NetIQ® AppManager®

Upgrade and Migration Guide

June 2016

Legal Notice

Contents

About this Book and the Library

The NetIQ AppManager Suite (AppManager) is a comprehensive solution for managing, diagnosing, and analyzing performance, availability, and server health for a broad spectrum of operating environments, applications, and server hardware.

AppManager provides system and application administrators with a central, easy-to-use console to view critical resources across the enterprise. With AppManager, administrative staff can monitor computer and application resources, check for potential problems, initiate and automate responsive actions, automate routine tasks, and gather performance data for real-time and historical reporting and analysis.

Intended Audience

This guide is intended for organizations that have a previous version of AppManager installed and want to upgrade to the latest version. It includes tips and recommendations for a smooth upgrade of all AppManager components. This guide assumes you are already familiar with AppManager components and the installation process.

If you are new to AppManager, have installed AppManager but never Control Center, or want to perform a fresh installation instead of an upgrade, see the *Installation Guide for AppManager*.

Other Information in the Library

The library provides the following information resources:

Installation Guide

Provides detailed planning and installation information.

Administrator Guide

Provides information about maintaining an AppManager management site, managing security, using scripts to handle AppManager tasks, and leveraging advanced configuration options.

Control Center User Guide

Provides information about managing groups of computers, including running jobs, responding to events, creating reports, and working with the Control Center console.

Operator Console User Guide

Provides information for system and network administrators working with the AppManager Operator Console.

Module management guides

Provide information about installing and monitoring specific applications with AppManager.

NetIQ UNIX Agent documentation

Provides information about installing, upgrading, and configuring the NetIQ UNIX Agent and UNIX Agent Manager.

About NetIQ Corporation

We are a global, enterprise software company, with a focus on the three persistent challenges in your environment: Change, complexity and risk—and how we can help you control them.

Our Viewpoint

Adapting to change and managing complexity and risk are nothing new

In fact, of all the challenges you face, these are perhaps the most prominent variables that deny you the control you need to securely measure, monitor, and manage your physical, virtual, and cloud computing environments.

Enabling critical business services, better and faster

We believe that providing as much control as possible to IT organizations is the only way to enable timelier and cost effective delivery of services. Persistent pressures like change and complexity will only continue to increase as organizations continue to change and the technologies needed to manage them become inherently more complex.

Our Philosophy

Selling intelligent solutions, not just software

In order to provide reliable control, we first make sure we understand the real-world scenarios in which IT organizations like yours operate — day in and day out. That's the only way we can develop practical, intelligent IT solutions that successfully yield proven, measurable results. And that's so much more rewarding than simply selling software.

Driving your success is our passion

We place your success at the heart of how we do business. From product inception to deployment, we understand that you need IT solutions that work well and integrate seamlessly with your existing investments; you need ongoing support and training post-deployment; and you need someone that is truly easy to work with — for a change. Ultimately, when you succeed, we all succeed.

Our Solutions

- Identity & Access Governance
- Access Management
- Security Management
- Systems & Application Management
- Workload Management
- Service Management

Contacting Sales Support

For questions about products, pricing, and capabilities, contact your local partner. If you cannot contact your partner, contact our Sales Support team.

Worldwide:	www.netiq.com/about_netiq/officelocations.asp
United States and Canada:	1-888-323-6768
Email:	info@netiq.com
Web Site:	www.netiq.com

Contacting Technical Support

For specific product issues, contact our Technical Support team.

Worldwide:	www.netiq.com/support/contactinfo.asp
North and South America:	1-713-418-5555
Europe, Middle East, and Africa:	+353 (0) 91-782 677
Email:	support@netiq.com
Web Site:	www.netiq.com/support

Contacting Documentation Support

Our goal is to provide documentation that meets your needs. If you have suggestions for improvements, click **Add Comment** at the bottom of any page in the HTML versions of the documentation posted at www.netiq.com/documentation. You can also email Documentation-Feedback@netiq.com. We value your input and look forward to hearing from you.

Contacting the Online User Community

Qmunity, the NetIQ online community, is a collaborative network connecting you to your peers and NetIQ experts. By providing more immediate information, useful links to helpful resources, and access to NetIQ experts, Qmunity helps ensure you are mastering the knowledge you need to realize the full potential of IT investments upon which you rely. For more information, visit http://community.netiq.com.

1 Preparing to Upgrade

This chapter describes items to consider before you upgrade existing AppManager Windows components to version 8.*x*. For information about upgrading the NetIQ UNIX agent, see the NetIQ UNIX Agent documentation, which is included in the AppManager UNIX download package.

1.1 Planning Your Upgrade

The following checklist outlines the basic steps for planning an upgrade and provides references to detailed information.

Step		**Reference**
□	1. Review the features available with AppManager version 8.*x*.	Section 1.2, “Understanding New Features,” on page 12
□	2. Decide whether to upgrade components all at once or on an as-needed basis.	Section 1.3, “Understanding Supported Upgrade Scenarios,” on page 17
□	3. Ensure that the components you want to upgrade meet system requirements and upgrade prerequisites.	◆ Section 1.4, “Understanding Changes to Supported Operating Systems and Databases,” on page 19 ◆ Section 1.5, “Understanding Upgrade Prerequisites,” on page 19 ◆ *Installation Guide for AppManager*
□	4. Ensure that any version 7.*x* AppManager repositories (QDBs) or Control Center repositories (CCDBs) you plan to upgrade are on a supported version of Microsoft SQL Server. If not, migrate them to a supported version.	Chapter 2, “Migrating Version 7.x Repositories,” on page 25
□	5. Schedule components for upgrade in the correct order.	Section 1.6, “Understanding the Recommended Order for Upgrading Components,” on page 22
□	6. Run the setup program to generate a pre-installation check report and resolve issues.	◆ Section 1.7, “Understanding Changes to the Upgrade Process,” on page 22 ◆ Section 1.8, “Understanding Upgrade Methods,” on page 22 ◆ Section 1.9, “Starting an Upgrade and Generating a Pre-Installation Check Report,” on page 23

1.2 Understanding New Features

When deciding which components to upgrade, it is important to consider the features available with version 8.*x* and the requirements for using those features.

1.2.1 Logical Servers

AppManager 8.2 introduces entities known as **logical servers** that allow you to:

- View details about specific entities in the managed environment through new default management groups.

  AppManager 8.2 includes a new default management group, **AM Logical Servers**, that includes all physical computers as well as specific entities on those computers as top-level objects. These entities are known as logical servers and include the following object types:

  - The following VMware vCenter resources:
    - ESX hosts
    - Virtual machines
    - Datacenters
    - Clusters
    - Datastores
    - Datastore clusters
  - Root application objects that allow access to other objects within the application hierarchy (for example, SQL Servers or IIS servers)
  - SQL Server databases and Oracle databases
  - Applications you are monitoring with the AppManager Response Time for Windows (Windows-RT) module
  - Network and SNMP Toolkit devices

  Including both the physical computers and logical servers as top-level objects allows you to view details about the computers as a whole or about only specific entities on those computers. The **AM Logical Servers** management group is divided into the following management groups:

  - **All Logical Servers** displays all physical computers and logical servers that Control Center manages.
  - **Databases** displays all SQL Server databases and Oracle databases that Control Center manages.
  - **Network Devices** displays all network devices that Control Center manages.
  - **Virtual Infrastructure** displays the following resources that Control Center manages:
    - ESX hosts
    - Virtual machines
    - Datacenters
    - Clusters
    - Datastores
    - Datastore clusters

The new management groups include only physical computers and logical servers that belong to QDBs you upgrade to version 8.2 or later. The management groups do not include objects that belong to QDBs that you choose not to upgrade.

For more information about the new management groups, see the *Control Center User Guide for AppManager*.

- Create rule-based management groups made up of specific entities on a physical computer.

  When you created rule-based management groups with previous versions of AppManager, Control Center displayed the complete discovered hierarchy of entities for each computer selected as a group member. To manage specific entities on the computer, you had to navigate the complete hierarchy to locate the specific entity you wanted to investigate. In addition, you could not isolate the entities into their own groups.

  With AppManager 8.2 and later, you can create rule-based management groups consisting of physical computers, logical servers, or both. The ability to create rule-based management groups made up of logical servers gives you more flexibility in defining your view of the managed environment. For example, you can manage SQL Servers by location or manage a select set of SQL Server databases without having to view all of the discovered objects on the SQL Server.

  For QDBs you choose not to upgrade to version 8.2 or later, rules you create to select logical servers will not select objects from those QDBs.

  For more information about creating rule-based management groups, see the *Control Center User Guide for AppManager*.

- Assign custom properties to additional object types.

  Previously, you could only assign custom properties to physical computers. With AppManager version 8.2 and later, you can assign custom properties to both physical computers and logical servers.

  By assigning custom properties to logical servers, you can create rule-based management groups that allow you to dynamically manage logical servers in addition to physical computers. For more information about using custom properties with logical servers, see the *Control Center User Guide for AppManager*.

- View charts and data streams for specific objects in **Servers** views.

  Previously, you could only view charts and data streams at the physical computer level. With AppManager version 8.2 and later, when you select an object in a **Servers** view in the Control Center console (regardless of whether the object is a physical computer or a logical server), the **Charts** tab displays only the charts and data streams for that specific object. For example, if you select a Virtual Center server, you will now see only the charts and data streams related to the virtual server rather than all charts and data streams for the physical host.

1.2.2 Discovery Improvements

With AppManager 8.*x*, Discovery Knowledge Scripts allow you to specify the type of discovery you want to perform when a discovery job runs. You can specify a full discovery or a delta discovery. With delta discovery, version 8.*x* agents can detect changes to your environment since the last discovery and send only the changes to the management server and QDB, allowing you to run discovery jobs on a repeating schedule without overloading the management server and QDB. For most Discovery Knowledge Scripts, the default is to run the discovery job one time. You can adjust the schedule to suit your needs. NetIQ Corporation recommends that you run discovery jobs no more frequently than every 15 minutes. You can also choose to generate events that provide details about the discovered changes and view the events from the Control Center console. For more information about performing discoveries, see the *Control Center User Guide for AppManager*.

Version 8.*x* agents can only communicate with version 8.*x* QDBs and management servers. Before you upgrade an agent, upgrade the QDBs and management servers with which the agent communicates. For more information about supported upgrade scenarios, see Section 1.3, "Understanding Supported Upgrade Scenarios," on page 17. For more information about the order for upgrading AppManager components, see Section 1.6, "Understanding the Recommended Order for Upgrading Components," on page 22.

1.2.3 Security Improvements

AppManager 8.*x* offers improved security by supporting Federal Information Processing Standards (FIPS) and simplifying security administration within Control Center. For more information about FIPS support, see "FIPS Compliance" on page 14." For more information about improved Control Center security administration, see "Simplified Control Center Security Administration" on page 15.

FIPS Compliance

AppManager 8.*x* offers improved protection against security threats and compliance with United States federal government standards by supporting FIPS.

AppManager FIPS compliance consists of the following components:

- FIPS-compliant algorithms that AppManager uses for the **Encrypted communications only** and **Authentication and encrypted communications** security levels
- Control Center console FIPS-only compliance flag

AppManager implements FIPS-compliant algorithms for the **Encrypted communications only** and **Authentication and encrypted communications** security levels. If your site uses the **Encrypted communications only** security level, AppManager encrypts data transmissions between agents and management servers, but does not require agents to authenticate the management servers with which they communicate. If your site uses the **Authentication and encrypted communications** security level, AppManager encrypts data transmissions between agents and management servers and requires agents to authenticate management servers before they transmit data. FIPS compliance does not affect unencrypted communications.

AppManager retains proprietary encryption algorithms for backward compatibility with earlier versions of AppManager and supports a mix of FIPS-compliant and non-FIPS-compliant components. For the **Encrypted communications only** and **Authentication and encrypted communications** security levels, FIPS-compliant components communicate with each other using FIPS-compliant algorithms and communicate with non-FIPS-compliant components using proprietary AppManager encryption algorithms.

The Control Center console offers an option to use only FIPS-compliant security algorithms for the **Encrypted communications only** and **Authentication and encrypted communications** security levels. Because you activate FIPS at the QDB level, the setting affects management servers and agents that are connected to the QDB. If you want to activate the option to use only FIPS-compliant security algorithms, upgrade the QDB, the management servers connected to the QDB, and the agents that communicate with the management servers.

WARNING: If you implement this option, AppManager no longer supports a mixed security environment. If your AppManager environment includes older agents that use secure communications, the older agents will not be able to send events and data to management servers.

SQL Server authentication is not FIPS-compliant. If you plan to activate the option to use only FIPS-compliant security algorithms, install repositories, management servers, and agents to use Windows authentication and configure Kerberos delegation to use Windows authentication. For more information about implementing the Control Center console FIPS-only compliance flag, see the *Control Center User Guide for AppManager*. For more information about configuring Kerberos delegation, see Section 5.7, "Configuring Kerberos Delegation for a Distributed Control Center Environment," on page 65.

Simplified Control Center Security Administration

AppManager version 8.*x* offers the following improvements to Control Center security administration:

- Global permission sets

  Global permission sets allow you to assign permissions that apply to all management groups in Control Center. Previously, only members of the Control Center Administrators group had access to all management groups. The new global permission sets reduce the need to add users to the Administrators group in order to give them permissions in all management groups.

- Permission inheritance by child management groups

  Permissions you assign to a management group automatically apply to each child of that management group. With this change, when you assign a user group to a management group, the user group will have the same permissions on each child of the management group. Permission inheritance eliminates the need to assign user groups or permission sets individually on each child management group, and allows administrators to delegate access for specific user groups to only their portion of the management group hierarchy.

- Removal of dependencies between permissions

  Previously, permission to perform certain activities depended on permission to perform other activities. For example, permission to create jobs required permission to check in and check out Knowledge Scripts. Removal of these dependencies makes it easier for administrators to delegate activities without providing more permissions than necessary.

- Separation of combined permissions

  To allow for separation of responsibilities, some permissions that were combined are now separate permissions. For example, the Allowed to start/stop/close existing jobs permission is now three separate permissions: **Allowed to start jobs**, **Allowed to stop jobs**, and **Allowed to close** jobs.

- Permissions for delegating management group administration

  New permissions have been added to allow administrators to delegate administration of a section of the management group hierarchy to specific users or user groups without having to give them full administrator permissions. Examples include the **Allowed to create and modify management groups and folders** and **Allowed to move management groups and folders** permissions.

- Knowledge Script category permissions

  Knowledge Script category permissions have been added to Control Center to allow administrators to use the Control Center console to control access to specific Knowledge Script categories. Previously, administrators had to use the Operator Console to control access to Knowledge Script categories.

- Creation of read-only QDB accounts in Control Center

You can use the Control Center console to create users and register them with QDBs, and then assign the users to user groups with the required permissions. Registering a user allows the user to access the QDB according to their associated user group and permission sets. If a user only needs to view event and job details and charts, you can assign the user to a group with the **Read Only** permission set.

- Default groups and permission sets

  The Control Center console has a default set of user groups and permission sets that you can copy or modify to develop your own groups and permission sets. Using the default groups and permission sets can reduce the amount of time administrators spend implementing Control Center security.

For more information about Control Center security, see the *Administrator Guide for AppManager*.

1.2.4 Control Center Improvements

With AppManager version 8.*x*, Control Center includes functions that previously were only available through the Operator Console and also includes new features that are not available with the version 8.*x* Operator Console.

You can now use Control Center to perform the following functions:

- Find and delete objects in a tree view.
- View servers, their sub-objects, and details in a tree view.
- View object status in a tree view.
- Drag and drop Knowledge Scripts to create jobs.
- Restrict access to Knowledge Script categories, either globally or per management group.

Control Center also offers new features that are not available with the Operator Console. With a version 8.*x* Control Center, you have the ability to perform the following activities:

- Configure Discovery Knowledge Scripts to run more frequently and collect only information about changes to your environment since the last discovery. For most Discovery Knowledge Scripts, the default is to run the discovery job one time. You can adjust the schedule to suit your needs. NetIQ Corporation recommends that you run discovery jobs no more frequently than every 15 minutes.
- Monitor the health of AppManager components.
- View information about the agent and application-monitoring support installed on a Windows computer.

These new features are only available for upgraded AppManager components. If you choose to maintain existing version 7.*x* QDBs, management servers, and agents in your upgraded Control Center environment, these features will not be available for those components. For more information about maintaining existing AppManager components in an upgraded Control Center environment, see Section 1.3.2, “Upgrading Components in a Multiple-QDB, Control Center Environment,” on page 17.

For more information about installing Control Center in an environment in which you previously only used the Operator Console, see the *Installation Guide for AppManager*. For more information about upgrading existing Control Center components, see Chapter 5, “Upgrading Control Center Components,” on page 61.

For more information about Control Center, see the *Control Center User Guide for AppManager*.

1.3 Understanding Supported Upgrade Scenarios

You do not have to upgrade all of your AppManager components to version 8.*x* at the same time; however, components you do not upgrade might not support new features. This section provides information about the configurations AppManager 8.*x* supports.

1.3.1 Upgrading Management Servers and Agents in a Single QDB Environment

If you have only one QDB in your environment, you must upgrade the QDB and the primary and secondary management servers that connect to the QDB. When you upgrade a management server, you also upgrade the agent on the management server computer. Otherwise, you can upgrade agents on an as-needed basis.

While a version 8.*x* management server can communicate with version 6.0.*x*, 7.0.*x*, and 8.0.*x* agents, version 8.*x* agents cannot communicate with earlier versions of the management server.

For more information about upgrading QDBs and management servers, see Chapter 3, “Upgrading Management Site Components,” on page 45.

For more information about upgrading agents, see Chapter 4, “Upgrading and Migrating Agent Components,” on page 55.

1.3.2 Upgrading Components in a Multiple-QDB, Control Center Environment

With Control Center version 8.*x*, the primary QDB must be a version 8.*x* QDB. You can either create a new version 8.*x* primary QDB or upgrade your existing primary QDB to version 8.*x*. If you create a new primary QDB, in order for the product to function correctly, you must add at least one agent computer to the QDB and discover the computer. With a version 8.*x* primary QDB, you can maintain existing non-primary QDBs, management servers, and agents as long as they meet the requirements described in the following table.

Existing Component	AppManager Version	Microsoft SQL Server Version
Non-primary QDB	7.0.1, with Hotfix 7011821 or later To obtain the hotfix, visit the AppManager Suite Hotfixes Web site.	2005 Standard or Enterprise edition Service Pack 2 or later (32-bit)
	7.0.3, with Hotfix 7011821 or later To obtain the hotfix, visit the AppManager Suite Hotfixes Web site.	2005 Standard or Enterprise edition Service Pack 2 or later (64-bit)
	AppManager 7 Platform Update, with Hotfix 7011821 or later To obtain the hotfix, visit the AppManager Suite Hotfixes Web site.	One of the following (32-bit or 64-bit): ◆ 2008 Standard or Enterprise edition Service Pack 1 or later ◆ 2008 R2 Standard or Enterprise edition
	8.0.2, with Hotfix 7011822 or later To obtain the hotfix, visit the AppManager Suite Hotfixes Web site.	One of the following (32-bit or 64-bit): ◆ 2005 Standard or Enterprise edition Service Pack 2 or later ◆ 2008 Standard or Enterprise edition Service Pack 1 or later ◆ 2008 R2 Standard or Enterprise edition
	8.0.3, with Hotfix 7011823 or later To obtain the hotfix, visit the AppManager Suite Hotfixes Web site.	
Management server	One of the following: ◆ 7.0.1 ◆ AppManager 7 Platform Update ◆ 8.0.2 ◆ 8.0.3	Not applicable
Agent	One of the following: ◆ 7.0.1 ◆ 7.0.2 ◆ 7.0.25 ◆ 8.0.1 ◆ 8.0.2	Not applicable

Once you have a version 8.*x* primary QDB, you can upgrade non-primary QDBs on an as-needed basis. When you upgrade a non-primary QDB, you must also upgrade the primary and secondary management servers that connect to that QDB. When you upgrade a management server, you also upgrade the agent on the management server computer. Otherwise, you can upgrade agents on an as-needed basis.

