Anyone that has been involved in ongoing administration of systems would tell you that these and other problems would occur from time to time, even when the tasks are executed by experienced and reliable administrators. Automating routine tasks can assist greatly in making sure that they are performed consistently each time they are executed.

Proactive Management

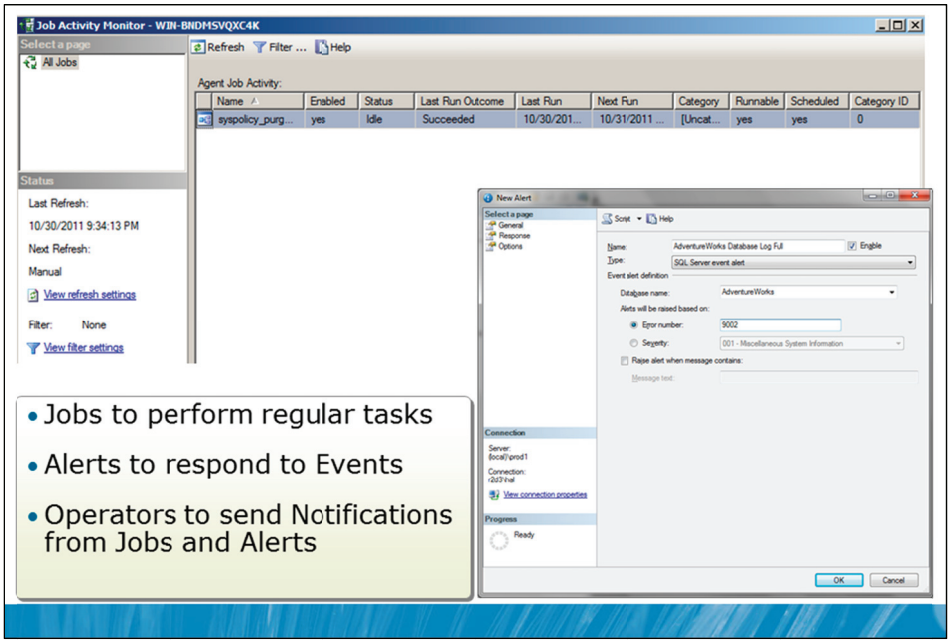
Once routine tasks are automated, it is easy for a situation where the routine execution of the tasks fails but no administrator ever notices that the task is failing. For example, there are many tragic tales on the SQL Server community support forums from administrators that automated the backup of their databases, and where they did not notice that the backups had been failing for a long time, until they needed one of the backups.

As well as automating your routine tasks, you need to ensure that you create notifications that tell you when the tasks fail, even if you cannot imagine a situation where the tasks could fail. For example, you may have created a backup strategy that creates database backups in a given folder. The job may run reliably for years until another administrator inadvertently deletes or renames the target folder. You need to know as soon as this problem occurs so that you can rectify the situation.

A more proactive administrator will try to detect potential problems before they occur. For example, rather than receiving a notification that a job failed because a disk was full, an administrator might schedule regular checks of available disk space and make sure that a notification is received when available free space is starting to get too low. SQL Server provides alerts on system and performance conditions.

Question: What tasks need to be automated on the systems in your organization?

Available Options for Automating SQL Server Management



Key Points

The primary method for automation of management, administrative, and other routine tasks when working with SQL Server 2012 is to use SQL Server Agent.

Framework for SQL Server Agent

The management framework that is supplied by SQL Server Agent is based on two core objects:

- Jobs that are used to automate tasks
- Alerts that are used to respond to events

Jobs can be used to schedule a wide variety of task types, including tasks that are required for the implementation of other SQL Server features. For example, the Replication, Change Data Capture (CDC), Data Collection, and Policy Based Management (PBM) features of SQL Server create SQL Server Agent jobs. (Data Collection will be discussed in Module 18).

Note Replication, CDC, and PBM are advanced topics that are out of scope for this course.

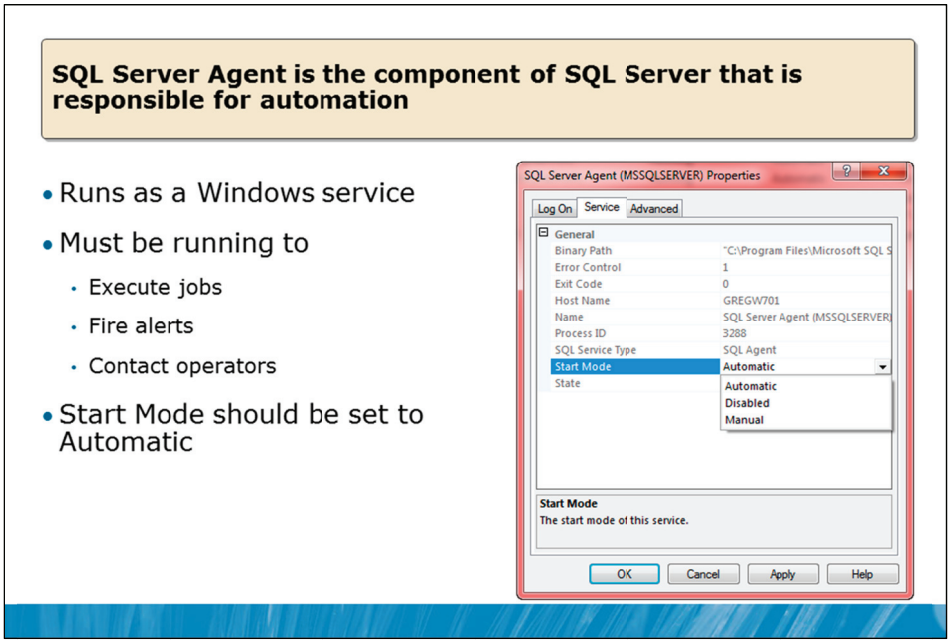
The alerting system that is provided by SQL Server Agent is capable of responding to a wide variety of alert types, including SQL Server Error Messages, SQL Server Performance Counter Events, and Windows Management Instrumentation (WMI) Alerts.

In response to an alert, an action can be configured, such as the execution of a SQL Server Agent job or sending a notification to an administrator. In SQL Server Agent, administrators that can be notified are called operators. Operators are commonly notified by the use of SMTP based email. (Alerts are discussed in Module 15).

Note that there are other SQL Server features that can be used to automate complex monitoring requirements. The Extended Events feature is an example of this but is out of the scope of this training.

Question: Can you think of events that might occur on a SQL Server system that you would want to be alerted about?

Overview of SQL Server Agent



Key Points

As mentioned earlier in this module, SQL Server Agent is the component of SQL Server that is responsible for automating SQL Server administrative tasks. SQL Server Agent runs as a Windows service.

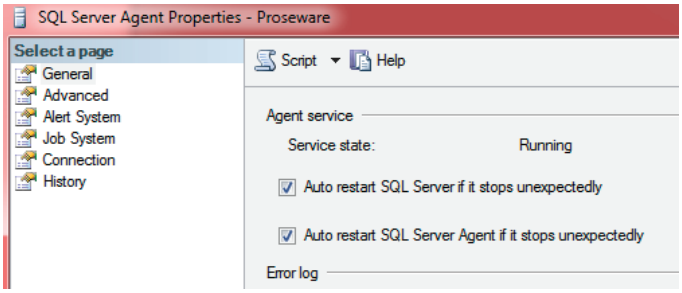
Starting SQL Server Agent

For SQL Server Agent to perform its main role of executing jobs and firing alerts, SQL Server Agent needs to be running constantly. Because of this, SQL Server Agent is typically configured to start automatically when the operating system starts. Note that the default option during SQL Server installation is for SQL Server Agent to be started manually. The design of the SQL Server installation process ensures that services and components are not installed or started unless they are required. This means that the default installation option needs to be changed if there is a need for SQL Server Agent to be running on your system.

The start mode for SQL Server Agent is configured in the properties of the SQL Server Agent service in SQL Server Configuration Manager as shown on the slide. Note that three start modes are configurable:

Start Mode	Description
Automatic	The service will start when the operating system starts.
Disabled	The service will not start, even if you attempt to start it manually.
Manual	The service needs to be manually started.

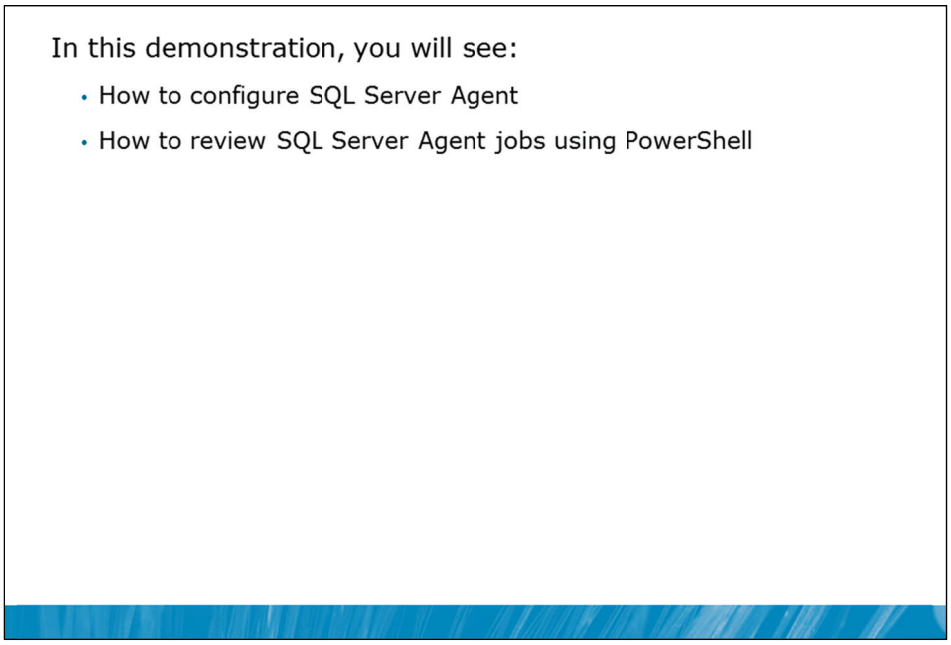
In addition, you can configure the SQL Server Agent service to restart automatically if it stops unexpectedly. The automatic restart option is set in the properties page for the SQL Server Agent in SSMS, as shown below:



To restart automatically, the SQL Server Agent service account must be a member of the local Administrators group for the computer that SQL Server is installed on but this is not considered a best practice. A better option would be to use an external monitoring tool such as System Center Operations Manager to monitor and restart the SQL Server Agent service if necessary.

Question: Why should SQL Server Agent service be always configured to start up automatically?

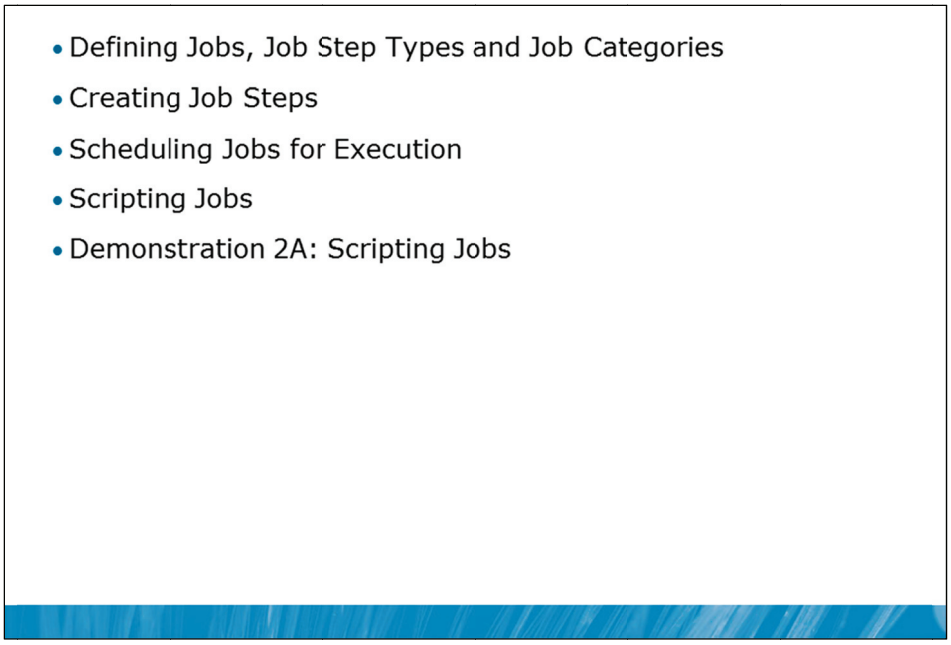
Demonstration 1A: Working with SQL Server Agent



Demonstration Steps

1. Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
2. In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, click **SQL Server Management Studio**. In the Connect to Server window, type **Proseware** and click **Connect**. From the **File** menu, click **Open**, click **Project/Solution**, navigate to **D:\10775A_Labs\10775A_13_PRJ\10775A_13_PRJ.ssmssln** and click **Open**.
3. From the **View** menu, click **Solution Explorer**. Open and execute the **00 – Setup.sql** script file from within Solution Explorer.
4. Open the **11 – Demonstration 1A.sql** script file.
5. Follow the instructions contained within the comments of the script file.

Lesson 2
Working with SQL Server Agent



You have seen that SQL Server Agent is the primary tool for automating tasks within SQL Server. Database administrators need to be proficient at creating and configuring SQL Server Agent jobs. Jobs can be created to implement a variety of different types of task and can be categorized for ease of management.

Creating a job involves creating a series of steps that the job will execute, along with the workflow that determines which steps should be executed, and in which order.

Once the steps that a job needs to take have been determined, you need to determine when the job will be executed. Most SQL Server Agent jobs are run on defined schedules. SQL Server provides the ability to create a flexible set of schedules that can be shared between jobs.

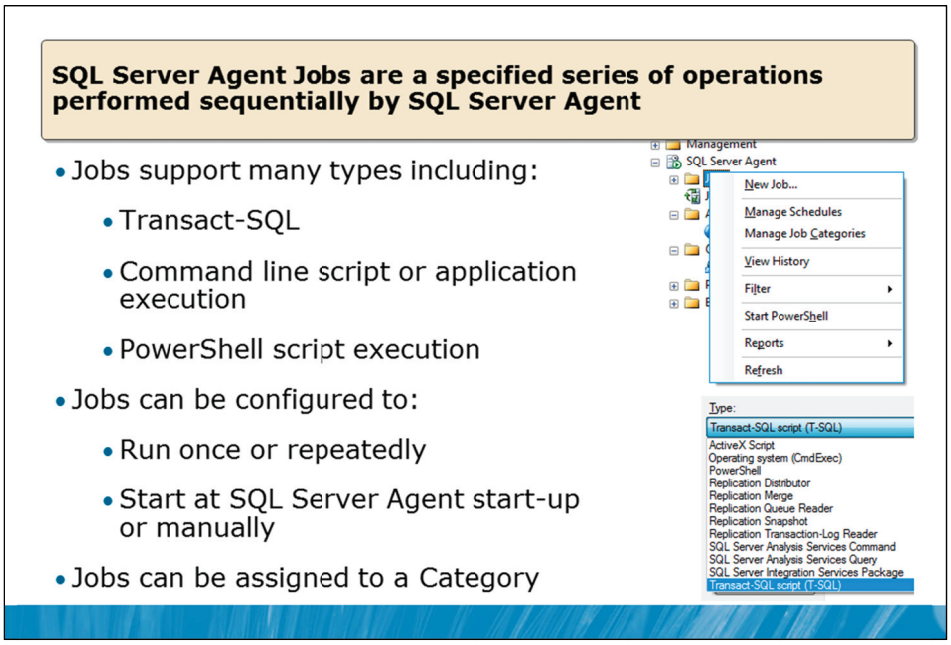
It is important to learn to script jobs that have been created. The scripting of jobs allows the jobs to be quickly recreated if a failure occurs, and allows the jobs to be recreated in other environments. For example, the jobs may have been created in a test environment but need to be executed in a production environment.

Objectives

After completing this lesson, you will be able to:

- Define jobs, job types and job categories.
- Create job steps.
- Schedule jobs for execution.
- Script jobs.

Defining Jobs, Job Step Types and Job Categories



Key Points

SQL Server Agent jobs are comprised of a series of operations that need to be performed in order. In most jobs, the steps are performed sequentially but an administrator can exercise control over the order of the steps.

By configuring the action to occur on the success and failure of each job step, a workflow can be created that determines the overall logic flow of the job.

Job Step Types

Note that every job step has an associated type that defines the kind of operation to run. The most commonly used types are:

- Executing a command line script, batch of commands, or application.
- Executing a T-SQL statement.
- Executing a Windows PowerShell® script.
- Executing a SQL Server Integration Services Package
- Executing Analysis Services commands and queries.

Note While the ability to execute ActiveX® scripts has been retained for backwards compatibility, this option is deprecated and should not be used for new development.

Note Other specialized job step types are used by features of SQL Server such as Replication. Replication is an advanced topic that is out of scope for this course.

Job Schedules

One or more schedules can be defined for every job. Schedules can be defined as recurring, but also other schedule types provide for one time execution or for execution when SQL Server Agent first starts.

You may need to create multiple schedules for a job when the required recurrence pattern for the job is complex and cannot be accommodated within a single job schedule.

While each schedule can also be shared between many jobs, it is important to avoid having too many jobs starting at the same time.

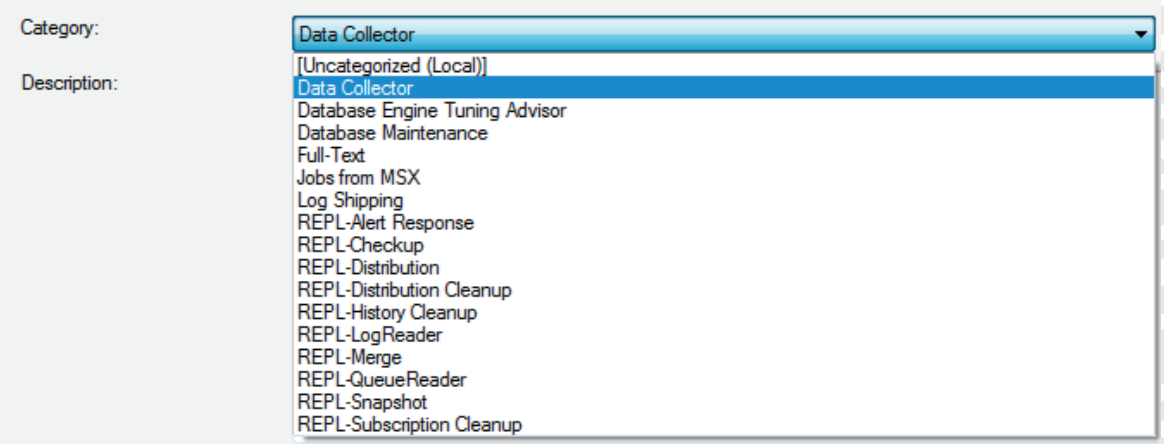
Creating Jobs

You can use SQL Server Management Studio to create jobs or you can execute the sp_add_job system stored procedure. Creating a job requires the execution of a number of additional system stored procedures, to add steps and schedules to the job.

The job definition is stored in the msdb database, along with all of SQL Server Agent configuration.

Job Categories

Jobs can be placed into categories. SQL Server has a number of built-in categories as shown below but you can add your own categories;



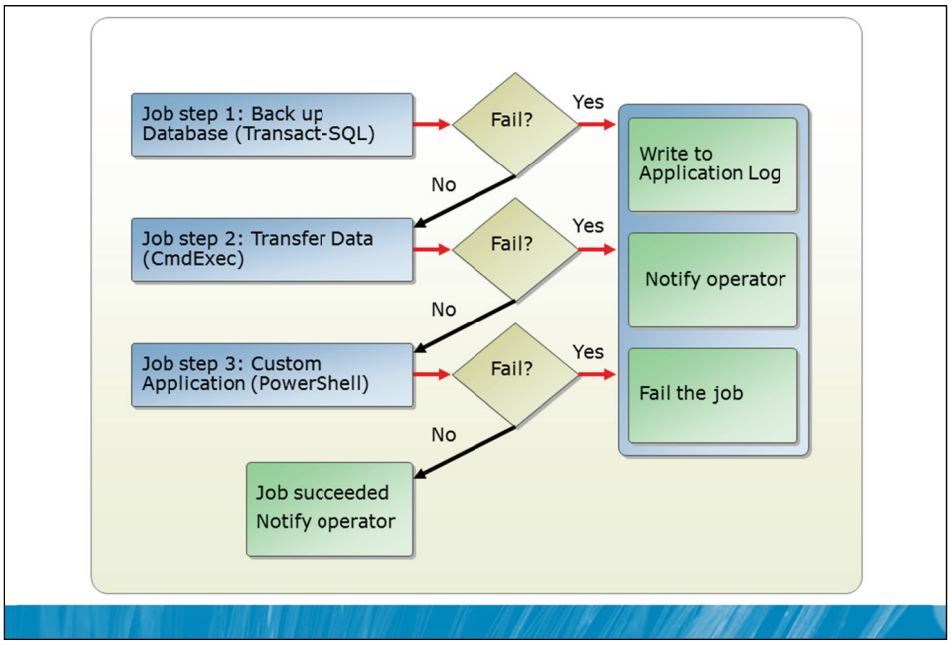
Job categories can be useful when you need to perform actions that are associated with jobs in a specific category. For example, it would be possible to create a job category called "SQL Server 2005 Policy Check" and to write a PowerShell script to execute all the jobs in that category against your SQL Server 2005 servers.

Note A detailed discussion on the use of PowerShell is an advanced topic that is out of scope for this course.

Note It is also important to consider the security account that each type of job step requires for execution. Module 14 discusses security for SQL Server Agent.

Question: Can you think of a use for job categories?

Creating Job Steps



Key Points

Use SQL Server Management Studio or execute the sp_add_jobstep system stored procedure to define each job step that is required to automate a task. Only one execution type can be defined for each job step, but each step of a job can have a different type.

Job Step Workflow

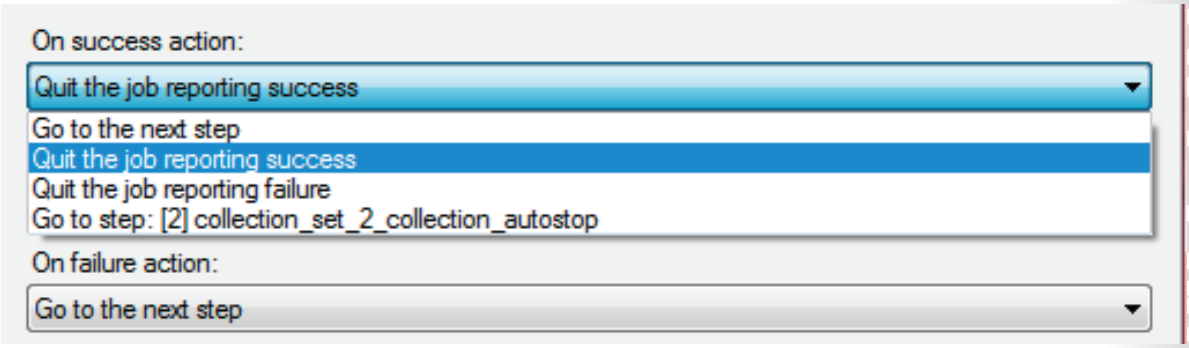
Every step has an outcome that defines whether the step has succeeded or failed. Note the list of job steps and the "On Success" and "On Failure" options for each step in the following job:

Job step list:

S	Name	Type	On Success	On Failure
1	collection_set_2_collection_collect	Operating system (Cmd...	Quit the job reporting suc...	Go to the next step
2	collection_set_2_collection_autostop	Transact-SQL script (T-...	Quit the job reporting fail...	Quit the job reporting failure

By default, SQL Server advances to the next job step upon success and stops upon failure of a job step but job steps can continue with any step defined in the job upon success or failure, to define a special workflow.

In the Advanced properties of each job step, an action can be configured for both success and failure as shown below:



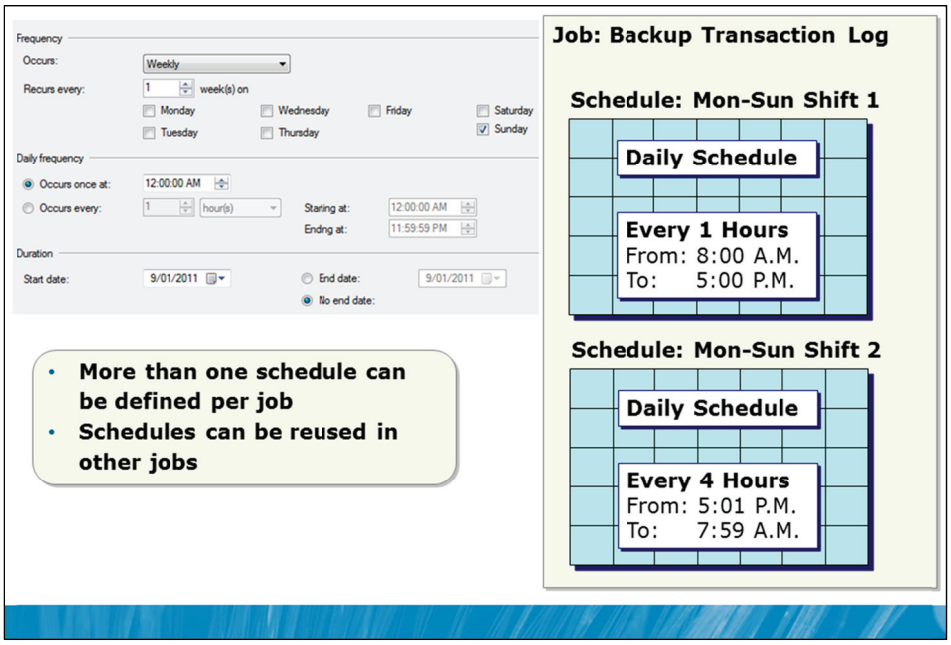
By configuring the action to occur on the success and failure of each job step, a workflow can be created that determines the overall logic flow of the job. Note that as well as each job step having a defined outcome, the overall job reports an outcome. This means that even though some job steps might succeed, the overall job might still report failure.

Retrying Job Steps

You can specify the number of times that SQL Server should attempt to retry execution of a job step if the step fails. You also can specify the retry intervals (in minutes). For example, if the job step requires a connection to a remote server, you could define several retry attempts in case the connection fails.

Question: Which operations should not be grouped together in a job?

Scheduling Jobs for Execution



Key Points

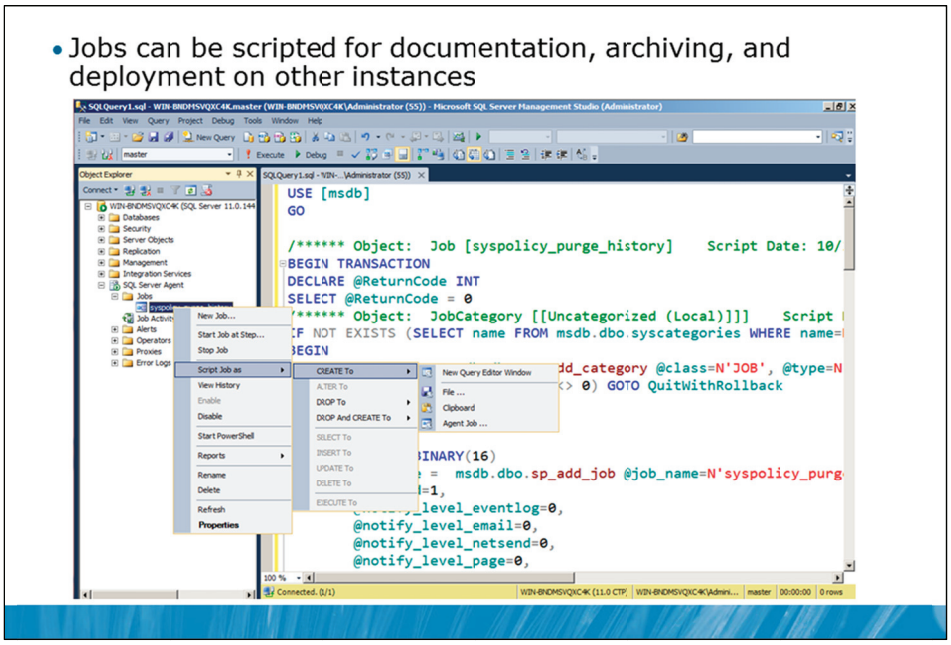
Schedules are used to start jobs at requested times. One or more schedules can be associated with a job. Schedules are assigned names and schedules can be shared across multiple jobs. Apart from standard recurring schedules, a number of special recurrence types are also defined:

- One time execution.
- Start automatically when SQL Server Agent starts.
- Start whenever the CPU becomes idle.

Even though a job may have multiple schedules, SQL Server will limit the job to a single concurrent execution. If you try to run a job manually while it is running as scheduled, SQL Server Agent refuses the request. If a job is still running when it is scheduled to run again, SQL Server Agent refuses to run it again.

Question: What could be changed if the database in the example above does not need hourly backups during weekend?

Scripting Jobs



Key Points

Jobs are typically first created in SSMS as there are a several system stored procedures that need to be executed to define a single job. However, it is important that existing jobs be scripted in T-SQL for the following reasons:

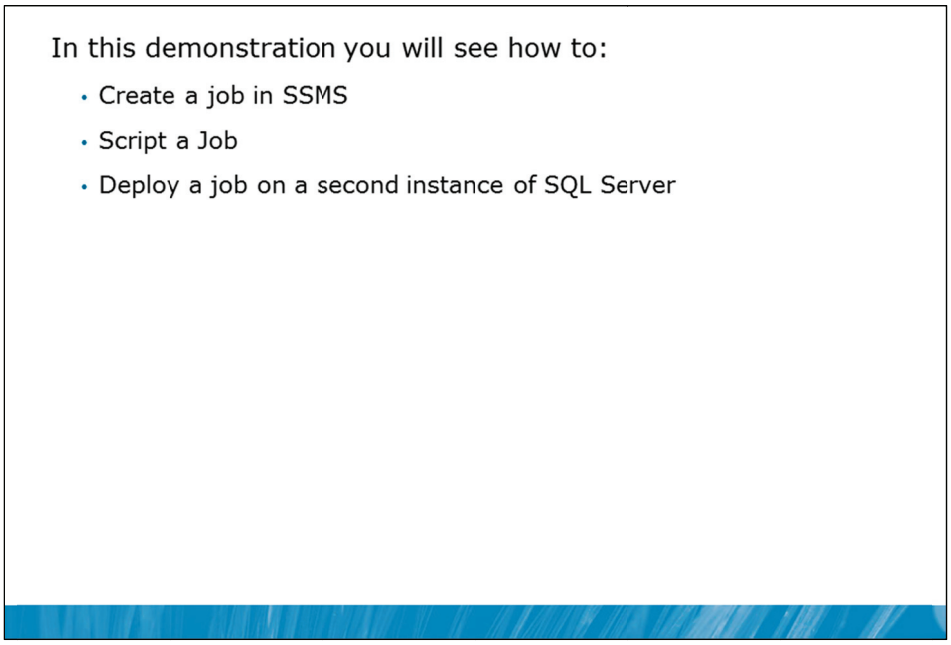
- Scripts of jobs can be used in documentation and can be archived into source code control systems.
- Jobs can easily be recreated after a failure if necessary, if scripts of the jobs have been created.
- There is also a common requirement to be able to create a job in one environment (such as a test environment) and to deploy the job in another environment (such as a production environment). The ability to script jobs allows for easier deployment of the jobs.
- Scripts of jobs could be used when performing side by side upgrades of SQL Server systems.

Note More than one job can be selected in Object Explorer Details when scripting.

Other more advanced options are available for scripting jobs, such as the use of SQL Server Management Objects (SMO). SMO can be used in conjunction with .NET programming in languages such as Microsoft Visual Basic® or C#, and can be used in conjunction with PowerShell.

Question: In which scenarios might it be useful to script more than one job at a time?