For more information about creating a new version 8.*x* QDB, see the *Installation Guide for AppManager*. For more information about upgrading existing QDBs and management servers to version 8.*x*, see Chapter 3, “Upgrading Management Site Components,” on page 45. For more information about upgrading agents, see Chapter 4, “Upgrading and Migrating Agent Components,” on page 55.

For more information about upgrading Control Center to version 8.*x*, see Chapter 5, “Upgrading Control Center Components,” on page 61.

1.4 Understanding Changes to Supported Operating Systems and Databases

Both AppManager version 8.*x* and Hotfix 7011821, which is a prerequisite hotfix for version 7.*x* non-primary QDBs you want to maintain with Control Center version 8.*x*, remove support for the following operating systems and databases:

- Microsoft Windows 2000
- Microsoft SQL Server 2000

AppManager version 8.*x* adds support for the following operating systems and databases:

- For all components, Microsoft Windows Server 2008 Standard and Enterprise editions (32-bit and 64-bit)
- For all components, Microsoft Windows Server 2008 R2 Standard and Enterprise editions
- For all components, Microsoft Windows Server 2012 Standard and Datacenter editions (GUI mode only)
- For the Control Center console, Operator Console, Operator Web Console, and Windows agent, Microsoft Windows 7 Business and Enterprise editions (32-bit and 64-bit)
- For the Control Center console, Operator Console, Operator Web Console, and Windows agent, Microsoft Windows 8 (32-bit and 64-bit)
- For the QDB and CCDB:
  - Microsoft SQL Server 2008 Standard and Enterprise editions Service Pack 1 or later (32-bit and 64-bit)
  - Microsoft SQL Server 2008 R2 Standard and Enterprise editions (32-bit and 64-bit)
  - Microsoft SQL Server 2012 Standard and Enterprise editions (32-bit and 64-bit)

For more information about supported operating systems, databases, and system requirements, see the *Installation Guide for AppManager*.

1.5 Understanding Upgrade Prerequisites

This section describes prerequisites your existing AppManager components must meet before you upgrade them.

1.5.1 Understanding Microsoft SQL Server Versions Supported for Upgrade and Migration

Version 8.*x* QDBs and CCDBs must be hosted on one of the following versions of Microsoft SQL Server:

- For the QDB, Microsoft SQL Server 2005 Standard or Enterprise edition Service Pack 2 or later (32-bit or 64-bit)
- For the CCDB, Microsoft SQL Server 2005 Standard or Enterprise edition Service Pack 3 or later (32-bit or 64-bit)
- Microsoft SQL Server 2008 Standard or Enterprise edition Service Pack 1 or later (32-bit or 64-bit)
- Microsoft SQL Server 2008 R2 Standard or Enterprise edition (32-bit or 64-bit)
- Microsoft SQL Server 2012 Standard or Enterprise edition (32-bit or 64-bit)

Before you can upgrade a QDB or CCDB hosted on Microsoft SQL Server 2000 to AppManager 8.*x*, you must migrate the repository to a supported version of Microsoft SQL Server. Once the repository is hosted on a supported version of SQL Server, you can upgrade it to version 8.*x*. For more information about migrating a repository before upgrading it to version 8.*x*, see Chapter 2, “Migrating Version 7.x Repositories,” on page 25.

AppManager 8.*x* supports migrating version 8.*x* QDBs and CCDBs hosted on Microsoft SQL Server 2005 to Microsoft SQL Server 2008, 2008 R2, or 2012 and version 8.*x* QDBs and CCDBs hosted on SQL Server 2008 or 2008 R2 to SQL Server 2012. For more information about upgrading a QDB, see Chapter 3, “Upgrading Management Site Components,” on page 45. For more information about upgrading a CCDB, see Chapter 5, “Upgrading Control Center Components,” on page 61. For more information about migrating an upgraded repository, see Appendix A, “Migrating Version 8.x Repositories,” on page 77.

1.5.2 Understanding AppManager Versions Supported for Upgrade

Before you upgrade AppManager components, ensure that the components meet version prerequisites. The following table lists the components and the AppManager versions supported for upgrade.

Component	Supported AppManager Versions for Upgrade
QDB	◆ AppManager 7 Platform Update ◆ 8.0.*x*
Management server	
Web management server	
Agent	◆ 7.0.1 ◆ 7.0.2 ◆ 7.0.25 ◆ 8.0.*x*

Component	Supported AppManager Versions for Upgrade
CCDB	◆ 7.0.4 ◆ AppManager 7 Platform Update ◆ 8.0.*x*
Command queue service (CQS)	
Deployment Service	
Deployment Web Service	
Control Center console	
Operator Console	◆ AppManager 7 Platform Update ◆ 8.0.*x*

For more information about obtaining a supported version, contact Technical Support.

If you have a report agent, before you upgrade, you can run the `CompVersion` report Knowledge Script to generate a report detailing the version number of AppManager components installed on all computers in an AppManager site. For more information about running the Knowledge Script, see the *AppManager Knowledge Script Reference Guide*.

If you do not have a report agent, use the methods described in the following table to obtain component version information.

Component	Steps to Take
Operator Console	1. Start the Operator Console. 2. Select **Help > About AppManager Operator Console**.

QDB	1. In the Operator Console, select **Extensions > Repository Browser**.
	2. From the Tables list, select **Version**.
	3. Run the following query:
	`SELECT * FROM Version`
	4. Scroll through the results until you find the following record:
	`MachineName: computer name`
	`Component: Repository`
	`Version: current QDB version`

Management server	1. In the Operator Console, right-click the server. 2. Select **Troubleshooter > Management Service Info > Connectivity**.
Agent	1. In the Operator Console, right-click the computer whose agent you want to check. 2. Select **Troubleshooter > Client Resource Monitor Info > Connectivity**.

Control Center components	Use the Registry Editor to view the following key:
	`HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager`
	`\Control Center\1.0`

1.6 Understanding the Recommended Order for Upgrading Components

NetIQ Corporation recommends upgrading AppManager components in the following order:

1. Primary QDB
2. Management server
3. Web management server
4. Windows agent
5. Modules
6. CCDB
7. Control Center and Deployment services
8. Control Center console
9. Non-primary QDBs
10. Operator Console
11. Jobs

For the computer on which you run the setup program, the program automatically detects the components available for upgrade and upgrades those components in the recommended order. If you are upgrading components on multiple computers, schedule the upgrades according to the recommended order. For example, if you are upgrading the management server and agent on different computers, run the setup program on the management server before you run it on the agent computer.

If you do not upgrade Control Center components in the recommended order, the Deployment Service will not start after the upgrade.

1.7 Understanding Changes to the Upgrade Process

With AppManager 8.*x*, you can upgrade QDBs and CCDBs on remote SQL Servers. You no longer have to run the setup program on the SQL Server.

For all components except the QDB and CCDB, when you run the setup program, it automatically detects the components available for upgrade on that computer. Because you can upgrade QDBs and CCDBs on remote SQL Servers, the setup program does not detect if there is a QDB or CCDB available for upgrade. Instead, you select options to upgrade those components.

1.8 Understanding Upgrade Methods

You can upgrade AppManager components interactively or silently from a command prompt. For information about silently upgrading components, see the *Installation Guide for AppManager*. Regardless of whether you choose to upgrade components interactively or silently, ensure that the installation path contains only ASCII characters. If a component is already installed in a path that contains non-ASCII characters, uninstall the component, and then install it in a supported path.

To interactively upgrade components, NetIQ Corporation recommends using the AppManager setup program. When you run `Setup.exe`, the setup program runs a pre-installation check script to verify system requirements and then runs individual Windows Installer packages for the components you selected to upgrade. During the upgrade process, the setup program automatically installs the runtime libraries required for the selected components. For more information about running `Setup.exe` to generate a pre-installation check report and interactively upgrade components, see Section 1.9, “Starting an Upgrade and Generating a Pre-Installation Check Report,” on page 23.

You can run Windows Installer packages for individual components instead of using the AppManager setup program. If you use this method to upgrade components, you must complete the following tasks before you run the Windows Installer packages:

- Manually install the required runtime libraries for the components you are upgrading.

  For more information about installing the runtime libraries, see the *Installation Guide for AppManager*.
- (Conditional) If the computer from which you will run the Windows Installer package has the User Access Control (UAC) feature enabled, ensure that the user who will perform the installation is authorized to run the package.

  For more information about ensuring the user is authorized, see the *Installation Guide for AppManager*.

If you choose to upgrade the Control Center services by running the Windows Installer package instead of the AppManager setup program, after the upgrade, the NetIQ AppManager Client Resource Monitor (`NetIQmc`) and NetIQ AppManager Client Communication Manager (`NetIQccm`) services will not start until you upgrade the agent. For more information about upgrading Control Center components, see Chapter 5, “Upgrading Control Center Components,” on page 61.

1.9 Starting an Upgrade and Generating a Pre-Installation Check Report

The upgrade steps in this section are common to all AppManager components, except the UNIX agent. For more information about upgrading the UNIX agent, see the NetIQ UNIX Agent documentation, which is included in the AppManager UNIX download package.

The upgrade process does not change the settings from your previous AppManager installation. If you want to change the installation settings, uninstall the components and then perform a new installation. For more information about performing a new installation, see the *Installation Guide for AppManager*.

To start an upgrade and generate a pre-installation check report:

1. (Conditional) If deployment tasks are in the Waiting for Approval state, approve or reject the tasks before you start the upgrade. Otherwise, the tasks will fail.
2. (Conditional) If you are upgrading only a QDB or CCDB, run `Setup.exe` on the computer from which you want to perform the upgrade.
3. (Conditional) If you are upgrading components other than the QDB or CCDB, run `Setup.exe` on the computer where the components you want to upgrade are installed.
4. On the Welcome window, view the components available for upgrade on the computer.
5. (Conditional) If you want to upgrade a QDB or CCDB, select the appropriate option.
6. Click **Next**.

7. On the Confirmation window, click the link to view the pre-installation check report.

   The AppManager pre-installation check script verifies system requirements and generates a report that summarizes the results. For each requirement, the report provides information about how your environment meets or does not meet the requirement and the check result. The following results are possible:

   - Passed - Your environment passed the check.
   - Warning - Your environment passed the check, but configuration issues exist.
   - Failed - Your environment failed the check.

8. (Conditional) If your environment passed all requirements, click **Next**.

   The AppManager setup program launches the individual component setup programs.

9. (Conditional) If your environment did not pass all requirements, resolve issues and re-generate the pre-installation check report.

10. Complete the upgrade steps for the components you want to upgrade.

    The following table provides references to detailed information about upgrading AppManager components.

For more information about upgrading the...	See...
QDB	Chapter 3, “Upgrading Management Site Components,” on page 45
Management server	Chapter 3, “Upgrading Management Site Components,” on page 45
Web management server	Chapter 3, “Upgrading Management Site Components,” on page 45
Windows agent	Chapter 4, “Upgrading and Migrating Agent Components,” on page 55
CCDB	Chapter 5, “Upgrading Control Center Components,” on page 61
Control Center and Deployment services	Chapter 5, “Upgrading Control Center Components,” on page 61
Control Center console	Chapter 5, “Upgrading Control Center Components,” on page 61

2 Migrating Version 7.*x* Repositories

This chapter describes how to migrate a version 7.*x* AppManager repository (QDB) or Control Center repository (CCDB) to a new computer before upgrading to AppManager 8.*x*.

2.1 Understanding Supported 7.*x* Repository Migration Scenarios

You can migrate a 7.*x* QDB or CCDB to a new computer running the following operating system and Microsoft SQL Server combinations:

- Microsoft Windows Server 2003 or 2003 R2 with SQL Server 2005 (32-bit and 64-bit)
- Microsoft Windows Server 2008 with SQL Server 2008 or 2008 R2 (32-bit and 64-bit)
- Microsoft Windows Server 2008 R2 with SQL Server 2008 or 2008 R2 (64-bit)

2.2 Understanding the Migration and Upgrade Process

The recommended method for migrating and upgrading a repository is to install a new, empty version 7.*x* repository on the new computer, close connections to the repository you will migrate, create a backup copy, restore the backup copy over the empty repository on the new computer, and then upgrade the repository to version 8.*x* on the new computer. This method allows you to retain the original repository as a fallback until you are satisfied the upgraded repository on the new computer functions as expected.

The following checklist outlines the migration and upgrade process and provides references to detailed information.

Step		Reference
☐	1. Prepare for repository migration.	Section 2.3, “Preparing to Migrate the Repositories,” on page 26
☐	2. Restore the backup copy over the new, empty repository on the new computer.	Section 2.4, “Restoring Repositories on the New Computer,” on page 33
☐	3. Configure the restored repository on the new computer.	Section 2.5, “Configuring Restored Repositories,” on page 34
☐	4. Update components and services that connect to the repository.	Section 2.6, “Updating Connected Components and Services,” on page 38
☐	5. Verify that the migration was successful.	Section 2.7, “Verifying Successful Migration,” on page 42

Step		Reference
❑	6. Upgrade the restored repository to version 8.*x* on the new computer.	(Conditional) If you migrated a QDB, Chapter 3, “Upgrading Management Site Components,” on page 45 (Conditional) If you migrated a CCDB, Chapter 5, “Upgrading Control Center Components,” on page 61

2.3 Preparing to Migrate the Repositories

Before you migrate a repository to a new computer, install a new, empty version 7.*x* repository on the new computer, record information about the repository you will migrate, close connections to it, and create a backup copy.

To prepare for repository migration:

1 On the new computer, install a new AppManager 7.*x* QDB or CCDB. During installation, when you specify the repository name, specify the same name as the repository you will migrate.

The following table lists the AppManager version to install based on the operating system and Microsoft SQL Server combination on the new computer. For information about obtaining the AppManager versions listed, contact Technical Support.

If the new computer is running...	And the processor type is...	Install AppManager version...
Microsoft Windows Server 2003 or 2003 R2 with SQL Server 2005	32-bit	(Conditional) If you are migrating the QDB, 7.0.1 with Hotfix 72040 or later (Conditional) If you are migrating the CCDB, 7.0.4 with Hotfix 71949 or later
Microsoft Windows Server 2003 or 2003 R2 with SQL Server 2005	64-bit	(Conditional) If you are migrating the QDB, 7.0.3 with Hotfix 72040 or later (Conditional) If you are migrating the CCDB, 7.0.4 with Hotfix 71949 or later
Microsoft Windows Server 2008 with SQL Server 2008 or 2008 R2	32-bit or 64-bit	(Conditional) If you are migrating the QDB, AppManager 7 Platform Update (Conditional) If you are migrating the CCDB, AppManager 7 Platform Update
Microsoft Windows Server 2008 R2 with SQL Server 2008 or 2008 R2	64-bit	

2 (Conditional) If you installed an AppManager 7 Platform Update QDB on the new computer, also install an AppManager 7 Platform Update management server.

The AppManager 7 Platform Update QDB requires an AppManager 7 Platform Update management server. The AppManager 7 Platform Update does not support upgrading the management server. You have the following options for installing an AppManager 7 Platform Update management server:

- Uninstall the existing management server, upgrade the operating system to a supported version, and then install an AppManager 7 Platform Update management server and point it to the QDB you will migrate.
- Move to an AppManager 7 Platform Update management server on a new computer, which requires installing a Platform Update management server on the new computer, pointing it to the QDB you will migrate, and then pointing any existing agents that communicated with the old management server to the new Platform Update management server.

  For more information about moving the existing management server, see Section 2.3.1, "Moving to a Platform Update Management Server on a New Computer," on page 30.

3 On each computer, to ensure that the Distributed Transaction Coordinator (DTC) security settings are the same, complete the following steps:

WARNING: If the settings are not the same, the migration will fail.

3a In the Component Services application in Administrative Tools, expand `Component Services\Computers\My Computer\Distributed Transaction Coordinator`.

3b Right-click **Local DTC** and select **Properties**.

3c On the **Security** tab, note the settings.

3d (Conditional) If the settings on the new computer do not match the settings on the old computer, adjust the settings on the new computer and restart it.

4 On the SQL Server that hosts the repository you will migrate, to note the properties for SQL Server logins with access to the repository, complete the following steps:

4a In Microsoft SQL Server Management Studio, expand *`SQL_Server_Name`*`\Databases`.

4b Right-click the repository and select **New Query**.

4c In the query window, type the following command, and then click **Execute**:

```
SELECT  name,
        CASE WHEN type = 'S' THEN 'SQL'
             ELSE 'Windows'
        END AS 'Type'
FROM    sys.database_principals
WHERE   type IN ( 'S', 'U' )
        AND name != 'dbo'
        AND default_schema_name IS NOT NULL
        AND default_schema_name != 'guest'
ORDER BY name ASC
```

4d Expand `Databases\`*`Repository_Name`*`\Security\Users` and compare the accounts listed in the results table for the query to the accounts in the `Users` folder.

4e With the exception of the `probe` account, for each account that appears in both the results table and the `Users` folder, right-click the user in the `Users` folder and select **Properties**.

It is not necessary to note the properties for the `probe` user.

4f On the **General** page, note the **Login name** and **Database role membership**.

4g On the **Securables** page, note the **Explicit permissions**.

With the exception of the `probe` account, you will recreate the SQL Server logins after you restore the repository on the new computer.

5 (Conditional) If you changed the schedule for any NetIQ SQL Server jobs, to note the schedule settings for each modified job, complete the following steps on the SQL Server that hosts the repository you will migrate:

5a In Microsoft SQL Server Management Studio, expand `SQL Server Agent\Jobs`.

5b Right-click the job and select **Properties**.

5c On the **Schedules** page, select the job schedule, and then click **Edit**.

5d On the Job Schedule Properties window, note the settings, and then click **OK**.

6 (Conditional) If you will migrate a CCDB that manages remote QDBs, to note the linked server properties for the remote QDBs, complete the following steps on the SQL Server that hosts the CCDB you will migrate:

6a In Microsoft SQL Server Management Studio, expand `Server Objects\Linked Servers`.

6b Right-click a linked QDB and select **Properties**.

6c On the **General** page, note the linked server name and server type.

6d On the **Security** page, note each local login defined and how Control Center makes the connection.

A login can:

- **Be made without using a security context.** If this option is selected, Control Center connects without using any login and password.
- **Be made using the login's current security context**. If this option is selected, Control Center uses the Log On As account for the SQL Server Agent service to log in to the remote QDB.
- **Be made using this security context**. If this option is selected, it implies you checked the **Use SQL Server authentication** option when you added the QDB to Control Center. When you restore the SQL Server link on the new CCDB computer, provide the same SQL Server user name and password you provided when you added the QDB to Control Center.

6e On the **Server Options** page, note the **RPC** and **RPC Out** values.

You will restore the SQL Server links after you restore the CCDB on the new computer.

7 (Conditional) If you will migrate a QDB that is not an AppManager 7 Platform Update QDB, apply Hotfix 72040 or later to the QDB.

To obtain the hotfix, visit the AppManager Suite Hotfixes Web site.

8 (Conditional) If you will migrate a CCDB that is not an AppManager 7 Platform Update CCDB, apply Hotfix 71949 or later to the CCDB.

To obtain the hotfix, visit the AppManager Suite Hotfixes Web site.

9 On each computer, to ensure that the SQL Server collation order, sort order, and character set are the same, complete the following steps:

WARNING: If the settings are not the same, the migration will fail.

9a In Microsoft SQL Server Management Studio, right-click the SQL Server instance and select **Properties**.

9b On the **General** page, note the **Server Collation** setting, and then click **OK** to close the Properties window.

9c Right-click the SQL Server instance and select **New Query**.

9d In the query window, type the following command, and then click **Execute**:

```
sp_helpsort
```

The sort order and character set is displayed in the results table.

When you install SQL Server, the collation order is set by default according to the locale of the operating system. You can use advanced installation options to change the collation order. If the collation order is not the same, re-install SQL Server on the new computer and set the collation order to be the same as the collation order on the old computer.

10 (Conditional) If you will migrate the QDB, complete the following steps to close connected services:

10a Click **Start** > **Administrative Tools** > **Services**.

10b For each of the following services, right-click the service and select **Stop**:

- On the SQL Server that hosts the QDB, SQL Server Agent service
- On primary and secondary management servers that connect to the QDB, NetIQ AppManager Management Service
- (Conditional) If you are running NetIQ AppManager Performance Profiler version 4.0.2 or later, on the SQL Server that hosts AppManager Performance Profiler, Analytics service

  Stop the Analytics service at the same time you stop the NetIQ AppManager Management Service.
- On primary and secondary management servers that connect to the QDB, NetIQ AppManager Client Communication Manager and NetIQ AppManager Client Resource Monitor services
- (Conditional) If Control Center manages the QDB, on the command queue service computer, NetIQ AppManager Control Center Command Queue Service
- (Conditional) If Control Center manages the QDB, on the SQL Server that hosts the CCDB, SQL Server Agent service
- (Conditional) If report agents connect to the QDB, on the agent computers, NetIQ AppManager Client Communication Manager and NetIQ AppManager Client Resource Monitor services

If a service is set to automatically restart when it stops, disable the service.

11 (Conditional) If you will migrate the CCDB, complete the following steps to close connected services:

11a Click **Start** > **Administrative Tools** > **Services**.

11b For each of the following services, right-click the service and select **Stop**:

- On the command queue service computer, NetIQ AppManager Control Center Command Queue Service
- On the SQL Server that hosts the CCDB, SQL Server Agent service
- On the Deployment Service computer, NetIQ AppManager Deployment Service
- On the Deployment Web Service computer, World Wide Web Publishing Service that manages the Deployment Web Service and the Web Depot virtual directories

12 (Conditional) If you will migrate the QDB, stop AppManager Connectors that connect directly to it, such as the AppManager Connector for Micromuse Netcool/OMNIbus or AppManager Connector for Security Manager.

13 Close any AppManager consoles, such as the Control Center console and Operator Console, that connect to the repository.

14 (Conditional) If you will migrate the QDB and it is a data source for Analysis Center, complete the following steps on the Data Mart computer to stop the Analysis Center ETL job:

14a In Microsoft SQL Server Management Studio, expand `SQL_Server_Name\SQL Server Agent\Jobs`.

14b Navigate to the ETL job.

14c Right-click the job and select **Disable**.

15 To verify that there are no open connections to the repository you will migrate, complete the following steps:

15a On the repository computer, in Microsoft SQL Server Management Studio, expand `SQL_Server_Name\Databases`.

15b Right-click the repository and select **New Query.**

15c In the query window, type the following command, and then click **Execute**:

```
USE master
GO
Exec sp_who2
GO
```

15d In the results table, check the **DBName** column for the repository name. The column should not contain entries for the repository.

16 (Conditional) If the **DBName** column contains entries for the repository, complete the following steps for each entry:

16a In the **SPID** column for the row in which the repository name appears, note the SPID number.

16b Right-click the repository and select **New Query**.

16c In the query window, type the following command, and then click **Execute**:

```
kill SPID_Number
```

17 Repeat Step 15 on page 30 and Step 16 on page 30 until the **DBName** column does not contain entries for the repository.

18 Create a backup copy of the repository you will migrate.

For more information about creating backup copies, see Section 2.3.2, “Creating Backup Copies of the Repositories,” on page 32.

After you create a backup copy of the repository you will migrate, you can restore the backup copy over the new, empty repository. For more information about restoring the repository, see Section 2.4, “Restoring Repositories on the New Computer,” on page 33.

2.3.1 Moving to a Platform Update Management Server on a New Computer

This section describes the steps required to move an existing management server to an AppManager 7 Platform Update management server on a new computer.

To move the management server:

1 To allow agent computers that communicate with the management server to temporarily allow communication with anonymous management servers, run the AMAdmin_SetAllowMS Knowledge Script and set the **New hostname(s) for AllowMS** parameter to an asterisk (*).

For more information about the Knowledge Script, see the *AppManager Knowledge Script Reference Guide*.

2 (Conditional) If the following registry keys on the agent computers are not set to an asterisk (*), use the NTAdmin_RegistrySet Knowledge Script to add the name of the new management server computer to the key values:

On this type of operating system...	Add the management server name to...
32-bit	◆ `HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQmc\Security\AllowDosCmd` ◆ `HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQmc\Security\AllowMS` ◆ `HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQmc\Security\AllowReboot`
64-bit	◆ `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\NetIQ\AppManager\4.0\NetIQmc\Security\AllowDosCmd` ◆ `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\NetIQ\AppManager\4.0\NetIQmc\Security\AllowMS` ◆ `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\NetIQ\AppManager\4.0\NetIQmc\Security\AllowReboot`

Otherwise, certain actions will not be allowed after you move the management server. For example, the `AllowReboot` registry key will no longer allow Action_RebootSystem. For information about the parameters to specify in the Knowledge Script, see the *AppManager Knowledge Script Reference Guide*.

3 On the new computer, install an AppManager 7 Platform Update management server and point it to the QDB you will migrate.

For more information about installing an AppManager 7 Platform Update management server, see the *AppManager 7 Platform Update Release Notes*, included in the Platform Update download package.

4 On the new computer, install the Windows agent and AppManager for Microsoft Windows module included with Hotfix 72616 or later.

To obtain the hotfix, visit the AppManager Suite Hotfixes Web site.

For more information about installing the agent and module, see the documentation included in the hotfix download package.

5 For each agent that communicates with the management server, complete the following steps:

5a Run the AMAdmin_SetAllowMS Knowledge Script and update the **New hostname(s) for AllowMS** parameter with the name of the new management server computer.

5b Run the AMAdmin_SetPrimaryMS Knowledge Script to update the management server name.

Depending on whether the management server is primary or secondary for the agent, update the primary or secondary management server name.

For more information about the Knowledge Scripts, see the *AppManager Knowledge Script Reference Guide*.

6 On the old computer, use the Add or Remove Programs application in Control Panel to uninstall the management server.

After you move the management server to the new computer, continue to Step 3 on page 27 of Section 2.3, “Preparing to Migrate the Repositories,” on page 26.

2.3.2 Creating Backup Copies of the Repositories

This section describes how to use Microsoft SQL Server Management Studio to create backup copies of the QDB and CCDB before you migrate them to the new computer.

To create a backup copy of the QDB or CCDB:

1 In Microsoft SQL Server Management Studio, expand `SQL_Server_Name\Databases`.

2 Right-click the repository and select **Tasks** > **Back Up**.

3 On the **General** page, complete the following steps:

  3a From the **Database** list, select the repository.

  3b Note the setting in the **Recovery model** field.

  3c From the **Backup type** list, select **Full**.

  3d For **Backup component**, select the **Database** radio button.

4 On the **Options** page, complete the following steps:

  4a Select the **Back up to the existing media set** radio button.

  4b (Conditional) If you want to add this backup to existing backups, select the **Append to the existing backup set** radio button.

  4c (Conditional) If you want to discard existing backups, select the **Overwrite all existing backup sets** radio button.

5 Click **OK** to start the backup.

6 (Conditional) If the **Recovery model** is set to **FULL** on the **General** page, after the backup completes, complete the following steps to add a backup device for the transaction log and to back up the transaction log:

  6a Right-click the repository and select **New Query**.

  6b In the query window, type the following command, and then click **Execute**:

```
USE master
EXEC sp_addumpdevice 'disk', 'dump device log',
'C: \Repository_NameBACKUP\Repository_Name_Log.bak'
GO
BACKUP LOG Repository_Name TO Dump_Device_Log
GO
sp_dropdevice 'Dump_Device_Log'
GO
```

  where *Repository_Name* is the name of the QDB or CCDB you are backing up and *Dump_Device_Log* is the name of the backup device or file

7 Copy the backup file to the computer where you will restore the QDB or CCDB.

After you copy the backup file to the new computer, you can restore the backup copy over the new, empty repository on the new computer. For more information about restoring the repository, see Section 2.4, “Restoring Repositories on the New Computer,” on page 33.

2.4 Restoring Repositories on the New Computer

After creating a backup copy and installing a new AppManager 7.*x* QDB or CCDB on the new computer, use Microsoft SQL Server Management Studio to restore the backup copy over the new, empty repository on the new computer.

If you are migrating both the QDB and the CCDB, restore the QDB first.

To restore repositories on the new computer:

1 On the new computer, stop the SQL Server Agent service.

2 To ensure that the repository you are restoring is not the default database for the account you are using to perform the restore, complete the following steps:

2a In Microsoft SQL Server Management Studio, expand `SQL_Server_Name\Security\Logins`.

2b Right-click the account you are using and select **Properties**.

2c On the **General** page, note the selection in the **Default database** list.

3 (Conditional) If the default database for the account you are using is the repository you are restoring, either change the default database for the account, or log in to SQL Server Management Studio with a different account.

4 To ensure that no users are connected to the new repository, complete the following steps:

4a Expand `SQL_Server_Name\Databases`.

4b Right-click the repository and select **New Query.**

4c In the query window, type the following command, and then click **Execute**:

```
USE master
GO
Exec sp_who2
GO
```

4d In the results table, check the **DBName** column for the repository name. The column should not contain entries for the repository.

5 (Conditional) If the **DBName** column contains entries for the repository, to close the open connections, complete the following steps for each connection:

5a In the **SPID** column for the row in which the repository name appears, note the SPID number.

5b Right-click the repository and select **New Query**.

5c In the query window, type the following command, and then click **Execute**:

```
kill SPID_Number
```

6 Repeat Step 4 on page 33 and Step 5 on page 33 until the **DBName** column does not contain entries for the repository.

7 Right-click the repository and select **Tasks** > **Restore** > **Database**.

8 On the **General** page, complete the following steps:

8a Select the **From device** radio button and click the button to specify the backup device.

8b On the Specify Backup window, select **File** from the **Backup media** list, and then click **Add**.

8c On the Locate Backup File window, browse to the location where you saved the backup copy, select the backup file, and then click **OK**.

8d Click **OK** to return to the **General** page.

8e Select the backup set to restore.

9 On the **Options** page, complete the following steps:

9a Under **Restore options**, select the **Overwrite the existing database (WITH REPLACE)** check box.

9b Under **Recovery state**, select the **RESTORE WITH RECOVERY** radio button.

9c Click **OK** to restore the repository.

10 After the restore completes, restart the SQL Server Agent service.

After you restore the repository, additional configuration is required. For more information about the configuration tasks, see Section 2.5, “Configuring Restored Repositories,” on page 34.

2.5 Configuring Restored Repositories

After you restore the repository on the new computer, additional configuration is required to ensure proper operation.

To configure the restored repository:

1 To verify that the compatibility level of the restored repository is set to the appropriate version of SQL Server, in Microsoft SQL Server Management Studio, complete the following steps:

1a Expand *SQL_Server_Name*\Databases.

1b Right-click the restored repository and select **Properties**.

1c On the **Options** page, ensure that the compatibility level is set to the appropriate version of SQL Server.

For example, if the restored repository is hosted on SQL Server 2008, ensure that the compatibility level is set to SQL Server 2008 (100).

2 To identify the SQL Server user accounts you must recreate for the repository, complete the following steps:

2a Expand *SQL_Server_Name*\Databases.

2b Right-click the repository and select **New Query**.

2c In the query window, type the following command, and then click **Execute**:

```
SELECT  name,
        CASE WHEN type = 'S' THEN 'SQL'
             ELSE 'Windows'
        END AS 'Type'
FROM    sys.database_principals
WHERE   type IN ( 'S', 'U' )
        AND name != 'dbo'
        AND default_schema_name IS NOT NULL
        AND default_schema_name != 'guest'
ORDER BY name ASC
```

2d Expand *Repository_Name*\Security\Users and compare the accounts listed in the results table for the query to the accounts in the Users folder. Note the accounts that appear in both locations.

3 To recreate the SQL Server logins for the repository, complete the following steps for each account you noted in Step 2 on page 34:

3a Right-click the repository and select **New Query**.

3b In the query window, type the following command, and then click **Execute**:

```
sp_dropuser 'User_Name'
```

3c Expand `SQL_Server_Name\Security\Logins`.

3d With the exception of the `probe` account, for each account you removed in Step 3b on page 35, right-click **Logins** and select **New Login**.

3e Configure the login with the properties you noted in Step 4 on page 27 of Section 2.3, "Preparing to Migrate the Repositories," on page 26.

3f (Conditional) If a repository account you removed already exists as a SQL Server account, use AppManager Security Manager (for the QDB) or the Control Center console (for the CCDB) to assign the accounts to the repository.

4 To verify that the repository owner is `netiq`, complete the following steps:

4a Right-click the restored repository and select **New Query**.

4b In the query window, type the following command, and then click **Execute**:

```
sp_helpdb 'Repository_Name'
```

5 (Conditional) If the repository owner is not `netiq`, to change the repository owner, complete the following steps:

5a Right-click the repository and select **New Query**.

5b In the query window, type the following command, and then click **Execute**:

```
sp_changedbowner 'netiq'
```

6 (Conditional) If you are migrating the QDB, complete the following steps to reassign SQL Server jobs specified to run using the `netiq` account:

6a Log in to SQL Server Management Studio with the `netiq` account.

6b Expand `SQL_Server_Name\Databases`.

6c Right-click the QDB and select **New Query**.

6d In the query window, type the following command, and then click **Execute**:

```
EXEC task_util
go
EXEC task_utilExt
go
```

6e (Conditional) If you previously changed the default schedule for any jobs, reset the job schedules with the properties you noted in Step 5 on page 28 of Section 2.3, "Preparing to Migrate the Repositories," on page 26.

7 (Conditional) If you are migrating the CCDB, complete the following steps to reassign SQL Server jobs specified to run using the `netiq` account:

7a Log in to SQL Server Management Studio with the `netiq` account.

7b Expand `SQL_Server_Name\SQL Server Agent\Jobs`.

7c For each NetIQ CC job, right-click the job and select **Properties**.

7d On the **General** page, note the owner.

7e (Conditional) If the owner is not `netiq`, change the owner to `netiq`, and then click **OK**.

8 (Conditional) If you are migrating the QDB, complete the following steps to update the QDB computer name in the `Version` table:

8a Right-click the QDB and select **New Query**.

8b In the query window, type the following command, and then click **Execute**:

```
update dbo.Version
set MachineName = 'New_Computer_Name'
where Component = 'Repository'
select *
from Version
where Component = 'Repository'
```

9 (Conditional) If you are migrating the QDB and previously created charts for use in the Chart Console, complete the following steps to enable the charts for use with the restored QDB:

9a Right-click the QDB and select **New Query**.

9b In the query window, type the following command, and then click **Execute**:

```
update dbo.blob set comment = replace(comment,
'_Old_SQL_Server_Name\instance\', '_New_SQL_Server_Name\instance\')
from dbo.blob
where charindex('_Old_SQL_Server_Name\instance\', comment) > 0
```

10 (Conditional) If you are migrating the QDB and Control Center manages it, complete the following steps to update the `DataSource` and `CC_Parameter` tables:

10a Right-click the QDB and select **New Query**.

10b In the query window, type the following command, and then click **Execute**:

```
update dbo.DataSource
set DataSourceName = 'New_SQL_Server_Name\instance:Repository_Name',
ServerName = 'newSQLServername\instance',
DatabaseName = 'Repository_Name'
```

10c Right-click the QDB and select **New Query**.

10d In the query window, type the following command, and then click **Execute**:

```
update dbo.CC_Parameter
set ValueStr = 'New_SQL_Server_Name\instance.Repository_Name'
where ValueStr = 'Old_SQL_Server_Name\instance.Repository_Name'
```

11 (Conditional) If you are migrating the CCDB and it manages remote QDBs, complete the following steps to restore SQL Server links to the remote QDBs:

11a Expand `SQL_Server_Name\Server Objects`.

11b Right-click the `Linked Servers` folder and select **New Linked Server**.

11c On the **General** page, in the **Linked server** field, specify the name and instance, if applicable, of the SQL Server that hosts the QDB for which you are restoring the link.

11d On the **General** page, for **Server type**, select the **SQL Server** radio button.

11e On the **Security** page, to add local logins defined before you migrated the CCDB, click **Add**.

11f On the **Server Options** page, set the **RPC** and **RPC Out** values to **True**.

11g Click **OK** to restore the SQL Server link.

12 (Conditional) If you are migrating the CCDB to a computer in a different domain or security environment (for example, from a production environment to a lab environment), complete the following steps to add your Windows account to the Administrator group in the restored CCDB:

12a Expand `SQL_Server_Name\Databases\Repository_Name`.

12b Right-click the CCDB and select **New Query**.

12c Copy and paste the SQL script provided below into the query window.

```
DECLARE @user NVARCHAR(256),
    @sid VARBINARY(85),
    @type INT,
    @group UNIQUEIDENTIFIER,
    @accountid INT,
    @SQLError NVARCHAR(2000)
SET nocount ON
SELECT @user = 'Domain\User_Name'
SELECT @type = 0
SELECT @sid = SUSER_SID(@user)
IF @sid IS NULL
    PRINT 'The user ' + @user
        + ' must first be added to SQL using the SQL tools'
ELSE
    BEGIN
        SELECT @group = GroupID
        FROM    dbo.[Group]
        WHERE   GroupName = 'Administrator'
        IF @group IS NULL
            PRINT 'Administrator Group was not found'
        ELSE
            BEGIN
                SELECT @accountid = AccountID
                FROM    dbo.Account
                WHERE   UPPER(AccountName) = UPPER(@user)
                        AND AccountSID = @sid
                IF @accountid IS NULL
                    BEGIN
                        BEGIN TRY
                            INSERT dbo.Account
                            VALUES (
                                      @user,
                                      'sysadmin',
                                      0,
                                      @type,
                                      GETUTCDATE(),
                                      @sid
                                    )
                            SELECT @accountid = @@IDENTITY
                        END TRY
                        BEGIN CATCH
                            SELECT @SQLError = ERROR_MESSAGE()
                          PRINT 'Failed to add user ' + @user + '. Error: '
                                + @SQLError
                        END CATCH
                    END
                IF @accountid IS NOT NULL
                    BEGIN
                        IF NOT EXISTS ( SELECT  1
                                        FROM    dbo.AccountInGroup
                                        WHERE   AccountID = @accountid
                                                AND GroupID = @group )
                            BEGIN
                                BEGIN TRY
                                    INSERT dbo.AccountInGroup
                                    VALUES (
                                              @group,
                                              @accountid,
                                              0,
                                              GETUTCDATE()
                                            )
                                    PRINT 'User ' + @user
                                        + ' added to Administrators group'
                                END TRY
                                BEGIN CATCH
                                    SELECT @SQLError = ERROR_MESSAGE()
                                    PRINT 'Failed to add user ' + @user
                                       + ' to Administrator group. Error: '
                                        + @SQLError
                                END CATCH
                            END
                        ELSE
                            PRINT 'User ' + @user
                                + ' is already in the Administrator group'
                    END
            END
    END
```

12d Edit the following line with the domain and user name of your Windows account:

```
SELECT @user = 'Domain\User_Name'
```

12e To run the script, click **Execute**.

When the script completes, the following message is displayed in the Messages pane:

```
User Domain\User_Name added to Administrators group
```

After you configure the restored repository, update components and services that connect to it. For more information about the components and services to update, see Section 2.6, “Updating Connected Components and Services,” on page 38.

2.6 Updating Connected Components and Services

After you configure the restored repository on the new computer, update components and services that connect to it.

To update components and services:

1 (Conditional) If you migrated the QDB, complete the following steps to update the primary management server and each secondary management server that connects to it:

1a (Conditional) If you customized any management server port or persistent IOC settings, use the Windows Registry Editor to back up the following registry keys on the management server computer:

On this type of operating system...	Back up...
32-bit	◆ HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQms\NetIQmc Port ◆ HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQms\Port ◆ HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQms\Unix Port ◆ HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQms\Config\Persistent IOC ◆ HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQms\Config\PIOC Map File Path

On this type of operating system...	Back up...
64-bit	◆ `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\NetIQ\AppManager\4.0\NetIQms\NetIQmc Port` ◆ `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\NetIQ\AppManager\4.0\NetIQms\Port` ◆ `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\NetIQ\AppManager\4.0\NetIQms\Unix Port` ◆ `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\NetIQ\AppManager\4.0\NetIQms\Config\Persistent IOC` ◆ `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\NetIQ\AppManager\4.0\NetIQms\Config\PIOC Map File Path`

1b From the `AppManager\bin` folder in the location where you installed the management server, type the following command to re-register the management server service:

```
netiqms -r
QDBms:QDB_Name:Windows_or_SQL_Account_User_Name:password:SQL_Server_Name\
instance -ur -i
```

1c Restore persistent IOC settings.