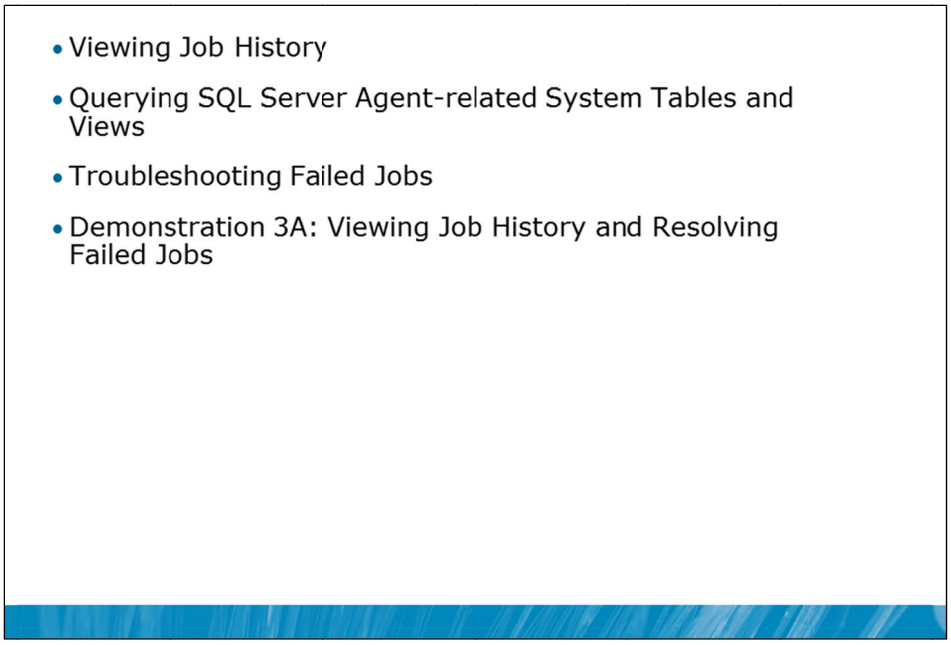
Demonstration 2A: Scripting Jobs



Demonstration Steps

1. If Demonstration 1A was not performed:
   - Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
   - In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, click **SQL Server Management Studio**. In the Connect to Server window, type **Proseware** and click **Connect**. From the **File** menu, click **Open**, click **Project/Solution**, navigate to **D:\10775A_Labs\10775A_13_PRJ\10775A_13_PRJ.ssmssln** and click **Open**.
   - From the **View** menu, click **Solution Explorer**. Open and execute the **00 – Setup.sql** script file from within Solution Explorer.
2. Open the **21 – Demonstration 2A.sql** script file.
3. Follow the instructions contained within the comments of the script file.

Lesson 3
Managing SQL Server Agent Jobs



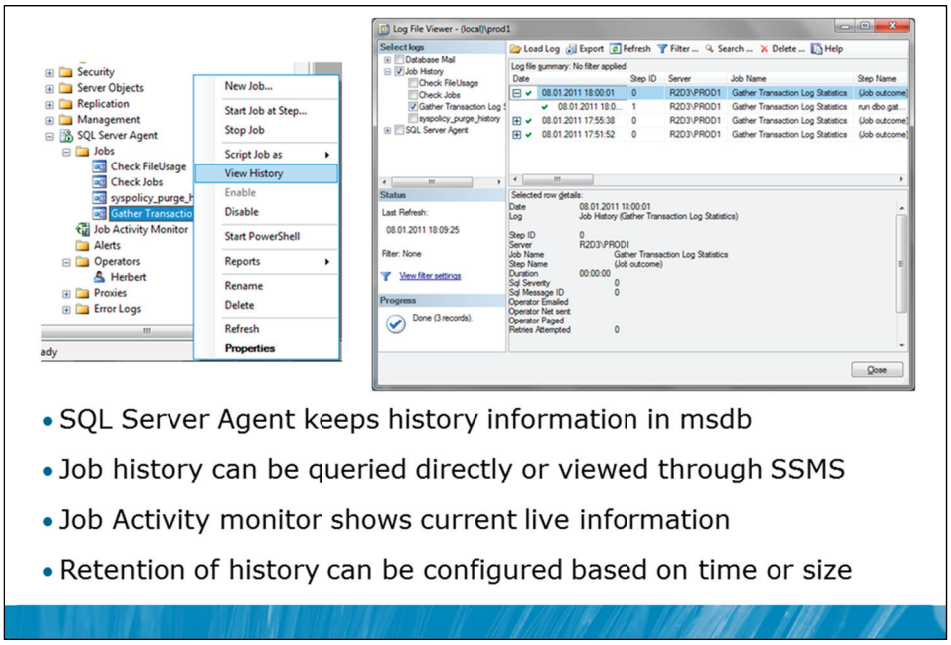
While the automation of routine administrative and other tasks is important, it is equally important to ensure that those tasks continue to execute as expected. SQL Server provides detail regarding previous and failed executions by maintaining history in tables that are contained in the msdb database. This lesson will show you how to query the history tables and provide you with an approach for troubleshooting jobs that fail or that do not perform as expected.

Objectives

After completing this lesson, you will be able to:

- View job history.
- Query SQL Server Agent-related system tables and views.
- Troubleshoot failed jobs.

Viewing Job History



Key Points

SQL Server Agent keeps track of job outcomes in system tables in the msdb database. As well as recording the outcome of entire jobs, SQL Server Agent records the outcome of each job step.

You can choose to write job outcomes to the Windows Application log and to the SQL Server log by setting notification properties for each job. The history tables in msdb are always written regardless of the configuration for log output.

Viewing Job History

Each job and each job step has an outcome. If a job fails, the failing job step outcome needs to be reviewed to see the reason why the job step failed. The history for each job can be viewed in SSMS but can also be retrieved programmatically by directly querying the system tables. Writing queries to retrieve job history will be shown in the next topic in this lesson.

The most recent 1000 entries for job history are retained by default but the retention period for job history entries can be configured based on either age or the total size of the job history data, using the property window for the SQL Server Agent service.

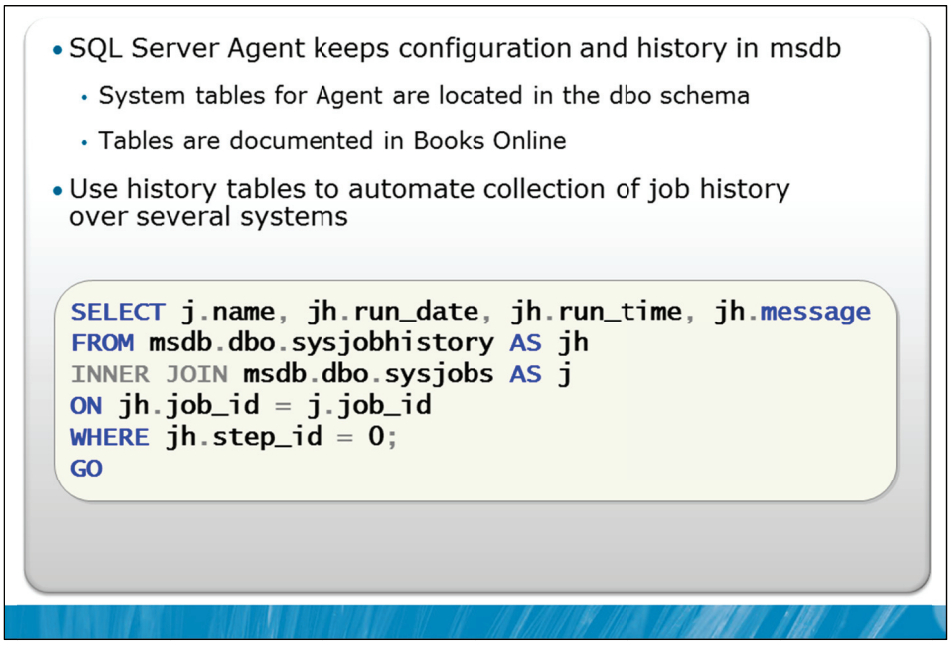
Object Explorer in SSMS also provides a Job Activity Monitor. The Job Activity Monitor offers a view of currently executing jobs and of data showing the results of the previous execution along with the scheduled time for the next execution of the job. An example of the data provided by the Job Activity Monitor is shown below:

Agent Job Activity:

Name	Enabled	Status	Last Run Out...	Last Run	Next Run	Categ
collection_set_1_noncached_co...	yes	Idle	Succeeded	9/01/2011 12:00:00 PM	9/01/2011 6:00:00 PM	Data (
collection_set_2_collection	yes	Executi...	Cancelled	11/12/2010 9:19:53 AM	not scheduled	Data (
collection_set_2_upload	yes	Idle	Succeeded	9/01/2011 1:00:00 PM	9/01/2011 1:15:00 PM	Data (
collection_set_3_collection	yes	Executi...	Cancelled	11/12/2010 9:19:53 AM	not scheduled	Data (
collection_set_3_upload	yes	Idle	Succeeded	9/01/2011 1:00:00 PM	9/01/2011 1:15:00 PM	Data (
mdw_purge_data_[MDW]	yes	Idle	Succeeded	8/01/2011 2:00:00 AM	10/01/2011 2:00:00 AM	Data (
syspolicy_purge_history	yes	Idle	Succeeded	8/01/2011 2:00:00 AM	10/01/2011 2:00:00 AM	[Unca
sysutility_get_cache_tables_data...	yes	Idle	Succeeded	8/01/2011 12:01:00 AM	10/01/2011 12:01:00 AM	[Unca
sysutility_get_cache_tables_data...	yes	Idle	Succeeded	9/01/2011 1:01:00 PM	9/01/2011 2:01:00 PM	[Unca
sysutility_get_views_data_into_c...	yes	Idle	Succeeded	9/01/2011 1:00:00 PM	9/01/2011 1:15:00 PM	[Unca

Question: How would a corrupt msdb database affect SQL Server Agent?

Querying SQL Server Agent-related System Tables and Views



Key Points

As mentioned earlier in this module, information about the configuration of SQL Server Agent and its objects such as jobs, alerts, schedules and operators is written to system tables in the msdb database. These objects are contained in the dbo schema and can be directly queried from there.

Job history is written to the dbo.sysjobhistory table and a list of jobs is written to the dbo.sysjobs table.

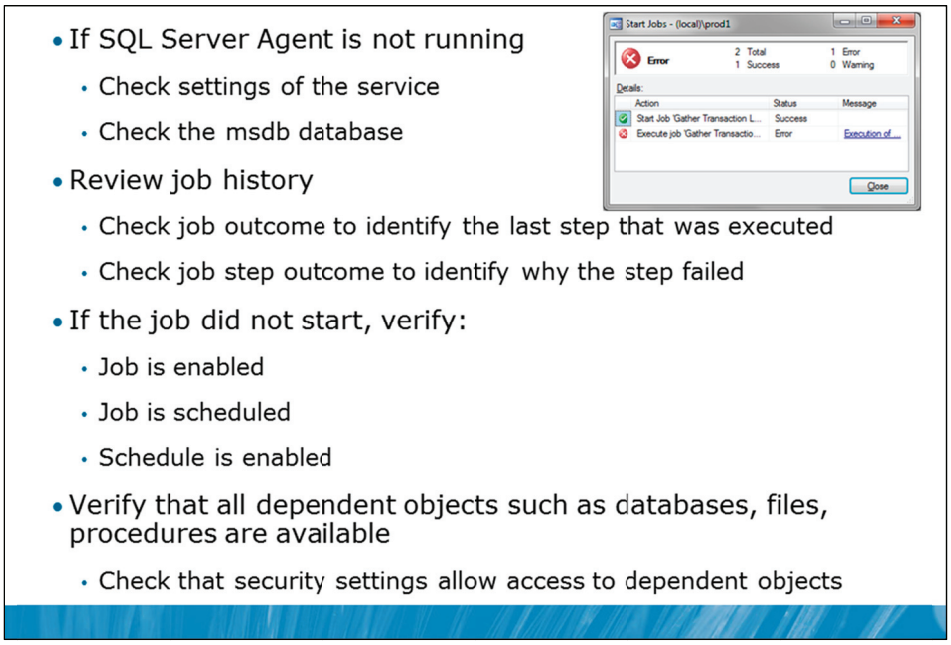
Example Query

In the example shown on the slide, the date and time that a job was last run, and the outcome are queried from the dbo.sysjobhistory table in msdb. The query joins to the dbo.sysjobs table to retrieve the name of the job.

Note that the WHERE clause specifies a step_id of zero. Job steps begin at one, not zero, but an entry in the dbo.sysjobhistory table is made with a job step_id of zero to record the overall outcome of the job. The outcome of individual job steps can be obtained by querying step_id values greater than zero.

Question: Why would querying the job history tables be important?

Troubleshooting Failed Jobs



Key Points

Jobs do not always execute as expected and sometimes they will fail to execute at all. It is important to follow a consistent process when attempting to work out why a job is failing.

There are four basic steps that need to be followed when troubleshooting jobs: checking SQL Server Agent status, reviewing job history, checking job execution, and checking access to dependencies.

Checking SQL Server Agent Status

If SQL Server Agent is not running, no jobs will be executed. Make sure the service is set to start automatically and attempt to start the service manually. If the service still will not start, check the following:

- Make sure that the service account for the service is valid, that the password for the account has not changed, and that the account is not locked out. If any of these items are incorrect, the service will not start but details about the problem will be written to the System event log on the computer.
- Check that the msdb database is online. If the msdb database is corrupt, suspect or offline, SQL Server Agent will not start.

Review Job History

Review the job outcome to identify the last step run. If the job failed because a job step failed, which is the most common situation, the error of the job step cannot be seen at this level. It is necessary to then review the individual job step outcome for the failed job.

Checking Job Execution

If SQL Server Agent is running but an individual job will not execute, check the following items:

- Make sure that the job is enabled. Disabled jobs will not run.
- Make sure that the job is scheduled. Perhaps the schedule is incorrect or the time for the next scheduled execution has not occurred yet.
- Make sure that the schedule is enabled. Both jobs and schedules can be disabled. A job will not run on a schedule that is disabled.

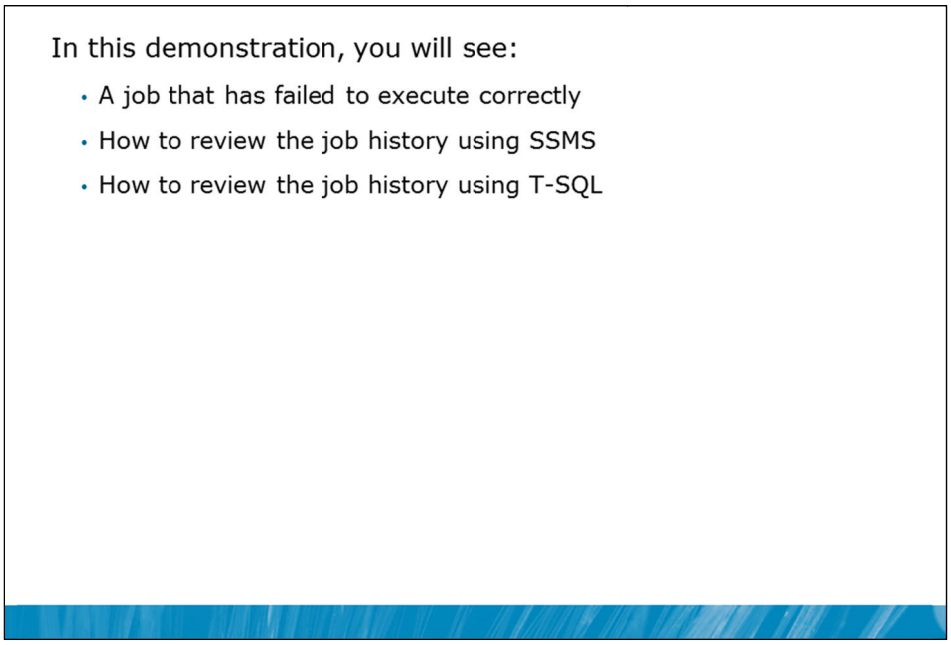
Check Access to Dependencies

Verify that all dependent objects such as databases, files, and procedures are available. In Demonstration 2A, you saw a situation where the job on the second SQL Server instance did not run because the objects required to run the job were not present.

Jobs often run in a different security context to the user that creates them. Incorrect security settings are a common problem that causes job execution to fail. The security context for job steps is discussed in more detail in Module 14.

Question: When migrating a job from test to production, what else would be required apart from moving the job itself?

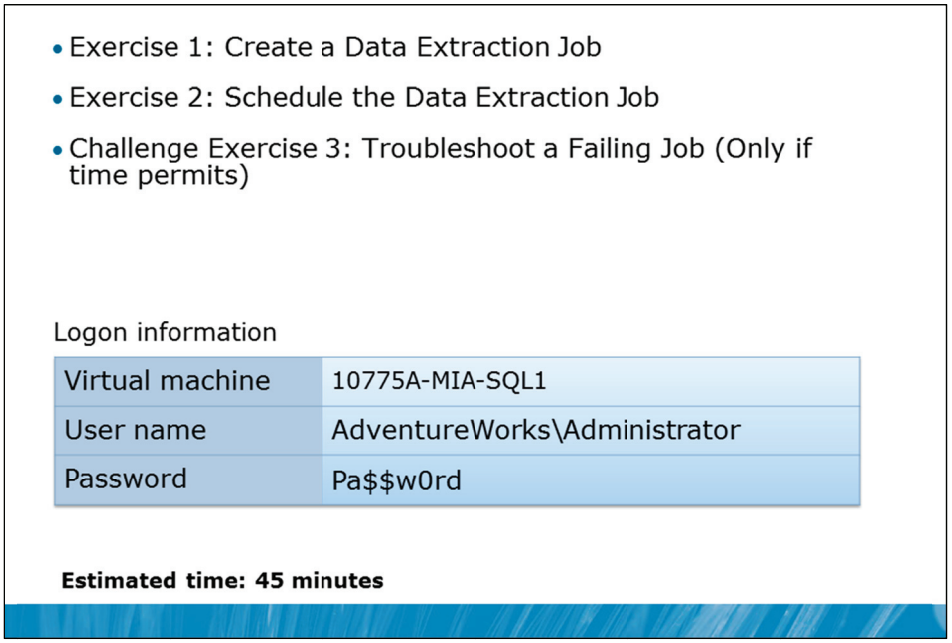
Demonstration 3A: Viewing Job History and Resolving Failed Jobs



Demonstration Steps

1. If Demonstration 2A was not performed:
   - Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
   - In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, click **SQL Server Management Studio**. In the Connect to Server window, type **Proseware** and click **Connect**. From the **File** menu, click **Open**, click **Project/Solution**, navigate to **D:\10775A_Labs\10775A_13_PRJ\10775A_13_PRJ.ssmssln** and click **Open**.
   - Open and execute the **00 – Setup.sql** script file from within Solution Explorer.
   - From the **View** menu, click **Solution Explorer**. Open the script file **21 – Demonstration 2A.sql** and follow the steps in the script file.
2. Open the **31 – Demonstration 3A.sql** script file.
3. Follow the instructions contained within the comments of the script file.

Lab 13: Automating SQL Server Management



Lab Setup

For this lab, you will use the available virtual machine environment. Before you begin the lab, you must complete the following steps:

1. Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
2. In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, and click **SQL Server Management Studio**.
3. In the Connect to Server window, type **Proseware** in the **Server name** text box.
4. In the **Authentication** drop-down list box, select **Windows Authentication** and click **Connect**.
5. In the **File** menu, click **Open**, and click **Project/Solution**.
6. In the Open Project window, open the project **D:\10775A_Labs\10775A_13_PRJ\10775A_13_PRJ.ssmssln**.
7. From the **View** menu, click **Solution Explorer**. In Solution Explorer, double-click the query **00-Setup.sql**. When the query window opens, click **Execute** on the toolbar.

Lab Scenario

There are a number of routine tasks to be performed on the Proseware instance. Previously these tasks have been performed manually and the lack of consistency in performing these tasks has caused issues for the organization. On the new instance, you need to automate these tasks using SQL Server Agent.

There is also a report about an existing SQL Server Agent job that is not performing as expected. If you have time, you need to resolve the issues with the job.

Exercise 1: Create a Data Extraction Job

Scenario

In Module 8 you created an SSIS data extraction package. The extraction process identifies prospects that have not been contacted recently. The output of this extraction process is used for planning marketing activities during the week. You need to create a job to execute the SSIS package.

The main tasks for this exercise are as follows:

1. Create the required job.
2. Test that the job executes without error.

▶ Task 1: Create the required job

- Create the required job. Call the job "Extract Uncontacted Prospects". The job needs to execute the SSIS package "Weekly Extract of Prospects to Contact" which is located on the Proseware server instance.

▶ Task 2: Test that the job executes without error

- Using Object Explorer, start the job and make sure it executes correctly.

Results: After this exercise, you should have created the data extraction job.

Exercise 2: Schedule the Data Extraction Job

Scenario

You have created a job to perform the extraction of prospects that need to be contacted. The information provided by this job is used two times during the week. A meeting is held at 9AM each Monday morning to plan the marketing activities for the week. A second planning meeting is held at 7PM on Tuesday evenings. You need to make sure that an updated list is available shortly before each meeting. You should schedule the extraction job to run each Monday at 8:30AM and each Tuesday at 6:30PM.

The main task for this exercise is as follows:

1. Schedule the data extraction job.

▶ Task 1: Schedule the data extraction job

- Create a new job schedule for each Monday at 8.30AM.
- Create a new job schedule for each Tuesday at 6.30PM.
- Assign the schedule to the data extraction job.

Results: After this exercise, you should have applied multiple schedules to the data extraction job.

Challenge Exercise 3: Troubleshoot a Failing Job (Only if time permits)

Scenario

On the Proseware server, a new SQL Server Agent job called Extract Long Page Loads was recently created. The job retrieves details of web pages that took a long time to load from the Marketing.WebLog table for further analysis by the web development team. The job is intended to run every Monday at 6AM. The job was implemented last week but no data retrieval appears to have occurred this week. You need to investigate and correct any issues with this job.

The main task for this exercise is as follows:

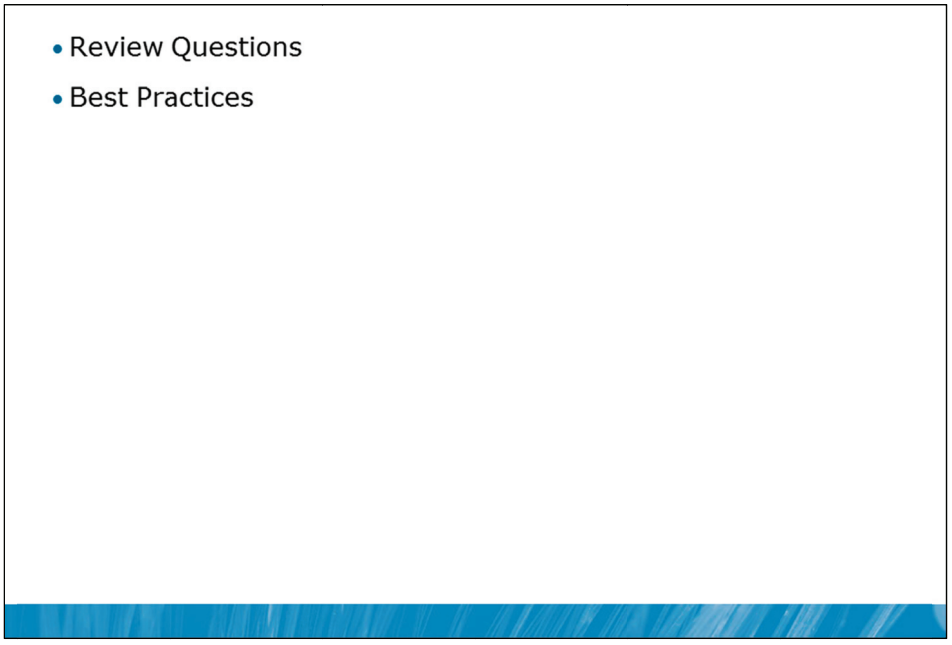
1. Troubleshoot the failing job.

▶ Task 1: Troubleshoot the failing job

- Review the job history for the failing job and identify the cause of the failure.
- Correct the problem that is preventing the job from executing successfully.
- Test that the job now runs.
- Ensure that the job is correctly scheduled.

Results: After this exercise, you should have resolved the issues with the failing job.

Module Review and Takeaways



Review Questions

1. What functions do you currently perform manually that could be placed in a job?
2. How long is the job history kept in msdb?

Best Practices

1. Use SQL Agent jobs to schedule routine jobs.
2. Create custom categories to group your jobs.
3. Script your jobs for remote deployment.
4. Use job history to review job and job step outcomes.
5. Use Job Activity Monitor to real time monitor jobs.

Module 14

Configuring Security for SQL Server Agent

Contents:

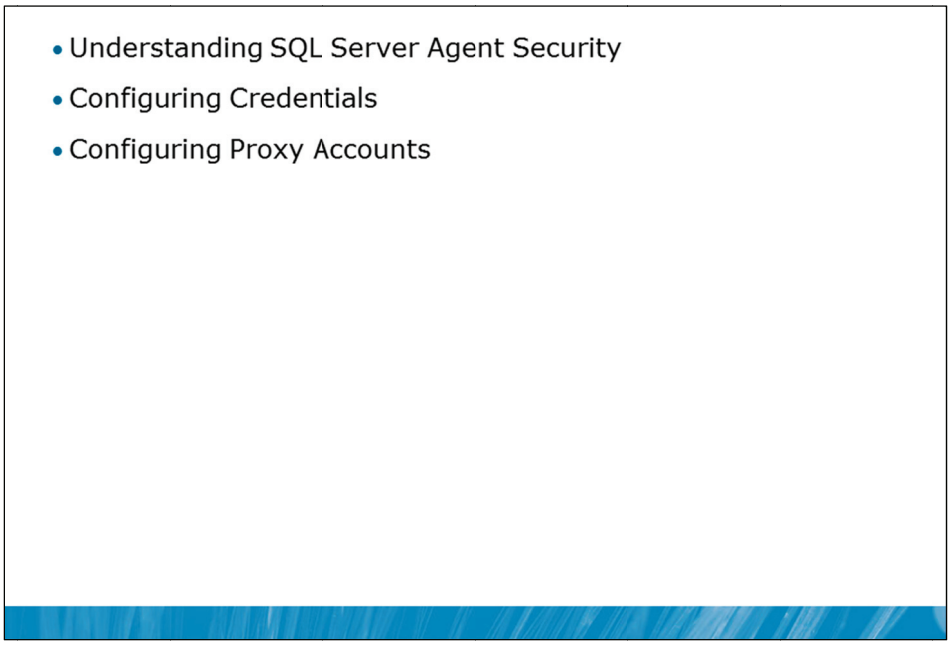
Lesson 1: Understanding SQL Server Agent Security **14-3**

Lesson 2: Configuring Credentials **14-13**

Lesson 3: Configuring Proxy Accounts **14-18**

Lab 14: Configuring Security for SQL Server Agent **14-24**

Module Overview



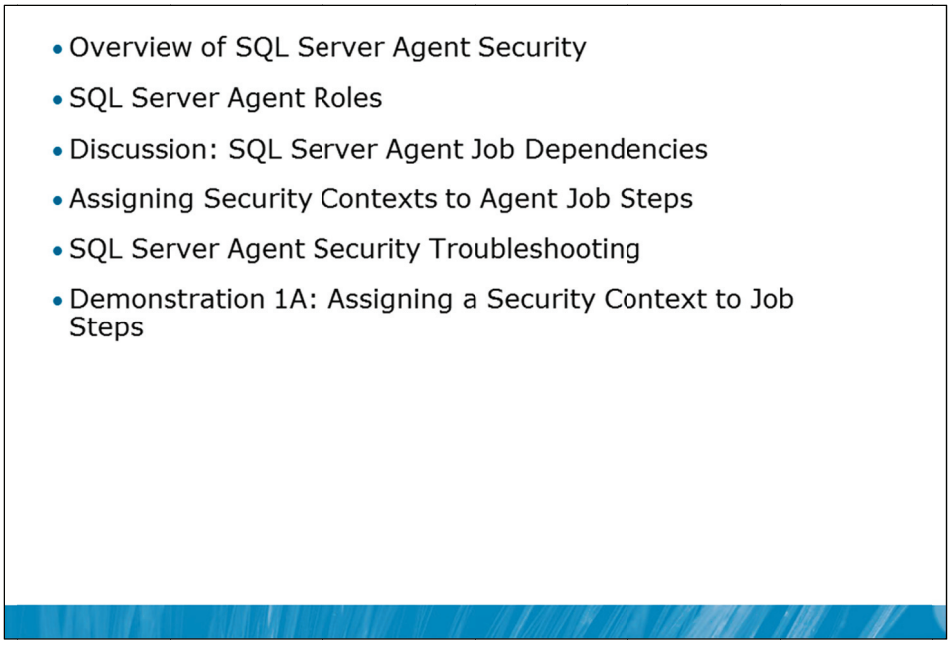
In earlier modules, you have seen the need to minimize the permissions that are granted to users, so that the users have only the permissions that they need to perform their tasks. The same logic applies to the granting of permissions to SQL Server Agent. While it is easy to execute all jobs in the context of the SQL Server Agent service account, and to configure that account as an administrative account, a poor security environment would result from doing this. It is important to understand how to create a minimal privilege security environment for jobs that run in SQL Server Agent.

Objectives

After completing this lesson, you will be able to:

- Explain SQL Server Agent security.
- Configure credentials.
- Configure Proxy accounts.

Lesson 1
Understanding SQL Server Agent Security



SQL Server Agent can be called upon to execute a wide variety of tasks. Many of the tasks that are executed by SQL Server Agent are administrative in nature but many other tasks that are executed by SQL Server Agent are performed on behalf of users. The need to be able to execute a wide variety of task types leads to the need for flexible security configuration.

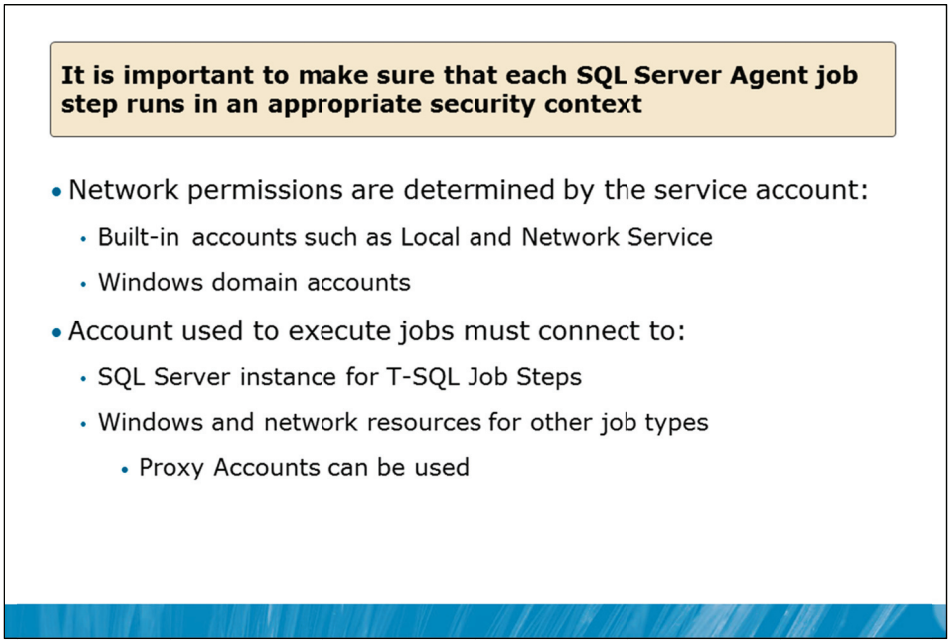
Jobs need to be able to access many types of objects. As well as objects that reside inside SQL Server, jobs often need to access external resources such as operating system files and folders. These operating system (and other) dependencies also require a configurable and layered security model, to avoid the need to grant too many permissions to the SQL Server Agent service account.