For more information about restoring persistent IOC settings, see Section 2.6.1, “Restoring Persistent IOC Settings,” on page 41.

1d (Conditional) If you have UNIX agents, restore the port setting that defines where the management server listens for communications from UNIX agents.

For more information about restoring the port setting, see Section 2.6.2, “Restoring the UNIX Port Setting,” on page 42.

1e Restore any registry keys you backed up.

1f Start the NetIQ AppManager Management Server service (`NetIQms`).

2 (Conditional) If you are migrating the CCDB, on each QDB Control Center manages, complete the following steps to enable the QDB to connect to the restored CCDB:

2a In Microsoft SQL Server Management Studio, expand *`SQL_Server_Name`*`\Databases`.

2b Right-click the QDB and select **New Query**.

2c In the query window, type the following command, and then click **Execute**:

```
UPDATE dbo.CC_CacheManager SET Name = 'New_CCDB_SQL_Server_Name\instance'
WHERE Name = 'Old_CCDB_SQL_Server_Name\instance'
```

3 (Conditional) If you are migrating the QDB and Control Center manages it, complete the following steps to update the QDB connection information in the CCDB:

3a (Conditional) If Control Center uses SQL authentication to communicate with the QDB, configure the QDB with the same SQL Server user account and permissions.

3b Log on to the Control Center console with an account that is a member of the Administrator group and has the `db_owner` database role for the QDB.

3c On the **File** menu, select **Manage Repositories.**

3d Select the QDB, and then click **Modify**.

3e Provide the following information, and then click **OK**:

- Name of the SQL Server and instance, if applicable, that hosts the QDB
- Name of the QDB

4 (Conditional) If you are migrating the CCDB, complete the following steps to update the command queue service:

4a Use the Control Center console to add the Windows user account for the command queue service as a Control Center administrator.

4b In Microsoft SQL Server Management Studio, right-click the CCDB and select **New Query**.

4c In the query window, to clear the previous command queue service settings from the CCDB, type the following commands, and then click **Execute**:

```
delete from Property where Scope = 'cqs'
exec InitialSQLJobs
exec SMVInitialSQLJob
```

4d On the command queue service computer, from the `AppManager\Control Center\bin` folder, open the `NQCQS.exe.config` file in a text editor.

4e Under `<appSettings>`, change the value of the `ServerName` parameter to specify the SQL Server and instance that hosts the restored CCDB. For example:

```
<appSettings>
  <add key="ServerName" value="MYSQLSERVER\INSTANCE1" />
  <add key="DBName" value="NQCCDB" />
```

4f Restart the command queue service to apply the changes.

4g (Conditional) If the command queue service does not start, in SQL Server Management Studio, right-click the CCDB and select **New Query**.

4h In the query window, type the following command, and then click **Execute**:

```
select * from Property where scope = 'cqs'
```

4i (Conditional) If the results table contains entries for the command queue service, right-click the CCDB and select **New Query**.

4j In the query window, type the following command, and then click **Execute**:

```
delete from Property where Scope = 'cqs'
```

4k Repeat Step 4g on page 40 through Step 4j on page 40 until the results table does not contain entries for the command queue service, and then restart the command queue service.

5 (Conditional) If you are migrating the CCDB, complete the following steps to update the Deployment Service:

5a (Conditional) If the Deployment Service will use different credentials or a different account to log on to the migrated CCDB, from the `AppManager\Control Center\bin` folder on the Deployment Service computer, issue the following command to change the account:

```
deploymentservice -setwindowsauth domain\username password
```

5b Use the Control Center console to add the Windows user account for the Deployment Service as a Control Center administrator.

5c On the Deployment Service computer, from the `AppManager\Control Center\bin` folder, open the `DeploymentService.exe.config` file in a text editor.

5d Under `<appSettings>`, change the value of the `ServerName` parameter to specify the SQL Server and instance that hosts the restored CCDB. For example:

```
<appSettings>
  <add key="ServerName" value="MYSQLSERVER\INSTANCE1" />
  <add key="DBName" value="NQCCDB" />
```

5e Restart the Deployment Service to apply the change.

6 (Conditional) If you are migrating the CCDB, complete the following steps to update the Deployment Web Service:

6a Use the Control Center console to add the Windows user account for the Deployment Web Service as a Control Center administrator.

6b On the Deployment Web Service computer, from the `AppManager\Control Center` folder, open the `Web.config` file in a text editor.

6c Under `<appSettings>`, change the value of the `ServerName` parameter to specify the SQL Server and instance that hosts the restored CCDB. For example:

```
<appSettings>
    <add key="ServerName" value="MYSQLSERVER\INSTANCE1">
```

6d Restart the World Wide Web Publishing Service to apply the change.

After you update the connected components and services, verify successful migration. For more information about verifying successful migration, see Section 2.7, “Verifying Successful Migration,” on page 42.

2.6.1 Restoring Persistent IOC Settings

Re-registering the management server service disables persistent IOC settings in the registry. This section describes how to restore the settings.

WARNING: Be careful when editing your Windows registry. If there is an error in your registry, your computer might become nonfunctional. If an error occurs, you can restore the registry to its state when you last successfully started your computer. For more information, see the Help for the Windows Registry Editor.

To restore persistent IOC settings:

1. Click **Start** > **Run**.
2. In the **Open** field, type `regedit`, and then click **OK**.
3. (Conditional) If the management server is installed on a 32-bit operating system, in the left pane of the Registry Editor, navigate to `HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQms\Config`.
4. (Conditional) If the management server is installed on a 64-bit operating system, in the left pane of the Registry Editor, navigate to `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\NetIQ\AppManager\4.0\NetIQms\Config`.
5. In the right pane, double-click **Persistent IOC**.
6. In the **Value data** field, set the value to **1**, and then click **OK**.
7. In the right pane of the Registry Editor, double-click **PIOC Map File Path**.

8 In the **Value data** field, set the value to the location of your persistent IOC files, and then click **OK**.

   Typically, the location is `Program Files\NetIQ\AppManager\dat\pioc`.

After you restore the persistent IOC settings, return to Step 1 on page 38 of Section 2.6, "Updating Connected Components and Services," on page 38.

2.6.2 Restoring the UNIX Port Setting

Re-registering the management server service resets the port setting that defines where the management server listens for communications from UNIX agents. This section describes how to restore the setting.

WARNING: Be careful when editing your Windows registry. If there is an error in your registry, your computer may become nonfunctional. If an error occurs, you can restore the registry to its state when you last successfully started your computer. For more information, see the Help for the Windows Registry Editor.

To restore the UNIX port setting:

1 Click **Start** > **Run**.

2 In the **Open** field, type `regedit`, and then click **OK**.

3 (Conditional) If the management server is installed on a 32-bit operating system, in the left pane of the Registry Editor, navigate to `HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQms`.

4 (Conditional) If the management server is installed on a 64-bit operating system, in the left pane of the Registry Editor, navigate to `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\NetIQ\AppManager\4.0\NetIQms`.

5 In the right pane, double-click **Unix Port**.

6 For **Base**, select **Decimal**.

7 In the **Value Data** field, set the value to the port number you specified when you installed AppManager, and then click **OK**.

   The default port where the management server listens for communications from UNIX agents is 9001.

After you restore the UNIX port setting, return to Step 1 on page 38 of Section 2.6, "Updating Connected Components and Services," on page 38.

2.7 Verifying Successful Migration

After you update components and services that connect to the migrated repository, verify successful migration.

To verify successful repository migration:

1 (Conditional) If you migrated a QDB, on the primary management server and each secondary management server that connects to the QDB, ensure that the following services are running:

   - NetIQ AppManager Management Service (`NetIQms`)
   - NetIQ AppManager Client Resource Monitor (`NetIQmc`)
   - NetIQ AppManager Client Communication Manager (`NetIQccm`)

2 (Conditional) If you migrated a CCDB, on the command queue service and Deployment Service computers, ensure that the following services are running:

- NetIQ AppManager Control Center Command Queue Service
- NetIQ AppManager Deployment Service

3 Log on to the Operator Console and verify that the primary management server appears in the tree view and is not disabled.

4 (Conditional) If the primary management server does not appear in the tree view, re-register the management server.

For more information about re-registering the management server, see Step 1 on page 38 of Section 2.6, “Updating Connected Components and Services,” on page 38.

5 Use the Operator Console and the Control Center console to start some jobs.

For example, start an NT_Discovery job. When that job completes, start an NT_CpuLoaded job and wait for it to complete.

6 (Conditional) If the jobs do not complete successfully, contact Technical Support.

After you verify successful migration, you can upgrade the migrated repository to version 8.*x*. For more information about upgrading the QDB, see Chapter 3, “Upgrading Management Site Components,” on page 45. For more information about upgrading the CCDB, see Chapter 5, “Upgrading Control Center Components,” on page 61. Ensure that you upgrade components in the correct order. For information about the order for upgrading components, see Section 1.6, “Understanding the Recommended Order for Upgrading Components,” on page 22.

3 Upgrading Management Site Components

This chapter describes how to upgrade a management site. A management site comprises one QDB and one or more management servers.

3.1 Understanding Upgrade Prerequisites

This section describes prerequisites your environment must meet before you upgrade management site components.

3.1.1 Understanding Component Versions Supported for Upgrade

Before you upgrade management site components, ensure that the components meet version prerequisites. If a component does not meet the prerequisite, you can either uninstall it and install a new version 8.*x* component, or you can first upgrade it to a version supported for upgrade. The following table lists the management site components and the AppManager versions supported for upgrade.

Component	Supported AppManager Versions for Upgrade
QDB	◆ AppManager 7 Platform Update ◆ 8.0.1 ◆ 8.0.2 ◆ 8.0.3
Management server	◆ AppManager 7 Platform Update ◆ 8.0.1 ◆ 8.0.2 ◆ 8.0.3

For more information about installing a new version 8.*x* QDB or management server, see the *Installation Guide for AppManager*.

When you upgrade a management server, you also upgrade the agent on the management server computer. For more information about upgrading agents, see Chapter 4, “Upgrading and Migrating Agent Components,” on page 55.

For more information about obtaining a version of AppManager that is supported for upgrade, contact Technical Support.

3.1.2 Understanding Microsoft SQL Server Versions Supported for Upgrade

Version 8.*x* QDBs must be hosted on one of the following versions of Microsoft SQL Server:

- Microsoft SQL Server 2005 Standard or Enterprise edition Service Pack 2 or later (32-bit or 64-bit)
- Microsoft SQL Server 2008 Standard or Enterprise edition Service Pack 1 or later (32-bit or 64-bit)
- Microsoft SQL Server 2008 R2 Standard or Enterprise edition (32-bit or 64-bit)
- Microsoft SQL Server 2012 Standard or Enterprise edition (32-bit or 64-bit)

Before you can upgrade a QDB hosted on Microsoft SQL Server 2000 to AppManager 8.*x*, you must migrate the QDB to a supported version of Microsoft SQL Server. Once the QDB is hosted on a supported version of SQL Server, you can upgrade it to version 8.*x*. For more information about migrating a QDB before upgrading it to version 8.*x*, see Chapter 2, "Migrating Version 7.x Repositories," on page 25.

AppManager 8.*x* supports migrating version 8.*x* QDBs hosted on Microsoft SQL Server 2005 to Microsoft SQL Server 2008, 2008 R2, or 2012 and 8.*x* QDBs hosted on SQL Server 2008 or 2008 R2 to SQL Server 2012. For more information about migrating an upgraded QDB, see Appendix A, "Migrating Version 8.x Repositories," on page 77.

3.1.3 Understanding SQL Server Compatibility Requirements

If a QDB was previously hosted on Microsoft SQL Server 2000 and the SQL Server compatibility level remains SQL Server 2000, the QDB might not function properly in Control Center. If the setup program detects the SQL Server compatibility level is set to SQL Server 2000, it sets the compatibility level to the appropriate version. The setup program only modifies the QDB. It will not modify any other databases.

3.2 Understanding Control Center Support for QDBs

A version 8.*x* CCDB only supports a version 8.*x* primary QDB. You can either create a new version 8.*x* primary QDB or upgrade your existing primary QDB to version 8.*x*. If you create a new primary QDB, in order for the product to function correctly, you must add at least one agent computer to the QDB and discover the computer. For more information about creating a new version 8.*x* primary QDB, see the *Installation Guide for AppManager*.

You can attach older QDBs to a version 8.*x* CCDB as non-primary QDBs. Older non-primary QDBs attached to a version 8.*x* CCDB must meet the hotfix and SQL Server version requirements described in the following table. The hotfixes are available on the AppManager Suite Hotfixes Web site.

Non-primary QDB Version	Microsoft SQL Server Version
7.0.1, with Hotfix 7011821 or later	2005 Standard or Enterprise edition Service Pack 2 or later (32-bit)
7.0.3, with Hotfix 7011821 or later	2005 Standard or Enterprise edition Service Pack 2 or later (64-bit)

Non-primary QDB Version	Microsoft SQL Server Version
AppManager 7 Platform Update, with Hotfix 7011821 or later	One of the following (32-bit or 64-bit): ◆ 2008 Standard or Enterprise edition Service Pack 1 or later ◆ 2008 R2 Standard or Enterprise edition
8.0.2, with Hotfix 7011822 or later	One of the following (32-bit or 64-bit): ◆ 2005 Standard or Enterprise edition Service Pack 2 or later ◆ 2008 Standard or Enterprise edition Service Pack 1 or later ◆ 2008 R2 Standard or Enterprise edition
8.0.3, with Hotfix 7011823 or later	

When you upgrade the CCDB to version 8.*x*, if the setup program detects a QDB hosted on Microsoft SQL Server 2000, the CCDB upgrade cannot continue.

3.3 Understanding the Order for Upgrading Management Site Components

If you have management site components installed on multiple computers, ensure that you upgrade components in the correct order. On the computer where you run the setup program, the program ensures you upgrade components in the correct order. However, when you have components installed on multiple computers, ensure that you upgrade the computers in order. For example, if the QDB and the management server are on different computers, upgrade the QDB before you upgrade the management server. For more information about the correct order for upgrading AppManager components, see Section 1.6, “Understanding the Recommended Order for Upgrading Components,” on page 22.

3.4 Discovering Upgraded 7.*x* Management Site Components for Health Monitoring

Once the setup program successfully upgrades a version 7.*x* management server, if an upgraded agent is already present, the setup program automatically runs the Discovery_AMHealth Knowledge Script to prepare the management site components for health monitoring in Control Center. Otherwise, the setup program runs the Knowledge Script after the agent upgrade. For more information about using Control Center to monitor the health of your AppManager components, see the *Control Center User Guide for AppManager*.

3.5 Backing Up Your Knowledge Scripts

If your organization modifies Knowledge Scripts or creates custom Knowledge Scripts, check your modified and custom Knowledge Scripts out of the QDB and copy them to a temporary location before you upgrade.

The setup program compares the Knowledge Scripts in the QDB to the Knowledge Scripts in your installation folder. If you made changes to a Knowledge Script in the installation folder without changing the Knowledge Script name, the setup program cannot identify the changes because the versions it will compare are the same.

Performing a formal checkout or checkout/check-in operation saves your modifications to the QDB and to the local disk before you upgrade. This is a precautionary step to ensure that you can restore customized Knowledge Scripts if you experience problems during the upgrade or Knowledge Scripts were changed through the Operator Console but the changes were not saved to the local disk copy.

If you modified a Knowledge Script and saved your changes in the local copy of the Knowledge Script file, you do not need to perform the checkout operation. For example, if you modified the General_ASCIILog Knowledge Script and saved the changes directly in the `General_AsciiLog.qml` file instead of modifying the Ascii Log Properties within the Operator Console, you do not need to perform the checkout operation. Your customized copy of the Knowledge Script is saved in the `\netiq\qdb_old\kp_old` folder after the upgrade.

3.6 Upgrading the QDB

This section describes the steps required to upgrade a QDB. For information about migrating a version 7.*x* QDB to a new computer before upgrading to version 8.*x*, see Chapter 2, “Migrating Version 7.x Repositories,” on page 25.

You can upgrade QDBs on remote SQL Servers. You no longer have to run the setup program on the SQL Server.

To upgrade the QDB:

1. Close connections to the QDB.

   For more information about the connections to close, see Section 2.3, “Preparing to Migrate the Repositories,” on page 26.

2. Create a QDB backup.

   For more information about creating a QDB backup, see Section 2.3.2, “Creating Backup Copies of the Repositories,” on page 32. Since you are not migrating the QDB to a different computer, it is not necessary to copy the backup files to a different location.

3. Start the upgrade and generate a pre-installation check report.

   For more information about generating the report, see Section 1.9, “Starting an Upgrade and Generating a Pre-Installation Check Report,” on page 23.

4. When you reach the Target SQL Server and Repository Name window, provide the following information and click **Next**:

   - Name of the SQL Server and, if applicable, instance that hosts the QDB you are upgrading. To specify a SQL Server instance, use the format *Server_Name\instance*.

- Name of the QDB you are upgrading.
- Account that can log in to the SQL Server for the upgrade. Ensure that the account is a member of the `sysadmin` SQL Server role.

The setup program prepares the existing QDB data for upgrade and checks in current Knowledge Scripts.

3.7 Upgrading Management Servers

This section describes the steps required to upgrade the management server on the same computer. For information about moving the management server to a new computer, see Section 3.7.1, “Moving the Management Server to a New Computer,” on page 50.

If you have only one QDB in your environment, you must upgrade the QDB and the primary and secondary management servers that connect to the QDB. When you upgrade a management server, you also upgrade the agent on the management server computer. Otherwise, you can upgrade agents on an as-needed basis.

While a version 8.*x* management server can communicate with version 7.0.*x* and 8.0.*x* agents, version 8.*x* agents cannot communicate with earlier management server versions.

For more information about supported management server versions in a multiple-QDB, Control Center environment, see Section 1.3.2, “Upgrading Components in a Multiple-QDB, Control Center Environment,” on page 17.

If you choose to upgrade by running the Windows Installer package for the management server instead of the AppManager setup program, after the upgrade, the NetIQ AppManager Client Resource Monitor (`NetIQmc`) and NetIQ AppManager Client Communication Manager (`NetIQccm`) services will not start until you upgrade the agent.

NetIQ Corporation does not recommend clustering the management server. Instead, you can install multiple management servers and designate them as primary and secondary to provide failover support. For more information about installing additional management servers and designating primary and secondary management servers, see the *Installation Guide for AppManager* and the *Administrator Guide for AppManager*. If you have a requirement to install a management server on Microsoft Cluster Service (MSCS), contact Technical Support.

To upgrade the management server on the same computer:

1. Ensure that the upgrade of the QDB to which the management server connects completed successfully.
2. Start the upgrade and generate a pre-installation check report.

   For more information about generating the report, see Section 1.9, “Starting an Upgrade and Generating a Pre-Installation Check Report,” on page 23.
3. Complete the management server setup program.

If you previously used Performance Monitor to monitor the operational health and performance of the management server and the agent on the management server computer, you must manually update the performance counters after the upgrade. For more information about manually updating the counters, see Section 3.7.2, “Updating Management Server Performance Counters,” on page 51 and Section 4.6.1, “Updating Agent Performance Counters,” on page 57.

Depending on your environment, you might need to update existing encryption keys after you upgrade the management server. For more information about updating encryption keys, see Section 3.7.3, “Using Existing Security Keys for Encrypted Communications,” on page 52.

You can also change the security level after you upgrade. For more information about changing the security level, see Section 3.7.4, “Changing the Security Level,” on page 52.

3.7.1 Moving the Management Server to a New Computer

This section describes how to move the management server to a new computer. You might want to move the management server if the current computer does not support upgrading to an operating system supported with AppManager 8.*x*.

Moving the management server to a new computer requires ensuring that it will be able to communicate with the QDB and agents after the move, installing a new version 8.*x* management server on the new computer, and uninstalling the management server from the old computer.

If you are also migrating the QDB with which the management server communicates to a new computer, complete the QDB migration before you move the management server. For more information about migrating a version 7.*x* QDB to a new computer, see Chapter 2, “Migrating Version 7.x Repositories,” on page 25.

To move the management server:

1 Upgrade the QDB with which the management server communicates to version 8.*x*.

For more information about upgrading the QDB, see Section 3.6, “Upgrading the QDB,” on page 48.