Objectives

After completing this lesson, you will be able to:

- You will be able to SQL Server Agent security.
- Describe SQL Server Agent roles.
- Assign security contexts to SQL Server Agent job steps.
- Troubleshoot SQL Server Agent security.

Overview of SQL Server Agent Security



Key Points

Like all services, the SQL Server Agent service has an identity within the Microsoft® Windows® operating system. The service startup account defines the Windows account in which the SQL Server Agent runs. The account that is used defines the permissions that the SQL Server Agent service has when accessing network resources.

Agent Service Account

For the SQL Server Agent service, you can use the built-in Local Service or Network Service account. Preferably, another specified Windows domain account can be used.

Note The Local System account option is provided for backward compatibility only and the Network Service is also not recommended for security reasons. Network Service has more capabilities than are required for the service account. An account with only the required permissions should be created and used instead.

A Windows domain account should be used and should be configured with the least possible privileges that will still allow operation. During the installation of SQL Server, a local group is created with a name in the following format:

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

This group is granted all the access privileges needed by the SQL Server Agent account. Note that this only includes the bare minimum permissions that the account needs for SQL Server Agent to function. When SQL Server Agent needs to access other resources in job steps, additional permissions are necessary.

When SQL Server Configuration Manager is used to assign an account to the service (which is the preferred and supported method for changing service accounts), SQL Server Configuration Manager places the account into the correct group. No additional special permission grants need to be made.

Note The SQL Server Agent account cannot use SQL Server Authentication for its connection to the SQL Server database engine.

SQL Server Agent jobs are executed in the context of the service account by default. The alternative is to create Proxy Accounts that will be used for job execution. Proxy Accounts will be described later in this module.

Question: What would cause a SQL Server Agent service account to need sysadmin privileges on the SQL Server instance?

SQL Server Agent Roles

- sysadmin fixed role members can administer SQL Server Agent
- Fixed database roles in the msdb control access for other users

Role	Description
SQLAgentUserRole	Control permission for jobs and schedules that they own
SQLAgentReaderRole	All permissions of the SQLAgentUserRole plus permission to view the list of all available jobs and job schedules
SQLAgentOperatorRole	Permission to manage local jobs, view properties for operators and proxies, and enumerate available proxies and alerts

Key Points

By default, only members of the sysadmin fixed server role can administer SQL Server Agent. As the SQL Server Agent job system can perform tasks that require access not only to SQL Server, but potentially also access to Windows and other network resources, it is important to control who can administer it. It is not considered good practice to make a login a member of the sysadmin fixed server role, if the only reason for their membership of that role is to administer SQL Server Agent.

Fixed Database Roles for SQL Server Agent

Fixed database roles in the msdb database are used to control access for non-sysadmin users. SQL Server 2012 includes fixed database roles for working with SQL Server Agent. The available roles, listed in order of increasing capability, are as follows:

- SQLAgentUserRole
- SQLAgentReaderRole
- SQLAgentOperatorRole

When users who are not members of one of these roles are connected to SQL Server® in SSMS, the SQL Server Agent node in Object Explorer is not visible. A user must be a member of at least one of these fixed database roles or they must be a member of the sysadmin fixed server role, before they can use SQL Server Agent.

SQLAgentUserRole

SQLAgentUserRole is the least privileged of the SQL Server Agent fixed database roles. Members of SQLAgentUserRole have permissions only on the local jobs and job schedules that they own. They cannot run multi-server jobs (master and target server jobs), and they cannot change job ownership to gain access to jobs that they do not already own. SQLAgentUserRole members can also view a list of available proxies in the Job Step Properties dialog box of SSMS.

SQLAgentReaderRole

SQLAgentReaderRole includes all the SQLAgentUserRole permissions as well as permissions to view the list of available multiserver jobs, their properties, and their history. Members of this role can also view the list of all available jobs and job schedules and their properties, not just the jobs and job schedules that they own. SQLAgentReaderRole members cannot change job ownership to gain access to jobs that they do not already own.

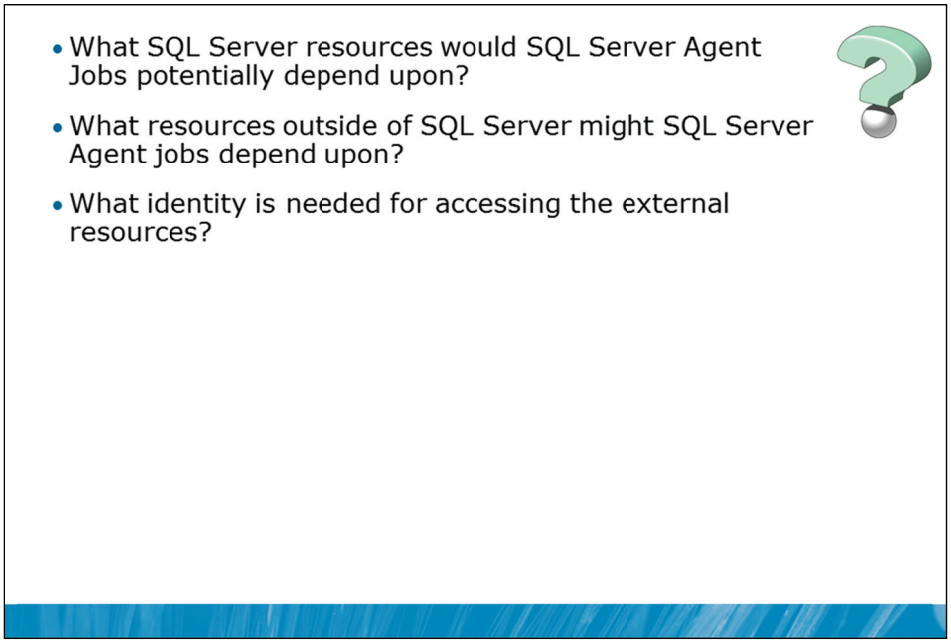
SQLAgentOperatorRole

SQLAgentOperatorRole is the most privileged of the SQL Server Agent fixed database roles. It includes all the permissions of SQLAgentUserRole and SQLAgentReaderRole. Members of this role can also view properties for operators and proxies, and enumerate the available proxies and alerts on the server.

SQLAgentOperatorRole members have additional permissions on local jobs and schedules. They can execute, stop, or start all local jobs, and they can delete the job history for any local job on the server. They can also enable or disable all local jobs and schedules on the server. To enable or disable local jobs or schedules, members of this role must use the sp_update_job and sp_update_schedule stored procedures, specifying the job name or schedule identifier parameter and the enabled parameter. If any other parameters are specified, execution of these stored procedures fails. SQLAgentOperatorRole members cannot change job ownership to gain access to jobs that they do not already own.

Question: Why should you be careful giving access to non sysadmin fixed roles members?

Discussion: SQL Server Agent Job Dependencies



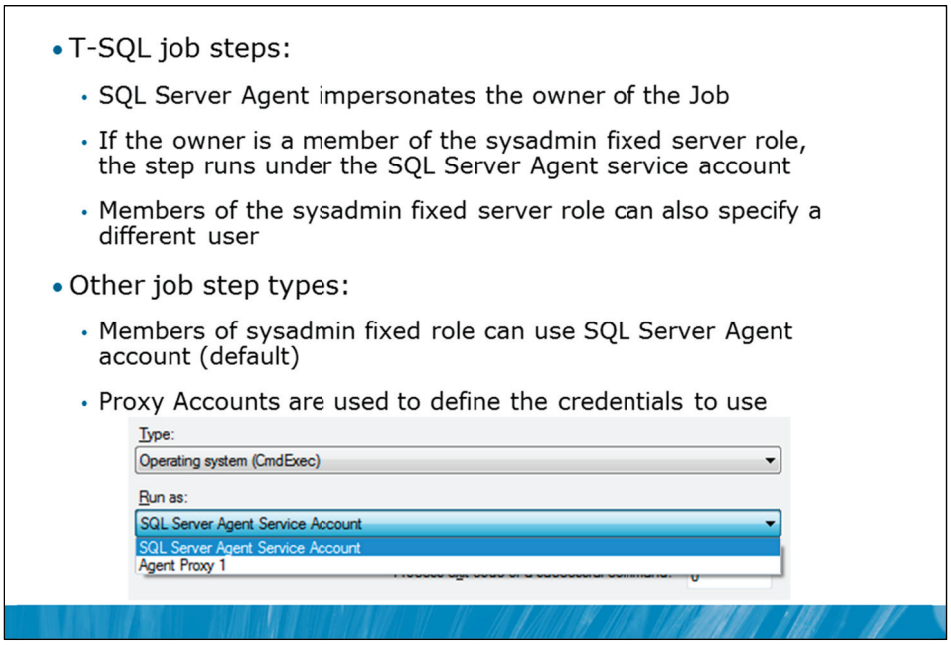
Discussion Topics

Question: What SQL Server resources would SQL Server Agent Jobs potentially depend upon?

Question: What resources outside of SQL Server might SQL Server Agent jobs depend upon?

Question: What identity is needed for accessing the external resources?

Assigning Security Contexts to Agent Job Steps



Key Points

Each job step can be assigned a security context. Job steps that execute T-SQL code need to be considered separately from other types of job steps.

T-SQL Job Steps

T-SQL job steps do not use SQL Server Agent proxies. When a T-SQL job step is executed, SQL Server Agent impersonates the owner of the job, except in the situation where the owner of the job step is a member of the sysadmin fixed server role. In that case, the job step will run in the security context of the SQL Server Agent service, unless the sysadmin chooses to have the step impersonate another user. Members of the sysadmin fixed server role can specify that the job step should impersonate another user.

Other Job Step Types

For job step types that are not T-SQL based, a different security model is used. For members of the sysadmin fixed server role, by default, other job step types still use the SQL Server Agent service account to execute job steps. Because many different types of job steps can be executed within SQL Server Agent, it is undesirable to execute them using this account. To provide for tighter control, a proxy system was introduced.

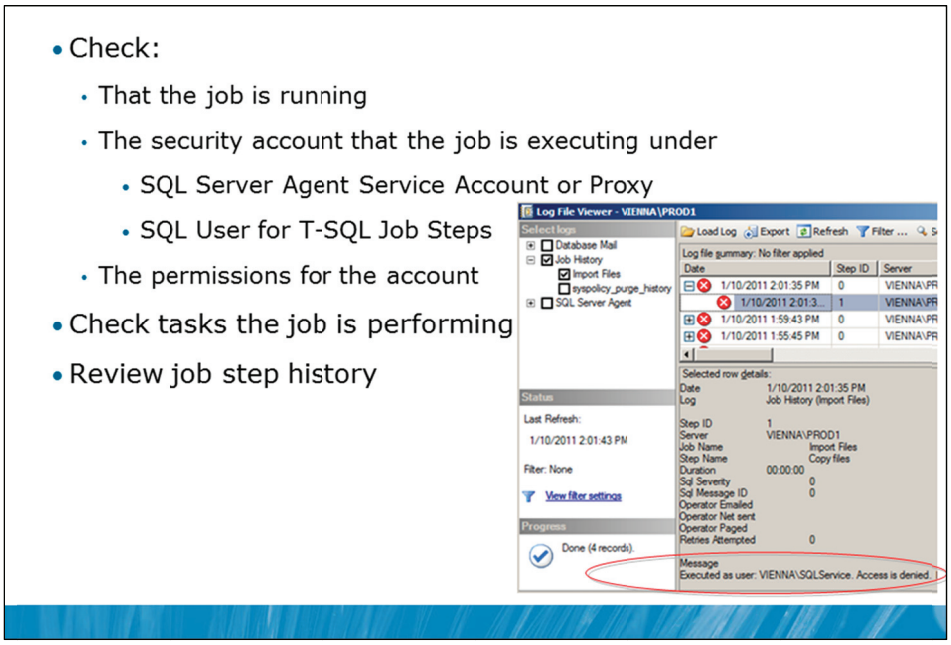
Proxy Accounts

A proxy account is used to associate a job step with a Windows identity, via an object called a Credential. Proxy accounts can be created for all available subsystems, except for T-SQL job steps. The use of proxy accounts means that different Windows identities can be used to perform the different tasks required in jobs and provides tighter security by avoiding the need for a single account to have all the permissions required to execute all jobs.

Credentials are discussed in Lesson 2 of this module and Proxy Accounts are discussed in Lesson 3.

Question: Why should a proxy account be used, even when the owner of the step is a member of the sysadmin fixed server role?

SQL Server Agent Security Troubleshooting



Key Points

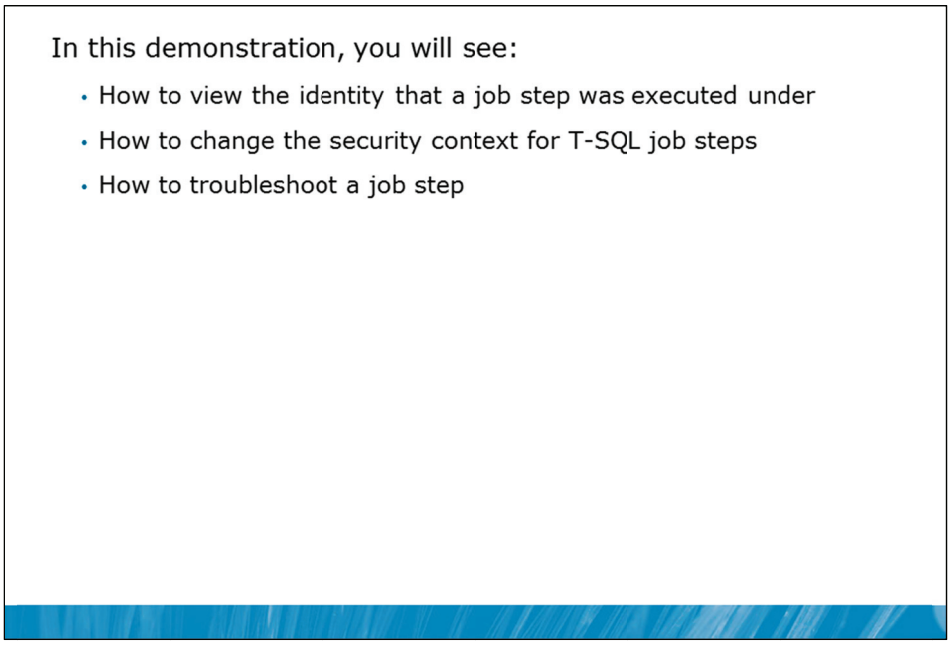
When SQL Server Agent jobs are not running as expected, security issues are a common cause of problems. To troubleshoot SQL Server Agent jobs, you should follow these steps:

1. Make sure that the job is in fact running. Look in the Job Activity log and check to see when the job has run. For each failure of a job that is indicated by a red X (as shown in the example on the slide), expand the job and find the job steps that are failing. The failing job steps will also have a red X icon.
2. Check the security account. When you click on the job step that is failing, at the bottom right hand side of the window, there is an indication of the security context that the job step ran under. Check the group membership for the account to make sure that the account should have all required permissions.
3. Check the tasks that the job step needs to perform. This includes any T-SQL objects that need to be accessed and any Windows files or resources that need to be accessed.
4. Check for each failing step, that the account that is being used to execute the step is able to access the resources that you have determined as necessary for the step.

Note Another very common problem that occurs is that job steps might specify local file paths instead of UNC paths. In general, jobs should use UNC paths to ensure they are sufficiently portable. That is, a job should be able to be migrated to another server if required.

Question: What might be the cause of a job that runs perfectly well on a test system but then fails when run on a production system?

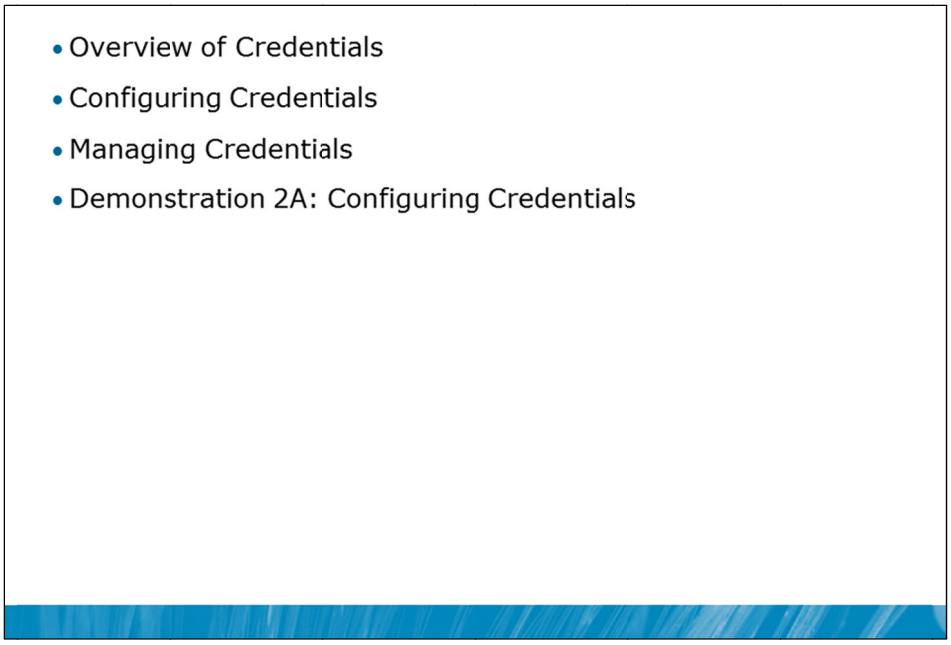
Demonstration 1A: Assigning a Security Context to Job Steps



Demonstration Steps

1. Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
2. In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, click **SQL Server Management Studio**. In the Connect to Server window, type **Proseware** and click **Connect**. From the **File** menu, click **Open**, click **Project/Solution**, navigate to **D:\10775A_Labs\10775A_14_PRJ\10775A_14_PRJ.ssmssln** and click **Open**.
3. From the **View** menu, click **Solution Explorer**. Open and execute the **00 – Setup.sql** script file from within Solution Explorer.
4. Open the **11 – Demonstration 1A.sql** script file.
5. Follow the instructions contained within the comments of the script file to execute each T-SQL batch contained in the file.

Lesson 2
Configuring Credentials



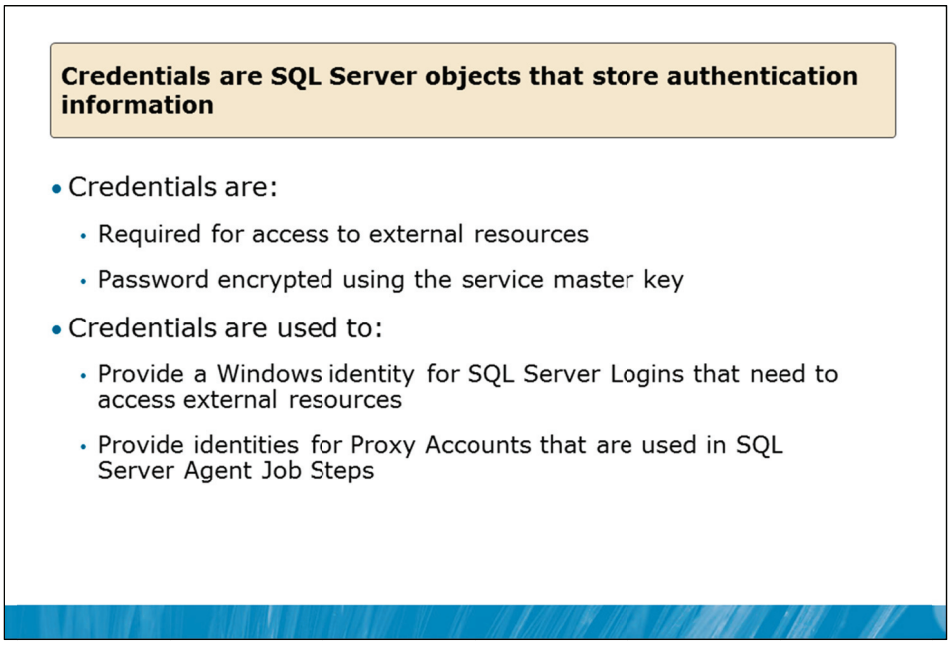
For SQL Server job steps to be able to access resources outside of SQL Server, the job steps must be executed in the security context of a Windows identity that has permission to access the required resources. Windows identities are separate from SQL Server identities, even though SQL Server can utilize Windows logins and groups. For a job step to be able to use a separate Windows identity, the job step needs to be able to logon as that identity and to be able to logon, the Windows user name and password needs to be stored somewhere. Credentials are SQL Server objects that are used to store Windows user names and passwords.

Objectives

After completing this lesson, you will be able to:

- Describe credentials.
- Configure credentials.
- Manage credentials.

Overview of Credentials



Key Points

A credential is a SQL Server object that contains the authentication information required to connect to a resource outside SQL Server. Most credentials contain a Windows user name and password.

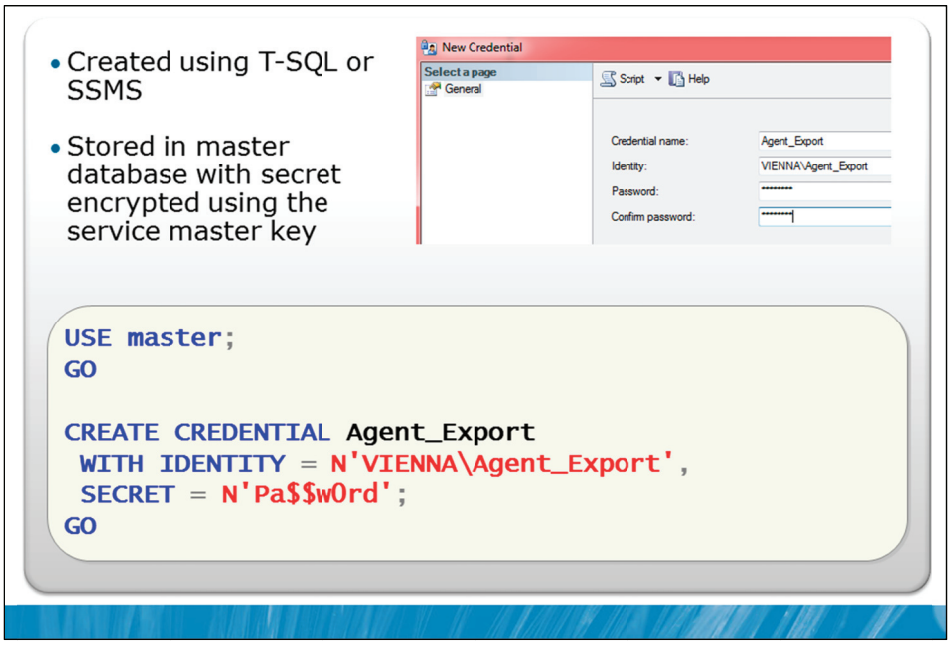
Credentials

A SQL Agent Proxy that can be used for job execution maps to a credential in SQL Server. In Lesson 3, you will see how to map a Proxy account to a credential.

SQL Server creates some system credentials that are associated with specific endpoints automatically. These automatically created credentials are called system credentials and have names that are prefixed with two hash signs (##).

Question: How does SQL Server accesses resources outside SQL Server, when the user is connected using a Windows login?

Configuring Credentials



Key Points

Credentials can be created using the T-SQL CREATE CREDENTIAL statement or by using the GUI in SSMS.

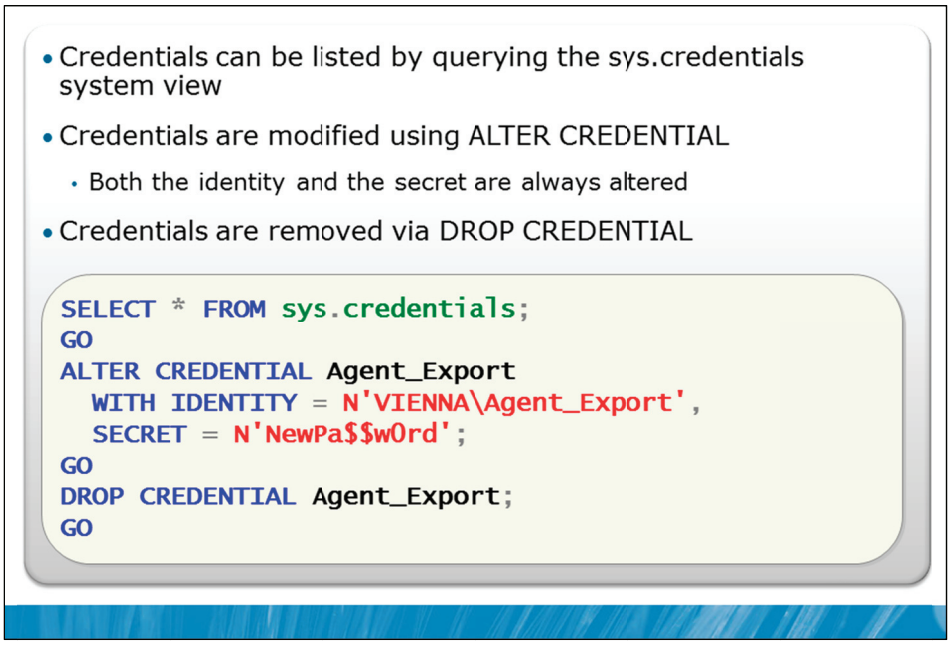
Configuring Credentials

The password for a credential is called a secret and is strongly encrypted and stored in the master database. When SQL Server first needs to perform any type of encryption, the SQL Server service generates a service master encryption key. The service master key is also used to protect the master keys for each database. (Not all databases have master keys).

Often an organization will have a policy that requires encryption keys to be replaced on a regular basis. If the service master key is regenerated, the secrets that are stored for credentials are automatically decrypted and re-encrypted using the new service master key.

Note Encryption in SQL Server is an advanced topic that is out of scope for this course. Note also that the encryption of secrets for credentials by an Extensible Key Management (EKM) Provider is also supported but also out of scope for this course.

Managing Credentials



Key Points

SQL Server provides the sys.credentials system view to provide catalog information about existing credentials. Consider the following query:

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

When executed, this query returns information similar to the following results:

Results | Messages

	credential_id	name	credential_identity	create_date	modify_date	target_type	target_id
1	65536	Agent_Export	VIENNA\Agent_Export	2011-01-11 11:36:26.863	2011-01-11 11:36:26.863	NULL	NULL

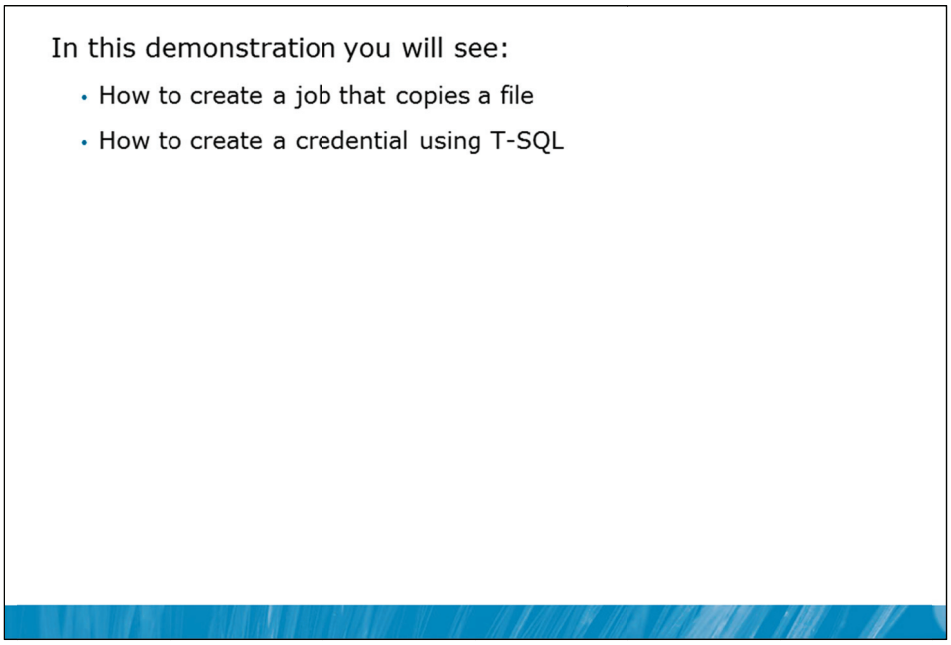
Modifying Credentials

The password for a Windows account could change over time. You can update a credential with new values by using the ALTER CREDENTIAL statement. In the example on the slide, notice how the Agent_Export credential is being updated. Both the user name and password (that is, the secret) are supplied in the ALTER CREDENTIAL statement. The ALTER CREDENTIAL command always updates both the identity and the secret.

Credentials are removed by the DROP CREDENTIAL statement.

> **Question:** What happens when the Windows user password that a credential maps to changes or expires?

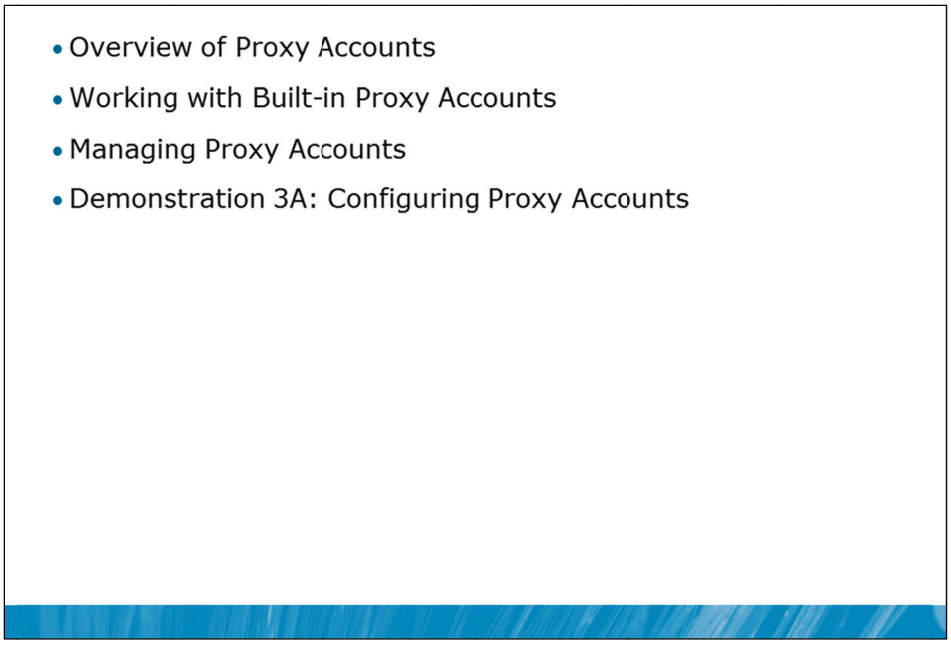
Demonstration 2A: Configuring Credentials



Demonstration Steps

1. If Demonstration 1A was not performed:
   - Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
   - In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, click **SQL Server Management Studio**. In the Connect to Server window, type **Proseware** and click **Connect**. From the **File** menu, click **Open**, click **Project/Solution**, navigate to **D:\10775A_Labs\10775A_14_PRJ\10775A_14_PRJ.ssmssln** and click **Open**.
   - From the **View** menu, click **Solution Explorer**. Open and execute the **00 – Setup.sql** script file from within Solution Explorer.
2. Open the **21 – Demonstration 2A.sql** script file.
3. Follow the instructions contained within the comments of the script file.