2 To allow agent computers that communicate with the management server to temporarily allow communication with anonymous management servers, run the AMAdmin_SetAllowMS Knowledge Script and set the **New hostname(s) for AllowMS** parameter to an asterisk (*).

For more information about the Knowledge Script, see the *AppManager Knowledge Script Reference Guide*.

3 (Conditional) If the following registry keys on the agent computers are not set to an asterisk (*), use the NTAdmin_RegistrySet Knowledge Script to add the name of the new management server computer to the key values:

```
HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQmc\Security\AllowDosCmd
HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQmc\Security\AllowMS
HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQmc\Security\AllowReboot
```

Otherwise, certain actions will not be allowed after you move the management server. For example, the `AllowReboot` registry key will no longer allow Action_RebootSystem. For information about the parameters to specify in the Knowledge Script, see the *AppManager Knowledge Script Reference Guide*.

4 Install a new version 8.*x* management server on the new computer.

For more information about installing a new version 8.*x* management server, see the *Installation Guide for AppManager*.

5 For each agent that communicates with the management server, run the AMAdmin_SetAllowMS Knowledge Script and update the **New hostname(s) for AllowMS** parameter with the name of the new management server computer.

6 For each agent that communicates with the management server, run the AMAdmin_SetPrimaryMS Knowledge Script to update the management server name.

Depending on whether the management server is primary or secondary for the agent, update the primary or secondary management server name.

For more information about the Knowledge Script, see the *AppManager Knowledge Script Reference Guide*.

7 Uninstall the management server from the old computer.

8 In the Operator Console tree view, select the old computer and press **Alt+F8**. Note the **ObjID** of the old management server computer.

9 In Microsoft SQL Server Management Studio, right-click the QDB with which the management server communicates and select **New Query**.

10 To change the status of the old management server computer to an agent computer so that you can remove it from the Operator Console, in the query window, type the following SQL statement and click **Execute**:

```
UPDATE dbo.Object
SET     Status = Status ^ 0x00000002
WHERE   ObjID = ObjID_of_Old_Management_Server
        AND Status & 0x00000002 != 0
```

where *ObjID_of_Old_Management_Server* is the **ObjID** you noted in Step 8 on page 51.

11 In the Operator Console, delete the old management server computer.

12 (Conditional) If the agent is still installed on the old computer, use Control Center to add the computer and rediscover it to establish a new **ObjID** for the computer.

13 (Conditional) If you edited registry keys in Step 3 on page 50, use the NTAdmin_RegistrySet Knowledge Script to remove the name of the old management server computer from the key values.

For information about the parameters to specify in the Knowledge Script, see the *AppManager Knowledge Script Reference Guide*.

If you previously used Performance Monitor to monitor the operational health and performance of the management server and the agent on the management server computer, you must manually update the performance counters after the upgrade. For more information about manually updating the counters, see Section 3.7.2, “Updating Management Server Performance Counters,” on page 51 and Section 4.6.1, “Updating Agent Performance Counters,” on page 57.

Depending on your environment, you might need to update existing encryption keys after you upgrade the management server. For more information about updating encryption keys, see Section 3.7.3, “Using Existing Security Keys for Encrypted Communications,” on page 52.

You can also change the security level after you upgrade. For more information about changing the security level, see Section 3.7.4, “Changing the Security Level,” on page 52.

3.7.2 Updating Management Server Performance Counters

The setup program does not update AppManager 7.*x* performance counters. To continue using the counters, you must manually update them.

To update the management server performance counters:

1 Open a Command Prompt and change directory to the `Windows\System32` (for 32-bit operating systems) or `Windows\SysWOW64` (for 64-bit operating systems) folder.

2 Type the following command and press **Enter**:

```
lodctr.exe "Installation_Drive_and_Folder\AppManager\bin\mscnt.ini"
```

For example, `lodctr.exe "C:\Program Files (x86)\NetIQ\AppManager\bin\mscnt.ini"`

On 64-bit operating systems, to view the counters, you must open Performance Monitor from the `Windows\SysWOW64` folder.

3.7.3 Using Existing Security Keys for Encrypted Communications

If your AppManager environment uses encrypted communications between the management server and agents, the upgraded management server uses existing encryption keys to communicate with existing and upgraded agents. For an upgraded management server to use an existing encryption key to communicate with a new version 8.*x* agent, you must use the `NQKeyGenWindows.exe` utility to export the key from the upgraded QDB and import it to the new agent. If the existing encryption key was generated using the NetIQ Encryption Utility, `rpckey.exe`, you must use the `NQKeyGenWindows.exe` utility to convert the older key file to the new key format before you import it to the new agent. The `NQKeyGenWindows.exe` utility is located in the `NetIQ\AppManager\bin` folder. For more information about the utility, see the *Administrator Guide for AppManager*.

To convert a key generated using the NetIQ Encryption Utility to the new key format:

1. To convert the older key file to the new key format, run the following command on the management server:

   ```
   NQKeyGenWindows -convert Old_Key_Location New_Key_Location
   ```

2. To check the key information into the QDB, run the following command on the management server:

   ```
   NQKeyGenWindows -db QDB_Name:User_Name:SQL_Server_Name\instance -change
   New_Key_Location
   ```

3. To set the desired security level, run the following command on the management server:

   ```
   NQKeyGenWindows -db QDB_Name:User_Name:SQL_Server_Name\instance -seclev level
   ```

4. Restart the management server.

To import an existing encryption key to a new agent:

1. To extract the agent portion of the key from the QDB, run the following command on the management server:

   ```
   NQKeyGenWindows -db QDB_Name:User_Name:SQL_Server_Name\instance -ckey
   Agent_Key_File_Location
   ```

2. To import the key to the new agent, run the following command on the agent computer:

   ```
   NQKeyGenWindows -agentchange Agent_Key_File_Location
   ```

3.7.4 Changing the Security Level

After you upgrade, if you want to change the security level for a management site from encrypted communications to cleartext communications, run the AMAdmin_AgentConfigSecurityLevel Knowledge Script.

To change the security level for a management site:

1. To change the security level for the agents within your management site, run the AMAdmin_AgentConfigSecurityLevel Knowledge Script.
2. Run the AMAdmin_AgentConfigSecurityLevel Knowledge Script again on each management server.

3. (Conditional) If you have not configured the QDB to store the security key information on each management server computer, edit the following Microsoft Windows registry key:

   ```
   \HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQMS\Config\
   RPC Encryption
   ```

   and change its value from **1** to **0**. You must restart the management server to apply your changes. For more information about how to configure the QDB to store the security key information, see Section 3.7.3, "Using Existing Security Keys for Encrypted Communications," on page 52.

   WARNING: Be careful when editing your Windows registry. If there is an error in your registry, your computer may become nonfunctional. If an error occurs, you can restore the registry to its state when you last successfully started your computer. For more information, see the Help for the Windows Registry Editor

4. (Conditional) If you used the `NQKeyGenWindows.exe` utility to store security information in the QDB, use that utility again and set the `-seclev` option to **0**.
5. Restart your management servers.

For more information about setting or changing the security level for an AppManager management site, see the *Administrator Guide for AppManager*.

3.8 Upgrading Web Management Servers and Operator Web Consoles

Upgrading the Web management server also upgrades the Operator Web Console. Before you start the upgrade, close connections to all Operator Web Consoles.

During the upgrade, the setup program requests permission to temporarily stop the IIS Admin Service and its dependent services. When the upgrade completes, the setup program restarts the services.

4 Upgrading and Migrating Agent Components

This chapter describes how to upgrade Windows agent components and how to migrate a version 7.*x* or 8.*x* Windows agent to a new version 8.*x* AppManager repository (QDB). For information about upgrading the UNIX agent, see the NetIQ UNIX Agent documentation, which is included in the AppManager UNIX download package.

For Windows computers, the agent consists of the following components:

- NetIQ AppManager Client Resource Monitor (`NetIQmc`) Windows service
- NetIQ AppManager Client Communication Manager (`NetIQccm`) Windows service
- Local repository
- AppManager for Microsoft Windows module

The setup program upgrades the agent services and local repository first, and then upgrades the AppManager for Microsoft Windows module.

4.1 Understanding Agent Versions Supported for Upgrade

You can upgrade the following versions of the Windows agent to version 8.*x*:

- 7.0.1
- 7.0.2
- 7.0.25
- 8.0.1
- 8.0.2

4.2 Understanding When to Upgrade

You do not have to upgrade all of your agents to version 8.*x* at the same time. When you upgrade a management server, you also upgrade the agent on the management server computer. Otherwise, you can upgrade agents on an as-needed basis, but version 7.*x* agents that you do not upgrade will not support the following features:

- Discovery improvements

  For more information about the discovery improvements available with AppManager version 8.*x*, see Section 1.2.2, “Discovery Improvements,” on page 13.
- Security improvements

  For more information about the security improvements available with AppManager version 8.*x*, see Section 1.2.3, “Security Improvements,” on page 14.

If you installed a new version 8.*x* QDB and want to maintain existing agents, you must migrate the agents to the new QDB. For more information about migrating an existing agent to a new QDB, see Section 4.9, “Migrating an Agent to a New QDB,” on page 58.

4.3 Understanding the Order for Upgrading Components

Before you upgrade an agent, upgrade the QDB and management servers with which the agent communicates. While a version 8.*x* management server can communicate with version 7.0.*x* and 8.0.*x* agents, version 8.*x* agents cannot communicate with earlier management server versions. For more information about upgrading QDBs and management servers, see Chapter 3, “Upgrading Management Site Components,” on page 45.

4.4 Discovering Upgraded AppManager 7.*x* Components for Health Monitoring

When you upgrade version 7.*x* components, after the setup program successfully upgrades the agent services and the AppManager for Microsoft Windows module, it automatically runs the Discovery_AMHealth Knowledge Script to prepare AppManager components for health monitoring in Control Center. For more information about using Control Center to monitor the health of your AppManager components, see the *Control Center User Guide for AppManager*.

4.5 Understanding Upgrade Methods

You can upgrade Windows agents by running the setup program locally on the agent computer, or you can use Control Center to upgrade agents on remote computers. For more information about using the setup program to upgrade agents, see Section 4.6, “Upgrading Agent Components on the Local Computer,” on page 56. For more information about using Control Center to upgrade agents, see Section 4.7, “Using Control Center to Upgrade Agents on Remote Computers,” on page 57.

The upgrade process does not change the settings from your previous agent installation and retains existing jobs, data, and events in the local repository.

4.6 Upgrading Agent Components on the Local Computer

This section describes the steps required to upgrade agent components on the local computer.

To upgrade agent components on the local computer:

1. Ensure that the upgrade of the QDB and management servers with which the agent communicates completed successfully.
2. Start the upgrade and generate a pre-installation check report.

   For more information about generating the report, see Section 1.9, “Starting an Upgrade and Generating a Pre-Installation Check Report,” on page 23.
3. Complete the agent setup program.

If you previously used Performance Monitor to monitor the operational health and performance of the agent, you must manually update the performance counters after the upgrade. For more information about manually updating the counters, see Section 4.6.1, “Updating Agent Performance Counters,” on page 57.

4.6.1 Updating Agent Performance Counters

The setup program does not update performance counters. To continue using the counters, you must manually update them.

To update the agent performance counters:

1. Open a Command Prompt and change directory to the `Windows\System32` (for 32-bit operating systems) or `Windows\SysWOW64` (for 64-bit operating systems) folder.
2. Type the following command and press **Enter**:

   ```
   regedt32.exe /S "Installation_Drive_and_Folder\AppManager\bin\mccnt.reg"
   ```

   For example:

   ```
   regedt32.exe /S "C:\Program Files (x86)\NetIQ\AppManager\bin\mccnt.reg"
   ```

3. Type the following command and press **Enter**:

   ```
   lodctr.exe "Installation_Drive_and_Folder\AppManager\bin\mccnt.ini"
   ```

   For example:

   ```
   lodctr.exe "C:\Program Files (x86)\NetIQ\AppManager\bin\mccnt.ini"
   ```

On 64-bit operating systems, to view the counters, you must open Performance Monitor from the `Windows\SysWOW64` folder.

4.7 Using Control Center to Upgrade Agents on Remote Computers

After you upgrade the QDB, management server, and Control Center components, you can use Control Center to upgrade agents on remote computers.

In Control Center, you will specify a user account to run the agent installation package. Ensure that the account has the required Group Policy object (GPO) setting. The account must be a member of the `Replace a process level token` policy, which determines the user accounts that can call the `CreateProcessAsUser()` application programming interface (API) so that one service can start another. By default, only local system accounts are members of the policy. You can edit the policy in the default domain controller GPO and in the local security policy of workstations and servers. The policy is located in the following path in the Microsoft Management Console:

```
Computer Configuration\Windows Settings\Security Settings\Local Policies\User
Rights Assignment
```

For more information about upgrading agents on remote Windows computers, see the *Control Center User Guide for AppManager*. For information about upgrading agents remotely on UNIX or Linux computers, see the NetIQ UNIX Agent documentation, which is included in the AppManager UNIX download package.

If you previously used Performance Monitor to monitor the operational health and performance of a version 7.*x* agent, you must manually update the performance counters after the upgrade. For more information about manually updating the counters, see Section 4.6.1, “Updating Agent Performance Counters,” on page 57.

4.8 Upgrading Jobs

After you upgrade the agent, existing jobs on the agent computer are not automatically upgraded to use the latest Knowledge Script functionality. Upgrading jobs to use the latest Knowledge Script version allows the jobs to take advantage of the latest Knowledge Script logic while maintaining existing job parameter values. Any associated graph data and event information are also retained if they have not changed. If the latest version of a Knowledge Script includes new parameters, for example, to create different events or data streams, the default values for the new parameters in the latest Knowledge Script are used.

For more information about upgrading jobs after you upgrade an agent, see Chapter 6, “Upgrading Ad Hoc Jobs,” on page 71.

4.9 Migrating an Agent to a New QDB

If you installed a new QDB and want to maintain existing agents for that QDB, you must migrate the agents to the new QDB. You can migrate version 7.*x* and version 8.*x* agents to new QDBs.

To migrate an agent to a new QDB:

1. Use Control Center to stop and then delete all ad hoc jobs and stop all monitoring policy jobs that are running on the agent computer.

   For information about using Control Center to stop and delete jobs, see the *Control Center User Guide for AppManager*.

2. (Conditional) If you are migrating a version 7.*x* agent, use the AMAdmin_SetDeploymentWebService Knowledge Script to update the agent with the name of the version 8.*x* Deployment Web Server.

   For information about running the Knowledge Script, see the *AppManager Knowledge Script Reference Guide*.

3. Use the AMAdmin_AgentConfigMSRestrictions Knowledge Script to edit the list of management servers that are authorized to communicate with the agent with the names of the management servers for the new QDB.

   For information about running the Knowledge Script, see the *AppManager Knowledge Script Reference Guide*.

4 On the agent computer, cold start both of the agent services. The following is an example of a batch job you can run to cold start the services:

```
REM Cold Start AM agent
@echo off
for /f %%a in (C:\BatchFiles\servers.txt) do call :restart %%a
goto :EOF
:restart
sc \\%1 stop netiqmc
sc \\%1 start netiqmc -oa
sc \\%1 stop netiqccm
sc \\%1 start netiqccm -oa
goto :EOF
:EOF
pause
exit
```

5 From a command prompt on the agent computer, run `NetIQctrl.exe` to start the `NetIQctrl` command-line program.

By default, `NetIQctrl.exe` is located in `AppManager\bin`.

6 Run the following command to verify that no management servers are currently communicating with the agent:

```
listms mc_hostname NetIQmc
```

where *mc_hostname* is the name of the agent computer

The list should be empty.

7 Use the Control Center console to delete the agent computer from the old QDB.

(Conditional) If you are migrating a version 7.*x* agent, use the version 7.0.4 Control Center console.

8 Use the Control Center console to add the agent computer to the new QDB.

For information about adding a computer to a QDB, see the *Control Center User Guide for AppManager*.

9 On the agent computer, delete any existing map queue files from the `AppManager\dat\mapque` folder. The following is an example of a batch job you can run to delete the files:

```
REM Warm Start AM agent with Mapque deletion
@echo off
for /f %%a in (C:\BatchFiles\servers.txt) do call :restart %%a
goto :EOF
:restart
sc \\%1 stop netiqmc
sc \\%1 stop netiqccm
cmd /c del \\%1\C$\"Program Files\NetIQ\AppManager\dat\mapque"\*.dat
cmd /c del \\%1\D$\"Program Files\NetIQ\AppManager\dat\mapque"\*.dat
cmd /c del \\%1\C$\"Program Files (x86)\NetIQ\AppManager\dat\mapque"\*.dat
cmd /c del \\%1\D$\"Program Files (x86)\NetIQ\AppManager\dat\mapque"\*.dat
sc \\%1 start netiqmc
sc \\%1 start netiqccm
goto :EOF
:EOF
pause
exit
```

10 (Conditional) If you are migrating a version 7.*x* agent, to force software inventory to be sent to the Deployment Web Service, warm start both of the agent services. Following is an example of a batch file you can use to warm start the services:

```
REM Warm Start AM agent
@echo off
for /f %%a in (C:\BatchFiles\servers.txt) do call :restart %%a
goto :EOF
:restart
sc \\%1 stop netiqmc
sc \\%1 start netiqmc
sc \\%1 stop netiqccm
sc \\%1 start netiqccm
goto :EOF
:EOF
pause
exit
```

11 In the Control Center console, view the properties of the agent computer in the **Server** view to verify that the correct primary and secondary management servers are assigned to the agent.

12 (Conditional) If the agent does not have primary and secondary management servers assigned to it, run the AMAdmin_SetPrimaryMS Knowledge Script to set the primary and secondary management servers.

For information about running the Knowledge Script, see the *AppManager Knowledge Script Reference Guide*.

13 From a command prompt on the agent computer, run `NetIQctrl.exe` to start the `NetIQctrl` command-line program.

By default, `NetIQctrl.exe` is located in `AppManager\bin`.

14 Run the following command to verify that the agent is only aware of management servers for the new QDB:

```
listms mc_hostname NetIQmc
```

where *mc_hostname* is the name of the agent computer

15 (Conditional) If you are migrating a version 7.*x* agent, in the **Deployment** view of the **Navigation** pane in the Control Center console, click **Deployment** and then click **Software Inventory** to verify that the agent sent software inventory to the version 8.*x* Deployment Web Service.

5 Upgrading Control Center Components

This chapter describes how to upgrade Control Center components. Control Center consists of the following components:

- CCDB
- Control Center and Deployment services, which includes the following services:
  - Command queue service
  - Deployment Service
  - Deployment Web Service
- Control Center console

To take advantage of the features available with AppManager version 8.*x*, you must upgrade your Control Center components. For more information about the features available with version 8.*x*, see Section 1.2, “Understanding New Features,” on page 12 and the *Control Center User Guide for AppManager*.

5.1 Understanding Upgrade Prerequisites

This section describes prerequisites your environment must meet before you upgrade Control Center components.

5.1.1 Understanding Control Center Versions Supported for Upgrade

You cannot upgrade version 7.0.1 or 7.0.3 Control Center components to version 8.*x*. To upgrade version 7.0.1 or 7.0.3 Control Center components to version 8.*x*, you can either uninstall them and install new version 8.*x* components, or you can first upgrade them to version 7.0.4 or AppManager 7 Platform Update.

For more information about installing new version 8.*x* Control Center components, see the *Installation Guide for AppManager*.

For more information about obtaining a version of AppManager that is supported for upgrade, contact Technical Support.

5.1.2 Understanding Microsoft SQL Server Versions Supported for Upgrade

Version 8.*x* CCDBs must be hosted on one of the following versions of Microsoft SQL Server:

- Microsoft SQL Server 2005 Standard or Enterprise edition Service Pack 3 or later (32-bit or 64-bit)
- Microsoft SQL Server 2008 Standard or Enterprise edition Service Pack 1 or later (32-bit or 64-bit)

- Microsoft SQL Server 2008 R2 Standard or Enterprise edition (32-bit or 64-bit)
- Microsoft SQL Server 2012 Standard or Enterprise edition (32-bit or 64-bit)

Before you can upgrade a CCDB hosted on Microsoft SQL Server 2000 to AppManager 8.*x*, you must migrate the CCDB to a supported version of Microsoft SQL Server. Once the CCDB is hosted on a supported version of SQL Server, you can upgrade it to version 8.*x*. For more information about migrating a CCDB before upgrading it to version 8.*x*, see Chapter 2, "Migrating Version 7.x Repositories," on page 25.