Lesson 3
Configuring Proxy Accounts



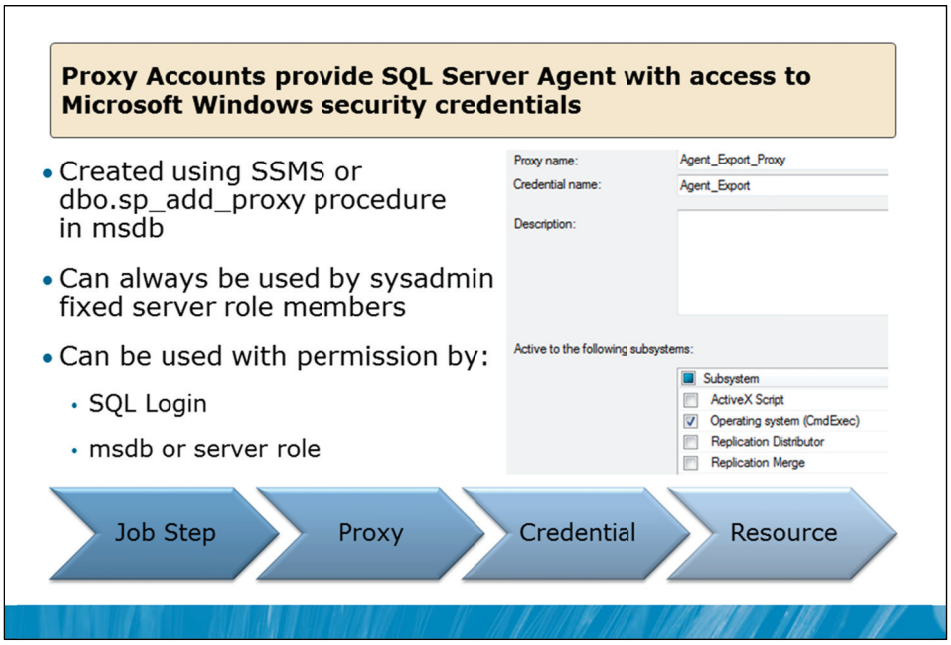
You saw in the last lesson that Credentials are used in SQL Server to store identities that are external to SQL Server. Mostly, these are Windows identities and Credentials are used to store their Windows user names and passwords. To enable a job step in a SQL Server Agent job to use a Credential, the job step is mapped to the Credential using a Proxy Account. There is a set of built-in Proxy Accounts and you can create Proxy Accounts manually.

Objectives

After completing this lesson, you will be able to:

- Describe Proxy Accounts.
- Work with built-in Proxy Accounts.
- Manage Proxy Accounts.

Overview of Proxy Accounts



Key Points

A SQL Server Proxy Account is used to define the security context that is used for a job step. A Proxy Account is typically used to provide SQL Server Agent with access to the security credentials for a Microsoft Windows user. Each Proxy Account can be associated with one or more subsystems.

A job step that uses the Proxy Account can access the specified subsystems by using the security context of the Windows user. Before SQL Server Agent runs a job step that uses a Proxy Account, SQL Server Agent impersonates the credentials defined in the Proxy Account, and then runs the job step by using that security context.

Note The Windows user that is specified in the credential must have the "Log on as a batch job" right on the computer on which SQL Server is running.

The creation of a Proxy Account does not change existing permissions for the Windows account that is specified in the credential. For example, you can create a Proxy Account for a Windows account that does not have permission to connect to an instance of SQL Server. Job steps that use that Proxy Account would be unable to connect to SQL Server.

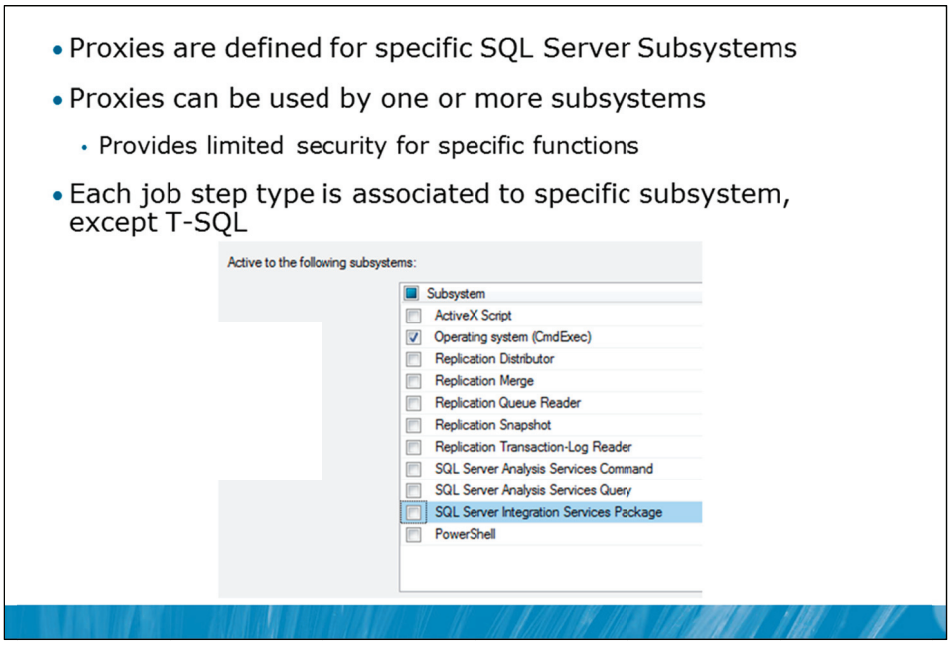
Note that a user must have permission to use a Proxy Account before they can specify the Proxy Account in a job step. By default, only members of the sysadmin fixed server role have permission to access all Proxy Accounts.

Permissions to access a Proxy Account can be granted to three types of security principals:

- SQL Server logins
- Server roles
- Roles within the msdb database.

Question: When should Proxy Accounts be used?

Working with Built-in Proxy Accounts



Key Points

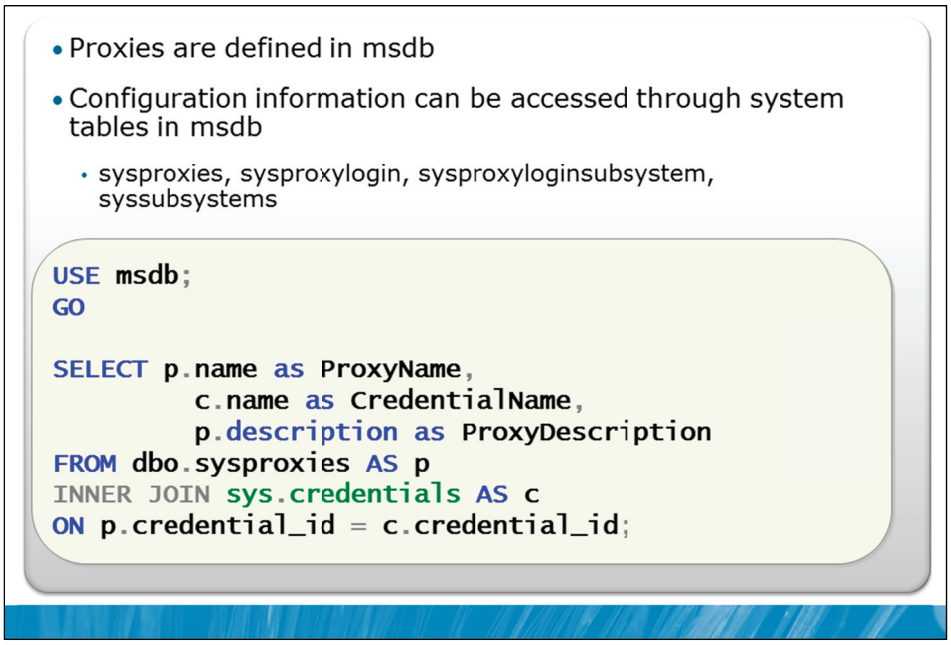
SQL Server Proxy Accounts are utilized by subsystems. A subsystem is a predefined object that represents a set of functionality available within SQL Server. Each Proxy Account can be associated with more than one subsystem.

Subsystems assist in providing security control because they segment the functions that are available to a Proxy Account. Earlier in this module, you saw that each job step runs in the context of a Proxy Account, except for T-SQL job steps. T-SQL job steps use the EXECUTE AS command to set the security context.

SQL Server Agent checks subsystem access for a Proxy Account each and every time that a job step runs. If the security environment has changed and the Proxy Account no longer has access to the subsystem, the job step fails.

Question: Why should Proxy Accounts not be assigned to all subsystems as a general rule?

Managing Proxy Accounts



Key Points

The configuration for SQL Server Agent is stored in the msdb database. Proxy Accounts are part of the SQL Server Agent, so the configuration for the Proxy Accounts is also stored in the msdb database.

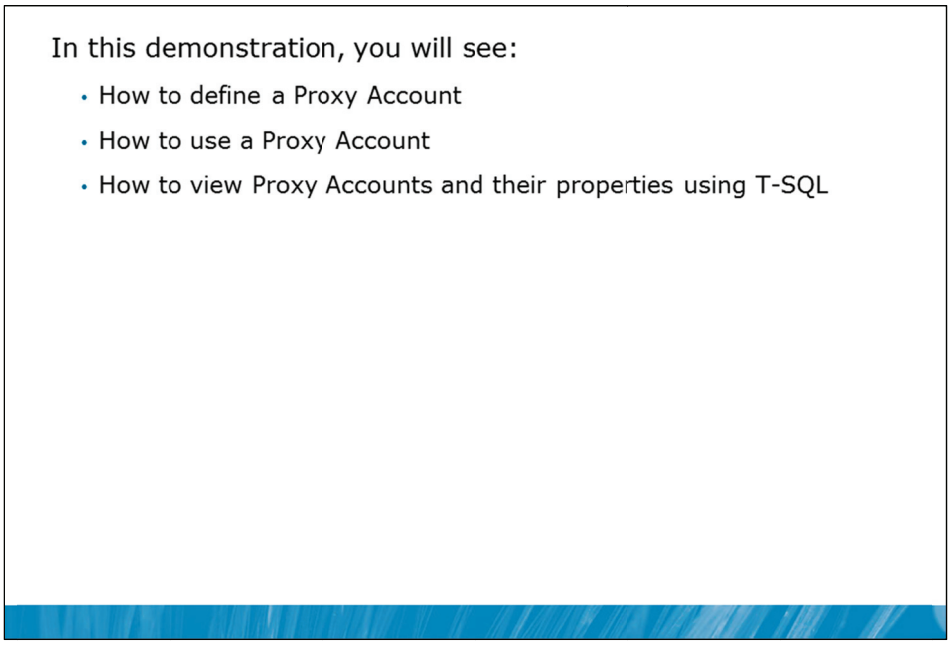
Details of the current Proxy Account configuration can be obtained through a set of system views that are shown in the following table:

System View	Description
dbo.sysproxies	Returns one row per proxy defined in SQL Server Agent
dbo.sysproxylogin	Returns which SQL Server logins are associated with each SQL Server Agent Proxy Account. Note that no entry for members of the sysadmin role is stored or returned
dbo.sysproxyloginsubsystem	Returns which SQL Server Agent subsystems are defined for each Proxy Account
dbo.syssubsystems	Returns information about all available SQL Server Agent proxy subsystems

You saw earlier that Credentials can be viewed via the sys.credentials system view. Credentials are stored in the master database, not in the msdb database.

Question: Why would Credentials be stored in the master database instead of the msdb database?

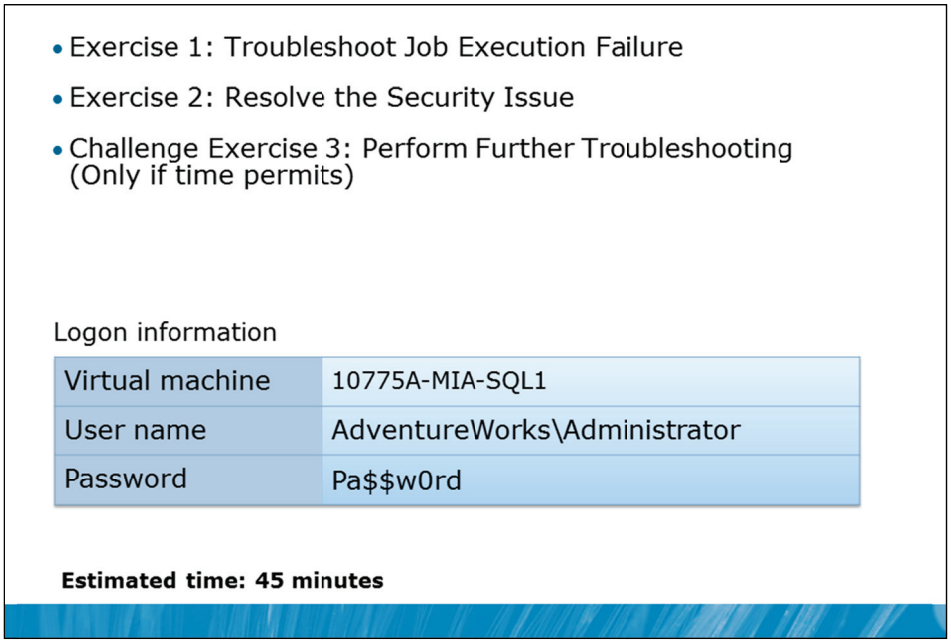
Demonstration 3A: Configuring Proxy Accounts



Demonstration Steps

1. If Demonstrations 2A were not performed:
   - Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
   - In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, click **SQL Server Management Studio**. In the Connect to Server window, type **Proseware** and click **Connect**. From the **File** menu, click **Open**, click **Project/Solution**, navigate to **D:\10775A_Labs\10775A_14_PRJ\10775A_14_PRJ.ssmssln** and click **Open**.
   - From the **View** menu, click **Solution Explorer**. Open and execute the **00 – Setup.sql** script file from within Solution Explorer.
   - Open the **21 – Demonstration 2A.sql** script file from within Solution Explorer and follow the instructions contained within the file.
2. Open the **31 – Demonstration 3A.sql** script file.
3. Follow the instructions contained within the comments of the script file.

Lab 14: Configuring Security for SQL Server Agent



Lab Setup

For this lab, you will use the available virtual machine environment. Before you begin the lab, you must complete the following steps:

1. Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
2. In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, and click **SQL Server Management Studio**.
3. In the Connect to Server window, type **Proseware** in the **Server name** text box.
4. In the **Authentication** drop-down list box, select **Windows Authentication** and click **Connect**.
5. In the **File** menu, click **Open**, and click **Project/Solution**.
6. In the Open Project window, open the project **D:\10775A_Labs\10775A_14_PRJ\10775A_14_PRJ.ssmssln**.
7. From the **View** menu, click **Solution Explorer**. In Solution Explorer, double-click the query **00-Setup.sql**. When the query window opens, click **Execute** on the toolbar.

Lab Scenario

You have deployed a job that extracts details of prospects that have not been contacted recently. You have also scheduled the job to run before each of the two marketing planning meetings that occur each week. The marketing team has deployed new functionality in Promote application to improve the planning processes. Rather than having the job scheduled, it is necessary for the Promote application to execute the job on demand.

The Promote application connects as a SQL login called PromoteApp. One of the other DBAs Terry Adams has attempted to configure SQL Server so that the PromoteApp login can execute the job. However he is unable to resolve why the job still will not run. In this lab you need to troubleshoot and resolve the problem.

Supporting Documentation

Actions that have already been taken by Terry Adams

1. Created a database user for the PromoteApp login in the msdb database.
2. Granted the PromoteApp database user permission to execute the msdb.dbo.sp_start_job stored procedure.
3. Added the PromoteApp database user to the SQLAgentOperatorRole database role.
4. Modified the Extract Uncontacted Prospects job to set the PromoteApp login as the owner of the job.
5. Created a Windows user called ExtractUser with a password of Pa$$w0rd.
6. Added the Windows user ExtractUser to the db_ssisoperator role within the msdb database.

Exercise 1: Troubleshoot Job Execution Failure

Scenario

You need to review the action that Terry Adams has already taken then also review the history log for the failing job. You need to determine why the job is failing.

The main task for this exercise is as follows:

1. Troubleshoot job execution failure.

▶ Task 1: Troubleshoot job execution failure

- Review the previous actions taken by Terry Adams as detailed in the supporting documentation for the exercise.
- View History log for the Extract Uncontacted Prospects job.
- Determine from the history the reason that the job is failing.

Results: After this exercise, you should have determined the reason that they job is failing.

Exercise 2: Resolve the Security Issue

Scenario

You have determined that a proxy account is required for the correct execution of the failing job step. You need to create and assign the proxy account then test to see if all issues have been resolved.

The main tasks for this exercise are as follows:

1. Create and assign proxy account.
2. Test to see if all problems have been resolved.

▶ Task 1: Create and assign proxy account

- Using SQL Server Management Studio, create a SQL Server credential called ExtractIdentity that is associated with the Windows user 1077XA-MIA-SQL\ExtractUser and with a password of Pa$$w0rd.
- Create a SQL Server proxy account called ExtractionProxy that is associated with the ExtractIdentity credential and which is active in the SQL Server Integration Services Package subsystem. Ensure that you grant permission to the PromoteApp login to use this credential.
- Assign the proxy account to the Extract Uncontacted Prospects job.

▶ Task 2: Test to see if all problems have been resolved

- Attempt to execute the Extract Uncontacted Prospects job.
- If the job fails, continue to Exercise 3 if you have time.

Results: After this exercise, you should have corrected a security issue with a job.

Challenge Exercise 3: Perform Further Troubleshooting (Only if time permits)

Scenario

After creating and assigning a proxy account to the job, the initial problem where SQL Server refused to execute job steps without a proxy has been resolved. However, the job still does not operate successfully. You should attempt to resolve the final issue.

The main task for this exercise is as follows:

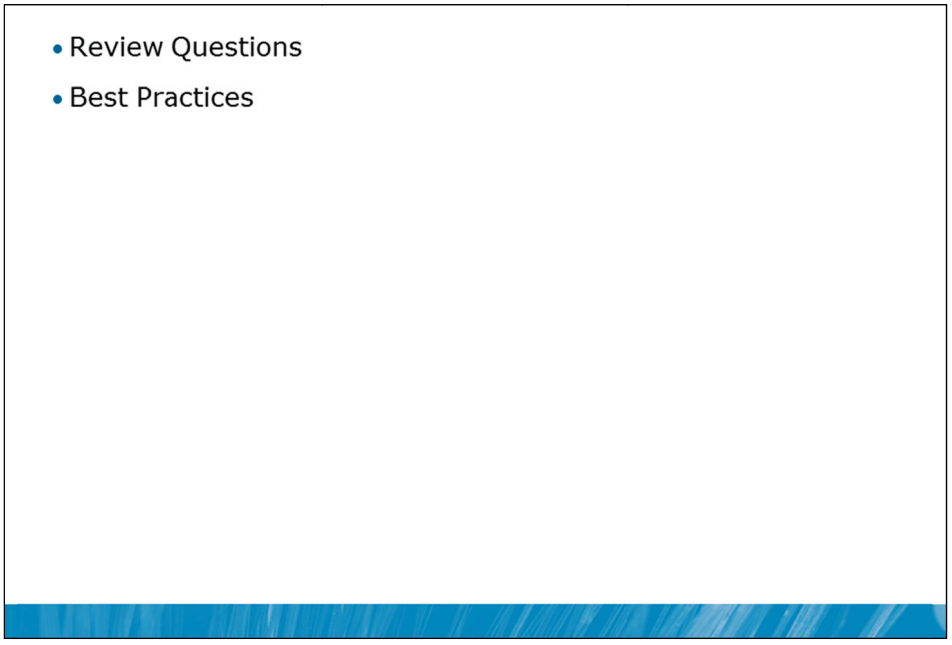
1. Perform further troubleshooting.

▶ Task 1: Perform further troubleshooting

- Locate and resolve further issues that are preventing the job from running successfully.
- Test that the job now runs successfully.

Results: After this exercise, you should have identified and resolved the remaining issues.

Module Review and Takeaways



Review Questions

1. What account types can be used to start SQL Server Agent service?
2. What can credentials be used for?

Best Practices related to a particular technology area in this module

1. Use a Windows domain user to start SQL Server Agent service account.
2. Use an account with least privileges.
3. Create Proxy Accounts with least permissions assigned for job execution.

Module 15

Monitoring SQL Server 2012 with Alerts and Notifications

Contents:

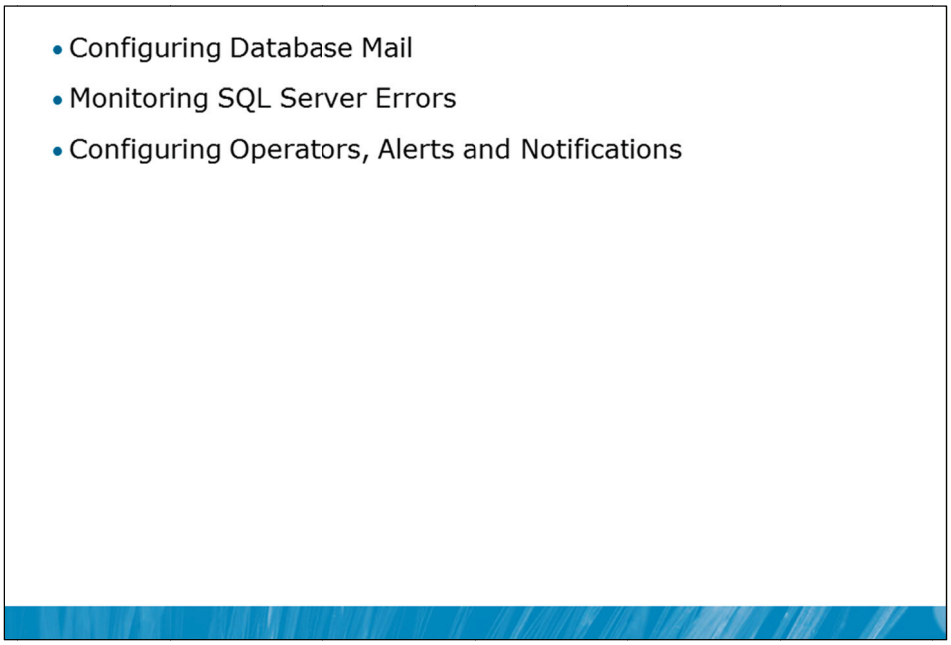
Lesson 1: Configuring Database Mail **15-3**

Lesson 2: Monitoring SQL Server Errors **15-11**

Lesson 3: Configuring Operators, Alerts and Notifications **15-18**

Lab 15: Monitoring SQL Agent Jobs with Alerts and Notifications **15-30**

Module Overview



Many database administrators work in a reactive mode where they respond when users complain that errors or problems are occurring. It is important to try to move from a reactive mode of operation to a more proactive mode.

One key aspect of managing Microsoft® SQL Server® in a proactive manner is to make sure that you are aware of events that occur in the server, as they happen. At first you might consider that this would still be a reactive approach. However, there are many types of issues that can arise that are not directly apparent to users of the database applications. SQL Server logs a wealth of information about issues and you can configure SQL Server to advise you automatically when these issues occur via alerts and notifications.

The most common way that SQL Server database administrators receive details of events of interest is via email. SQL Server can be configured to send mail via an existing SMTP mail server.

Objectives

After completing this lesson, you will be able to:

- Configure database mail.
- Monitor SQL Server errors.
- Configure operators, alerts and notifications.

Lesson 1
Configuring Database Mail

SQL Server needs to be able to advise administrators when issues arise that require the attention of the administrators. Electronic mail (email) is the most commonly used mechanism for notifications from SQL Server. The Database Mail feature of SQL Server is used to connect to an existing SMTP server, when SQL Server needs to send email.

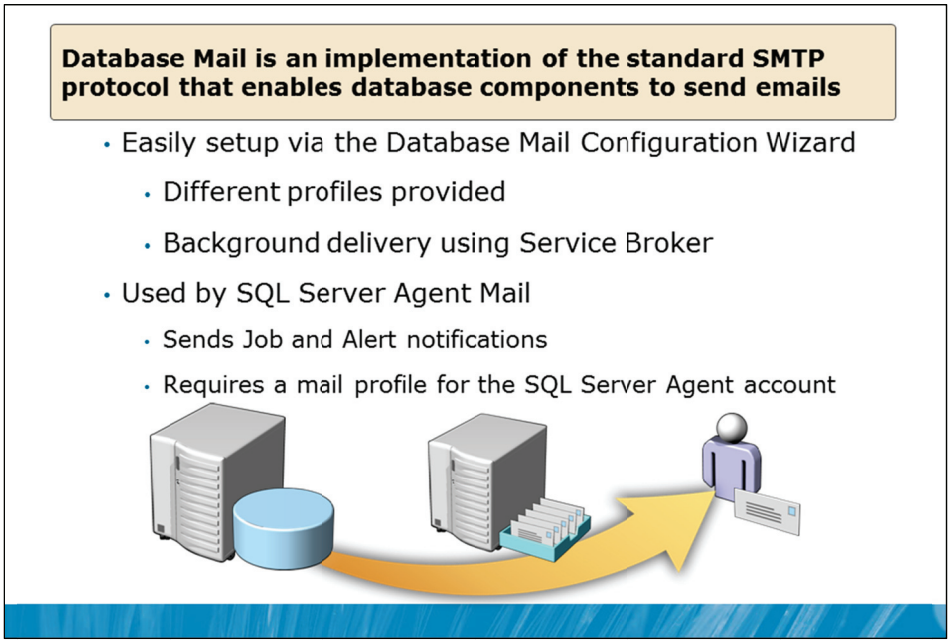
SQL Server can be configured with multiple email profiles and configured to control which users can utilize the email features of the product. It is important to be able to track and trace emails that have been sent. SQL Server allows you to configure a policy for the retention of emails.

Objectives

After completing this lesson, you will be able to:

- Describe database mail.
- Configure database mail profiles.
- Configure database mail security.
- Configure database mail retention.

Overview of Database Mail



Key Points

Database Mail sends email through a Simple Mail Transport Protocol (SMTP) server. There must be an available SMTP server on the network that accepts the mail.

Configuring Database Mail

To enable and configure Database Mail accounts and profiles, use the Database Mail Configuration Wizard.

While the configuration details for Database Mail are stored in the msdb database along with all other SQL Server Agent configuration data, SQL Server Agent caches profile information in memory, so that it is possible for SQL Server Agent to send email in situations where the SQL Server database engine is no longer available.

Database Mail can be used to send email as part of a SQL Server Agent job, in response to an alert being raised, or on behalf of a user by the execution of the sp_send_dbmail system stored procedure.

SQL Mail

Note that for backwards compatibility, earlier versions of SQL Server included a feature called SQL Mail. SQL Mail was a Messaging Application Programming Interface (MAPI)–based email feature that you could use to configure SQL Server to send and receive email via Microsoft Exchange Server or other MAPI-based email servers. SQL Mail is not supported on SQL Server 2012.

SMTP Relay

Most SMTP servers today are configured, by default, to deny all email relay. A server that is configured to permit relay is willing to accept email from another server even though the target for the email is not in the mail server's domain. The mail server then forwards the email to its final destination.

The refusal to relay email is important for avoiding spam related issues. Email servers that do not have this protection are called "open relay" servers and are a target for misuse. Blacklists for email servers that regularly send spam are maintained by many companies. By relaying email via other servers, it appears that those other servers are sending the spam, not the server that initially sent the spam. Because the most common configuration for mail servers is to deny all relaying activity, the SMTP server must be configured to permit the relay of emails from SQL Server if necessary.

Question: Why must mail administrators be included in discussions, when planning a database mail configuration?

Database Mail Profiles

- A Database Mail profile defines one or more email accounts
  - Defines configuration used to send mails
  - Allows multiple Database Mail Accounts for reliability
- Default profile for a login is used when not specified

Profile Type	Description
Private	Is accessible only to specific users or roles. A default private profile takes precedence over the default public profile
Public	Can be used by any user or role with permissions to use it

Key Points

A Database Mail profile is a collection of Database Mail accounts. At least one Database Mail account is required. If more than one Database Mail account is defined for a profile, the accounts are used in a predefined order in the attempt to send mails. The level of redundancy provided by the use of multiple email profiles can help to improve overall reliability.

Profiles can be private or public. Private profiles are strictly controlled and are only available to specified users or roles. By comparison, public profiles can be used by any user or role that has been given permission to use the profile.

Multiple Profiles

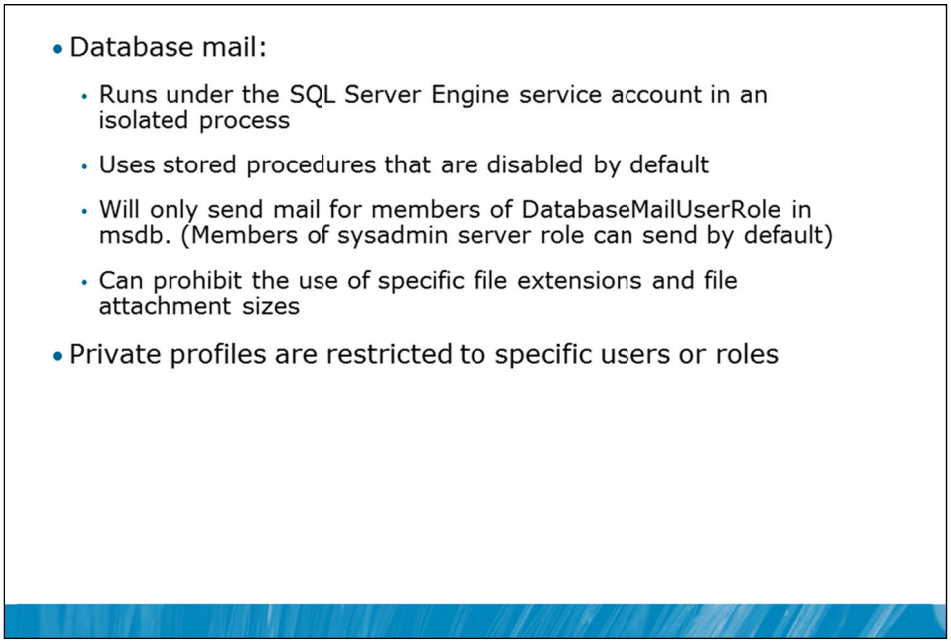
It is possible to create multiple configurations by the use of different profiles. For example a profile can be created to send mail to an internal SMTP server, using an internal email address, for mails sent by SQL Server Agent. A second profile could be created for use by a database application that needs to send external email notifications to customers.

Each database user might have access to several profiles. If no profile is specified when sending a mail, the default profile will be used. If both private and public profiles exist, precedence is given to a private default profile over a public default profile. If no default profile is specified or if a non-default profile should be used, the profile name must be specified as a parameter to the sp_send_dbmail system stored procedure as shown in the following code:

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

Question: If a user has access to several profiles, which profile is used when no profile is specified?

Database Mail Security



Key Points

The choice of service account for the SQL Server service is important when configuring Database Mail. If SQL Server is configured to run as the Local Service account, it does not have permission to make outgoing network connections. In this case, Database Mail cannot contact an email server located on a different computer.

Database Mail Stored Procedures

To minimize the security surface of SQL Server, the system extended stored procedures that are used for Database Mail are disabled by default. When you run the Database Mail Configuration Wizard, the procedures are enabled for you. If you wish to configure Database Mail manually, you can enable the Database Mail system extended stored procedures by setting the sp_configure option "Database Mail XPs" to the value 1.

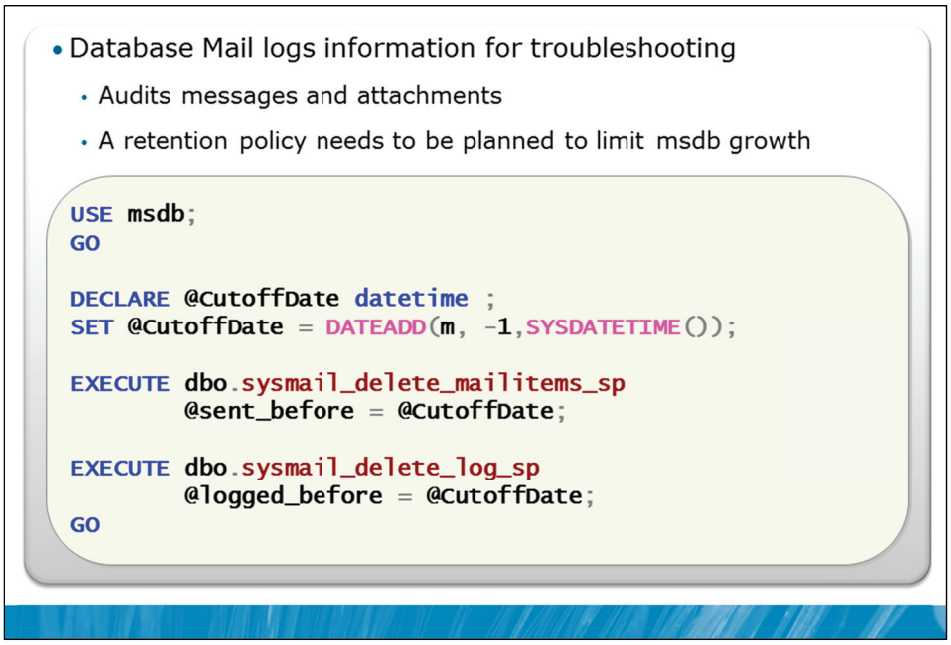
Security and Attachment Limitations

Not all SQL Server users are permitted to send emails. The ability to send emails is limited to members of the database role called DatabaseMailUserRole in the msdb database. Members of the sysadmin fixed server role can also send database mail.