AppManager 8.*x* supports migrating version 8.*x* CCDBs hosted on Microsoft SQL Server 2005 to SQL Server 2008, 2008 R2, or 2012 and 8.*x* CCDBs hosted on SQL Server 2008 or 2008 R2 to SQL Server 2012. For more information about migrating an upgraded CCDB, see Appendix A, "Migrating Version 8.x Repositories," on page 77.

5.1.3 Understanding SQL Server Compatibility Requirements

If a CCDB was previously hosted on Microsoft SQL Server 2000 and the SQL Server compatibility level remains SQL Server 2000, the CCDB upgrade cannot complete successfully. If the setup program detects that the SQL Server compatibility level is set to SQL Server 2000, it sets the compatibility level to the appropriate version. The setup program only modifies the CCDB. It does not modify any other databases.

5.2 Understanding QDB Requirements

A version 8.*x* CCDB only supports a version 8.*x* primary QDB. You can either create a new version 8.*x* primary QDB or upgrade your existing primary QDB to version 8.*x*. If you create a new primary QDB, in order for the product to function correctly, you must add at least one agent computer to the QDB and discover the computer. For more information about creating a new version 8.*x* primary QDB, see the *Installation Guide for AppManager*. For more information about upgrading a QDB, see Chapter 3, "Upgrading Management Site Components," on page 45.

You can attach older QDBs to a version 8.*x* CCDB as non-primary QDBs. Older non-primary QDBs attached to a version 8.*x* CCDB must meet the hotfix and SQL Server version requirements described in the following table. The hotfixes are available on the AppManager Suite Hotfixes Web site.

Non-primary QDB Version	Microsoft SQL Server Version
7.0.1, with Hotfix 7011821 or later	2005 Standard or Enterprise edition Service Pack 2 or later (32-bit)
7.0.3, with Hotfix 7011821 or later	2005 Standard or Enterprise edition Service Pack 2 or later (64-bit)
AppManager 7 Platform Update, with Hotfix 7011821 or later	One of the following (32-bit or 64-bit): ◆ 2008 Standard or Enterprise edition Service Pack 1 or later ◆ 2008 R2 Standard or Enterprise edition

Non-primary QDB Version	Microsoft SQL Server Version
8.0.2, with Hotfix 7011822 or later	One of the following (32-bit or 64-bit): ◆ 2005 Standard or Enterprise edition Service Pack 2 or later ◆ 2008 Standard or Enterprise edition Service Pack 1 or later ◆ 2008 R2 Standard or Enterprise edition
8.0.3, with Hotfix 7011823 or later	

When you upgrade the CCDB to version 8.*x*, if the setup program detects a QDB hosted on Microsoft SQL Server 2000, the CCDB upgrade cannot continue.

5.3 Understanding Agent Requirements

Because version 8.*x* Control Center and Deployment services require an agent, if the setup program detects that the services are available for upgrade but no agent is present, it automatically installs a version 8.*x* agent after it upgrades the services.

5.4 Understanding the Order for Upgrading Control Center Components

If you have Control Center components installed on multiple computers, ensure that you upgrade components in the correct order. On the computer where you run the setup program, the program ensures you upgrade components in the correct order. However, when you have components installed on multiple computers, ensure that you upgrade the computers in order. For example, if you are upgrading the CCDB and the command queue service and they reside on different computers, run the setup program on the CCDB computer before you run it on the command queue service computer.

Upgrade Control Center components in the following order:

1. CCDB
2. Command queue service
3. Deployment Service
4. Deployment Web Service
5. Control Center console

If you do not upgrade components in the recommended order, the Deployment Service will not start after the upgrade.

5.5 Discovering Upgraded 7.*x* Control Center Components for Health Monitoring

Once the setup program successfully upgrades a version 7.*x* command queue service, if an upgraded agent is already present, the setup program automatically runs the Discovery_AMHealth Knowledge Script to prepare Control Center components for health monitoring in Control Center.

Otherwise, the setup program runs the Knowledge Script after the agent upgrade. For more information about using Control Center to monitor the health of your AppManager components, see the *Control Center User Guide for AppManager*.

5.6 Understanding Changes to Control Center Settings and Functionality After Upgrading from 7.*x*

This section describes functionality changes to consider before upgrading version 7.*x* Control Center components. For the items described in this section, you might want to note your existing settings before upgrading.

5.6.1 Understanding Changes to Custom XML Menu Extensions

For custom XML menu extensions, Control Center version 8.*x* expects at least two levels and does not support extensions created for previous AppManager versions. When you upgrade from version 7.*x*, the setup program removes existing custom XML menu extensions. You must recreate the extensions after you upgrade.

5.6.2 Understanding Changes to Jobs and Events View Settings

With Control Center version 8.*x*, **Jobs** and **Events** views are organized hierarchically by default, with child items grouped under parent items in a collapsible list. Because the views are hierarchically organized, the default setting for **Jobs** views is **Do not show anything on a separate row** and the default setting for **Events** views is to display event messages on the same row as the event. If you previously configured a **Jobs** or **Events** view to display information on a separate row, the setting will not be preserved after you upgrade from version 7.*x*. To change the default settings after you upgrade, in the **Enterprise Layout** view of the **Navigation** pane, right-click the view you want to change and then select **View Properties** > **General**.

5.6.3 Understanding Changes to Creating Filters with Wildcard Characters

With Control Center version 8.*x*, when you create filters that use the asterisk (*), percent sign (%), and question mark (?) wildcard characters, use the `Is Like` and `Is Not Like` operators. With previous Control Center versions, filters with wildcard characters used the `Equal` and `Not Equal` operators.

5.6.4 Understanding Changes to Control Center Console Options

With previous versions of Control Center, changes you made to the Control Center console options applied to all console users. With Control Center version 8.*x*, you set Control Center console options for individual users. When you upgrade to version 8.*x* from version 7.*x*, the setup program resets the previous console options to the default values for version 8.*x*. Individual users can then change the default values as needed. For more information about changing the default options, see the *Control Center User Guide for AppManager*.

The setup program resets the following Control Center console options:

- Events

- General
- Servers
- Jobs
- Skin Selection

5.6.5 Understanding Permission Inheritance

To simplify Control Center security administration in AppManager 8.*x*, permissions you assign to a management group automatically apply to each child of that management group. Permission inheritance eliminates the need to assign user groups or permission sets individually on each child management group, and allows administrators to delegate access for specific user groups to only their portion of the management group hierarchy.

Permission inheritance applies to both new and existing management groups. If an existing user has different permissions on a parent management group and a child of that parent, or if a user has permissions on a parent management group but not children of that parent, after you upgrade, the user will have the same permissions on the parent and each child of that parent. For example, assume a user has the following permissions before you upgrade from version 7.*x*:

- View and edit jobs on parent management group 1
- View jobs on child management group 1A
- No permissions on child management group 1B

After you upgrade, the user will have permissions to view and edit jobs on management groups 1, 1A, and 1B.

5.7 Configuring Kerberos Delegation for a Distributed Control Center Environment

NetIQ Corporation recommends distributing Control Center components across computers to improve performance. If you plan to use Windows authentication to authenticate users between Control Center and the QDBs it manages in a distributed Control Center environment, configure Kerberos constrained delegation to ensure successful communication between Control Center components and QDBs. If Kerberos constrained delegation is not properly configured, connections between Control Center components and QDBs will fail with the following error:

```
Login failed for user 'NT AUTHORITY\ANONYMOUS LOGON'
```

To avoid this error, complete the following tasks:

- Prepare each QDB computer and the CCDB computer to authenticate using Kerberos.
- Configure the SQL Server and SQL Server Agent services for each QDB computer and the CCDB computer to be trusted for delegation.
- Configure the CCDB computer to impersonate the SQL Server and SQL Server Agent services for each QDB computer that connects to Control Center.

To prepare the QDB and CCDB computers to authenticate using Kerberos:

1. Set TCP/IP and Named Pipes as the preferred client protocols on the SQL Server and ensure that TCP/IP is listed first.
2. Determine the TCP dynamic port number the SQL Server and SQL Server Agent services are using and verify that it is not blocked by a firewall.

3. Ensure that the SQL Server and SQL Server Agent services are running under a domain account.
4. (Conditional) If you are running Microsoft Windows Server 2008, 2008 R2, or 2012, run the following commands to create the required service principal names:

   ```
   setspn -A MSSQLSvc/{Fully-
   qualified_Domain_Name_of_the_QDB_or_CCDB_Computer}:{SQL_Server_Name\instance}
   {Domain_Account_Name_Under_Which_the_SQL_Server_and_SQL_Server_Agent_Services_
   Run}
   ```

   ```
   setspn -A MSSQLSvc/{Fully-
   qualified_Domain_Name_of_the_QDB_or_CCDB_Computer}:{Port_on_Which_the_SQL_
   Server_and_SQL_Server_Agent_Services_Run}{Domain_Account_Name_Under_Which_the_
   SQL_Server_and_SQL_Server_Agent_Services_Run}
   ```

   ```
   setspn -A MSSQLSvc/
   {NETBIOS_Name_of_the_QDB_or_CCDB_Computer}:{SQL_Server_Name\instance}
   {Domain_Account_Name_Under_Which_the_SQL_Server_and_SQL_Server_Agent_Services_
   Run}
   ```

   ```
   setspn -A MSSQLSvc/
   {NETBIOS_Name_of_the_QDB_or_CCDB_Computer}:{Port_on_Which_the_SQL_
   Server_and_SQL_Server_Agent_Services_Run}
   {Domain_Account_Name_Under_Which_the_SQL_Server_and_SQL_Server_Agent_Services_
   Run}
   ```

To configure the SQL Server services to be trusted for delegation:

1. On the domain controller, in Active Directory Users and Computers, right-click the domain account under which the SQL Server and SQL Server Agent services run and select **Properties**.
2. On the **Delegation** tab, select the following options:
   - **Trust this user for delegation to specified services only**
   - **Use Kerberos only**
3. Click **Add**.
4. Click **Users and Computers**.
5. Enter the name of the domain account under which the SQL Server and SQL Server Agent services run and click **OK**.
6. Select the `MSSQLSvc` entries associated with the QDB or CCDB computer and click **OK**.
7. (Conditional) If the SQL Server and SQL Server Agent services will connect to Control Center across a firewall, run the following commands to register the required service principal names:

   ```
   setspn -A MSSQLSvc/{Fully-
   qualified_Domain_Name_of_the_QDB_or_CCDB_Computer}:{DNS_Service_Port}
   {Domain_Account_Name_Under_Which_the_SQL_Server_and_SQL_Server_Agent_Services_
   Run}
   ```

   ```
   setspn -A MSSQLSvc/
   {NETBIOS_Name_of_the_QDB_or_CCDB_Computer}:{DNS_Service_Port}
   {Domain_Account_Name_Under_Which_the_SQL_Server_and_SQL_Server_Agent_Services_
   Run}
   ```

   ```
   setspn -A MSSQLSvc/{Fully-
   qualified_Domain_Name_of_the_QDB_or_CCDB_Computer}:{Kerberos_Ticket_Granting_
   Service_Port}
   {Domain_Account_Name_Under_Which_the_SQL_Server_and_SQL_Server_Agent_Services_
   Run}
   ```

```
setspn -A MSSQLSvc/
{NETBIOS_Name_of_the_QDB_or_CCDB_Computer}:{Kerberos_Ticket_Granting_Service_
Port}
{Domain_Account_Name_Under_Which_the_SQL_Server_and_SQL_Server_Agent_Services_
Run}
```

```
setspn -A MSSQLSvc/{Fully-
qualified_Domain_Name_of_the_QDB_or_CCDB_Computer}:{Time_Service_Port}
{Domain_Account_Name_Under_Which_the_SQL_Server_and_SQL_Server_Agent_Services_
Run}
```

```
setspn -A MSSQLSvc/
{NETBIOS_Name_of_the_QDB_or_CCDB_Computer}:{Time_Service_Port}
{Domain_Account_Name_Under_Which_the_SQL_Server_and_SQL_Server_Agent_Services_
Run}
```

8. Restart the SQL Server and SQL Server Agent services on the QDB or CCDB computer.

To configure the CCDB computer to impersonate the SQL Server and SQL Server Agent services for connected QDB computers:

1. In the Local Security Policy application of Administrative Tools, select **Local Policies** > **User Rights Assignment**.
2. Right-click **Impersonate a client after authentication** and select **Properties**.
3. Click **Add User or Group**.
4. For each QDB computer that connects to Control Center, enter the name of the domain account under which the SQL Server and SQL Server Agent services run and click **OK**.

To verify that components are using Kerberos delegation:

1. On the command queue service and QDB computers, run the following command:

   ```
   osql -E -S {CCDB_SQL_Server_Name\instance}
   ```

2. From the osql command prompt, run the following query:

   ```
   select net_transport, auth_scheme from sys.dm_exec_connections where
   session_id=@@spid
   GO
   ```

   The query should return the values `TCP` and `KERBEROS`.
3. On the command queue service and CCDB computers, run the following command:

   ```
   osql -E -S {QDB_SQL_Server_Name\instance}
   ```

4. Repeat Step 2 on page 67.

5.8 Upgrading the CCDB

This section describes the steps required to upgrade a CCDB. For information about migrating a version 7.*x* CCDB to a new computer before upgrading to version 8.*x*, see Chapter 2, “Migrating Version 7.x Repositories,” on page 25.

You can upgrade CCDBs on remote SQL Servers. You no longer have to run the setup program on the SQL Server.

On a SQL Server or instance hosting multiple CCDBs, you must upgrade each CCDB to version 8.*x*. If you have a version 7.*x* CCDB on the same SQL Server or instance as a version 8.*x* CCDB, the Control Center console for the version 7.*x* CCDB might reflect the version 8.*x* command queue service in the status pane.

If you are upgrading a version 7.*x* CCDB, during the upgrade the setup program runs the `CC_ConvertTo3DES.exe` utility to update the AppManager encryption algorithm to be FIPS-compliant. If FIPS is enabled at the operating system level, the setup program cannot run the utility. If the setup program detects that FIPS is enabled at the operating system level, it gives you the option to disable FIPS and retry the upgrade, or continue the upgrade without updating the encryption algorithm. If you choose to continue without updating the encryption algorithm, you can run the utility from the `AppManager\Control Center\bin` folder after the upgrade. If FIPS is enabled at the operating system level, you must disable it before running the utility.

To upgrade the CCDB:

1. Close connections to the CCDB.

   For more information about the connections to close, see Section 2.3, “Preparing to Migrate the Repositories,” on page 26.

2. Create a CCDB backup.

   For more information about creating a CCDB backup, see Section 2.3.2, “Creating Backup Copies of the Repositories,” on page 32. Since you are not migrating the CCDB to a different computer, it is not necessary to copy the backup files to a different location.

3. Ensure that your primary QDB meets requirements.

   For more information about QDB requirements, see Section 5.2, “Understanding QDB Requirements,” on page 62.

4. Ensure that any existing components you plan to maintain in your version 8.*x* Control Center environment meet requirements.

   For more information about maintaining existing components, see Section 1.3.2, “Upgrading Components in a Multiple-QDB, Control Center Environment,” on page 17.

5. Ensure that there are no commands in the Control Center queue.

6. Start the upgrade and generate a pre-installation check report.

   For more information about generating the report, see Section 1.9, “Starting an Upgrade and Generating a Pre-Installation Check Report,” on page 23.

7. When you reach the Target SQL Server and Repository Name window, provide the following information and click **Next**:

   - Name of the SQL Server and, if applicable, instance that hosts the CCDB you are upgrading. To specify a SQL Server instance, use the format *Server_Name\instance*.
   - Name of the CCDB you are upgrading.
   - Account that can log in to the SQL Server for the upgrade. Ensure that the account is a member of the `sysadmin` SQL Server role.

   The setup program prepares the existing CCDB data for upgrade.

5.9 Upgrading Control Center and Deployment Services

This section describes the steps required to upgrade the command queue service, Deployment Service, and Deployment Web Service.

To upgrade the Control Center and Deployment services:

1. Ensure that the upgrade of the CCDB with which the services communicate completed successfully.
2. Start the upgrade and generate a pre-installation check report.

   For more information about generating the report, see Section 1.9, “Starting an Upgrade and Generating a Pre-Installation Check Report,” on page 23.
3. Complete the Control Center setup program.

The setup program stops the command queue service and the Deployment Service to perform the upgrade.

5.10 Upgrading the Control Center Console

This section describes the steps required to upgrade the Control Center console.

To upgrade the Control Center console:

1. Ensure that the CCDB and Control Center and Deployment services upgrades completed successfully.
2. Start the upgrade and generate a pre-installation check report.

   For more information about generating the report, see Section 1.9, “Starting an Upgrade and Generating a Pre-Installation Check Report,” on page 23.
3. Complete the Control Center console setup program.

6 Upgrading Ad Hoc Jobs

This chapter describes how to upgrade ad hoc jobs to use the latest Knowledge Scripts.

6.1 Understanding Upgrade Methods

After you upgrade the agent, existing ad hoc jobs on the agent computer are not automatically upgraded to use the latest Knowledge Script functionality. Upgrading ad hoc jobs to use the latest Knowledge Script version allows the jobs to take advantage of the latest Knowledge Script logic and properties. If you customized any job parameter values and want to retain the customizations, you have the option to only upgrade the Knowledge Script logic. Any associated graph data and event information are also retained if they have not changed. If the latest version of a Knowledge Script includes new parameters, for example, to create different events or data streams, the default values for the new parameters in the latest Knowledge Script are used.

You do not need to manually upgrade monitoring policy jobs. When a Knowledge Script that is a member of the Knowledge Script Group that started the job is upgraded, the changes are automatically propagated to the job.

The method you use to upgrade ad hoc jobs depends on whether the jobs are associated with default-named Knowledge Scripts or renamed Knowledge Scripts. You can upgrade the following types of jobs:

- Ad hoc jobs
- Ad hoc Knowledge Script Group jobs

You can use the Knowledge Script propagation feature in Control Center or run the AMAdmin_UpgradeJobs Knowledge Script to upgrade ad hoc jobs associated with default-named Knowledge Scripts. For more information about using the Knowledge Script propagation feature, see Section 6.2, “Upgrading Ad Hoc Jobs for Default-named Knowledge Scripts and Knowledge Script Group Members Using Propagation,” on page 71. For more information about using the AMAdmin_UpgradeJobs Knowledge Script, see Section 6.3, “Upgrading Jobs for Default-named Knowledge Scripts Using AMAdmin_UpgradeJobs,” on page 73.

You can use the Knowledge Script Propagation wizard in Control Center to upgrade ad hoc jobs associated with renamed Knowledge Scripts. For more information about upgrading ad hoc jobs associated with renamed Knowledge Scripts, see Section 6.4, “Upgrading Ad Hoc Jobs Associated with Renamed Knowledge Scripts and Knowledge Script Group Members,” on page 75.

6.2 Upgrading Ad Hoc Jobs for Default-named Knowledge Scripts and Knowledge Script Group Members Using Propagation

You can upgrade all ad hoc jobs or reports started by a particular default-named Knowledge Script or Knowledge Script Group member by propagating one or both of the following Knowledge Script elements:

- The actual script logic

If you select to propagate only the script logic, the propagation retains any changes you have made to the default values for job parameters. If the Knowledge Script includes a new parameter, the propagation adds the parameter. If a parameter has been removed from the Knowledge Script, the propagation deletes the parameter.

- The Knowledge Script properties, including changes to schedule, monitoring values, override values, actions, and advanced options

  If you have changed the default values for job parameters and you select to propagate Knowledge Script properties, the propagation replaces your customizations with the default parameter values specified in the Knowledge Script.

If you have configured an override value for a job parameter, the propagation replaces the override value with the override value specified in the Knowledge Script, if one was specified. If an override value is not specified in the Knowledge Script, the default parameter value is propagated.

All corresponding jobs are stopped and restarted with the changes. If you are managing more than one QDB with Control Center, the propagation applies to all corresponding jobs across all QDBs.