You can also limit both the types and size of attachments that can be included in emails that are sent by Database Mail. This limitation can be configured using the Database Mail Configuration Wizard or by calling the dbo.sysmail_configure_sp system stored procedure in the msdb database.

Question: Why can't database mail be used with a remote SMTP server when using the Local Service account for the database engine?

Database Mail Logs and Retention



Key Points

SQL Server logs messages in internal tables in the msdb database. Log messages can be viewed by querying the dbo.sysmail_log table. The level of logging that is carried out by SQL Server can be configured to one of following three levels:

Logging Level	Description
Normal	Only errors are logged
Extended	Errors, warnings, and informational messages are logged
Verbose	As per Extended plus success messages and a number of internal messages

You can configure the Logging Level parameter by using the Configure System Parameters dialog box of the Database Mail Configuration Wizard, or by calling the dbo.sysmail_configure_sp stored procedure in the msdb database.

Verbose level should only be used for troubleshooting purposes as it can generate a large volume of log entries.

Database Mail Tables and Views

Internal tables in the msdb database are used to hold the email messages and the attachments that are sent from Database Mail, together with the current status of each message. Database Mail updates these tables as each message is processed.

You can track the delivery status of an individual message by viewing information in the following views:

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

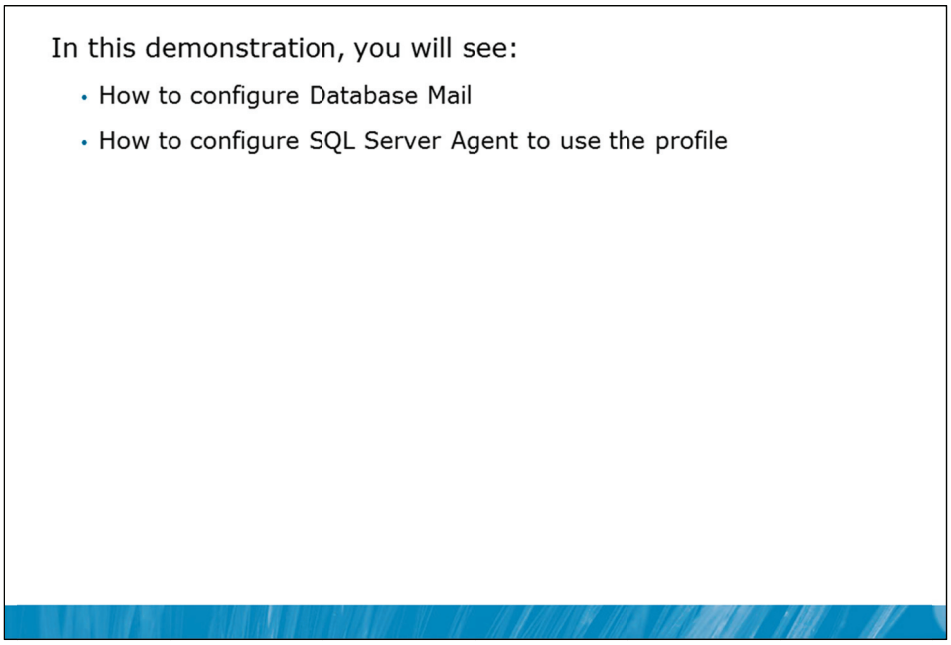
To see details of email attachments, query the dbo.sysmail_mailattachments view.

Retention Policy

Database Mail retains outgoing messages and their attachments in the msdb database. This means that there is a need to plan a retention policy for email messages and log entries. If the volume of Database Mail messages and related attachments is high, plan for substantial growth of the msdb database.

Periodically delete messages to regain space and to comply with your organization's document retention policies. For example, the example in the slide shows how to delete messages, attachments, and log entries that are more than one month old. You could schedule these commands to be executed periodically by creating a SQL Server Agent job.

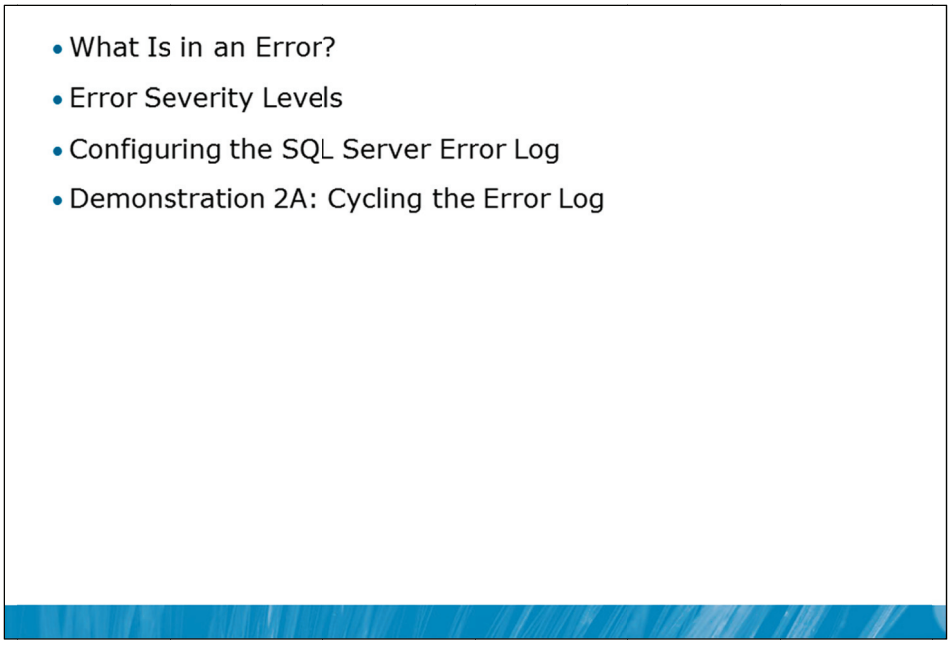
Demonstration 1A: Configuring Database Mail



Demonstration Steps

1. Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
2. In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, click **SQL Server Management Studio**. In the Connect to Server window, type **Proseware** and click **Connect**. From the **File** menu, click **Open**, click **Project/Solution**, navigate to **D:\10775A_Labs\10775A_15_PRJ\10775A_15_PRJ.ssmssln** and click **Open**.
3. From the **View** menu, click **Solution Explorer**. Open and execute the **00 – Setup.sql** script file from within Solution Explorer.
4. Open the **11 – Demonstration 1A.sql** script file.
5. Follow the instructions contained within the comments of the script file.

Lesson 2
Monitoring SQL Server Errors



It is important to understand the core aspects of errors as they apply to SQL Server. In particular, you need to consider:

- The nature of errors.
- The locations where errors can occur when T-SQL code is being executed.
- The data that is returned by errors.
- The severities that errors can exhibit.

Severe SQL Server errors are recorded in the SQL Server Error Log. It is important to know how to configure the log.

Objectives

After completing this lesson, you will be able to:

- Describe what an error is.
- Describe error severity levels.
- Configure the SQL Server error log.

What Is in an Error?

- Errors raised by the database engine have the following attributes:

Attribute	Description
Error number	Each error message has a unique error number
Error Message	String containing diagnostic info about the cause of the error
Severity	Indicates how serious the error is
State	Value used to determine the location in code at which an error occurred
Procedure Name	Name of the stored procedure or trigger in which the error occurred (if applicable)
Line Number	Indicates which line of a batch, stored procedure, trigger or function the error was fired

Key Points

An error is itself an object and has properties as shown in the table.

Error Attributes

It might not be immediately obvious that a SQL Server error (or sometimes called an exception) is itself an object. Errors return a number of useful properties (or attributes).

Error numbers are helpful when trying to locate information about the specific error, particularly when searching online for information about the error.

You can view the list of system-supplied error messages by querying the sys.messages catalog view:

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

When executed, this command returns the following:

Results | Messages

	message_id	language_id	severity	is_event_logged	text
1	21	1028	20	0	警告: 嚴重錯誤 %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	Warnung: Schwerwiegender Fehler %1! um %2!. Notieren Sie den Fehler und d
3	21	1033	20	0	Warning: Fatal error %d occurred at %S_DATE. Note the error and time, and c
4	21	1036	20	0	Avertissement : erreur irrécupérable %1! à %2!. Prenez note de l'erreur et de l'h
5	21	1040	20	0	Avviso: errore irreversibile %1! alle %2!. Prendere nota dell'errore e dell'ora in cu
6	21	1041	20	0	警告: %2! で致命的なエラー %1! が発生しました。エラーと発生時刻を記録してシステム
7	21	1042	20	0	경고: %2!에 오류 %1!이(가) 발생했습니다. 오류와 시각을 기록한 다음 시스템 관

Note that there are multiple messages with the same message_id. Error messages are localizable and can be returned in a number of languages. A language_id of 1033 is the English version of the message. You can see an English message in the third line of the output above.

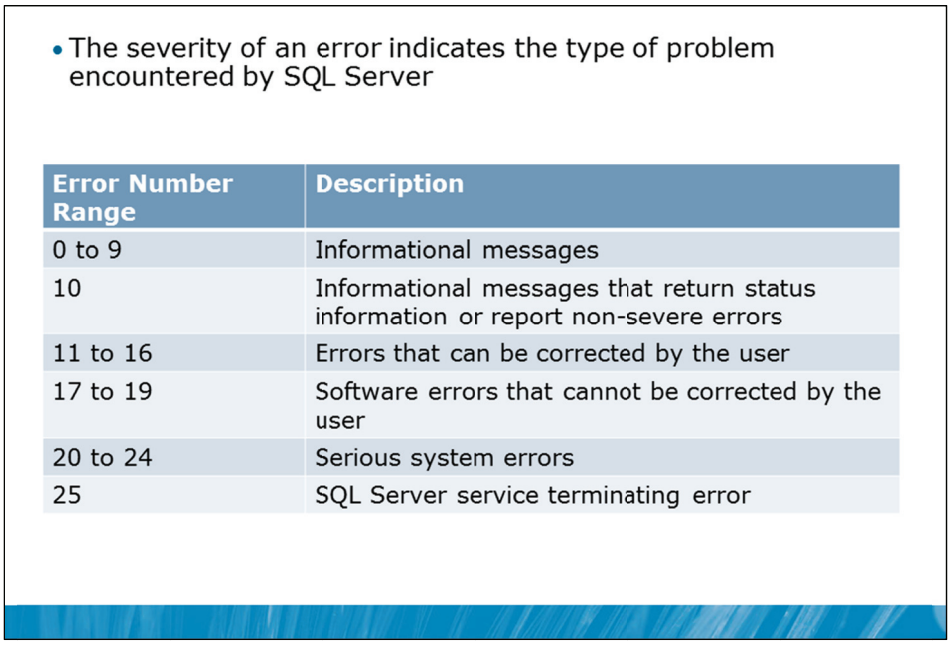
Severity indicates how serious the error is. It is described further in the next topic.

State is defined by the author of the code that raised the error. For example, if you were writing a stored procedure that could raise an error for a missing customer and there were five places in the code that this message could occur, you could assign a different state to each of the places where the message was raised. This would help later to troubleshoot the error.

Procedure name is the name of the stored procedure that that error occurred in and Line Number is the location within that procedure. In practice, line numbers are not very helpful and not always applicable.

Question: In which language is an error raised?

Error Severity



Key Points

The severity of an error indicates the type of problem encountered by SQL Server. Low severity values are informational messages and do not indicate true errors. Error severities occur in ranges.

Values from 0 to 10

Values from 0 to 9 are purely informational messages. When queries that raise these are executed in SQL Server Management Studio, the information is returned but no error status information is provided. For example, consider the following code executed against the AdventureWorks database:

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

When executed, it returns a count as expected. However, if you look on the Messages tab in SQL Server Management Studio, you will see the following:

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

(1 row(s) affected)
```

Note that no error really occurred but SQL Server is warning you that it ignored NULL values when counting the rows. Note that no status information is returned.

Severity 10 is the top of the informational messages.

Values from 11 to 16

Values from 11 to 16 are considered errors that the user can correct. Typically they are used for errors where SQL Server assumes that the statement being executed was in error.

Here are a few examples of these errors:

Error Severity Example
11 indicates that an object does not exist
13 indicates a transaction deadlock
14 indicates errors such as permission denied
15 indicates syntax errors

Values from 17 to 19

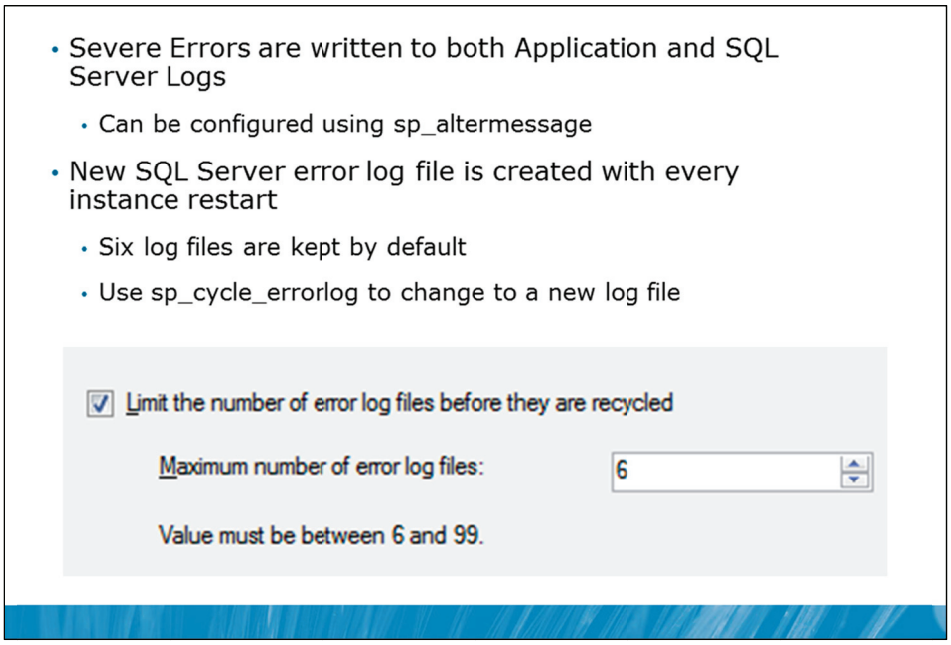
Values from 17 to 19 are considered serious software errors that the user cannot correct. For example, severity 17 indicates that SQL Server has run out of resources (memory, disk space, locks, etc.).

Values above 19

Values above 19 tend to be very serious errors that normally involve errors with either the hardware or SQL Server itself. It is common to ensure that all errors above 19 are logged and alerts generated on them.

Question: In which of the error number ranges shown on the slide, would you expect to see a syntax error?

Configuring the SQL Server Error Log



Key Points

Important messages (particularly those that would be considered as severe error messages) are logged to both the Windows Application Event Log and SQL Server Error Log. The sys.messages view shows the available error messages and indicates which messages will be logged by default. You can control the logging behavior of individual messages by using the sp_altermessage system stored procedure.

The SQL Server Error Log is located by default in the folder:

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

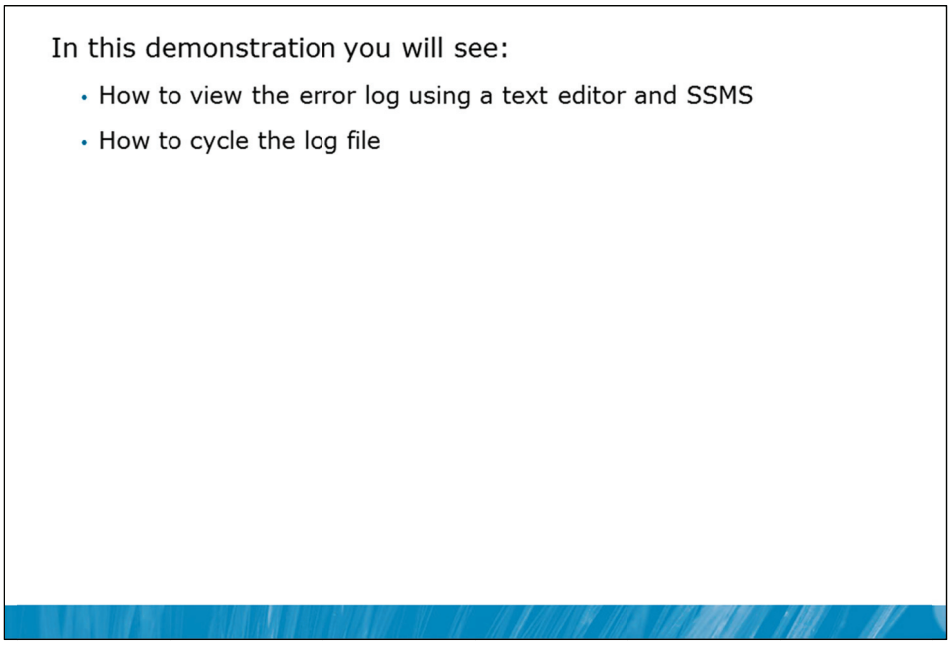
The log files are named ERRORLOG.n where n is the log file number. The log files are text files and can be viewed using any text editor or by using the Log Viewer provided in SSMS.

By default, SQL Server retains backups of the previous six logs and gives the most recent log backup the extension .1, the second most recent the extension .2, and so on. The current error log has no extension. The number of log files that should be retained can be configured in SSMS using the right-click Configure option from the SQL Server Logs node in Object Explorer.

Recycling Log Files

The log file cycles with every restart of the SQL Server instance. On occasions, you might want to remove excessively large log files. By using the system stored procedure sp_cycle_errorlog, you can close the existing log file and open a new log file on demand. If there is a regular need to recycle the log file, you could create a SQL Server Agent job to execute the sp_cycle_errorlog system stored procedure on a schedule. Cycling the log can let you stop the current error log becoming too large.

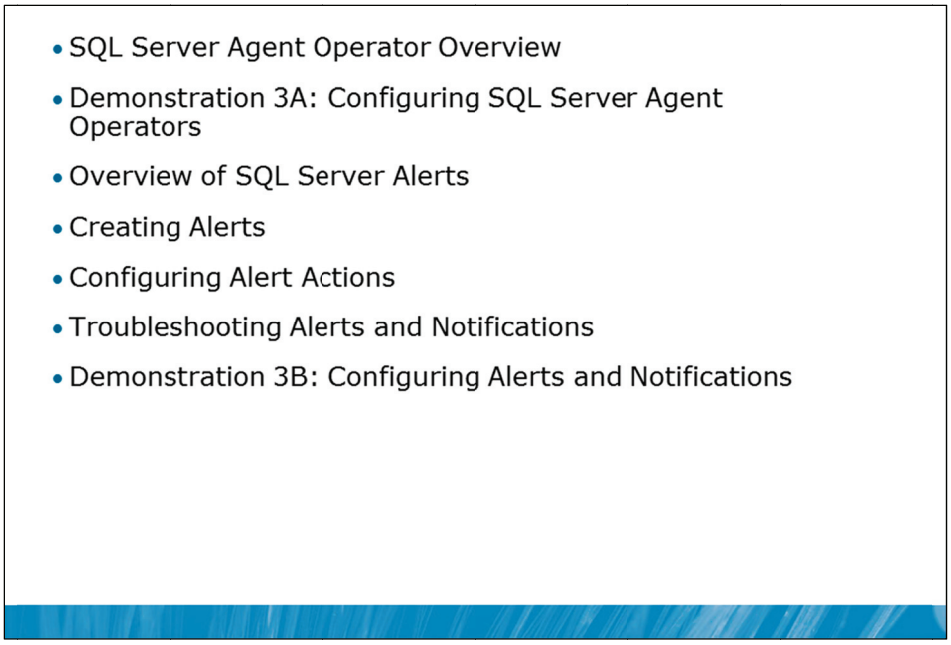
Demonstration 2A: Cycling the Error Log



Demonstration Setup

1. If Demonstration 1A was not performed:
   - Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
   - In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, click **SQL Server Management Studio**. In the Connect to Server window, type **Proseware** and click **Connect**. From the **File** menu, click **Open**, click **Project/Solution**, navigate to **D:\10775A_Labs\10775A_15_PRJ\10775A_15_PRJ.ssmssln** and click **Open**.
   - From the **View** menu, click **Solution Explorer**. Open and execute the **00 – Setup.sql** script file from within Solution Explorer.
2. Open the **21 – Demonstration 2A.sql** script file.
3. Follow the instructions contained within the comments of the script file.

Lesson 3
Configuring Operators, Alerts and Notifications



Earlier in this module, you have seen that it is important for SQL Server to be able to send messages to an administrator when events that need administrative attention occur.

Many SQL Server systems will have a number of administrators. SQL Server Agent allows you to configure Operators that are associated with one or more administrators and to determine when each of the operators should be contacted, along with the method that should be used for contacting the operator.

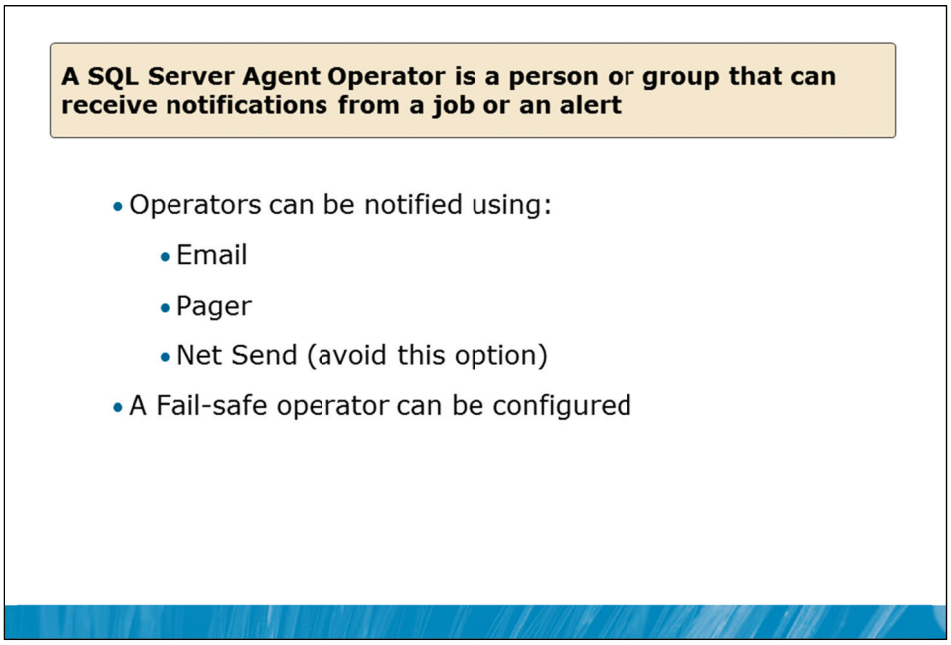
SQL Server can also detect many situations that might be of interest to administrators. You can configure Alerts that are based on SQL Server errors or on system events such as low disk space availability. SQL Server can then be configured to notify you of these situations.

Objectives

After completing this lesson, you will be able to:

- Describe the role of Operators in SQL Server Agent.
- Implement SQL Server alerts.
- Create alerts.
- Configure actions that need to occur in response to alerts.
- Troubleshoot alerts and notifications.

SQL Server Agent Operator Overview



Key Points

An Operator in SQL Server Agent is an alias for a person or a group of people that can receive electronic notifications when jobs complete or when alerts are raised.

Note Operators do not need to be Windows logins, SQL Server logins, or database users. For example, you could create an operator that is a reference to a pager address.

SQL Server Agent jobs can be configured to send messages when a job completes, when a job completes successfully, or when a job fails.

You can define new operators using either SSMS or the dbo.sp_add_operator system stored procedure. Once an operator is defined, the definition for the operator is visible through the dbo.sysoperators system table in the msdb database.

Contacting An Operator

You can configure three types of contact methods for each operator:

- Email: SMTP email address that notifications should be sent to. It is desirable to use group email addresses rather than individual email addresses where possible. It is possible to list multiple email addresses by separating them with a semicolon.
- Pager Email: SMTP email address that a message is sent to during specified times (and days) during a week.

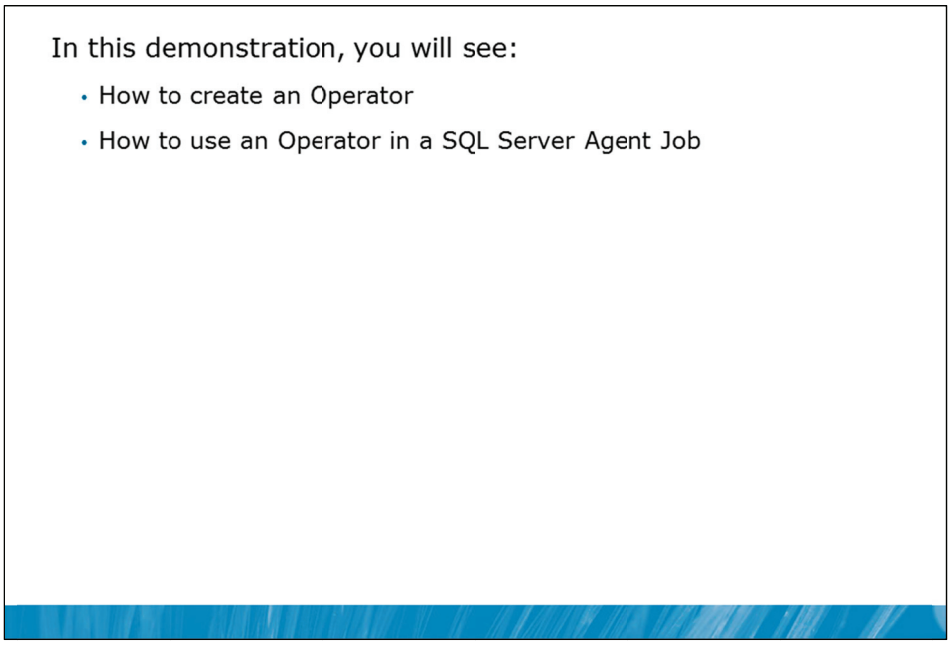
- Net Send address: Messenger address that a message is sent to.

Note The use of Net Send for notifications is deprecated and should not be used for new development as it will be removed in a future version of SQL Server. The Net Send option is not useful as it depends upon the Messenger service in Microsoft Windows®. That service is generally disabled on current systems.

A fail-safe operator can be defined to respond to an alert when pager notifications to other operators fail because of time limitations that have been configured. For example, if all operators are off duty when an alert is fired, the fail-safe operator will be contacted.

Existing Active Directory® users and groups can be used as operator groups if they are mail enabled groups.

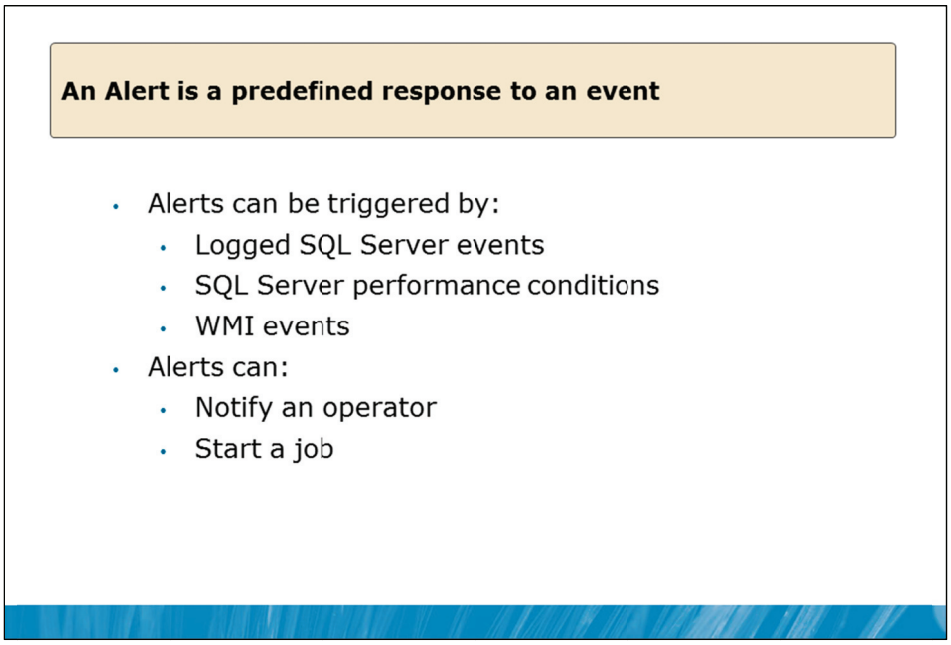
Demonstration 3A: Configuring SQL Server Agent Operators



Demonstration Steps

1. If Demonstration 1A was not performed:
   - Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
   - In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, click **SQL Server Management Studio**. In the Connect to Server window, type **Proseware** and click **Connect**. From the **File** menu, click **Open**, click **Project/Solution**, navigate to **D:\10775A_Labs\10775A_15_PRJ\10775A_15_PRJ.ssmssln** and click **Open**.
   - From the **View** menu, click **Solution Explorer**. Open and execute the **00 – Setup.sql** script file from within Solution Explorer.
2. Open the **31 – Demonstration 3A.sql** script file.
3. Follow the instructions contained within the comments of the script file.

Overview of SQL Server Alerts



Key Points

There are many events that can occur in a SQL Server system that are of interest to administrators. An Alert is a SQL Server object that defines a condition that requires attention and a response that should be taken when the event occurs. You can define alerts to execute a job or to notify an operator when a particular event occurs or even when a performance threshold is exceeded.

SQL Server Alerts

Events are generated by SQL Server and entered into the Windows Application Event Log. On startup, SQL Server Agent registers itself as a callback service with the Windows Application Event log. This means that SQL Server Agent will be directly notified by the application log when events of interest occur. This callback mechanism operates efficiently as it means that SQL Server Agent does not need to continuously read (or more formally "poll") the application log to find events of interest.

When SQL Server Agent is notified of a logged event, it compares the event to the alerts that have been defined. When SQL Server Agent finds a match, it fires an alert, which is an automated response to an event.

Note The error message must be written to the Windows Application Log to be used for SQL Server Agent Alerts.

Alerts Actions (Responses)

You can create alerts to respond to individual error numbers or to respond to all errors of a specific severity level. You can define the alert for all databases or for a specific database. You can define the time delay between responses.

Note It is considered good practice to configure notifications for all error messages with Severity Level 19 and above.

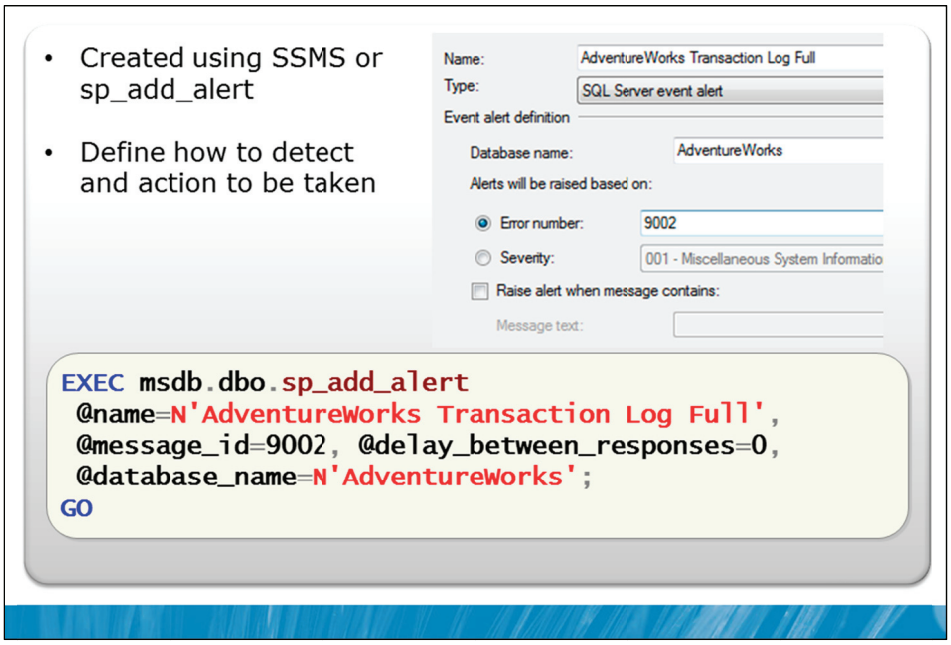
System Events

In addition to monitoring SQL Server events, SQL Server Agent can also monitor conditions that can be detected via Windows Management Instrumentation (WMI) events. The WQL queries that are written to retrieve the performance data are executed a few times each minute. As a result, it can take a few seconds for these alerts to fire.

Performance condition alerts can also be configured on any of the performance counters that SQL Server exposes.

Question: What events are you familiar with that should have a configured alert?

Create an Alert



Key Points

Alerts are created using the GUI in SSMS or by calling the dbo.sp_add_alert system stored procedure. When defining an alert, you can also specify a SQL Server Agent job that should be started when the alert occurs. In the example on the slide, the Job ID of a SQL Server Agent job that has already been created has been added to the definition of the alert. The job will be started when the alert fires.

The action that SQL Server Agent takes in response to the event or performance condition may include contacting an operator.

Logged Events

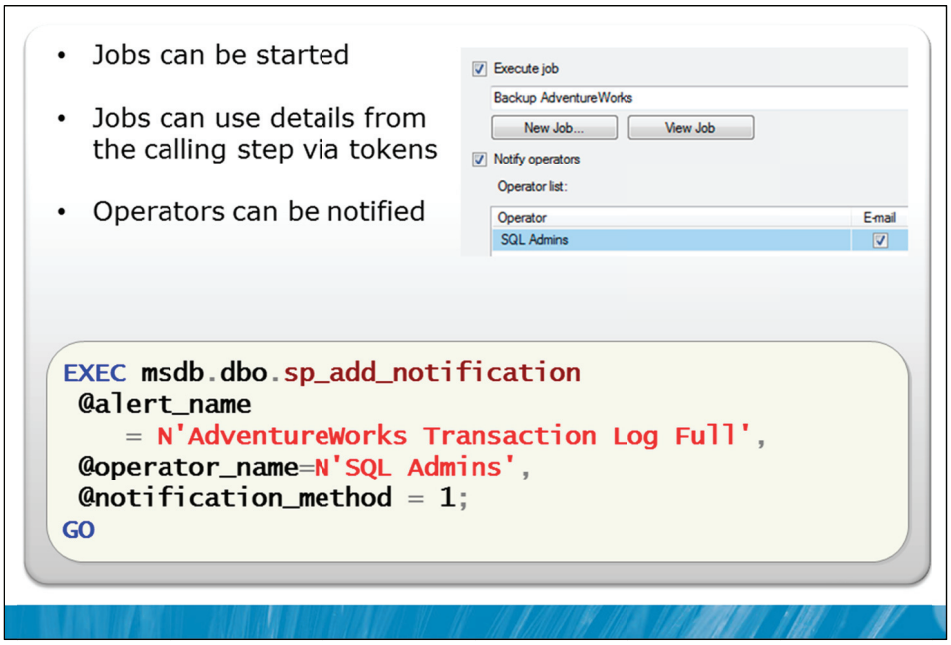
You have seen that alerts will only fire for SQL Server errors if the error messages are written to the Microsoft Windows Application Event log. In general, error severity levels from 19 to 25 are automatically written to the application log but this is not always the case. To check which messages are automatically written to the log, query the is_event_logged column in the sys.messages table.

Most events with severity levels less than 19 will only trigger alerts if you have used one of the following options:

- Modified the error message using the dbo.sp_altermessage system stored procedure to make the error message a logged message.
- Raised the error in code using the RAISERROR WITH LOG option.
- Used the xp_logevent system extended stored procedure to force entries to be written to the log.

Question: What type of alert would be needed to monitor free space in file the system?

Configuring Alert Actions



Key Points

When an alert fires, there are two actions that can be configured to respond to the alert:

- Execute a Job
- Notify Operators

Execute a Job

The execution of a SQL Server Agent job can be configured as a response to an alert. Only one job can be started. However, if you need to start multiple jobs when an alert occurs, create a new job that executes the other jobs and then configure the new job to respond to the alert.

The job to be executed can be configured when first creating the alert using dbo.sp_add_alert or by calling the dbo.sp_update_alert system stored procedure after the alert has already been created.

Notify Operators

Unlike the configuration of a job to run as part of the configuration of an alert, the list of operators to be notified when an alert fires is defined using the dbo.sp_add_notification system stored procedure.

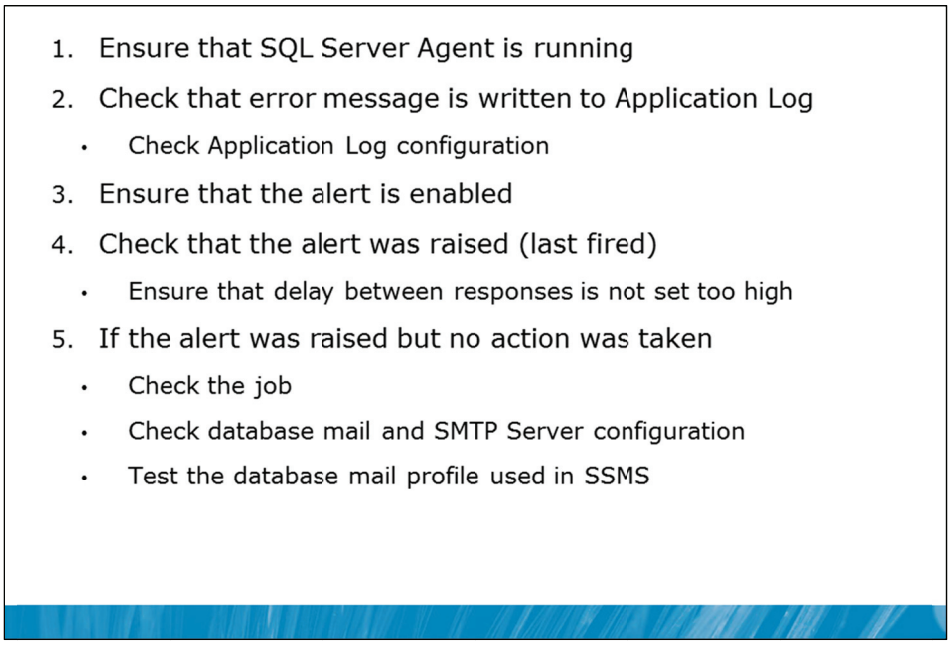
When sending messages to operators about alerts, it is important to be able to provide the operator with sufficient context about the alert so that they can determine the appropriate action to take. Tokens can be included in messages to add detail to the message. The special tokens available for working with alerts are shown in the following table:

Token	Description
A-DBN	Database Name
A-SVR	Server Name
A-ERR	Error Number
A-SEV	Error Severity
A-MSG	Error Message

Note that for security reasons this feature is disabled by default and can be enabled in the properties of SQL Server Agent.

Question: If notifications should be sent to a pager email address, what else should be configured?

Troubleshooting Alerts and Notifications



Key Points

When troubleshooting alerts and notifications, use the following process to identify the issues:

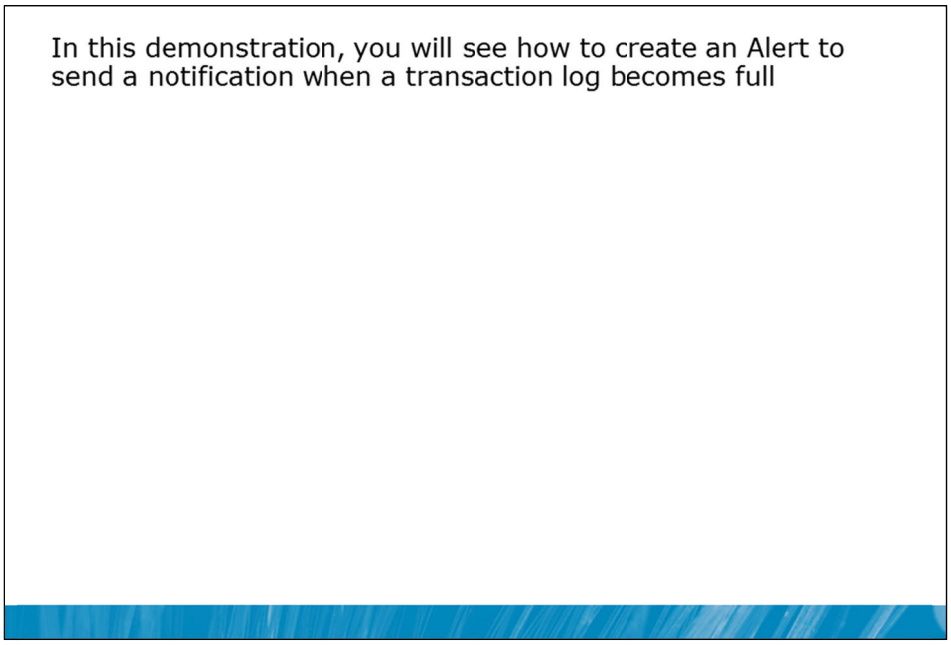
Step	Description
Ensure that SQL Server Agent is running	The Application Log will only send messages to SQL Server Agent when the Agent is running. The Application Log does not hold a queue of notifications to be made at a later time.
Check that the error message is written to Application Log.	For SQL Server Event Alerts, check that the error message is written to the Application Log and also make sure that the Application Log is configured with sufficient size to hold all event log details.
Ensure that the alert is enabled.	Alerts can be enabled or disabled and will not fire when disabled.
Check that the alert was raised.	If the alert does not appear to be raised, make sure that the setting for delay between responses is not set to too high a value.

(continued)

Step	Description
If the alert was raised but no action was taken	Check that the job that is configured to respond to the alert functions as expected. For operator notifications, check that Database Mail is working and that the SMTP Server configuration is correct. Test the Database Mail profile that is being used to send notifications by manually sending mail from the profile used by SQL Server Agent.

Question: Why might an error message not be written to the application log?

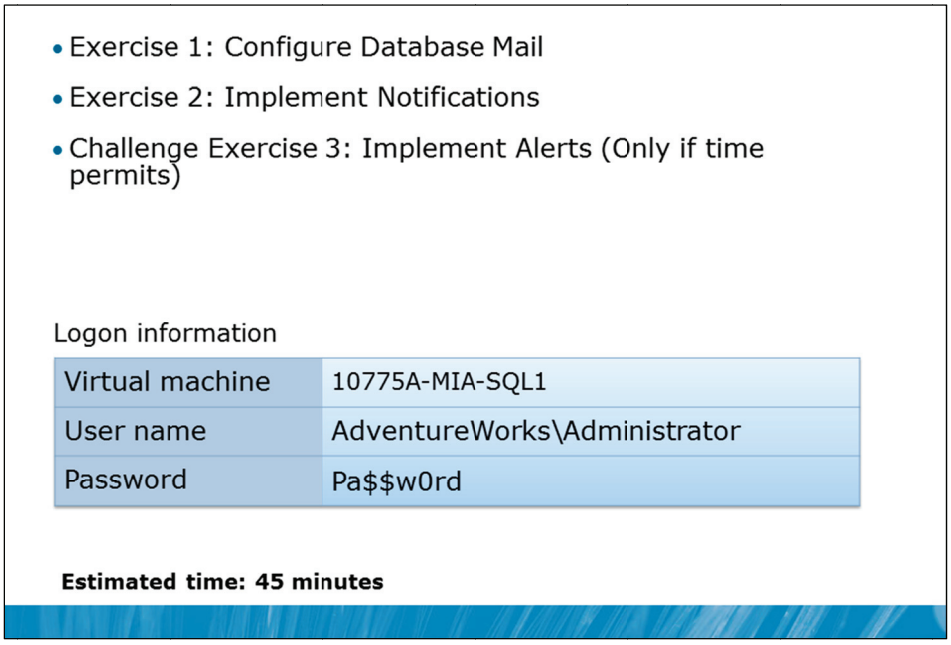
Demonstration 3B: Configuring Alerts and Notifications



Demonstration Steps

1. If Demonstration 1A or 3A was not performed:
   - Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
   - In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, click **SQL Server Management Studio**. In the Connect to Server window, type **Proseware** and click **Connect**. From the **File** menu, click **Open**, click **Project/Solution**, navigate to **D:\10775A_Labs\10775A_15_PRJ\10775A_15_PRJ.ssmssln** and click **Open**.
   - From the **View** menu, click **Solution Explorer**. Open and execute the **00 – Setup.sql** script file from within Solution Explorer.
   - Open the script file **11 – Demonstration 1A.sql** and follow the instructions in that file.
   - Open the script file **31 – Demonstration 3A.sql** and follow the instructions in that file.
2. Open the **32 – Demonstration 3B.sql** script file.
3. Follow the instructions contained within the comments of the script file.

Lab 15: Monitoring SQL Agent Jobs with Alerts and Notifications



Lab Setup

For this lab, you will use the available virtual machine environment. Before you begin the lab, you must complete the following steps:

1. Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
2. In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, and click **SQL Server Management Studio**.
3. In the Connect to Server window, type **Proseware** in the **Server name** text box.
4. In the **Authentication** drop-down list box, select **Windows Authentication** and click **Connect**.
5. In the **File** menu, click **Open**, and click **Project/Solution**.
6. In the Open Project window, open the project **D:\10775A_Labs\10775A_15_PRJ\10775A_15_PRJ.ssmssln**.
7. From the **View** menu, click **Solution Explorer**. In Solution Explorer, double-click the query **00-Setup.sql**. When the query window opens, click **Execute** on the toolbar.

Lab Scenario

You have configured automated management tasks using SQL Server Agent and have configured security for those tasks. You now need to configure alerts and notifications for your Proseware system. The IT Support team at AdventureWorks has a defined escalation policy for SQL Server systems. As Proseware is part of the group of companies owned by AdventureWorks, you need to implement the relevant parts of this policy.

The IT Support team has supplied you with details from the policy that they have determined are needed for your Proseware server. For some automated tasks, notifications need to be sent every time the tasks are completed, whether or not the tasks work as expected. For other tasks, notifications only need to be sent if the tasks fail for some reason.

Notifications at AdventureWorks are pager-based. You need to configure Database Mail within SQL Server so that SQL Server Agent can send notification emails to the pager system. There are two on-call DBAs allocated to your system from the AdventureWorks IT Support team. You need to configure these staff members as operators based on their current on-call work schedules and also configure a fail-safe operator for any time period where no team member is working.

If you have enough time, you should also configure SQL Server to alert you if severe errors occur on the server.

Supporting Documentation

Database Mail Configuration Parameters

Profile Name: Proseware SQL Server Agent Profile

SMTP Account	Item	Value
Main	Account Name	Proseware Administrator
	E-mail Address	prosewaresqladmin@adventureworks.com
	Display name	Proseware SQL Server Administrator
	Reply e-mail	prosewaresqladmin@adventureworks.com
	Server name	mailserver.adventureworks.com
Fail-safe	Account Name	AdventureWorks Administrator
	E-mail Address	adventureworkssqladmin@adventureworks.com
	Display name	AdventureWorks SQL Server Administrator
	Reply e-mail	adventureworkssqladmin@adventureworks.com
	Server name	mailserver.adventureworks.com

Public Profiles: Configure Proseware SQL Agent Profile as public and as default

Private Profiles: Configure SQL Server Agent Profile as the default profile for the SQL Server Agent service (AdventureWorks\PWService)

Maximum E-mail File Size: 4MB

On-call DBA Operator Requirements

- Senior DBA Jeff Hay is on-call via pager jeff.hay.pager@adventureworks.com for the entire 24 hours per day, seven days per week.
- DBA Palle Petersen is on-call via pager palle.petersen.pager@adventureworks.com for the entire 24 hours per day, seven days per week.
- Although there should always be a DBA on call a fail-safe pager address itsupport.pager@adventureworks.com should be configured. The IT Support operator should also be available 24 hours per day, seven days per week.

Job Notification Requirements

- Backup-related jobs must send notifications on completion, not just on failure. Notifications for backup-related jobs should be sent to Jeff Hay.
- System jobs do not need to send any notifications. System jobs are identified by the prefix sys.
- All other jobs should notify on failure only. Notifications for other jobs should be sent to Palle Petersen.

Severe Error Alerting Requirements

- Any error of severity 17 or 18 should be notified to all operators via pager.
- Error 9002 on any database should be notified to all operators via pager.

Exercise 1: Configure Database Mail

Scenario

Notifications at AdventureWorks are pager-based. You need to configure Database Mail within SQL Server so that SQL Server Agent can send notification emails to the pager system.

The main tasks for this exercise are as follows:

1. Configure database mail.
2. Test that database mail operates.

▶ Task 1: Configure database mail

- Review the database mail configuration parameters in the supporting documentation for the exercise.
- Configure database mail as per supplied parameters.

▶ Task 2: Test that database mail operates

- Send a test email using the right-click option on the database mail node in Object Explorer.
- From Solution Explorer, open and execute the script file 51 – Lab Exercise 1.sql to view outgoing mail items.

Results: After this exercise, you should have configured and tested database mail.

Exercise 2: Implement Notifications

Scenario

The IT Support team at AdventureWorks has a defined escalation policy for SQL Server systems. As Proseware is part of the group of companies owned by AdventureWorks, you need to implement the relevant parts of this policy.