Before you propagate the Knowledge Script properties of a report, ensure that you have specified a value for all of the required parameters. For example, make sure you update a report script to include parameter values that are not displayed in the **Values** tab of the Knowledge Script Properties window.

To upgrade ad hoc jobs for default-named Knowledge Scripts and Knowledge Script Group members using propagation:

1. In the **Enterprise Layout** view of the **Navigation** pane in the Control Center console, click the appropriate **Knowledge Scripts** view.
2. In the view pane, click the Knowledge Script or Knowledge Script Group you want.
3. (Conditional) If you are propagating a Knowledge Script, in the **Tasks** pane click **Propagate > Knowledge Script To Ad Hoc Jobs**.
4. (Conditional) If you are propagating a Knowledge Script Group, in the **Tasks** pane click **Propagate > Knowledge Script Group To Ad Hoc Jobs**.
5. (Conditional) If you are propagating a Knowledge Script Group, in the Knowledge Script Group Propagation window, select the members to propagate.
6. In the Knowledge Script Propagation window, select the components of the Knowledge Script that you want to propagate to associated ad hoc jobs, and then click **OK**:

Select...	To propagate...
Propagate knowledge script code to ad hoc jobs	The logic of the Knowledge Script.
Propagate knowledge script properties to ad hoc jobs	The Knowledge Script properties, including schedule, monitoring values, actions, and advanced options.

6.3 Upgrading Jobs for Default-named Knowledge Scripts Using AMAdmin_UpgradeJobs

The AMAdmin_UpgradeJobs Knowledge Script upgrades all child jobs for one or more parent jobs. You can select the parent jobs you want to upgrade based on the following criteria:

- **Knowledge Script**—Upgrades all ad hoc jobs started by the specified Knowledge Script.
- **Knowledge Script Category**—Upgrades all ad hoc jobs started by the specified Knowledge Script category.
- **Parent Job Identifier**—Upgrades all ad hoc child jobs that belong to the specified Parent Job ID.
- **Monitoring Policy**—Upgrades all policy-based jobs started by the specified Knowledge Script Group. If you are using a Knowledge Script Group in one or more monitoring policies, AMAdmin_UpgradeJobs upgrades jobs in all of the affected monitoring policies. This option does not upgrade ad hoc jobs started by a Knowledge Script Group.

The Windows user account for the agent services on the computer where you run this Knowledge Script must belong to the AppManager Administrator role. To see a list of valid AppManager administrators, in AppManager Security Manager expand **AppManager Roles** and click **Administrator**.

For more information about the AMAdmin_UpgradeJobs Knowledge Script, see the *AppManager Knowledge Script Reference Guide*.

6.3.1 Performing an Instant-check Query Before Upgrading Jobs

Before you upgrade jobs using the AMAdmin_UpgradeJobs Knowledge Script, perform an instant-check query to identify jobs that have not been upgraded. The instant-check query identifies jobs by AppManager version and displays both Microsoft Windows and UNIX jobs. The query also identifies the agent version for each job.

You can perform the following instant-check queries:

- **Out-of-date parent jobs**—Identifies parent jobs associated with a Knowledge Script that is not the latest version.
- **Up-to-date parent jobs**—Identifies parent jobs using the latest version of a Knowledge Script.
- **Old parent jobs with no upgrade**—Identifies jobs associated with an old Knowledge Script for which there is no new version. If the query returns any parent job IDs, the associated Knowledge Script has either been discontinued, or is a Knowledge Script you created or customized under a new name without creating a new version in the QDB. If the query returns no parent job IDs, no parent jobs are using out-of-date Knowledge Scripts.
- **Agent build IDs**—Lists the agent build number on each computer. The query can help you identify agents to upgrade.
- **Monitoring-policy jobs**—Identifies all jobs that are part of a monitoring policy. The query lists the jobs according to the view or server group associated with the monitoring policy and then sorts them by Knowledge Script Group.

  You cannot upgrade backlevel UNIX jobs that are policy-based. After you upgrade the backlevel UNIX agent to the latest version, remove the existing backlevel policy-based jobs and create new policy-based jobs.

6.3.2 Identifying Knowledge Scripts with New Parameters

Before you upgrade jobs, use the **Generate report** option to identify Knowledge Scripts with new parameters. The report provides detailed information about the changes to the Knowledge Script, including a list of new or changed parameters and the default values for those parameters.

Each time you run the AMAdmin_UpgradeJobs Knowledge Script, regardless of whether you choose to upgrade jobs or only generate the report, AppManager reports job upgrade information and saves the report in the `\NetIQ\Temp\NetIQ_debug\`*`Computer_Name`*`\jobupgrade` folder. For more information about viewing the reports, see Section 6.5, “Verifying That a Job Has Been Upgraded,” on page 76.

6.3.3 Viewing Sample Job Upgrade Reports

Each time you run an AMAdmin_UpgradeJobs Knowledge Script job, the Knowledge Script creates the following reports in the `\NetIQ\Temp\NetIQ_Debug\`*`Computer_Name`*`\jobupgrade` folder:

- `Upgradejob_`*`id`*`.txt`, where *id* is the AM_AdminUpgradeJobs job ID

  This report lists the jobs that are eligible for upgrade.
- `Upgradejob_`*`id`*`.rpt`, where *id* is the AM_AdminUpgradeJobs job ID

  This report provides detailed information about the changes that will be applied to jobs that are eligible for upgrade.
- `Upgradejob_`*`id`*`.log`, where *id* is the UpgradeJobs job ID

  This report lists the job IDs of the jobs eligible for upgrade and references the corresponding `.rpt` and `.log` files for more information.

If the child of a specified parent job is running on an agent that has not been upgraded to the current version and you specified the **Restricted** upgrade option, the `UpgradeJob_`*`id`*`.txt` file displays information similar to the following example:

```
Connected to SQL Server : RACKR14 repository QDB.
Time stamp: 03/03/11 14:20:47
  [Child Job] [Parent Job] [Build ID]  [Computer\KS]
2 5.0.1 agent(s) found.
2 6.0.2 agent(s) found.
Parent job 436 is skipped because under restricted mode, there cannot be any non-
8.0 agents.
Upgrade is finished.
Please check upgradejob_1343.rpt and upgradejob_1343.log located in
D:\NetIQ\Temp\NetIQ_Debug\RACKR14\jobupgrade.
Time stamp: 03/03/11 14:20:47
```

If the child of a specified parent job can be upgraded with parameter changes, the `UpgradeJob_`*`id`*`.rpt` file displays information similar to the following example:

```
Connected to SQL Server : RACKR14 repository QDB.
Time stamp: 03/03/11 15:14:30
************************************************************
Parent job 54 can be upgraded under force mode.
2 5.0.1 agent(s) found.
2 6.0.2 agent(s) found.
1)
Child job ID = 55
Parent job ID = 54
KS name = NT_CpuLoaded
Machine name = RACKN08
Version = 5.0
Job 55 can be upgraded.
The following parameters in the existing job are not found in the new version of
```

```
the KS:
1) Event? (y/n)
Existing value is y.
2) Collect Data? (y/n)
Existing value is y.
3) Overall Load? (y/n)
Existing value is y.
4) Cpu Threshold >
Existing value is 0
. . .
14) Threshold - Processor queue length
Default value is 0
Check for OldParameter tag
1) Create event if total system CPU is high?
Default value is y
OldParameter tag value = ?DO_EVENT="y" ((AND)) DO_OVERALL="y":"y":"n".
New StringValue = "y"
. . .
4) Severity - Individual CPU
Default value is 15
OldParameter tag value = ?DO_EVENT="y" ((AND)) DO_OVERALL="n":Severity:$default$.
No matching value, will keep original.
. . .
10) Threshold - Individual CPU
Default value is 98
OldParameter tag value = ?DO_OVERALL="n":TH_UTIL:$default$.
No matching value, will keep original.
```

If the child of a specified parent job cannot be upgraded because the agent on which it is running is from a version no longer supported (such as 6.0), the entry is similar to the following example:

```
Parent job 1536 cannot be upgraded under restricted mode.
29 6.0 agents are found.
Please upgrade these agents and restart the upgrade process.
```

In this case, upgrade the agent or use the **Force** option to upgrade the jobs on the older agent.

6.4 Upgrading Ad Hoc Jobs Associated with Renamed Knowledge Scripts and Knowledge Script Group Members

You can use the Knowledge Script Propagation wizard in Control Center to upgrade jobs associated with Knowledge Scripts you created by copying and renaming default Knowledge Scripts. You can propagate the new or modified Knowledge Script to:

- Renamed Knowledge Scripts and ad hoc jobs that were started from those renamed Knowledge Scripts.
- Renamed Knowledge Scripts within Knowledge Script Groups.
- Only renamed Knowledge Scripts.

You can upgrade one or both of the following Knowledge Script elements:

- The actual script logic

  If you select to propagate only the script logic, the propagation retains any changes you have made to the default values for job parameters. If the Knowledge Script includes a new parameter, the propagation adds the parameter. If a parameter has been removed from the Knowledge Script, the propagation deletes the parameter.

- The Knowledge Script properties, including changes to schedule, monitoring values, override values, actions, and advanced options

  If you have changed the default values for job parameters and you select to propagate Knowledge Script properties, the propagation replaces your customizations with the default parameter values specified in the Knowledge Script.

If you have configured an override value for a job parameter, the propagation replaces the override value with the override value specified in the Knowledge Script, if one was specified. If an override value is not specified in the Knowledge Script, the default parameter value is propagated.

You cannot undo propagation of new code and properties to renamed Knowledge Scripts. If you update the jobs associated with renamed Knowledge Scripts, all corresponding jobs are stopped and restarted with the changes. If you are managing more than one QDB with Control Center, the propagation applies to all corresponding jobs across all QDBs.

To upgrade ad hoc jobs associated with renamed Knowledge Scripts and Knowledge Script Group members:

1. In the **Enterprise Layout** view of the **Navigation** pane in the Control Center console, click the appropriate **Knowledge Scripts** view.
2. In the view pane, click the default Knowledge Script you want and then in the **Tasks** pane click **Propagate > Knowledge Script To Renamed Knowledge Scripts**.
3. Use the Knowledge Script Propagation wizard to select the renamed Knowledge Scripts you want to upgrade, where to propagate the Knowledge Script changes, and the Knowledge Script elements to propagate. For more information about completing these actions, see the Help in the wizard.

6.5 Verifying That a Job Has Been Upgraded

To verify that a job has been upgraded, customize the Control Center console **Jobs** view to display the Knowledge Script version.

To display the Knowledge Script version:

1. In the **Jobs** view, right-click a column heading and select **Customize Columns**.
2. In the **All Available Columns** list, select **Knowledge Script Version** and click **Add**.
3. Use the **Move Up** and **Move Down** buttons to adjust the order in which the columns are displayed.

A Migrating Version 8.*x* Repositories

This appendix describes how to migrate a version 8.*x* QDB or CCDB to Microsoft SQL Server 2008, 2008 R2, or 2012, including migrating to a new computer.

A.1 Understanding the Migration and Upgrade Process

The recommended method for migrating a version 8.*x* repository is to install a new, empty version 8.*x* repository on the new SQL Server, close connections to the repository you will migrate, create a backup copy, and restore the backup copy over the empty repository on the new SQL Server. After you restore the repository, additional tasks are necessary to ensure proper operation.

If you migrate a repository from a 32-bit computer to a 64-bit computer, you must disconnect the 32-bit repository.

The following checklist outlines the migration process and provides references to detailed information.

Step		Reference
❑	1. Prepare for repository migration.	Section A.2, "Preparing to Migrate the Repositories," on page 77
❑	2. Restore the backup copy over the new, empty repository on the new SQL Server.	Section A.3, "Restoring Repositories on the New SQL Server," on page 82
❑	3. Configure the restored repository on the new SQL Server.	Section A.4, "Configuring Restored Repositories," on page 83
❑	4. Update components and services that connect to the repository.	Section A.5, "Updating Connected Components and Services," on page 87
❑	5. Verify that the migration was successful.	Section A.6, "Verifying Successful Migration," on page 91

A.2 Preparing to Migrate the Repositories

Before you migrate a repository to a new version of Microsoft SQL Server, install a new, empty version 8.*x* repository on the new SQL Server, record information about the repository you will migrate, close connections to it, and create a backup copy.

To prepare for repository migration:

1 On the new SQL Server, install a new version 8.*x* QDB or CCDB. During installation, when you specify the repository name, specify the same name as the repository you will migrate.

For more information about installing a new repository, see the *Installation Guide for AppManager*.

2 (Conditional) If you are migrating a QDB and Control Center manages it, ensure that Control Center uses Windows authentication to connect to the QDB for the migration.

If Control Center currently uses SQL Server authentication to connect to the QDB, use Control Center to change the authentication method. You can change back to SQL Server authentication after the migration. For information about changing the authentication method Control Center uses to connect to a QDB, see the *Control Center User Guide for AppManager*.

3 On each computer, to ensure that the Distributed Transaction Coordinator (DTC) security settings are the same, complete the following steps:

WARNING: If the settings are not the same, the migration will fail.

3a In the Component Services application in Administrative Tools, expand `Component Services\Computers\My Computer\Distributed Transaction Coordinator`.

3b Right-click **Local DTC** and select **Properties**.

3c On the **Security** tab, note the settings.

3d (Conditional) If the settings on the new computer do not match the settings on the old computer, adjust the settings on the new computer and restart it.

4 On the old SQL Server, to note the properties for SQL Server logins with access to the repository, complete the following steps:

4a In Microsoft SQL Server Management Studio, expand *`SQL_Server_Name`*`\Databases`.

4b Right-click the repository and select **New Query**.

4c In the query window, type the following command, and then click **Execute**:

```
SELECT  name,
        CASE WHEN type = 'S' THEN 'SQL'
             ELSE 'Windows'
        END AS 'Type'
FROM    sys.database_principals
WHERE   type IN ( 'S', 'U' )
        AND name != 'dbo'
        AND default_schema_name IS NOT NULL
        AND default_schema_name != 'guest'
ORDER BY name ASC
```

4d Expand `Databases\`*`Repository_Name`*`\Security\Users` and compare the accounts listed in the results table for the query to the accounts in the `Users` folder.

4e For each account that appears in both the results table and the `Users` folder, right-click the user in the `Users` folder and select **Properties**.

4f On the **General** page, note the **Login name** and **Database role membership**.

4g On the **Securables** page, note the **Explicit permissions**.

You will recreate the SQL Server logins after you restore the repository on the new SQL Server.

5 (Conditional)If you changed the schedule for any NetIQ SQL Server jobs, on the SQL Server that hosts the repository you will migrate, complete the following steps to note the schedule settings for each modified job:

5a In Microsoft SQL Server Management Studio, expand `SQL Server Agent\Jobs`.

5b Right-click the job and select **Properties**.

5c On the **Schedules** page, select the job schedule, and then click **Edit**.

5d On the Job Schedule Properties window, note the settings, and then click **OK**.

6 (Conditional) If you will migrate a CCDB that manages remote QDBs, on the SQL Server that hosts the CCDB you will migrate, complete the following steps to note the linked server properties for the remote QDBs:

6a In Microsoft SQL Server Management Studio, expand `Server Objects\Linked Servers`.

6b Right-click a linked QDB and select **Properties**.

6c On the **General** page, note the linked server name and server type.

6d On the **Security** page, note each local login defined and how Control Center makes the connection.

A login can:

- **Be made without using a security context.** If this option is selected, Control Center connects without using any login and password.
- **Be made using the login's current security context**. If this option is selected, Control Center uses the Log On As account for the SQL Server Agent service to log in to the remote QDB.
- **Be made using this security context**. If this option is selected, it implies you checked the **Use SQL Server authentication** option when you added the QDB to Control Center. When you restore the SQL Server link on the new CCDB computer, provide the same SQL Server user name and password you provided when you added the QDB to Control Center.

6e On the **Server Options** page, note the **RPC** and **RPC Out** values.

You will restore the SQL Server links after you restore the CCDB on the new computer.

7 On each computer, to ensure that the SQL Server collation order, sort order, and character set are the same, complete the following steps:

WARNING: If the settings are not the same, the migration will fail.

7a In Microsoft SQL Server Management Studio, right-click the SQL Server instance and select **Properties**.

7b On the **General** page, note the **Server Collation** setting, and then click **OK** to close the Properties window.

7c Right-click the SQL Server instance and select **New Query**.

7d In the query window, type the following command, and then click **Execute**:

```
sp_helpsort
```

The sort order and character set is displayed in the results table.

When you install SQL Server, the collation order is set by default according to the locale of the operating system. You can use advanced installation options to change the collation order. If the collation order is not the same, re-install SQL Server on the new computer and set the collation order to be the same as the collation order on the old computer.

8 (Conditional) If you will migrate the QDB, complete the following steps to close connected services:

8a Click **Start** > **Administrative Tools** > **Services**.

8b For each of the following services, right-click the service and select **Stop**:

- On the SQL Server that hosts the QDB, SQL Server Agent service
- On primary and secondary management servers that connect to the QDB, NetIQ AppManager Management Service

- (Conditional) If you are running NetIQ AppManager Performance Profiler version 4.0.2 or later, on the SQL Server that hosts AppManager Performance Profiler, Analytics service

  Stop the Analytics service at the same time you stop the NetIQ AppManager Management Service.
- On primary and secondary management servers that connect to the QDB, NetIQ AppManager Client Communication Manager and NetIQ AppManager Client Resource Monitor services
- (Conditional) If Control Center manages the QDB, on the command queue service computer, NetIQ AppManager Control Center Command Queue Service
- (Conditional) If Control Center manages the QDB, on the SQL Server that hosts the CCDB, SQL Server Agent service
- (Conditional) If report agents connect to the QDB, on the agent computers, NetIQ AppManager Client Communication Manager and NetIQ AppManager Client Resource Monitor services

If a service is set to automatically restart when it stops, disable the service.

9 (Conditional) If you will migrate the CCDB, complete the following steps to close connected services:

9a Click **Start** > **Administrative Tools** > **Services**.

9b For each of the following services, right-click the service and select **Stop**:

- On the command queue service computer, NetIQ AppManager Control Center Command Queue Service
- On the SQL Server that hosts the CCDB, SQL Server Agent service
- On the Deployment Service computer, NetIQ AppManager Deployment Service
- On the Deployment Web Service computer, World Wide Web Publishing Service that manages the Deployment Web Service and the Web Depot virtual directories

10 (Conditional) If you will migrate the QDB, stop AppManager Connectors that connect directly to it, such as the AppManager Connector for Micromuse Netcool/OMNIbus or AppManager Connector for Security Manager.

11 Close any AppManager consoles, such as the Control Center console and Operator Console, that connect to the repository.

12 (Conditional) If you will migrate the QDB and it is a data source for Analysis Center, complete the following steps on the Data Mart computer to stop the Analysis Center ETL job:

12a In Microsoft SQL Server Management Studio, expand *SQL_Server_Name*`\SQL Server Agent\Jobs`.

12b Navigate to the ETL job.

12c Right-click the job and select **Disable**.

13 To verify that there are no open connections to the repository you will migrate, complete the following steps:

13a On the repository computer, in Microsoft SQL Server Management Studio, expand *SQL_Server_Name*`\Databases`.

13b Right-click the repository and select **New Query.**

13c In the query window, type the following command, and then click **Execute**:

```
USE master
GO
Exec sp_who2
GO
```

13d In the results table, check the **DBName** column for the repository name. The column should not contain entries for the repository.

14 (Conditional) If the **DBName** column contains entries for the repository, complete the following steps for each entry:

14a In the **SPID** column for the row in which the repository name appears, note the SPID number.

14b Right-click the repository and select **New Query**.

14c In the query window, type the following command, and then click Execute:

```
kill SPID_Number
```

15 Repeat Step 13 on page 80 and Step 14 on page 81 until the **DBName** column does not contain entries for the repository.

16 Create a backup copy of the repository you will migrate.

For more information about creating backup copies, see Section A.2.1, “Creating Backup Copies of the Repositories,” on page 81.

After you create a backup copy of the repository you will migrate, you can restore the backup copy over the new, empty repository. For more information about restoring the repository, see Section A.3, “Restoring Repositories on the New SQL Server,” on page 82.

A.2.1 Creating Backup Copies of the Repositories

This section describes how to use Microsoft SQL Server Management Studio to create backup copies of the QDB and CCDB before you migrate them to the new computer.

To create a backup copy of the QDB or CCDB:

1 In Microsoft SQL Server Management Studio, expand *SQL_Server_Name*\Databases.

2 Right-click the repository and select **Tasks** > **Back Up**.

3 On the **General** page, complete the following steps:

3a From the **Database** list, select the repository.

3b Note the setting in the **Recovery model** field.

3c From the **Backup type** list, select **Full**.

3d For **Backup component**, select the **Database** radio button.

4 On the **Options** page, complete the following steps:

4a Select the **Back up to the existing media set** radio button.

4b (Conditional) If you want to add this backup to existing backups, select the **Append to the existing backup set** radio button.

4c (Conditional) If you want to discard existing backups, select the **Overwrite all existing backup sets** radio button.

5 Click **OK** to start the backup.

6 (Conditional) If the Recovery model is set to **FULL** on the **General** page, after the backup completes, complete the following steps to add a backup device for the transaction log and to back up the transaction log:

  6a Right-click the repository and select **New Query**.

  6b In the query window, type the following command, and then click **Execute**:

```
USE master
EXEC sp_addumpdevice 'disk', 'Dump_Device_Log',
'C: \Repository_NameBACKUP\Repository_Name_Log.bak'
GO
BACKUP LOG Repository_Name TO Dump_Device_Log
GO
sp_dropdevice 'Dump_Device_Log'
GO
```

  where *Repository_Name* is the name of the QDB or CCDB you are backing up and *Dump_Device_Log* is the name of the backup device or file

7 Copy the backup file to the computer where you will restore the QDB or CCDB.

After you copy the backup file to the new computer, you can restore the backup copy over the new, empty repository on the new computer. For more information about restoring the repository, see Section A.3, “Restoring Repositories on the New SQL Server,” on page 82.

A.3 Restoring Repositories on the New SQL Server

After installing a new version 8.*x* repository on the new SQL Server and creating a backup copy of the repository you will migrate, use Microsoft SQL Server Management Studio to restore the backup copy over the new, empty repository on the new SQL Server.

If you are migrating both the QDB and the CCDB, restore the QDB first.

To restore repositories on the new computer:

1 On the new computer, stop the SQL Server Agent service.

2 To ensure that the repository you are restoring is not the default database for the account you are using to perform the restore, complete the following steps:

  2a In Microsoft SQL Server Management Studio, expand `SQL_Server_Name\Security\Logins`.

  2b Right-click the account you are using and select **Properties**.

  2c On the **General** page, note the selection in the **Default database** list.

3 (Conditional) If the default database for the account you are using is the repository you are restoring, either change the default database for the account, or log in to SQL Server Management Studio with a different account.

4 To ensure that no users are connected to the new repository, complete the following steps:

  4a Expand `SQLServer_Name\Databases`.

  4b Right-click the repository and select **New Query.**

4c In the query window, type the following command, and then click **Execute**:

```
USE master
GO
Exec sp_who2
GO
```

4d In the results table, check the **DBName** column for the repository name. The column should not contain entries for the repository.

5 (Conditional) If the **DBName** column contains entries for the repository, to close the open connections, complete the following steps for each connection:

5a In the **SPID** column for the row in which the repository name appears, note the SPID number.

5b Right-click the repository and select **New Query**.

5c In the query window, type the following command, and then click **Execute**:

```
kill SPID_Number
```

6 Repeat Step 4 on page 82 and Step 5 on page 83 until the **DBName** column does not contain entries for the repository.

7 Right-click the repository and select **Tasks** > **Restore** > **Database**.

8 On the **General** page, complete the following steps:

8a Select the **From device** radio button and click the button to specify the backup device.

8b On the Specify Backup window, select **File** from the **Backup media** list, and then click **Add**.

8c On the Locate Backup File window, browse to the location where you saved the backup copy, select the backup file, and then click **OK**.

8d Click **OK** to return to the **General** page.

8e Select the backup set to restore.

9 On the **Options** page, complete the following steps:

9a Under **Restore options**, select the **Overwrite the existing database (WITH REPLACE)** check box.

9b Under **Recovery state**, select the **RESTORE WITH RECOVERY** radio button.

9c Click **OK** to restore the repository.

10 After the restore completes, restart the SQL Server Agent service.

After you restore the repository, additional configuration is required. For more information about the configuration tasks, see Section A.4, “Configuring Restored Repositories,” on page 83.

A.4 Configuring Restored Repositories

After you restore the repository on the new SQL Server, additional configuration is required to ensure proper operation.

To configure the restored repository:

1 To verify that the compatibility level of the restored repository is set to the appropriate version of SQL Server, complete the following steps in Microsoft SQL Server Management Studio:

1a Expand *`SQL_Server_Name`*`\Databases`.

1b Right-click the restored repository and select **Properties**.

1c On the **Options** page, ensure that the compatibility level is set to the appropriate version of SQL Server.

For example, if the restored repository is hosted on SQL Server 2008, ensure that the compatibility level is set to SQL Server 2008 (100).

2 To identify the SQL Server user accounts you must recreate for the repository, complete the following steps:

2a Expand `SQL_Server_Name\Databases`.

2b Right-click the repository and select **New Query**.

2c In the query window, type the following command, and then click **Execute**:

```
SELECT  name,
        CASE WHEN type = 'S' THEN 'SQL'
             ELSE 'Windows'
        END AS 'Type'
FROM    sys.database_principals
WHERE   type IN ( 'S', 'U' )
        AND name != 'dbo'
        AND default_schema_name IS NOT NULL
        AND default_schema_name != 'guest'
ORDER BY name ASC
```

2d Expand `Repository_Name\Security\Users` and compare the accounts listed in the results table for the query to the accounts in the `Users` folder. Note the accounts that appear in both locations.

3 To recreate the SQL Server logins for the repository, complete the following steps for each account you noted in Step 2 on page 84:

3a Right-click the repository and select **New Query**.

3b In the query window, type the following command, and then click **Execute**:

```
sp_dropuser 'User_Name'
```

3c Expand `SQL_Server_Name\Security\Logins`.

3d With the exception of the `probe` account, for each account you removed in Step 3b on page 84, right-click **Logins** and select **New Login**.

3e Configure the login with the properties you noted in Step 4 on page 78 of Section A.2, “Preparing to Migrate the Repositories,” on page 77.

3f (Conditional) If a repository account you removed already exists as a SQL Server account, use AppManager Security Manager (for the QDB) or the Control Center console (for the CCDB) to assign the accounts to the repository.

4 To verify the repository owner, complete the following steps:

4a Right-click the restored repository and select **New Query**.

4b In the query window, type the following command, and then click **Execute**:

```
sp_helpdb 'Repository_Name'
```

5 (Conditional) If the repository owner is not correct, to change the owner, type the following command, and then click **Execute**:

```
sp_changedbowner 'Repository_Owner'
```

6 (Conditional) If you restored the repository on a different computer or with a different name, complete the following steps to restore SQL Server jobs in the restored repository:

6a Expand the SQL Server Agent and then expand the `Jobs` folder.

6b (Conditional) If you are restoring SQL Server jobs in the QDB, verify that the following jobs are in the `Jobs` folder:

- NetIQ AMDC Daily
- NetIQ Archive Event
- NetIQ Chart Console Backup
- NetIQ Chart Console Restore
- NetIQ Daily
- NetIQ Dynamic View
- NetIQ Hourly
- NetIQ License Audit
- NetIQ Minutely
- NetIQ Monitoring Policy
- NetIQ MS HealthCheck
- NetIQ Purge Archived Event
- NetIQ PurgeData
- NetIQ Rule Based Dynamic View
- NetIQ Update MG Server-Membership
- NetIQ Uphold Parameter Overrides
- NetIQ VSG Modtime Update
- NetIQ Weekly

6c (Conditional) If you are restoring SQL Server jobs in the CCDB, verify that the following jobs are in the `Jobs` folder:

- NetIQ CC Daily Task
- NetIQ CC Half-Hourly Task
- NetIQ CC Hourly Task
- NetIQ CC Manage SQL Jobs
- NetIQ CC SMV Hourly Task

6d (Conditional) If you previously customized the schedule for a job, customize the schedule again.

7 (Conditional) If you are migrating the QDB and Control Center manages it, complete the following steps to update the `ComponentCurrentVersion` table with the new SQL Server information for the restored QDB:

7a In SQL Server Management Studio on the CCDB computer, right-click the CCDB and select **New Query**.

7b In the query window, to obtain the `DataSourceIntID` value for the QDB, type the following command, and then click **Execute**:

```
Select DataSourceIntID from dbo.ComponentCurrentVersion
where ComponentName = 'NetIQ AppManager Repository'
and MachineName = 'Old_SQL_Server_Name\instance'
```

7c In the results table, note the `DataSourceIntID` value. You will need this value for Step 7e on page 86.

7d Right-click the CCDB and select **New Query**.

7e In the query window, type the following command, and then click **Execute**:

```
update dbo.ComponentCurrentVersion
set MachineName = 'New_QDB_SQL_Server_Name\instance'
where ComponentName = 'NetIQ AppManager Repository'
and DataSourceIntID = 'DataSourceIntID_Value_from_Step 7c on page 86'
```

8 (Conditional) If you are migrating the CCDB, type the following command and then click **Execute** to update the `ComponentCurrentVersion` table with the new SQL Server information:

```
update dbo.ComponentCurrentVersion
set MachineName = 'New_SQL_Server_Name\instance'
where ComponentName = 'NetIQ AppManager Control Center Repository'
```

9 (Conditional) If you are migrating the CCDB and restored it on a different computer, complete the following steps to update the `Version` table with the new computer name:

9a Right-click the restored CCDB and select **New Query**.

9b In the query window, type the following command, and then click **Execute**:

```
update dbo.Version
set MachineName = 'New_Computer_Name'
where Component = 'CCDB'
```

10 (Conditional) If you are migrating the CCDB and it manages remote QDBs, complete the following steps to restore SQL Server links to the remote QDBs:

10a Expand *`SQL_Server_Name`*`\Server Objects`.

10b Right-click the `Linked Servers` folder and select **New Linked Server**.

10c On the **General** page, in the **Linked server** field, specify the name and instance, if applicable, of the SQL Server that hosts the QDB for which you are restoring the link.

10d On the **General** page, for **Server type**, select the **SQL Server** radio button.

10e On the **Security** page, to add local logins defined before you migrated the CCDB, click **Add**.

10f On the **Server Options** page, set the **RPC** and **RPC Out** values to **True**.

10g Click **OK** to restore the SQL Server link.

After you configure the restored repository, update components and services that connect to it. For more information about the components and services to update, see Section A.5, “Updating Connected Components and Services,” on page 87.

A.5 Updating Connected Components and Services

After you configure the restored repository, update components and services that connect to it.

To update components and services:

1 (Conditional) If you are migrating the QDB, complete the following steps to update the primary management server and each secondary management server that connects to it:

1a (Conditional) If you customized any management server port or persistent IOC settings, use the Windows Registry Editor to back up the following registry keys on the management server computer:

On this type of operating system...	Back up...
32-bit	◆ `HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQms\NetIQmc Port` ◆ `HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQms\Port` ◆ `HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQms\Unix Port` ◆ `HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQms\Config\Persistent IOC` ◆ `HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQms\Config\PIOC Map File Path`
64-bit	◆ `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\NetIQ\AppManager\4.0\NetIQms\NetIQmc Port` ◆ `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\NetIQ\AppManager\4.0\NetIQms\Port` ◆ `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\NetIQ\AppManager\4.0\NetIQms\Unix Port` ◆ `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\NetIQ\AppManager\4.0\NetIQms\Config\Persistent IOC` ◆ `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\NetIQ\AppManager\4.0\NetIQms\Config\PIOC Map File Path`

1b From the `AppManager\bin` folder in the location where you installed the management server, type the following command to re-register the management server service:

```
netiqms -r
QDBms:QDB_Name:Windows_or_SQL_Account_User_Name:password:SQL_Server_Name\
instance -ur -i
```

1c Restore persistent IOC settings.

For more information about restoring persistent IOC settings, see Section A.5.1, “Restoring Persistent IOC Settings,” on page 89.

1d (Conditional) If you have UNIX agents, restore the port setting that defines where the management server listens for communications from UNIX agents.

For more information about restoring the port setting, see Section A.5.2, “Restoring the UNIX Port Setting,” on page 90.

1e Restore any registry keys you backed up.

1f Start the NetIQ AppManager Management Server service (`NetIQms`).

2 (Conditional) If you are migrating the CCDB, on each QDB Control Center manages, complete the following steps to enable the QDB to connect to the restored CCDB:

2a In Microsoft SQL Server Management Studio, right-click the QDB and select **New Query**.

2b In the query window, type the following command, and then click **Execute**:

```
UPDATE dbo.CC_CacheManager SET Name = 'New_CCDB_SQL_Server_Name\instance'
WHERE Name = 'Old_CCDB_SQL_Server_Name\instance'
```

3 (Conditional) If you are migrating the QDB and Control Center manages it, complete the following steps to update the QDB connection information in the CCDB:

3a (Conditional) If Control Center uses SQL authentication to communicate with the QDB, configure the QDB with the same SQL Server user account and permissions.

3b Log on to the Control Center console with an account that is a member of the Administrator group and has the `db_owner` database role for the QDB.

3c On the **Global Tasks** tab of the ribbon, click **Manage Repositories.**

3d Select the QDB, and then click **Modify**.

3e Provide the following information, and then click **OK**:

- Name of the SQL Server and instance, if applicable, that hosts the QDB
- Name of the QDB
- Whether to use Windows or SQL Server authentication
- (Conditional) If you select SQL Server authentication, SQL Server account information

3f Click **Close** to close the Manage Repositories window.

4 (Conditional) If you are migrating the CCDB, complete the following steps to update the command queue service:

4a Use the Control Center console to add the Windows user account for the command queue service as a Control Center administrator.

4b In Microsoft SQL Server Management Studio, right-click the CCDB and select **New Query**.

4c In the query window, to clear the previous command queue service settings from the CCDB, type the following commands, and then click **Execute**:

```
delete from Property where Scope = 'cqs'
exec InitialSQLJobs
exec SMVInitialSQLJob
```

4d On the command queue service computer, from the `AppManager\Control Center\bin` `folder`, open the `NQCQS.exe.config` file in a text editor.

4e Under `<appSettings>`, change the value of the `ServerName` parameter to specify the SQL Server and instance that hosts the restored CCDB, and change the value of the `DBName` parameter to specify the restored CCDB name. For example:

```
<appSettings>
  <add key="ServerName" value="MYSQLSERVER\INSTANCE1" />
  <add key="DBName" value="CCDB_Name" />
```

4f Restart the command queue service to apply the changes.

5 (Conditional) If you are migrating the CCDB, complete the following steps to update the Deployment Service:

5a (Conditional) If the Deployment Service will use different credentials or a different account to log on to the migrated CCDB, from the `AppManager\Control Center\bin` folder on the Deployment Service computer, issue the following command to change the account:

```
deploymentservice -setwindowsauth Domain\User_Name password
```

5b Use the Control Center console to add the Windows user account for the Deployment Service as a Control Center administrator.

5c On the Deployment Service computer, from the `AppManager\Control Center\bin folder`, open the `DeploymentService.exe.config` file in a text editor.

5d Under `<appSettings>`, change the value of the `ServerName` parameter to specify the SQL Server and instance that hosts the restored CCDB, and change the value of the `DBName` parameter to specify the restored CCDB name. For example:

```
<appSettings>
  <add key="ServerName" value="MYSQLSERVER\INSTANCE1" />
  <add key="DBName" value="CCDB_Name" />
```

5e Restart the Deployment Service to apply the change.

6 (Conditional) If you are migrating the CCDB, complete the following steps to update the Deployment Web Service:

6a Use the Control Center console to add the Windows user account for the Deployment Web Service as a Control Center administrator.

6b On the Deployment Web Service computer, from the `AppManager\Control Center\web` folder, open the `Web.config` file in a text editor.

6c Under `<appSettings>`, change the value of the `ServerName` parameter to specify the SQL Server and instance that hosts the restored CCDB, and change the value of the `DBName` parameter to specify the restored CCDB name. For example:

```
<appSettings>
    <add key="ServerName" value="MYSQLSERVER\INSTANCE1">
<add key="DBName" value="CCDB_Name" />
```

6d Restart the World Wide Web Publishing Service to apply the change.

After you update the connected components and services, verify successful migration. For more information about verifying successful migration, see Section A.6, “Verifying Successful Migration,” on page 91.

A.5.1 Restoring Persistent IOC Settings

Re-registering the management server service disables persistent IOC settings in the registry. This section describes how to restore the settings.

WARNING: Be careful when editing your Windows registry. If there is an error in your registry, your computer may become nonfunctional. If an error occurs, you can restore the registry to its state when you last successfully started your computer. For more information, see the Help for the Windows Registry Editor.

To restore persistent IOC settings:

1 Click **Start** > **Run**.

2 In the **Open** field, type `regedit`, and then click **OK**.

3 (Conditional) If the management server is installed on a 32-bit operating system, in the left pane of the Registry Editor, navigate to `HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQms\Config`.

4 (Conditional) If the management server is installed on a 64-bit operating system, in the left pane of the Registry Editor, navigate to `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\NetIQ\AppManager\4.0\NetIQms\Config`.

5 In the right pane, double-click **Persistent IOC**.

6 In the **Value data** field, set the value to **1**, and then click **OK**.

7 In the right pane of the Registry Editor, double-click **PIOC Map File Path**.

8 In the **Value data** field, set the value to the location of your persistent IOC files, and then click **OK**.

Typically, the location is `Program Files\NetIQ\AppManager\dat\pioc`.

After you restore the persistent IOC settings, return to Step 1 on page 87 of Section A.5, "Updating Connected Components and Services," on page 87.

A.5.2 Restoring the UNIX Port Setting

Re-registering the management server service resets the port setting that defines where the management server listens for communications from UNIX agents. This section describes how to restore the setting.

WARNING: Be careful when editing your Windows registry. If there is an error in your registry, your computer may become nonfunctional. If an error occurs, you can restore the registry to its state when you last successfully started your computer. For more information, see the Help for the Windows Registry Editor.

To restore the UNIX port setting:

1 Click **Start** > **Run**.

2 In the **Open** field, type `regedit`, and then click **OK**.

3 (Conditional) If the management server is installed on a 32-bit operating system, in the left pane of the Registry Editor, navigate to `HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0\NetIQms`.

4 (Conditional) If the management server is installed on a 64-bit operating system, in the left pane of the Registry Editor, navigate to `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\NetIQ\AppManager\4.0\NetIQms`.

5 In the right pane, double-click **Unix Port**.

6 For **Base**, select **Decimal**.

7 In the **Value Data** field, set the value to the port number you specified when you installed AppManager, and then click **OK**.

The default port where the management server listens for communications from UNIX agents is 9001.

After you restore the UNIX port setting, return to Step 1 on page 87 of Section A.5, "Updating Connected Components and Services," on page 87.

A.6 Verifying Successful Migration

After you update components and services that connect to the migrated repository, verify successful migration.

To verify successful repository migration:

1. (Conditional) If you migrated a QDB, on the primary management server and each secondary management server that connects to the QDB, ensure that the following services are running:
   - NetIQ AppManager Management Service (`NetIQms`)
   - NetIQ AppManager Client Resource Monitor (`NetIQmc`)
   - NetIQ AppManager Client Communication Manager (`NetIQccm`)
2. (Conditional) If you migrated a CCDB, on the command queue service and Deployment Service computers, ensure that the following services are running:
   - NetIQ AppManager Control Center Command Queue Service
   - NetIQ AppManager Deployment Service
3. Log on to the Operator Console and verify that the primary management server appears in the tree view and is not disabled.
4. (Conditional) If the primary management server does not appear in the tree view, re-register the management server.

   For more information about re-registering the management server, see Step 1 on page 87 of Section A.5, “Updating Connected Components and Services,” on page 87.
5. Use the Operator Console and the Control Center console to start some jobs.

   For example, start an NT_Discovery job. When that job completes, start an NT_CpuLoaded job and wait for it to complete.
6. (Conditional) If the jobs do not complete successfully, contact Technical Support.