The IT Support team has supplied you with details from the policy that they have determined are needed for your Proseware server. For some automated tasks, notifications need to be sent every time the tasks are completed, whether or not the tasks work as expected. For other tasks, notifications only need to be sent if the tasks fail for some reason.

Notifications at AdventureWorks are pager-based. There are two on-call DBAs allocated to your system from the AdventureWorks IT Support team. You need to configure these staff members as operators based on their current on-call work schedules and also configure a fail-safe operator for any time period where no team member is working.

The main tasks for this exercise are as follows:

1. Review the requirements.
2. Configure the required operators.
3. Configure SQL Server Agent Mail.
4. Configure and Test Notifications in SQL Server Agent Jobs.

▶ Task 1: Review the requirements

- Review the supplied requirements in the supporting documentation for the exercise. In particular, note any required operators.

▶ Task 2: Configure the required operators

- Configure the required operators that you determined were required in Task 1. The supporting documentation includes details of how the operators need to be configured.

▶ Task 3: Configure SQL Server Agent Mail

- Configure SQL Server Agent to use the mail profile that you created in Exercise 1.
- Configure SQL Server Agent to use the IT Support fail-safe operator that you configured in Task 2.

▶ Task 4: Configure and Test Notifications in SQL Server Agent Jobs

- Configure notifications for jobs as per the requirements in the supporting documentation.
- Test the notifications by executing all non-system jobs and reviewing the mail item sent.

Results: After this exercise, you should have configured SQL Server Agent operators, and job notifications.

Challenge Exercise 3: Implement Alerts (Only if time permits)

Scenario

If you have enough time, you should also configure SQL Server to alert you if severe errors occur on the server.

The main task for this exercise is as follows:

1. Configure and test alerts.

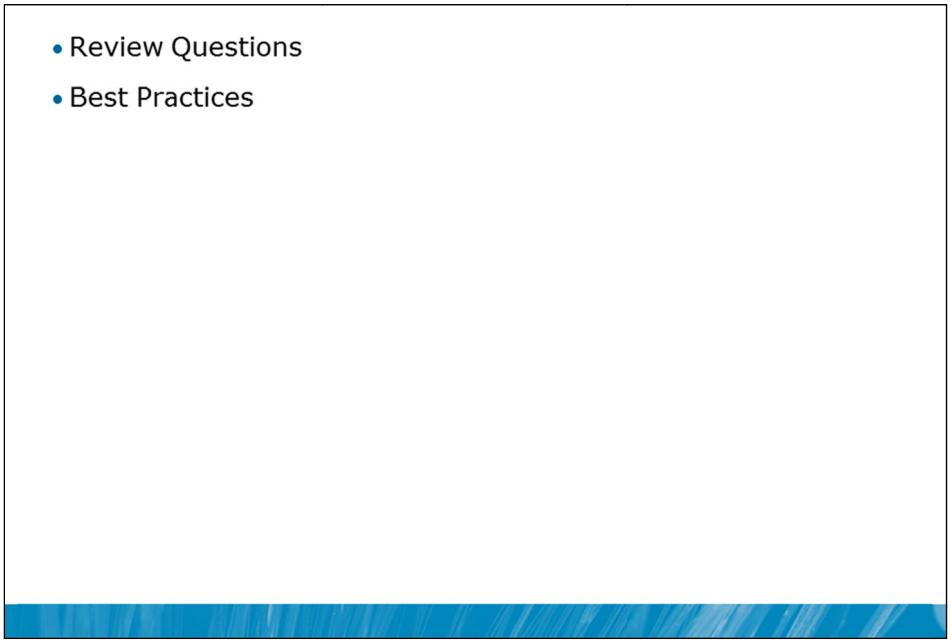
▶ Task 1: Configure and test alerts

- Review the supporting documentation for the alerting requirements.
- Configure the required alerts.
- Execute the script 71 – Lab Exercise 3.sql to test the alerting functionality.

Note The script will return error 9002.

Results: After this exercise, you should have configured and tested SQL Server alerts.

Module Review and Takeaways



Review Questions

1. What is an Operator in SQL Server Agent terminology?
2. What is the lowest error severity that appears as an error message in SSMS?

Best Practices

1. Use Database Mail and not SQL Mail.
2. Configure different profiles for different usage scenarios.
3. Provide limited access to the ability to send emails from the database engine.
4. Implement a retention policy for database mail log and mail auditing.
5. Create operators to send notifications about Jobs and Alerts.
6. Define Alerts for severe error messages.

Module 16

Performing Ongoing Database Maintenance

Contents:

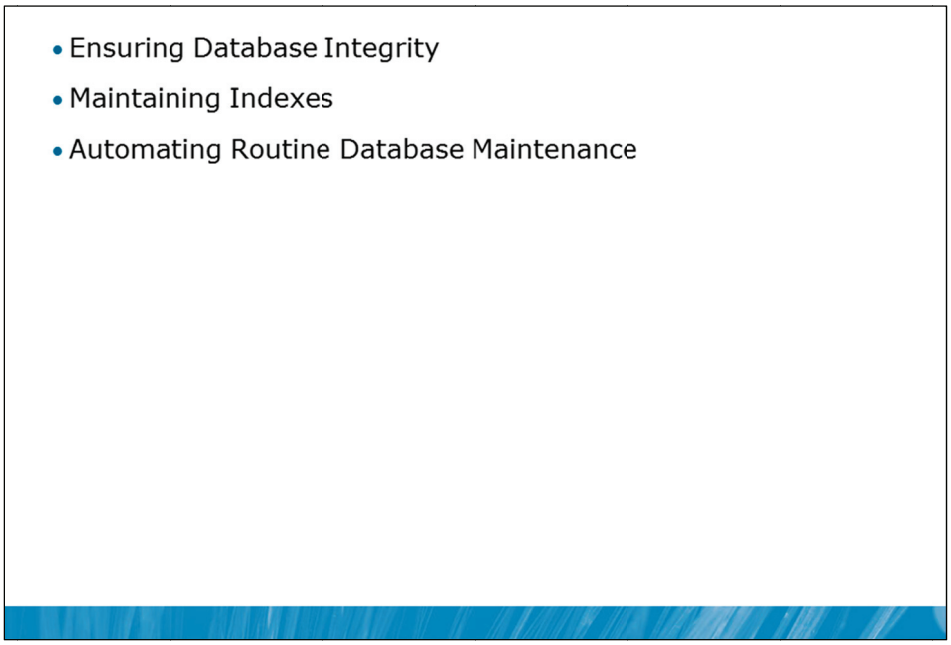
Lesson 1: Ensuring Database Integrity **16-3**

Lesson 2: Maintaining Indexes **16-12**

Lesson 3: Automating Routine Database Maintenance **16-26**

Lab 16: Performing Ongoing Database Maintenance **16-30**

Module Overview



The Microsoft® SQL Server® database engine is very capable of running indefinitely without any ongoing maintenance. However obtaining the best outcomes from the database engine requires a schedule of routine maintenance operations.

Database corruption is relatively rare but one of the most important tasks in the ongoing maintenance of a database is to check that no corruption has occurred in the database. Recovering from corruption depends upon detecting the corruption soon after it occurs.

SQL Server indexes can also continue to work without any maintenance but they will perform better if any fragmentation that occurs within them is periodically removed.

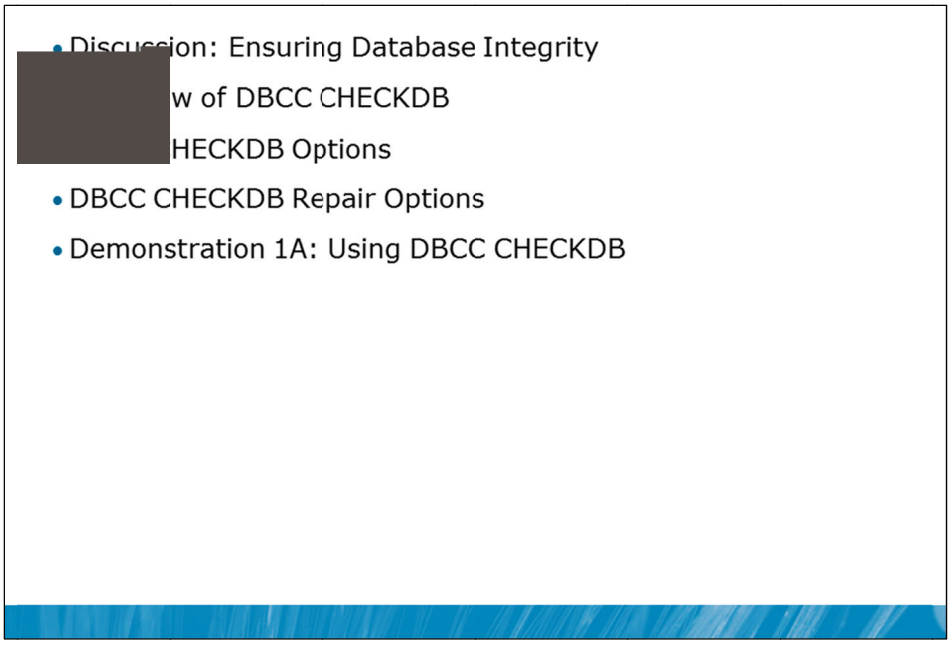
SQL Server includes a Maintenance Plan Wizard to assist in creating SQL Server Agent jobs that perform these and other ongoing maintenance tasks.

Objectives

After completing this lesson, you will be able to:

- Ensure database integrity.
- Maintain indexes.
- Automate routine database maintenance.

Lesson 1
Ensuring Database Integrity



It is particularly rare for the database engine to cause corruption directly. However, the database engine depends upon the hardware platform that it runs upon and that platform can cause corruption. In particular, issues in the memory and I/O subsystems can lead to corruption within databases.

If you do not detect corruption soon after it has occurred, further (and significantly more complex or troublesome) issues can arise. For example, there is little point attempting to recover a corrupt database from a set of backups where every backup contains a corrupted copy of the database.

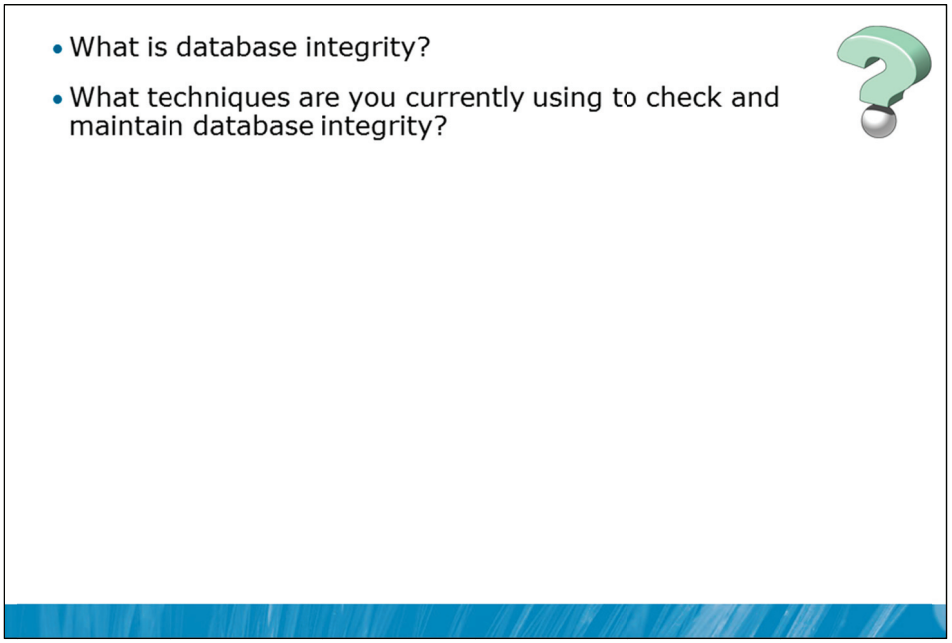
The DBCC CHECKDB command can be used to detect, and in some circumstances correct, database corruption. It is important that you are familiar with how DBCC CHECKDB is used.

Objectives

After completing this lesson, you will be able to:

- Use DBCC CHECKDB.
- Explain the most common DBCC CHECKDB options.
- Explain how to use the DBCC CHECKDB repair options.

Discussion: Ensuring Database Integrity

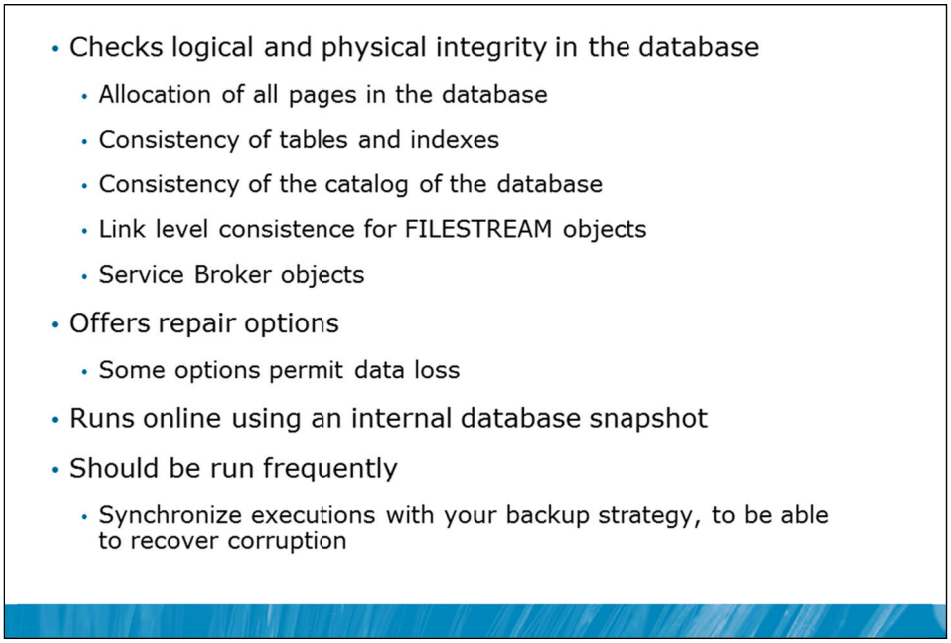


Discussion Topics

Question: What is database integrity?

Question: What techniques are you currently using to check and maintain database integrity?

Overview of DBCC CHECKDB



Key Points

DBCC is a utility that is supplied with SQL Server that provides a large number of management facilities. In earlier documentation, you may see it referred to as the Database Consistency Checker. While checking the consistency of databases by using the CHECKDB option is a primary use of DBCC, it has many other uses. In current versions of the product, it is referred to as the Database Console Commands utility, to more closely reflect the wider variety of tasks that it can be used for.

DBCC CHECKDB

The CHECKDB option in the DBCC utility makes a particularly thorough check of the structure of a database, to detect almost all forms of potential corruption. The series of functions that are contained within DBCC CHECKDB are also available as options that can be performed separately if required.

The most important of these options is shown in the following table:

Option	Description
DBCC CHECKALLOC	Checks the consistency of disk space allocation structures for a specified database.
DBCC CHECKTABLE	Checks the pages associated with a specified table and the pointers between pages that are associated with the table. DBCC CHECKDB executes DBCC CHECKTABLE for every table in the database.

(continued)

Option	Description
DBCC CHECKCATALOG	Checks the database catalog by performing logical consistency checks on the metadata tables in the database. These metadata tables are used to hold information that describes both system and user tables and other database objects. DBCC CHECKCATALOG does not check user tables.

DBCC CHECKDB also performs checks on other types of objects such as the links for FILESTREAM objects and consistency checks on the Service Broker objects.

Note FILESTREAM and Service Broker are advanced topics that are out of scope for this course.

Repair Options

Even though DBCC CHECKDB has repair options, it is not always possible to repair a database without data loss. Usually, the best option for database recovery is to restore the database. This means that the execution of DBCC CHECKDB should be synchronized with your backup retention policy, to make sure that you can always restore a database from an uncorrupted database and that all required log backups since that time are available.

Online Concurrent Operations

DBCC CHECKDB can take a long time to execute and consumes considerable I/O and CPU resources. For this reason, DBAs often need to run it while the database is in use.

In versions of SQL Server prior to SQL Server 2005, it was possible to receive indications of corruption where no corruption was present if DBCC CHECKDB was executed while the database was being used concurrently by other users. Since SQL Server 2005, DBCC CHECKDB operates using internal database snapshots to make sure that the utility works with a consistent view of the database. If DBCC CHECKDB reports corruption, it needs to be investigated.

If the performance needs for the database activity that needs to run while DBCC CHECKDB is executing are too high, running DBCC CHECKDB against a restored backup of your database would be a better (but far from ideal) option than not running DBCC CHECKDB at all.

Disk Space

The use of an internal snapshot causes DBCC CHECKDB to need additional disk space. DBCC CHECKDB creates hidden files (using NTFS Alternate Streams) on the same volumes as the database files are located. Sufficient free space on the volumes must be available for DBCC CHECKDB to run successfully. The amount of disk space required on the volumes depends upon how much data is changed during the execution of DBCC CHECKDB.

DBCC CHECKDB also uses space in tempdb while executing. To provide an estimate of the amount of space required in tempdb, DBCC CHECKDB offers an ESTIMATEONLY option.

Backups and DBCC CHECKDB

It is considered a good practice to run DBCC CHECKDB on a database prior to performing a backup of the database. This check helps to ensure that the backup contains a consistent version of the database.

Question: Why is it vital to run DBCC CHECKDB regularly?

DBCC CHECKDB Options

Option	Description
PHYSICAL_ONLY	Only checks the physical integrity with less overhead
NOINDEX	Does not perform logical checks on nonclustered indexes
EXTENDED_LOGICAL_CHECKS	Performs additional logical checks of indexed views, spatial and XML indexes
TABLOCK	Uses locks instead of database snapshots
ALL_ERRORMSGS	Returns all error messages instead of the default action that returns the first 200
NO_INFOMSGS	Returns only error messages and no informational messages
ESTIMATEONLY	Estimates the amount of tempdb space that is required for execution

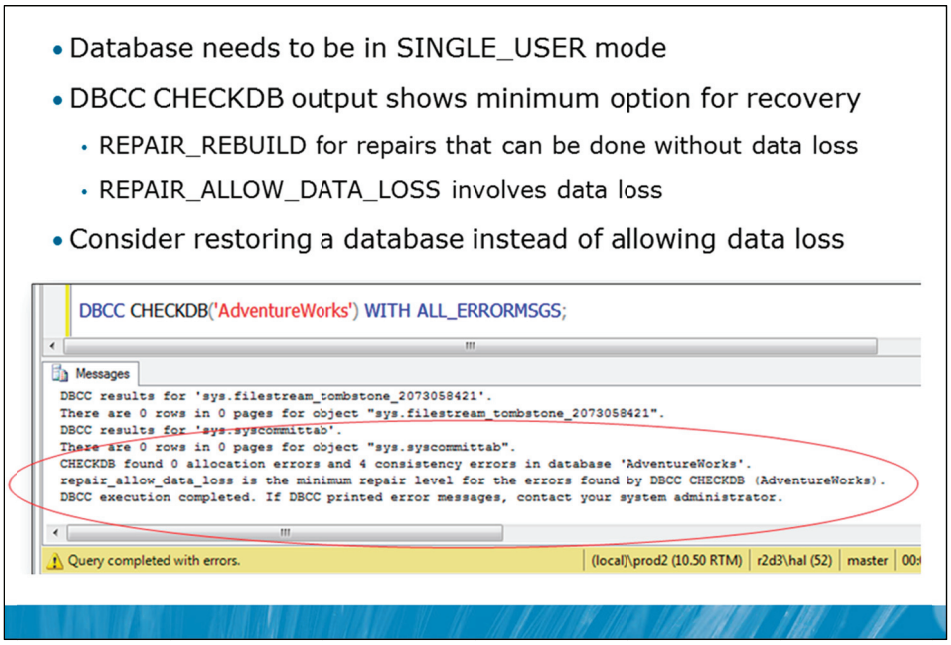
Key Points

DBCC CHECKDB provides a number of options that alter its behavior while it is executing.

- The PHYSICAL_ONLY option is often used on production systems because it substantially reduces the time taken to run DBCC CHECKDB on large databases. If you regularly use the PHYSICAL_ONLY option, you still need to periodically run the full version of the utility. How often you perform the full version would depend upon specific business requirements.
- The NOINDEX option specifies that intensive checks of nonclustered indexes for user tables should not be performed. This also decreases the overall execution time but does not affect system tables because integrity checks are always performed on system table indexes. The assumption that you are making when using the NOINDEX option is that you can rebuild the nonclustered indexes if they become corrupt.
- The EXTENDED_LOGICAL_CHECKS can only be performed when the database is in database compatibility level 100 (SQL Server 2008) or above. It performs detailed checks of the internal structure of objects such as CLR user-defined data types and spatial data types.
- The TABLOCK option is used to request that DBCC CHECKDB takes a table lock on each table while performing consistency checks rather than using the internal database snapshots. This reduces the disk space requirements at the cost of preventing other users from updating the tables.
- The ALL_ERRORMSGS and NO_INFOMSGS options only affect the output from the command but not the operations performed by the command.
- The ESTIMATEONLY option estimates the space requirements in tempdb as discussed in the previous topic.

Question: Which DBCC CHECKDB option might be used on very large production systems?

DBCC CHECKDB Repair Options



Key Points

As well as providing details of errors that have been found, the output of DBCC CHECKDB shows the repair option that would be needed to correct the problem. In the example on the slide, four consistency errors were found and the REPAIR_ALLOW_DATA_LOSS option would be needed to repair the database.

Repair Options

DBCC CHECKDB offers two repair options. For both options, the database needs to be in single user mode. The options are:

- REPAIR_REBUILD rebuilds indexes. Corrupt data pages are removed. This option only works with certain mild forms of corruption and does not involve data loss.
- REPAIR_ALLOW_DATA_LOSS will almost always produce data loss. It deallocates the corrupt pages and changes other pages that reference the corrupt pages. After the operation is complete, the database will be consistent, but only from a physical database integrity point of view. Significant loss of data could have occurred. Repair operations also do not consider any of the constraints that may exist on or between tables. If the specified table is involved in one or more constraints, it is recommended that you execute DBCC CHECKCONSTRAINTS after the repair operation is complete.

The use of DBCC CHECKDB is shown in Demonstration 1A.

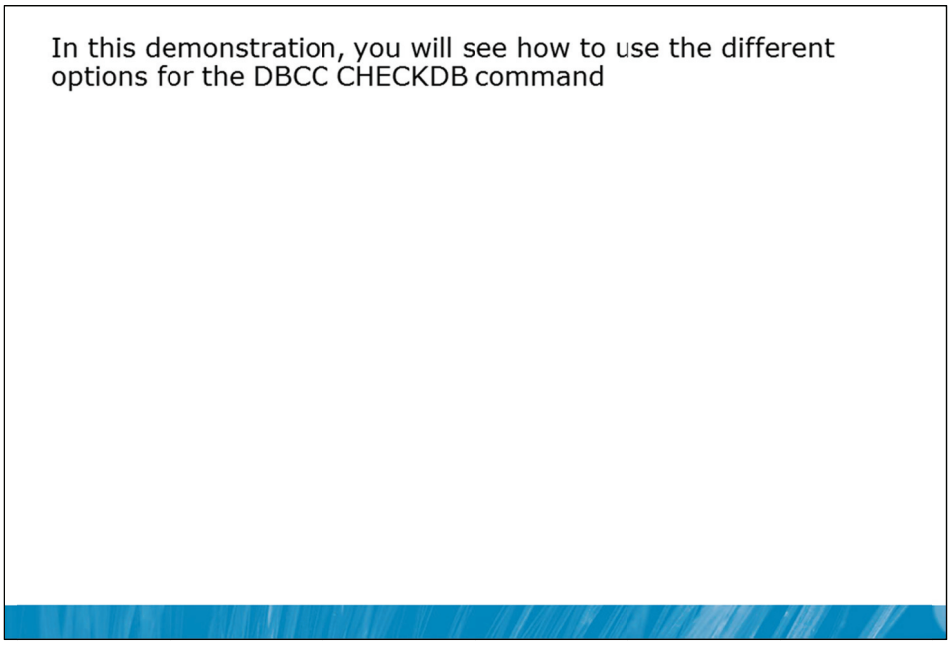
You should backup a database before performing any repair option. Repairing a database should be an option of last resort. When a database is corrupt, it is typically better to restore the database from a backup, after solving the cause of the corruption. Unless you find and resolve the reason for the corruption, it may well happen again soon after. Corruption in SQL Server databases is mostly caused by failures in I/O or memory subsystems.

Transaction Log Corruption

If the transaction log becomes corrupt, a special option called an emergency mode repair can be attempted, but it is strongly recommended to restore the database in that situation. An emergency mode repair should only be used when no backup is available.

Question: Why would it be preferable to restore a database rather than using REPAIR_ALLOW_DATA_LOSS?

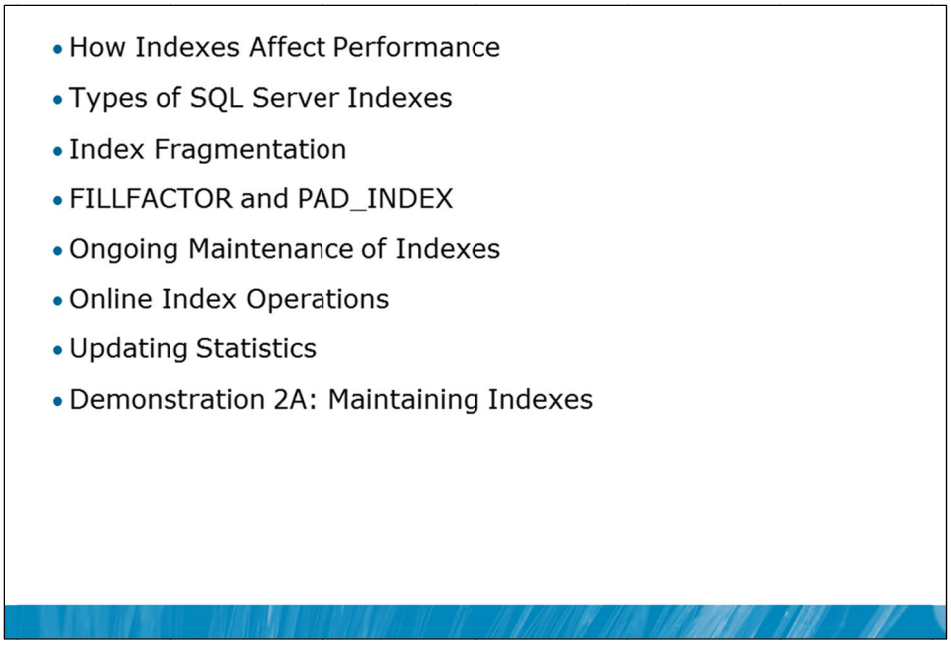
Demonstration 1A: Using DBCC CHECKDB



Demonstration Steps

1. Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
2. In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, click **SQL Server Management Studio**. In the Connect to Server window, type **Proseware** and click **Connect**. From the **File** menu, click **Open**, click **Project/Solution**, navigate to **D:\10775A_Labs\10775A_16_PRJ\10775A_16_PRJ.ssmssln** and click **Open**.
3. From the **View** menu, click **Solution Explorer**. Open and execute the **00 – Setup.sql** script file from within Solution Explorer.
4. Open the **11 – Demonstration 1A.sql** script file.
5. Follow the instructions contained within the comments of the script file.

Lesson 2
Maintaining Indexes



Another important aspect of SQL Server that requires ongoing maintenance for optimal performance is the management of indexes. Indexes are used to speed up operations where SQL Server needs to access data in a table. Over time, indexes can become fragmented and the performance of database applications that use the indexes will be reduced. Defragmenting or rebuilding the indexes will restore the performance of the database.

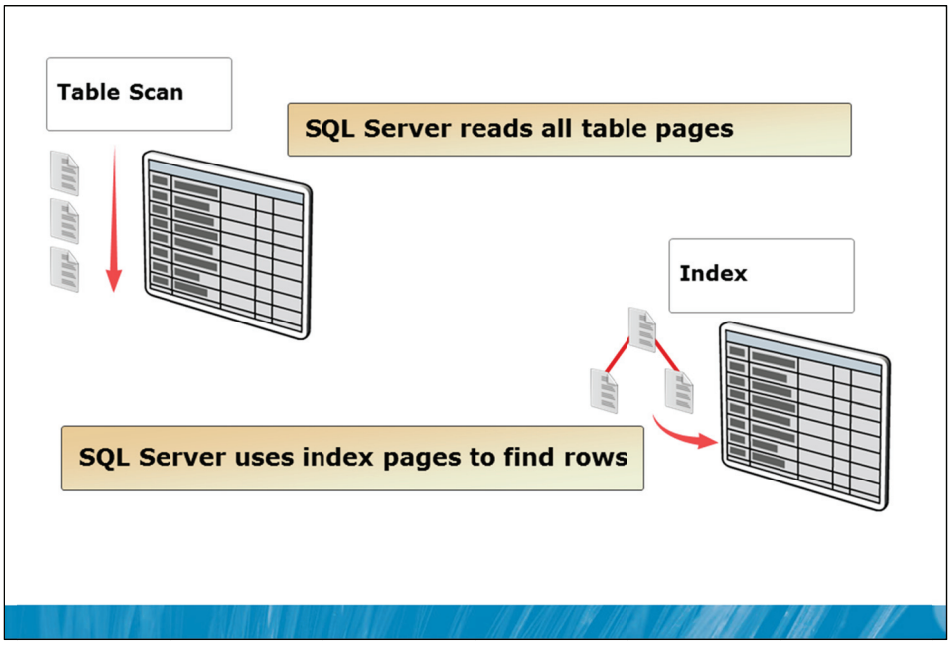
Index management options are often included in regular database maintenance plan schedules. Before learning how to set up the maintenance plans, it is important to understand more about how indexes work and how they are maintained.

Objectives

After completing this lesson, you will be able to:

- Describe how indexes affect performance.
- Describe the different types of SQL Server indexes.
- Describe how indexes become fragmented.
- Use FILLFACTOR and PAD_INDEX.
- Explain the ongoing maintenance requirements for indexes.
- Implement online index operations.
- Describe how statistics are created and used by SQL Server.

How Indexes Affect Performance



Key Points

SQL Server can access data in a table by reading all the pages of the table (known as a table scan) or by using index pages to locate the required rows.

Indexes

Whenever SQL Server needs to access data in a table, it makes a decision about whether to read all the pages of the table or whether there are one or more indexes on the table that would reduce the amount of effort required in locating the required rows.

Queries can always be resolved by reading the underlying table data. Indexes are not required but accessing data by reading large numbers of pages is usually considerably slower than methods that use appropriate indexes.

Indexes can help to improve searching, sorting, and join performance but they can impact data modification performance, they require ongoing management, and they require additional disk space.

On occasion, SQL Server will create its own temporary indexes to improve query performance. However, doing so is up to the optimizer and beyond the control of the database administrator or programmer, so these temporary indexes will not be discussed in this module. The temporary indexes are only used to improve a query plan, if no proper indexing already exists.

In this module, you will consider standard indexes created on tables. SQL Server includes other types of index:

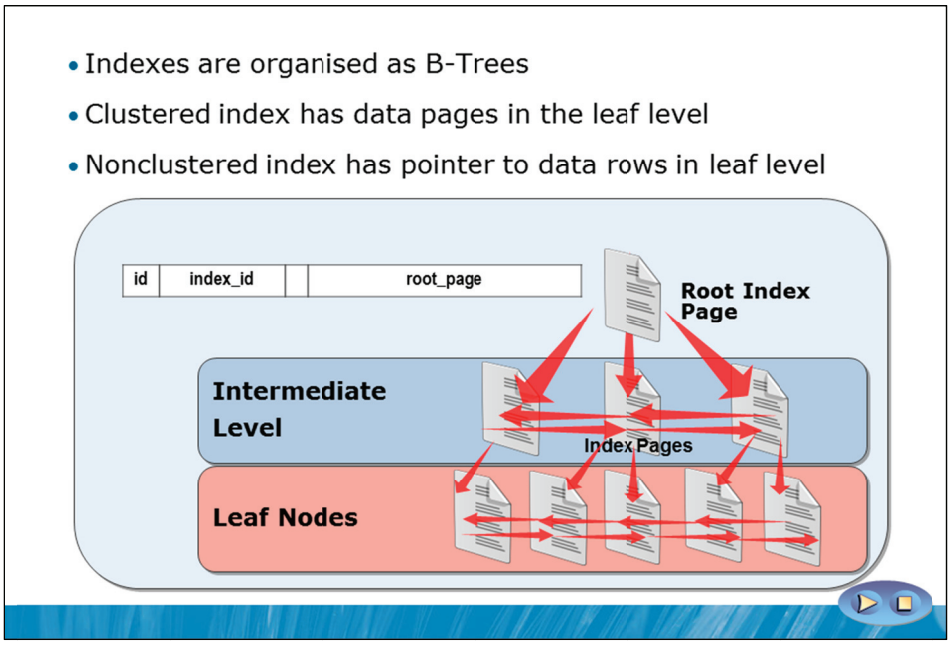
- Integrated full-text search (iFTS) uses a special type of index that provides flexible searching of text.
- Spatial indexes are used with the GEOMETRY and GEOGRAPHY data types.
- Primary and secondary XML indexes assist when querying XML data.

- Columnstore indexes are used typically used in large data warehouses. The tables essentially become read-only while the columnstore indexes are in place.

Note iFTS, Spatial, XML indexes, and Columnstore indexes are out of scope for this course but iFTS, Spatial, and XML indexes are described in course 10776A: Developing Microsoft SQL Server 2012 Databases, along with greater detail on standard clustered and nonclustered indexes.

Question: When might a table scan be more efficient than using an index?

Types of SQL Server Indexes



Key Points

Rather than storing rows of a data as a heap, tables can be designed with an internal logical ordering. This is known as a clustered index.

Clustered Index

A table with a clustered index has a predefined order for rows within a page and for pages within the table. The order is based on a key made up of one or more columns. The key is commonly called a clustering key.

Because the rows of a table can only be in a single order, there can be only a single clustered index on a table. An Index Allocation Map entry is used to point to a clustered index. Clustered indexes are always index id = 1.

There is a common misconception that pages in a clustered index are "physically stored in order". While this is possible in rare situations, it is not commonly the case. If it was true, fragmentation of clustered indexes would not exist. SQL Server tries to align physical and logical order while creating an index but disorder can arise as data is modified.

Index and data pages are linked within a logical hierarchy and also double-linked across all pages at the same level of the hierarchy to assist when scanning across an index. For example, imagine a table with ten extents and with allocated page numbers 201 to 279 all linked in order. (Each extent contains eight pages). If a page needed to be placed into the middle of the logical order, SQL Server finds an extent with a free page or allocates a new extent for the index. The page is logically linked into the correct position but it could be located anywhere within the database pages.

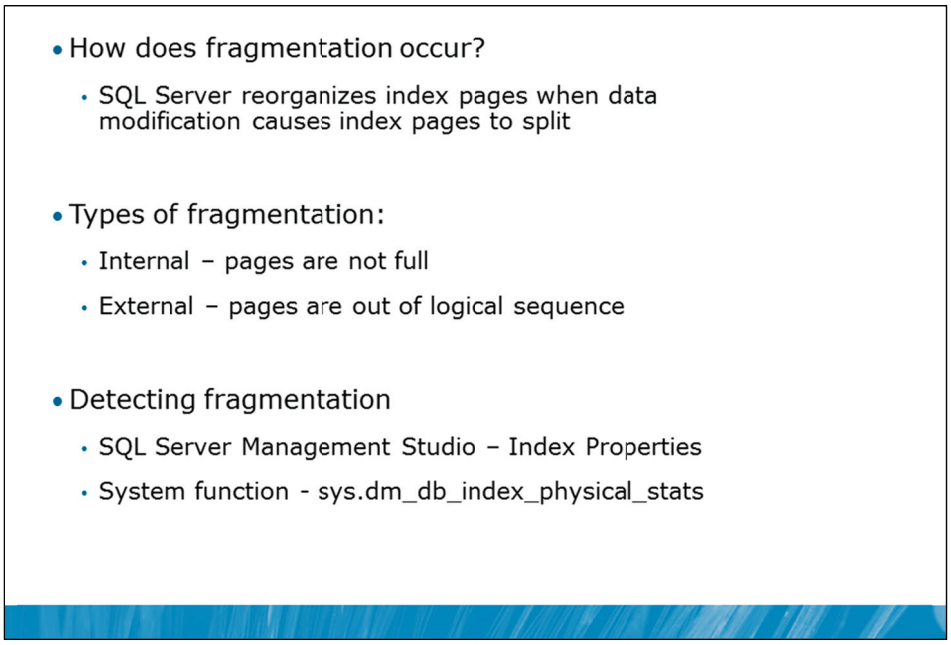
Nonclustered Index

A nonclustered index is a type of index that does not affect the layout of the data in the table in the way that a clustered index does.

If the underlying table is a heap (that is, it has no clustered index), the leaf level of a nonclustered index contains pointers to where the data rows are stored. The pointers include a file number, a page number, and a slot number on the page.

If the underlying table has a clustered index (that is, the pages and the data are logically linked in the order of a clustering key), the leaf level of a nonclustered index contains the clustering key that is then used to seek through the pages of the clustered index to locate the desired rows.

Index Fragmentation



Key Points

Index fragmentation is the inefficient use of pages within an index. Fragmentation occurs over time as data is modified.

Index Fragmentation

For operations that read data, indexes perform best when each page of the index is as full as possible. While indexes may initially start full (or relatively full), modifications to the data in the indexes can cause the need to split index pages. Adding a new index entry to the end of an index is easy but the process is more complicated if the entry needs to be made in the middle of an existing full index page.

Internal vs. External Fragmentation

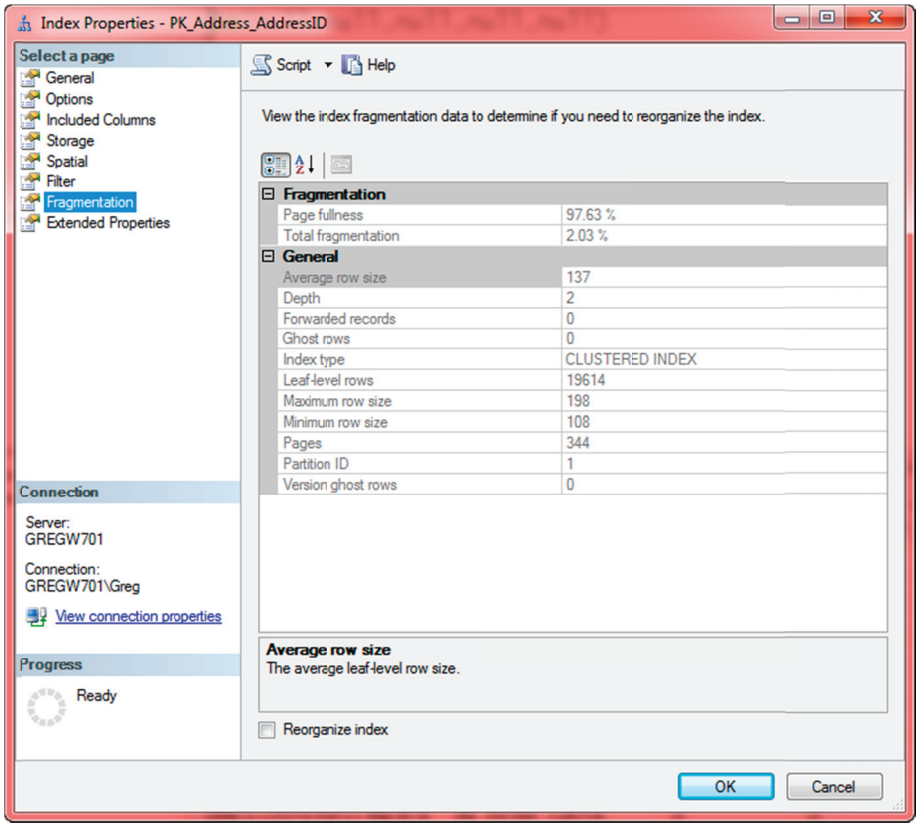
Internal fragmentation occurs when pages are not holding as much data as they are capable of holding. This often occurs when a page is split during an insert operation and can also occur when an update operation causes a row to be moved to another page. In either situation, empty space is left within pages.

External fragmentation occurs when pages that are logically sequenced are not held in sequenced page numbers. If a new index page needs to be allocated, it would be logically inserted into the correct location in the list of pages but could well be placed at the end of the index. That means that a process that needs to read the index pages in order would need to follow pointers to locate the pages and the process would involve accessing pages that are not sequential within the database.

Detecting Fragmentation

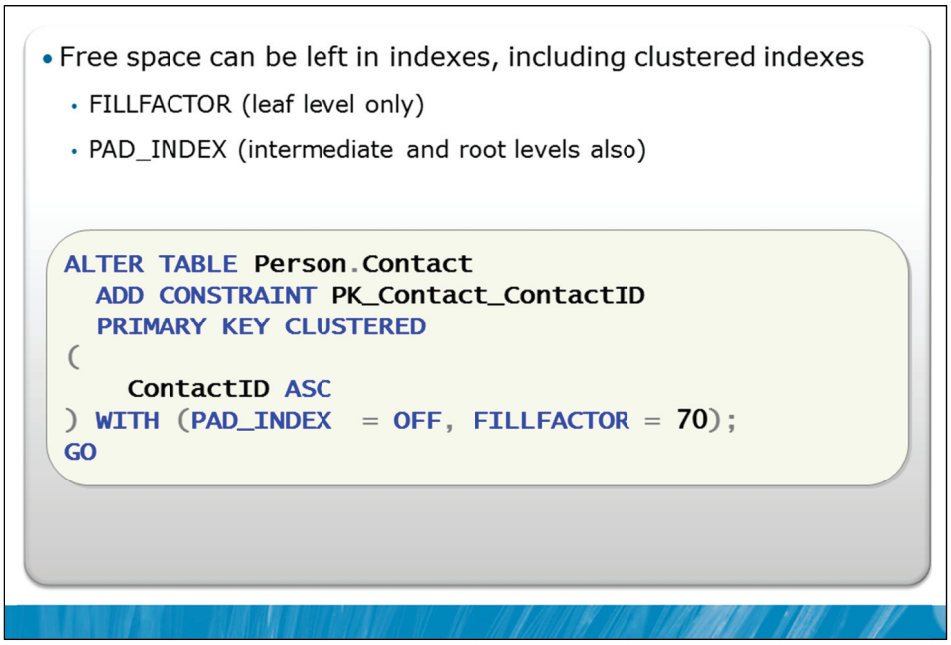
SQL Server provides a useful measure in the avg_fragmentation_in_percent column of the sys.dm_db_index_physical_stats dynamic management view.

SQL Server Management Studio also provides details of index fragmentation in the properties page for each index as shown in the following screenshot from the AdventureWorks database:



Question: Why does fragmentation affect performance?

FILLFACTOR and PAD_INDEX



Key Points

The FILLFACTOR and PAD_INDEX options are used to provide free space within index pages. This can improve INSERT and UPDATE performance in some situations but often to the detriment of SELECT operations.

FILLFACTOR and PAD_INDEX

The availability of free space in an index page can have a significant effect on the performance of index update operations. If an index record must be inserted and there is no free space, a new index page must be created and the contents of the old page split across the two pages. This can affect performance if it happens too frequently.

The performance impacts of page splits can be alleviated by leaving empty space on each page when creating an index, including a clustered index. This is achieved by specifying a FILLFACTOR value. FILLFACTOR defaults to 0, which means "fill 100%". Any other value (including 100) is taken as the percentage of how full each page should be. For the example in the slide, this means 70% full and 30% free space on each page.

Note The difference between the values 0 and 100 can seem confusing. While both values lead to the same outcome, 100 indicates that a specific FILLFACTOR value has been requested. The value 0 indicates that no FILLFACTOR has been specified.

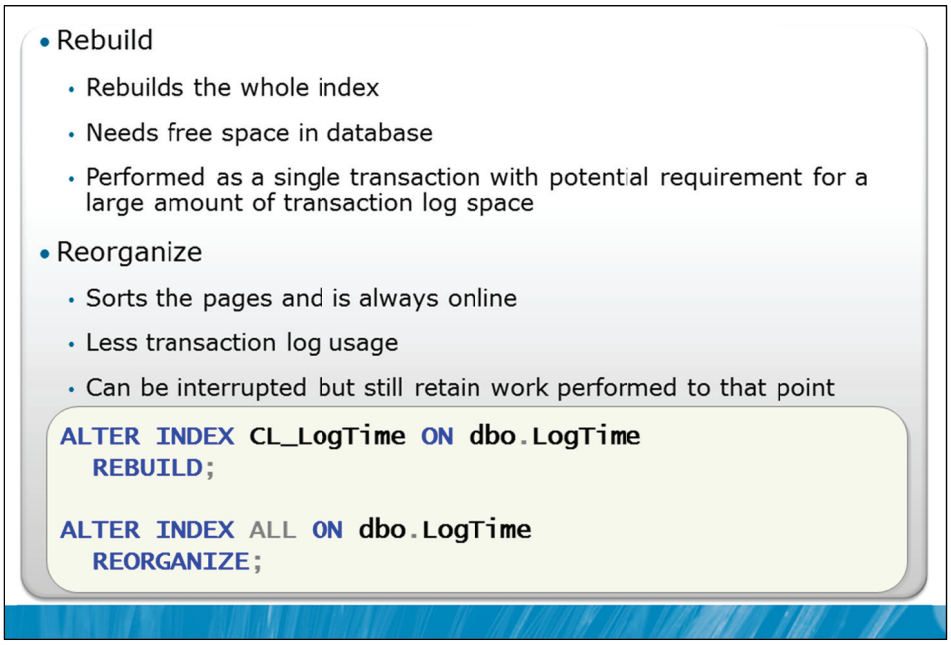
FILLFACTOR only applies to leaf level pages in an index. PAD_INDEX is an option that, when enabled, causes the same free space to be allocated in the non-leaf levels of the index.

Question: While you could avoid many page splits by setting a FILLFACTOR of 50, what would be the downside of doing this?

Question: When would a FILLFACTOR of 100 be useful?

Question: What is the significance of applying a FILLFACTOR on a clustered index versus a non-clustered index?

Ongoing Maintenance of Indexes



Key Points

As indexes are updated during data modifications, over time the indexes can become fragmented. SQL Server provides two options for removing fragmentation from clustered and nonclustered indexes:

- Rebuild
- Reorganize

Rebuild

Rebuilding an index drops and re-creates the index. This removes fragmentation, reclaims disk space by compacting the pages based on the specified or existing fill factor setting, and reorders the index rows in contiguous pages. When the option ALL is specified, all indexes on the table are dropped and rebuilt in a single operation. If any part of the operation fails, the entire operation is rolled back.

Because rebuilds are performed as single operations and are logged, a single rebuild operation can use a large amount of space in the transaction log. It is possible to perform the rebuild operation as a minimally-logged operation when the database is in BULK_LOGGED or SIMPLE recovery model. A minimally-logged rebuild operation uses much less space in the transaction log and completes faster.

Free space needs to be available when rebuilding indexes.

Reorganize

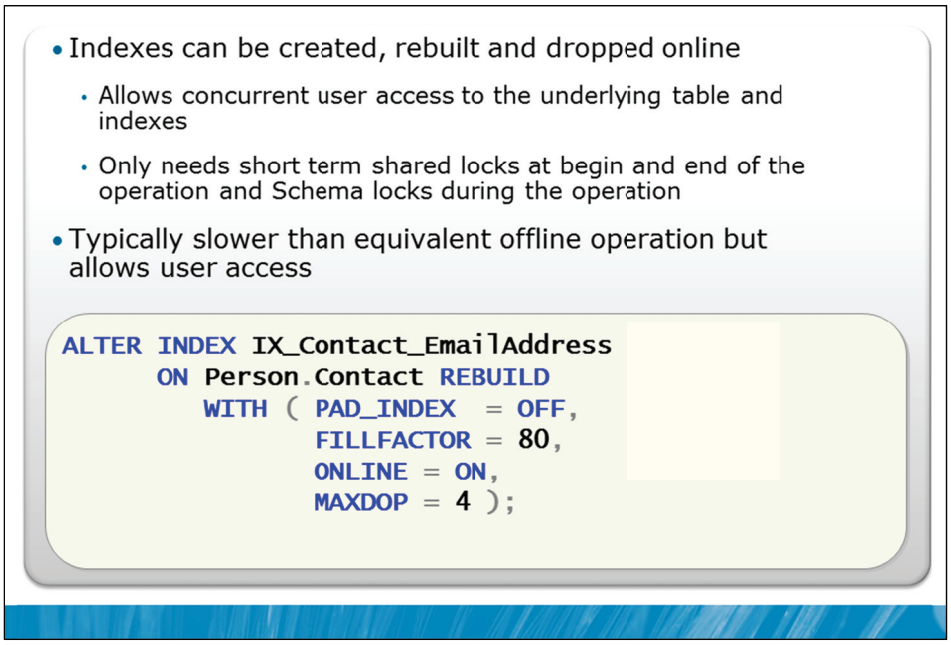
Reorganizing an index uses minimal system resources. It defragments the leaf level of clustered and nonclustered indexes on tables by physically reordering the leaf-level pages to match the logical, left to right order of the leaf nodes. Reorganizing an index also compacts the index pages. The compaction is based on the existing fill factor value. It is possible to interrupt a reorganize without losing the work performed so far. For example, this means that on a large index, partial reorganization could be performed each day.

For heavily fragmented indexes (> 30%) rebuilding is usually the most appropriate option to use.

SQL Server maintenance plans include options to rebuild or reorganize indexes. If maintenance plans are not used, it is important to build a job that performs defragmentation of the indexes in your databases.

Question: What is typically the best option to defragment big indexes and tables (clustered indexes)?

Online Index Operations



Key Points

For most organizations, the primary reason for purchasing the Enterprise edition of SQL Server is that those editions can perform index operations online, while users are accessing the database. This is very important because many organizations have no available maintenance time windows during which to perform database maintenance operations such as index rebuilds.

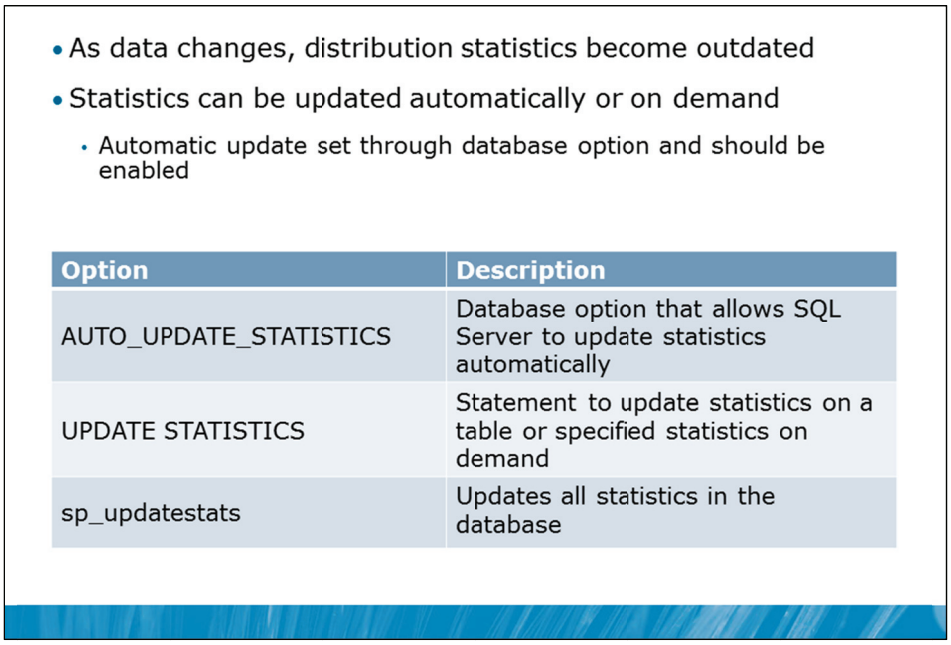
When performing an online index rebuild operation, SQL Server creates a temporary mapping index that tracks data changes that occur while the index rebuild operation is occurring. For consistency, SQL Server takes a very brief shared lock on the object at the beginning of the operation and again at the end. During the online rebuild operation, schema locks are held to prevent metadata changes. This means that users cannot change the structure of the table using commands such as ALTER TABLE while the online index rebuild operation is occurring.

Because of the extra work that needs to be performed, online index rebuild operations are typically slower than their offline counterparts.

Note Some indexes cannot be rebuilt online, including clustered indexes with large object data or nonclustered indexes that include large object data.

Question: When would online index operations be most important?

Updating Statistics



Key Points

One of the main tasks performed by SQL Server when it is optimizing queries that it needs to execute, is deciding which indexes to use. SQL Server makes decisions about which indexes to use based upon statistics that it keeps about the distribution of the data in the index.

Statistics should mostly be updated automatically by SQL Server and AUTO_UPDATE_STATISTICS is enabled in all databases by default. It is recommended that you do not disable this option.

Alternatives to Auto-updating Statistics

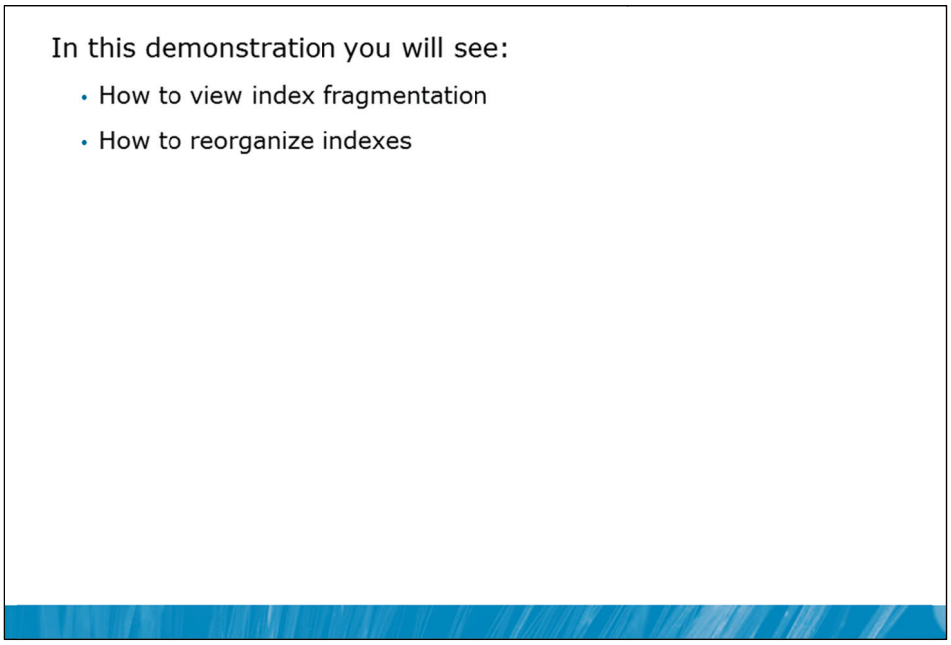
For large tables, the AUTO_UPDATE_STATISTICS_ASYNC option instructs SQL Server to update statistics asynchronously instead of delaying query execution, where it would have otherwise updated an outdated statistic that it required for query compilation.

Statistics can also be updated on demand. Executing the command UPDATE STATISTICS against a table causes all statistics on the table to be updated.

The system stored procedure sp_updatestats can be used to update all statistics in a database.

Question: Why might you decide to update statistics out of hours instead of automatically?

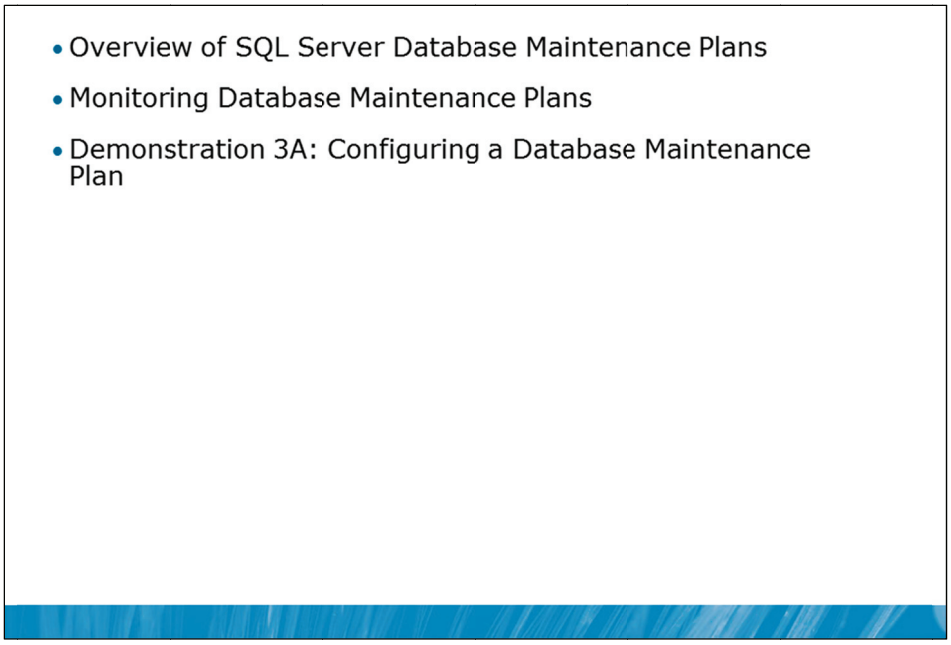
Demonstration 2A: Maintaining Indexes



Demonstration Steps

1. If Demonstration 1A was not performed:
   - Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
   - In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, click **SQL Server Management Studio**. In the Connect to Server window, type **Proseware** and click **Connect**. From the **File** menu, click **Open**, click **Project/Solution**, navigate to **D:\10775A_Labs\10775A_16_PRJ\10775A_16_PRJ.ssmssln** and click **Open**.
   - From the **View** menu, click **Solution Explorer**. Open and execute the **00 – Setup.sql** script file from within Solution Explorer.
   - Open and execute the **11 – Demonstration 1A.sql** script file.
2. Open the **21 – Demonstration 2A.sql** script file.
3. Follow the instructions contained within the comments of the script file.

Lesson 3
Automating Routine Database Maintenance



You have now seen how to manually perform some of the common database maintenance tasks that need to be executed on a regular basis. SQL Server provides a Database Maintenance Plan Wizard that can be used to create SQL Server Agent jobs that perform the most common database maintenance tasks.

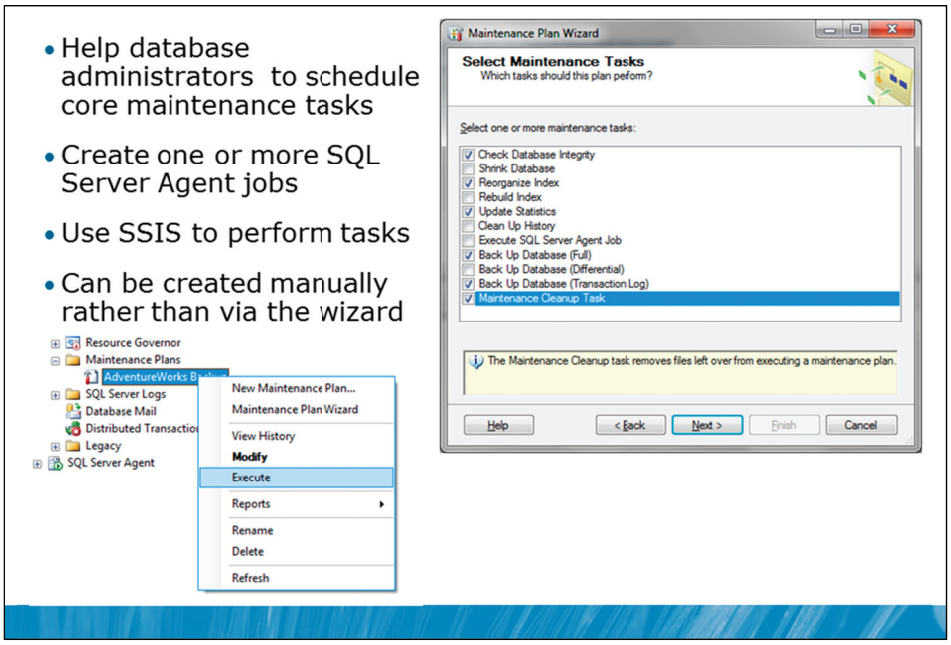
While the Database Maintenance Plan Wizard makes this process easy to set up, it is important to realize that you could use the output of the wizard as a starting point for creating your own maintenance plans, or you could create plans from scratch.

Objectives

After completing this lesson, you will be able to:

- Configure SQL Server database maintenance plans.
- Monitor database maintenance plans.

Overview of SQL Server Database Maintenance Plans



Key Points

The SQL Server Maintenance Plan Wizard creates SQL Server Agent jobs that perform routine database maintenance tasks and schedules those jobs to ensure that your database is regularly backed up, performs well, and is checked for inconsistencies. The wizard creates SQL Server Integration Services packages that are executed by SQL Server Agent tasks.

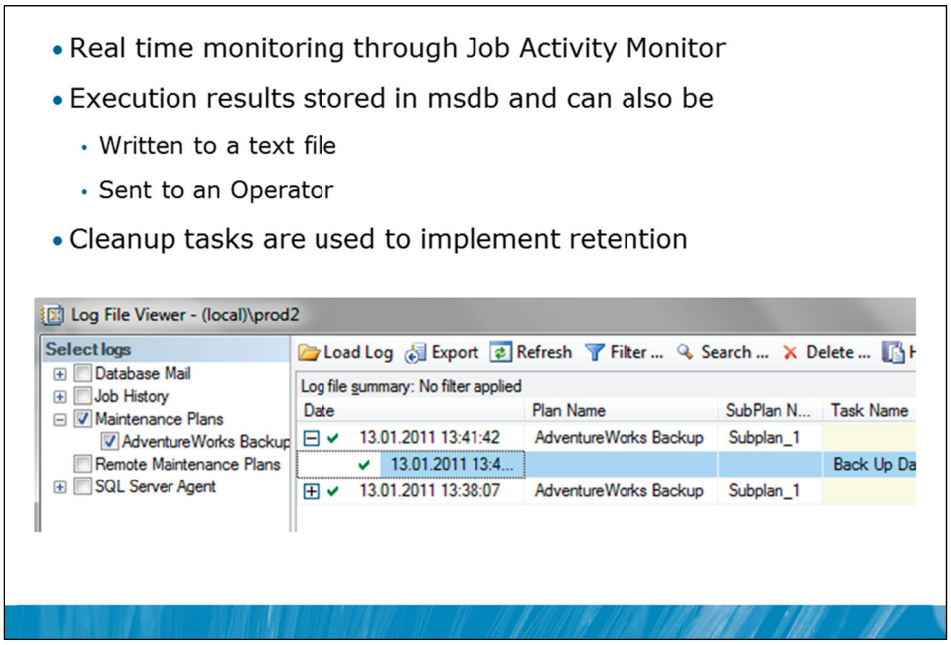
You can schedule many maintenance tasks to run automatically, including:

- Backing up the database and transaction log files. Database and log backups can be retained for a specified period and then automatically deleted.
- Running SQL Server Agent jobs that perform a variety of actions.
- Compacting data files by removing empty database pages.
- Performing internal consistency checks of the data and data pages within the database to make sure that a system or software problem has not damaged data.
- Reorganizing the information on the data pages and index pages by rebuilding indexes.
- Updating index statistics to make sure the query optimizer has up-to-date information about the distribution of data values in the tables.

Note Maintenance plans can be created using one schedule for all tasks or with individual schedules for every selected task.

Question: What types of maintenance tasks should be automated?

Monitoring Database Maintenance Plans



Key Points

SQL Server database maintenance plans are implemented using SQL Server Agent jobs that run SQL Server Integration Services (SSIS) packages. Because they use SQL Server Agent jobs, the maintenance plans can be monitored using the standard Job Activity Monitor in SSMS. As with other SQL Server Agent jobs, job history is written but maintenance plans record additional information.

Results from Maintenance Plans

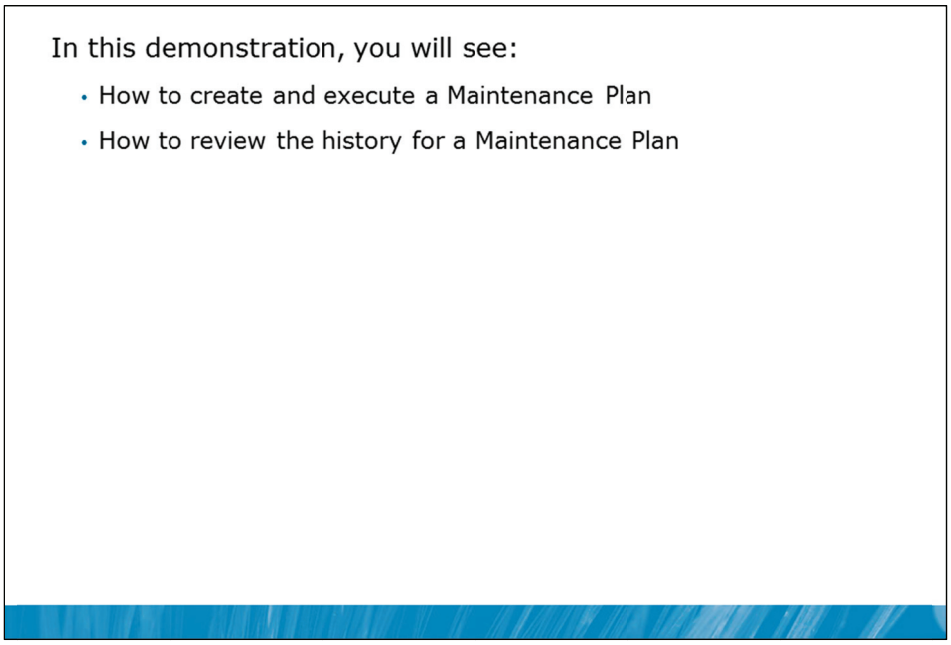
The results generated by the maintenance tasks are written to the maintenance plan tables dbo.sysmaintplan_log and dbo.sysmaintplan_log_detail in the msdb database. The entries in these tables can be viewed by querying those tables directly using T-SQL or by using the Log File Viewer.

In addition, text reports can be written to the file system and can also be sent automatically to operators that have been defined in SQL Server Agent.

Note that the cleanup tasks that are part of the maintenance plans are used to implement a retention policy for backup files, job history, maintenance plan report files, and msdb database table entries.

Question: Are maintenance plan history records cleaned up automatically?

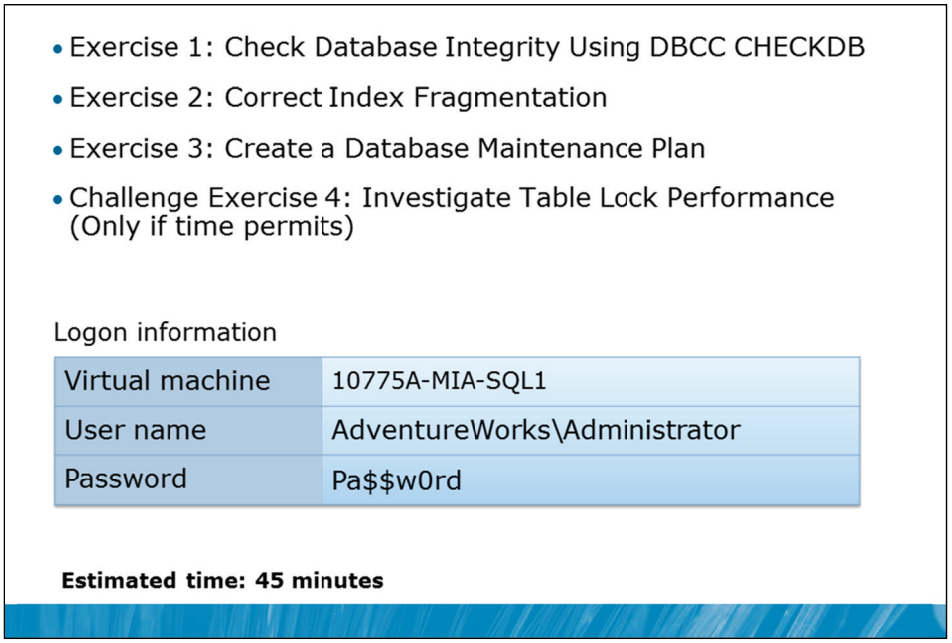
Demonstration 3A: Configuring a Database Maintenance Plan



Demonstration Steps

1. If Demonstration 1A was not performed:
   - Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
   - In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, click **SQL Server Management Studio**. In the Connect to Server window, type **Proseware** and click **Connect**. From the **File** menu, click **Open**, click **Project/Solution**, navigate to **D:\10775A_Labs\10775A_16_PRJ\10775A_16_PRJ.ssmssln** and click **Open**.
   - From the **View** menu, click **Solution Explorer**. Open and execute the **00 – Setup.sql** script file from within Solution Explorer.
   - Open and execute the **11 – Demonstration 1A.sql** script file.
2. Open the **31 – Demonstration 3A.sql** script file.
3. Follow the instructions contained within the comments of the script file.

Lab 16: Performing Ongoing Database Maintenance



Lab Setup

For this lab, you will use the available virtual machine environment. Before you begin the lab, you must complete the following steps:

1. Revert the virtual machines as per the instructions in D:\10775A_Labs\Revert.txt.
2. In the virtual machine, click **Start**, click **All Programs**, click **Microsoft SQL Server 2012**, and click **SQL Server Management Studio**.
3. In the Connect to Server window, type **Proseware** in the **Server name** text box.
4. In the **Authentication** drop-down list box, select **Windows Authentication** and click **Connect**.
5. In the **File** menu, click **Open**, and click **Project/Solution**.
6. In the Open Project window, open the project **D:\10775A_Labs\10775A_16_PRJ\10775A_16_PRJ.ssmssln**.
7. From the **View** menu, click **Solution Explorer**. In Solution Explorer, double-click the query **00-Setup.sql**. When the query window opens, click **Execute** on the toolbar.

Lab Scenario

There has been a disk failure in the I/O subsystem. The disk has been replaced but you want to check the consistency of your existing databases. You will execute DBCC CHECKDB to verify the logical and physical integrity of all databases on the Proseware instance.

You have identified fragmentation in a number of tables in the MarketDev database and you are sure that performance is decreasing as the amount of fragmentation increases. You will rebuild the indexes for any of the main database tables that are heavily fragmented.

You have also identified a degradation of performance in the application when proper index maintenance has not been performed. You want to ensure that there is an early detection of any consistency issues in the MarketDev database and that the index maintenance is automatically executed on a scheduled basis. To make sure this regular maintenance occurs, you will create a Database Maintenance plan to schedule these operations on a weekly basis.

While DBCC CHECKDB runs quite quickly, you are interested in the performance difference that might be achieved by using table locks instead of database snapshots during DBCC CHECKDB operations. If you have time, you will investigate the performance differences.

Supporting Documentation

Database Maintenance Plan Requirements

Item	Configuration
Plan Name	Proseware Weekly Maintenance
Schedule	Once per week for all tasks at 6PM Sunday night
Tasks required	Check Database Integrity for all databases on the Proseware server instance
	Rebuild indexes in the MarketDev database
Notes	The database integrity checks should include indexes
	When indexes in the MarketDev database are rebuilt, pages in the indexes should be 90% full
	As Proseware uses an Enterprise Edition license, online index rebuilds are supported and should be used
	Reports should be written to the folder L:\MKTG

Exercise 1: Check Database Integrity Using DBCC CHECKDB

Scenario

There has been a disk failure in the I/O subsystem. The disk has been replaced but you want to check the consistency of your existing databases. You will execute DBCC CHECKDB to verify the logical and physical integrity of all databases on the Proseware instance.

The main tasks for this exercise are as follows:

1. Check the consistency of the databases on the Proseware instance.
2. Correct any issues found.

▶ Task 1: Check the consistency of the databases on the Proseware instance

- Execute DBCC CHECKDB against all databases on the Proseware server instance. Note any databases that have errors.

▶ Task 2: Correct any issues found

- For any databases with errors, using the DBCC option to repair while allowing data loss. (Note that this is an extreme action that should only be undertaken in emergency situations where no backups are available to be restored).

Results: After this exercise, you should have used the DBCC CHECKDB command to check consistency on all databases on the Proseware instance and corrected any issues that were found.

Exercise 2: Correct Index Fragmentation

Scenario

You have identified fragmentation in a number of tables in the MarketDev database and you are sure that performance is decreasing as the amount of fragmentation increases. You will rebuild the indexes for any of the main database tables that are heavily fragmented.

The main tasks for this exercise are as follows:

1. Review the fragmentation of indexes in the MarketDev database to determine which indexes should be defragmented and which indexes should be rebuilt.
2. Defragment indexes as determined.
3. Rebuild indexes as determined.

▶ Task 1: Review the fragmentation of indexes in the MarketDev database to determine which indexes should be defragmented and which indexes should be rebuilt

- Write a query using sys.dm_db_index_physical_stats function to locate indexes that have more than 30% fragmentation.

▶ Task 2: Defragment indexes as determined

- Write a query to defragment the indexes that you determined had fragmentation levels above 30% but below 70%.

▶ Task 3: Rebuild indexes as determined

- Write a query to rebuild the indexes that you determined had fragmentation levels above 70%

Results: After this exercise, you should have rebuilt or defragmented any indexes with substantial fragmentation.

Exercise 3: Create a Database Maintenance Plan

Scenario

You have also identified a degradation of performance in the application when proper index maintenance has not been performed. You want to ensure that there is an early detection of any consistency issues in the MarketDev database and that the index maintenance is automatically executed on a schedule basis. To make sure this regular maintenance occurs, you will create a Database Maintenance plan to schedule these operations on a weekly basis.

The main task for this exercise is as follows:

1. Create the required database maintenance plan.

▶ Task 1: Create the required database maintenance plan

- Review the requirements for the exercise in the supporting documentation.
- Create a database maintenance plan that meets the requirements.

Results: After this exercise, you should have created the required database maintenance plan.

Challenge Exercise 4: Investigate Table Lock Performance (Only if time permits)

Scenario

While DBCC CHECKDB runs quite quickly, you are interested in the performance difference that might be achieved by using table locks instead of database snapshots during DBCC CHECKDB operations. If you have time, you will investigate the performance differences.

The main tasks for this exercise are as follows:

1. Execute DBCC CHECKDB using database snapshots.
2. Execute DBCC CHECKDB using table locks.

▶ Task 1: Execute DBCC CHECKDB using database snapshots

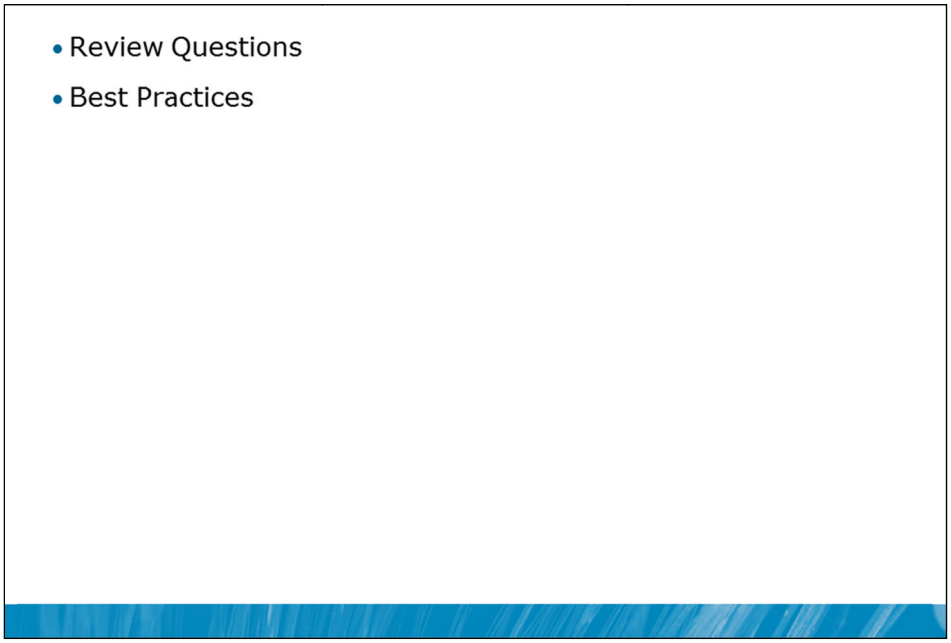
- Execute DBCC CHECKDB against all databases in the Proseware server instance using database snapshots (the default option).
- Record the total execution time.

▶ Task 2: Execute DBCC CHECKDB using table locks

- Execute DBCC CHECKDB against all databases in the Proseware server instance using table locks (TABLOCK option).
- Record the total execution time.
- Compare the execution time to the time recorded in Task 1.

Results: After this exercise, you should have compared the performance of DBCC CHECKDB when using database snapshots and table locks.

Module Review and Takeaways



Review Questions

1. What regular tasks should be implemented for read only databases?
2. What option should you consider using when running DBCC CHECKDB against large production databases?

Best Practices

1. Run DBCC CHECKDB regularly.
2. Synchronize DBCC CHECKDB with your backup strategy.
3. Consider RESTORE before repairing if corruption occurs.
4. Defragment your indexes when necessary.
5. Update statistics on schedule, if you don't want it to occur during normal operations.
6. Use Maintenance Plans to implement regular tasks.